БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра информатики

О.И.Костюкова

МЕТОДЫ ОПТИМИЗАЦИИ И УПРАВЛЕНИЯ

Оглавление

Люди оптимизируют. Авиационные компании составляют расписания для экипажей и самолетов так, чтобы минимизировать стоимость перевозок. Предприятия пытаются найти такой план выпуска продукции, чтобы максимизировать прибыль. При запуске ракеты, стараются выбрать такую траекторию, чтобы достичь цель за минимальное время.

Природа оптимизирует. Многие физические системы стремятся к такому состоянию, при котором расход энергии минимальный. Молекулы в изолированных химических системах вступают в реакцию с друг другом до тех пор, пока ни минимизируется полная потенциальная энергия их электронов. Лучи света выбирают путь, который минимизирует время прохождения этих лучей.

Оптимизация — важнейший инструмент в теории принятия решений и в анализе физических систем. Чтобы использовать этот инструмент, мы должны, прежде всего, определить некоторую целевую функцию, которая количественно характеризует цель, которую мы стремимся достичь в данной системе. Эта функция должна быть выражена через некоторую количественную меру: время, энергию, расстояние и т.д. Значение целевой функции зависит от конкретного выбора параметров (характеристик) рассматриваемой системы. Такие параметры называются переменными или неизвестными. Наша цель — найти такие значения неизвестных, которые оптимизируют целевую функцию. Часто на переменные накладываются ограничения. Например, мы не можем выпускать продукции больше, чем позволяют имеющиеся ресурсы, автомобиль может двигаться только с ограниченной скоростью (ограничения наложены либо правилами, либо техническими возможностями).

Процесс определения (формирования) целевой функции, переменных, влияющих на систему, и ограничений на эти переменные в данной системе называется моделированием.

Построение подходящей (соответствующей) модели — первый шаг — иногда наиболее важный - в оптимизационном процессе. Если модель получилась очень простой, то она может плохо отражать поведение реальной системы. С другой стороны, очень сложная модель делает невозможным (или очень трудным) процесс решения задачи.

Как только математическая модель построена, мы можем использовать оптимизационные алгоритмы для поиска решения. Обычно, модель и алгоритмы достаточно сложны и требуется серьезная вычислительная техника для реализации этих алгоритмов. Не существует универсальных оптимизационных алгоритмов. Но существует множество алгоритмов, каждый из которых применим к некоторому конкретному классу оптимизационных задач. И часто пользователь ответственен за выбор алгоритма, наиболее подходящего для его конкретной задачи. Этот выбор является важным элементом, т.к. от этого зависит, будет ли задача решена быстро или медленно и будет ли она вообще решена.

После того как выбранный алгоритм будет применен к рассматриваемой задаче, мы должны иметь инструмент, позволяющий нам проверить действительно ли алгоритм нашел решение нашей задачи. Во многих случаях имеются элегантные математические выражения, позволяющие проверить, что имеющийся набор переменных является решением задачи. Такие математические выражения называются критериями оптимальности. Если условия критерия оптимальности выполняются, то мы действительно нашли решение задачи и процесс решения прекращается. Если условия оптимальности не выполняются, они могут дать полезную информацию о том как текущий набор параметров можно улучшить.

Наконец математическая модель может быть улучшена с применение техники, такой как анализ чувствительности, который показывает чувствительность решения к изменению параметром в модели и данных.

Цель дисциплины “Методы оптимизации и управления” - дать студенту представление о моделях оптимизационных задач, дать основы теории математического и динамического программирования, изложить основные методы решения конечномерных экстремальных задач и задач оптимального управления. В результате изучения курса студенты должны уметь составлять математические модели основных типов экстремальных задач, проводить их теоретический анализ, использовать известные методы решения, реализовывать эти методы на ЭВМ и делать выводы по изучаемой задаче. Материал настоящего курса использует знания, полученные студентами при изучении курсов по линейной алгебре, математическому анализу, программированию на ЭВМ. Материал настоящего курса непосредственно связан и базируется на знании дисциплин "Математический анализ "Геометрия и алгебра "Методы численного анализа"(с разделом "Вычислительные методы алгебры"). В свою очередь дисциплина "Методы оптимизации"служит базой для изучения дисциплины "Исследование операций".

В данном курсе будут рассмотрены следующие классы оптимизационных задач и методы их численного решения

Общие задачи линейного программирования,

задачи транспортного типа,

задачи квадратичного программирования,

задачи выпуклого программирования,

задачи нелинейного программирования,

задачи оптимального управления.

Глава 1

Линейное программирование

Создание Данцигом в конце 1940-х годов симплекс-метода стало началом современной эры в оптимизации. Этот метод позволил экономистам формулировать большие модели и систематически и эффективно анализировать их. Открытие Данцига совпало с появлением цифровых компьютеров, и симплекс-метод стал одним из первых примеров удачного использования этой новой революционной технологии. С тех пор реализация симплекс-метода постоянно совершенствуется. Сегодня линейное программирование и симплекс-метод - наиболее широко используемые средства оптимизации, и, несмотря на то что в настоящее время появились новые классы эффективных методов (например, метод внутренней точки), актуальность и важность симплекс-метода гарантированы в обозримом будущем.

§1. Симплекс-метод

1. Постановка задачи. Нормальная, каноническая формы

Линейным программированием называется раздел математики, в котором исследуются задачи оптимизации линейных функций на множествах, определенных линейными равенствами и неравенствами.

Одним из важнейших физических прототипов математической модели задачи линейного программирования является производственная задача. Её содержательная (неформальная) формулировка такова.

На предприятии производится продукция n видов. При этом расходуются ресурсы m типов, имеющиеся в объемах $b_1, b_2, \ldots, b_m$.

Известны затраты a_{ij} i-го ресурса на изготовление единицы продукции j-го вида и прибыль c_j от реализации единицы продукции j-го вида. Требуется найти план производства с максимальной прибылью.

Обозначим через x_j количество единиц планируемой к выпуску продукции j-го типа, $j = \overline{1,n}$.

Тогда

$\sum_{j=1}^{n} a_{ij}x_j$ — расход i-го сырья при плане производства $(x_j, j = \overline{1,n})$, $i = \overline{1,m}$;

$\sum_{j=1}^{n} c_j x_j$ — прибыль от реализации плана производства $(x_j, j = \overline{1,n})$.

Таким образом, наша цель состоит в том, чтобы найти план производства $(x_j, j = \overline{1,n})$, который

- удовлетворяет ограничениям

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i, \quad i = \overline{1,m}; \tag{1.1}$$

$$x_j \ge 0, j = \overline{1,n}, \tag{1.2}$$

- максимизирует функцию

$$\sum_{j=1}^{n} c_j x_j. \tag{1.3}$$

Функция (1.3) называется *целевой функцией* или *критерием качества.*

Ограничения вида (1.1) называются *основными ограничениями*, а ограничения (1.2) – *прямыми ораничениями.*

Математическая модель решаемой задачи имеет вид

$$\sum_{j=1}^{n} c_j x_j \to \max;$$

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i, \ i = \overline{1,m}; \tag{1.4}$$

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

Задача (1.4) — это общий вид задачи линейного программирования **в нормальной форме** (запись покомпонентна).

Удобнее использовать матрично-векторную запись. Обозначим

$I = \{1, 2, \dots, m\}$, $J = \{1, 2, \dots, n\}$ — множества индексов,

$c = c(J) = (c_j, j \in J)'$, $b = b(I) = (b_i, i \in I)'$;

$A = A(I, J) = \begin{pmatrix} a_{ij}, j \in J \\ i \in I \end{pmatrix} = (A_j, j \in J)$, $x = x(J) = (x_j, j \in J)$ где A_j — m-вектор, $A'_j = (a_{ij}, i \in I)$, символ $'$ обозначает операцию транспонирования.

Запись $x \geq 0$ означает, что $x_j \geq 0, j \in J$. Используя введенные обозначения, задачу (1.4) можно записать в виде

$$c'x \to \max, \tag{1.5}$$
$$Ax \leq b, x \geq 0.$$

Задача (1.5) — задача линейного программирования в **нормальной форме** (векторно-матричная запись).

Кроме нормальной формы (1.5) существуют другие виды записи общей задачи линейного программирования. Например:

$$\begin{cases} c'x \to \max, \\ Ax = b,\ x \geq 0; \end{cases} \tag{1.6}$$

$$\begin{cases} c'x \to \max, \\ Ax = b,\ d_* \leq x \leq d^*, \end{cases} \tag{1.7}$$

$$\begin{cases} c'x \to \max, \\ A_1x = b_1,\ A_2x \leq b_2,\ x \geq 0, \end{cases} \tag{1.8}$$

$$\begin{cases} c'x \to \min, \\ \dots \end{cases} \tag{1.9}$$

Форма записи (1.6) называется **канонической формой** записи общей задачи линейного программирования. Между общими формами существует тесная взаимосвязь. Каждая из этих форм может быть сведена к другой, т.е. в некотором смысле эти формы эквивалентны.

В дальнейшем в качестве „ основной“, „базовой“ формы рассматривается каноническая форма (1.6). Покажем, как задача (1.5), записанная в нормальной форме может быть сведена к задаче в канонической форме.

Пусть задана задача вида (1.5). Требуется записать ее в виде (1.6).

Положим $\overline{n} = n + m$, $\bar{J} = \{1, 2, \ldots, n, n+1, \ldots, n+m\}$, $\overline{x} = (x_j, j \in \bar{J})$, $\overline{b} = b$, $\overline{c}' = (\bar{c}_j, j \in \bar{J})$, $\bar{c}_j = c_j, j = \overline{1, n}$; $\bar{c}_j = 0, j = \overline{n+1, n+m}$,

$\bar{A} = (A, E) = \begin{pmatrix} \overline{a}_{ij}, j \in \bar{J} \\ i \in I \end{pmatrix}$, $\overline{a}_{ij} = a_{ij}, j \in J, i \in I$; $\overline{a}_{i\,n+i} = 1, i = \overline{1, m}$; $\overline{a}_{ij} = 0$ при $j > n, j \neq i + n$.

С учетом введенных обозначений задачу (1.5) можно записать в эквивалентной канонической форме

$$\overline{c}'\bar{x} \to \max, \bar{A}\bar{x} = \bar{b},\ \bar{x} \geq 0.$$

Если исходная задача является задачей минимизации, то заменим c на $\bar{c} = -c$ и получаем эквивалентную задачу на максимум. Остальные типы переходов к эквивалентным формам записи рассмотрим на практике.

2. Базисный план

Рассмотрим задачу линейного программирования в канонической форме

$$c'x \to \max, Ax = b, x \geq 0, \tag{1.10}$$

где

$c \in \mathbb{R}^n$ — вектор стоимости, или вектор целевой функции;

$c'x$ — целевая функция или критерий качества;

$A \in \mathbb{R}^{m \times n}$ — матрица условий;

$b \in \mathbb{R}^m$ — вектор ограничений;

$A_j = A(I, j)$ — j-ый вектор условий.

n-вектор x, удовлетворяющий ограничениям задачи (1.10) называется **планом** (допустимым планом).

$X = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\}$ — **множество допустимых планов**,

$Ax = b$ — **основное ограничение**,

$x \geq 0$ — **прямое ограничение**.

Предположим пока, что

$$rankA = m \le n. \qquad (1.11)$$

Допустимый план x^0 называется оптимальным, если $c'x^0 = \max c'x$, где максимум вычисляется по всем допустимым планам $x \in X$.

Определение 1 *План* $x = x(J)$ *называется* **базисным**, *если существует такое подмножество индексов* $J_Б \subset J$, *что*

1) $|J_Б| = m$,

2) $x_j = 0, j \in J_Н = J \setminus J_Б$,

3) $\det A_Б \neq 0$, *где* $A_Б = (A_j, j \in J_Б)$.

Множество $J_{\text{Б}}$ — множество базисных индексов,

множество $J_{\text{Н}}$ — множество небазисных индексов,

$A_{\text{Б}} = (A_j, j \in J_{\text{Б}})$ — базисная матрица, $A_{\text{Н}} = (A_j, j \in J_{\text{Н}})$ — небазисная матрица,

$x_{\text{Б}} = (x_j, j \in J_{\text{Б}})$ — базисные компоненты плана,

$x_{\text{Н}} = (x_j, j \in J_{\text{Н}})$ — небазисные компоненты плана.

Определение 2 *Базисный план* x *называется* **невырожденным**, *если*

$$x_j > 0, j \in J_Б.$$

Примеры. 1. Пусть ограничения задачи имеют вид

$$\begin{cases} x_1 + x_2 + x_3 + x_4 = 1, \\ x_1 - x_2 + x_3 - x_4 = 1, \ x_j \ge 0, \ j = \overline{1,4}. \end{cases}$$

Вектор $x = (1, 0, 0, 0)$ — план.

Пара $(x, J_{\text{Б}} = \{1, 2\})$ — базисный план.

Пара $(x, J_{\text{Б}} = \{1, 4\})$ также — базисный план.

Пара $(x, J_{\text{Б}} = \{1, 3\})$ не является базисным планом.

Приведенные базисные планы вырожденные.

2. Рассмотрим ограничения вида

$$\begin{cases} 2x_1 - 3x_2 + x_3 - x_4 = 3, \\ -x_1 + 2x_2 + 0x_3 + 8x_4 = 6, \ x_j \ge 0, \ j = \overline{1,4}. \end{cases}$$

Пара из допустимого плана $x = (2, 0, 0, 1)$, подмножества индексов $J_{\text{Б}} = \{1, 4\}$ является базисный план (невырожденный).

Замечание 1 *Из определения следует, что для базисного плана выполняются следующие условия $x_Н = 0, Ax = b$. Следовательно, имеем*

$$Ax = A_Б x_Б + A_Н x_Н = A_Б x_Б = b.$$

Из последних соотношений получаем

$$x_Б = A_Б^{-1} b.$$

Следовательно, мы однозначно можем восстановить базисный план, если задано множество его базисных индексов $J_Б$:

$$x_Н = 0, x_Б = A_Б^{-1} b, x = (x_Б, x_Н).$$

3. Формула приращения целевой функции

Пусть x — базисный план с базисом $J_Б$ и базисной матрицей $A_Б = (A_j, j \in J_Б)$. Рассмотрим другой (не обязательно базисный) план $\overline{x} = (\overline{x}_j, j \in J)$.

Обозначим

$$\Delta x = \overline{x} - x.$$

Подсчитаем приращение целевой функции

$$c'\overline{x} - c'x = c'\Delta x. \tag{1.12}$$

По условию $Ax = b, A\overline{x} = b \Rightarrow A\Delta x = 0$. Обозначим $\Delta x_Б = \Delta x(J_Б)$, $\Delta x(J_Н) = \Delta x_Н$. Следовательно

$$A\Delta x = A_Б \Delta x_Б + A_Н \Delta x_Н = 0 \Rightarrow$$

$$\Delta x_Б = -A_Б^{-1} A_Н \Delta x_Н. \tag{1.13}$$

Перепишем (1.12) с учетом (1.13):

$$c'\Delta x = c'_Б \Delta x_Б + c'_Н \Delta x_Н = -(c'_Б A_Б^{-1} A_Н - c'_Н)\Delta x_Н = -\Delta'_Н \Delta x_Н,$$

где

$$\Delta'_Н = (c'_Б A_Б^{-1} A_Н - c'_Н)$$

или

$$\Delta'_Н = (u' A_Н - c'_Н) = (\Delta_j, j \in J_Н), \ \ \Delta_j = u' A_j - c_j, \ j \in J_Н, \tag{1.14}$$

$$u' = c'_{\text{Б}} A_{\text{Б}}^{-1}. \tag{1.15}$$

Вектор u из (1.15) называется m-вектором потенциалов, вектор $\Delta_{\text{н}}$ из (1.14) — вектором оценок.

Окончательно получаем формулу приращения

$$c'\overline{x} - c'x = -\Delta_{\text{н}}\Delta x_{\text{н}} = -\sum_{j \in J_{\text{Н}}} \Delta_j \Delta x_j. \tag{1.16}$$

4. Критерий оптимальности

Пусть x — базисный план с базисной матрицей $A_{\text{Б}}$. Является ли план x оптимальным?

Для ответа на этот вопрос подсчитаем вектор потенциалов (1.15) и вектор оценок (1.14).

Теорема 1 *(Критерий оптимальности). Рассмотрим базисный план $(x, J_Б)$. Неравенства*

$$\Delta_j \geq 0, j \in J_н, \tag{1.17}$$

достаточны, а в случае невырожденности базисного плана $(x, J_Б)$ и необходимы для оптимальности плана x.

Доказательство. Достаточность. По определению $x_{\text{Н}} = 0$ и по построению $\overline{x}_{\text{Н}} \geq 0, \Rightarrow \Delta x_{\text{Н}} = \overline{x}_{\text{Н}} - x_{\text{Н}} = \overline{x}_{\text{Н}} \geq 0$. Согласно формуле приращения (1.16) и неравенствам (1.17) имеем

$$c'\overline{x} - c'x = -\Delta'_{\text{Н}}\Delta x_{\text{Н}} \leq 0 \text{ или}$$
$$c'\overline{x} \leq c'x \text{ для любого плана } \overline{x}.$$

Следовательно, x — оптимальный план.

Необходимость. Пусть $(x, J_{\text{Б}})$ — оптимальный **невырожденный** базисный план. Предположим противное: соотношения (1.17) не выполняются, т.е. существует индекс $j_0 \in J_{\text{Н}}$, такой что

$$\Delta_{j_0} < 0. \tag{1.18}$$

Рассмотрим вектор $\overline{x} = x + \Delta x$, где Δx построим следующим образом. Небазисные компоненты этого вектора определим согласно правилу

$$\Delta x_{j_0} = \theta \geq 0, \Delta x_j = 0, j \in J_{\text{н}} \backslash j_0,$$

базисные компоненты найдем из условия $A\Delta x = 0$:

$$\Delta x_{\text{Б}} = -A_{\text{Б}}^{-1} A_{j_0} \theta = -\theta z,$$

где $z = A_{\text{Б}}^{-1} A_{j_0} = (z_i, i = \overline{1, m})'$.

По построению, вектор $\overline{x}$ удовлетворяет основному ограничению $A\overline{x} = b$ при всех значениях параметра θ.

Так как $x_{\text{н}} = 0$ и $\Delta x_{\text{н}} \geq 0$, то $\overline{x}_{\text{н}} \geq 0$ при любом $\theta \geq 0$. Из невырожденности плана x следует, что

$$x_{\text{Б}} > 0,$$

следовательно, при достаточно малых $\theta > 0$ имеет место неравенство

$$\overline{x}_{\text{Б}} = x_{\text{Б}} + \Delta x_{\text{Б}} = x_{\text{Б}} + \theta(-A_{\text{Б}}^{-1} A_{j_0}) = x_{\text{Б}} - \theta z \geq 0.$$

Из чего следует, что $\overline{x}$ — допустимый план при достаточно малых $\theta > 0$.

Подсчитаем приращение целевой функции на плане $\overline{x}$:

$$c'\overline{x} - c'x = -\Delta_{\text{н}}' \Delta x_{\text{н}} = -\Delta_{j_0} \theta > 0 \Rightarrow$$

$c'\overline{x} > c'x$, что противоречит оптимальности плана x. ⊠

5. Достаточное условие неограниченности целевой функции

Предположим,что для базисного плана x с базисным множеством $J_{\text{Б}}$ достаточное условие оптимальности (1.17) не выполняется, т.е. $\exists j_0 \in J_{\text{н}}$:

$$\Delta_{j_0} < 0.$$

Рассмотрим случай, когда все компоненты вектора $z = A_{\text{Б}}^{-1} A_{j_0}$ неположительны, т.е.

$$(z_i, i = \overline{1, m})' = A_{\text{Б}}^{-1} A_{j_0} \leq 0.$$

В этом случае легко видеть, что компоненты вектора $\overline{x} = (\overline{x}_{\text{Б}}, \overline{x}_{\text{Н}})$, построенного по правилам

$$\overline{x}_{\text{Н}} = \Delta x_{\text{Н}}, \quad \Delta x_{j_0} = \theta \geq 0, \Delta x_j = 0, j \in J_{\text{Н}} \backslash j_0,$$
$$\overline{x}_{\text{Б}} = x_{\text{Б}} - \theta A_{\text{Б}}^{-1} A_{j_0} = x_{\text{Б}} - \theta z,$$

неотрицательны и удовлетворяют условию $A\bar{x} = b$ при $\forall \theta \geq 0$. Следовательно, вектор $\overline{x}$ — допустимый план при $\forall \theta \geq 0$ и на нем приращение целевой функции равно

$$c'\overline{x} - c'x = -\Delta_{j_0} \theta > 0.$$

При $\theta \to \infty$ целевая функция неограниченно возрастает. Таким образом, мы доказали следующую теорему.

Теорема 2 *Если среди оценок $\Delta_{Н}$ базисного плана x существует отрицательная (Δ_{j_0}) и ей соответствует вектор $A_{Б}^{-1} A_{j_0} = z$ с неположительными компонентами, то целевая функция исходной задачи линейного программирования не ограничена сверху на множестве допустимых планов.*

6. Итерация

Продолжим анализ базисного плана x с базисной матрицей $A_{\text{Б}}$ и базисным множеством индексов $J_{\text{Б}} = \{j_1, j_2, \ldots, j_m\}$.

Рассмотрим случай, когда достаточное условие оптимальности (1.17) не выполняется, и ни для одного индекса $j_0 \in J_{\text{Н}}$ с $\Delta_{j_0} < 0$ не выполняется неравенство $z = A_{\text{Б}}^{-1} A_{j_0} \leq 0$, т.е. не выполняется и достаточное условие неограниченности целевой функции.

Построим вектор приращения плана Δx как при доказательстве критерия оптимальности:

$$\Delta x_{j_0} = \theta, \ \Delta x_j = 0, j \in J_{\text{Н}} \backslash j_0, \tag{1.19}$$

$$\Delta x_{\text{Б}} = -\theta A_{\text{Б}}^{-1} A_{j_0} = -\theta z, z = A_{\text{Б}}^{-1} A_{j_0} = (z_i, i = \overline{1, m})'.$$

Согласно формуле приращения для плана $\bar{x} = x + \Delta x$ имеем

$$c'\bar{x} = c'x - \Delta_{j_0}\theta, \text{ где } \Delta_{j_0} < 0$$

Отсюда видно, что чем больше $\theta > 0$, тем лучше (больше) значение целевой функции, следовательно, наша цель выбрать θ максимально большим при условии, что вектор $\bar{x}$ остается планом задачи. Мы отмечали выше, что

$$A\bar{x} = b \quad \text{при} \quad \forall\theta;$$
$$\bar{x}_{\text{н}} = \Delta x_{\text{н}} \geq 0 \quad \text{при} \quad \forall\theta \geq 0.$$

Для $\bar{x}_{\text{Б}}$ имеем

$$\bar{x}_{\text{Б}} = x_{\text{Б}} - \theta A_{\text{Б}}^{-1} A_{j_0} \qquad \text{или}$$
$$\bar{x}_{j_i} = x_{j_i} - \theta z_i, \quad i = 1, \ldots, m.$$

Напомним, что $z := A_{\text{Б}}^{-1} A_{j_0} = (z_i, i = 1, ..., m)'$, $J_{\text{Б}} = \{j_1, j_2, \ldots, j_m\}$.

Подсчитаем шаги $\theta_i, \ i = 1, ..., m:$

$$\theta_i = \begin{cases} x_{j_i}/z_i, \text{ если} & z_i > 0, \\ \infty, \text{ если} & z_i \leq 0, \end{cases} \quad i = 1, \ldots, m.$$

Очевидно, что θ_i — максимальное значение θ, при котором базисная компонента $\bar{x}_{j_i}$ остается неотрицательной, т.е. выполняется неравенство $\bar{x}_{j_i} \geq 0$. Найдем

$$\theta_0 = \theta_s = \min \theta_i, \quad i = 1, \ldots, m.$$

По построению, θ_0 — максимальное значение θ, при котором вектор $\bar{x}$ является планом.

Построим новый план $\bar{x} = x + \Delta x$, где вектор Δx найден по правилам (1.19) при $\theta = \theta_0:$

$$\bar{x}_{j_i} = x_{j_i} - \theta_0 z_i, \quad i = 1, \ldots, m,$$

$$\bar{x}_j = 0, J_{\text{н}} \setminus j_0, \ \bar{x}_{j_0} = \theta_0.$$

Значение целевой функции на новом плане $\bar{x}$ равно

$$c'\bar{x} = c'x - \Delta_{j_0}\theta_0 \geq c'x.$$

Причем, если x — невырожденный план (т.е. $x_j > 0, j \in J_{\text{Б}}$), то $\theta^0 > 0$ и, следовательно, $c'\bar{x} > c'x$.

Покажем, что $\bar{x}$ — базисный план. Среди компонент $\bar{x}_j, j \in J_{\text{н}}$, только одна компонента $\bar{x}_{j_0} = \theta_0$ может оказаться отличной от нуля. С другой стороны, среди компонент $\bar{x}_j, j \in J_{\text{Б}}$, одна компонента $\bar{x}_{j_s}$ обязательно равна нулю, ибо по построению

$$\bar{x}_{j_s} = x_{j_s} - \theta_0 z_s = x_{j_s} - x_{j_s} z_s / z_s = 0.$$

Таким образом, $n - m$ компонент

$$\bar{x}_j, j \in \bar{J}_{\text{н}} = (J_{\text{н}} \backslash j_0) \cup j_s$$

плана $\bar{x}$ равны нулю. Остальные переменные $\bar{x}_j, j \in \bar{J}_{\text{Б}} = (J_{\text{Б}} \backslash j_s) \cup j_0$ неотрицательны и им соответствует матрица условий

$$\bar{A}_{\text{Б}} = (A_j, \quad j \in \bar{J}_{\text{Б}}).$$

Покажем, что $\det \bar{A}_{\text{Б}} \neq 0$. Справедливо

Утверждение. *Пусть есть квадратная невырожденная матрица $A_Б = (A_j, j \in J_Б), \det A_Б \neq 0$, и обратная к ней матрица $A_Б^{-1}$. Рассмотрим матрицу $\bar{A}_Б = (A_j, j \in \bar{J}_Б), \bar{J}_Б = (J_Б \backslash j_s) \cup j_0$, получающуюся из $A_Б$ заменой столбца A_{j_s} на столбец A_{j_0}. Тогда*

1) *матрица $\bar{A}_Б$ невырождена тогда и только тогда, когда $\alpha := e_s{}' A_Б^{-1} A_{j_0} \neq 0$, где $e_s = (e_s(i), i = 1, \ldots, m)$, $e_s(i) = 0, i \neq s, e_s(s) = 1$.*

2) *если $\alpha \neq 0$, то*

$$\bar{A}_Б^{-1} = A_Б^{-1} - A_Б^{-1}(A_{j_0} - A_{j_s}) e_s' A_Б^{-1} / \alpha.$$

Подсчитаем число α для нашей матрицы $\bar{A}_{\text{Б}}$:

$$\alpha := e_s{}' A_{\text{Б}}^{-1} A_{j_0} = z_s$$

По построению (см. формулы подсчета θ_i, θ_0) имеем $z_s > 0$, следовательно, согласно Утверждению, имеем $\det \bar{A}_{\text{Б}} \neq 0$. Следовательно, вектор $\bar{x}$ — новый базисный план с новой базисной матрицей $\bar{A}_{\text{Б}}$ и новым базисным индексом $\bar{J}_{\text{Б}} = (J_{\text{Б}} \backslash j_s) \cup j_0$.

Описанный переход $(x, J_{\text{Б}}) \to (\bar{x}, \bar{J}_{\text{Б}})$ от старого базисного плана x переходим к новому $\bar{x}$ называется симплексной итерацией. Последовательность таких итераций называется Симплекс-методом.

7. Алгоритм

Из предыдущего видно, что в вычислениях основную роль играет матрица $A_{\text{Б}}^{-1}$, обратная к $A_{\text{Б}}$ (по ней можно восстановить и соответствующий базисный план).

Опишем алгоритм в терминах этой матрицы.

Рассмотрим задачу (1.10), считая, что выполняется условие (1.11).

Для реализации симплекс-метода кроме исходных данных задачи (1.10) (векторов c, b и матрицы A) на каждой итерации необходимо знать следующие параметры:

- текущий базисный план $x = (x_j, j \in J)$ (достаточно знать только его базисные компоненты $x_j, j \in J_{\text{Б}}$);
- соответствующее плану x множество базисных индексов
$$J_{\text{Б}} = \{j_1, j_2, \ldots, j_m\};$$
- $m \times m$ — матрицу $B = A_{\text{Б}}^{-1}$, обратную к базисной матрице $A_{\text{Б}} = (A_j, j \in J_{\text{Б}})$.

Опишем общую итерацию симплекс-метода по шагам.

Шаг 1. Вычислим вектор потенциалов
$$u' = c_{\text{Б}}{}'B, \text{ где } c'_{\text{Б}} = (c_j, j \in J_{\text{Б}}),$$
и оценки
$$\Delta_j = u'A_j - c_j, j \in J_{\text{н}} = J \backslash J_{\text{Б}}.$$

Шаг 2. Если $\Delta_j \geq 0, j \in J_{\text{н}}$, то STOP: вектор $x^0 = x$ является оптимальным планом задачи (1.10). В противном случае переходим к шагу 3.

Шаг 3. Выберем индекс $j_0 \in J_{\text{н}}$, для которого $\Delta_{j_0} < 0$. Построим вектор
$$z = (z_1, z_2, \ldots, z_m)' := BA_{j_0}.$$
Если $z_i \leq 0, i = 1, \ldots, m$, то STOP: задача (1.10) не имеет решения в силу неограниченности сверху целевой функции на множестве планов. В противном случае перейдем к шагу 4.

Шаг 4. Найдем минимум

$$\theta_0 = \min_{\substack{i=\overline{1,\,m},\\ z_i>0}} x_{j_i}/z_i$$

и выберем индекс $s, 1 \le s \le m$, для которого $z_s > 0$ и $x_{j_s}/z_s = \theta_0$.

Шаг 5. Построим новый базисный план $\bar{x} = (\bar{x}_j, j \in J)$ и соответствующий ему базис $\bar{J}_Б$ по правилам

$$\bar{x}_j = 0, j \in J_н \backslash j_0;\ \bar{x}_{j_0} = \theta_0;$$

$$\bar{x}_{j_i} = x_{j_i} - \theta_0 z_i;\ i = 1, ..., m,$$

$$\bar{J}_Б = (J_Б \backslash j_s) \cup j_0 = \{j_1, \ldots, j_{s-1}, j_0, j_{s+1}, \ldots, j_m\}.$$

Шаг 6. Вычислим матрицу $\bar{B}$, обратную к новой базисной матрице $\bar{A}_Б = (A_j, j \in \bar{J}_Б)$, по формуле $\bar{B} = MB$, где матрица $M \in \mathbb{R}^{m\times m}$ отличается от единичной $m \times m-$матрицы только s-м столбцом и имеет вид

$$M = (e_1, ..., e_{s-1}, d_s, e_{s+1}, ..., e_m),$$

где

$$d_s = -(z_1, z_2, \ldots, z_{s-1}, -1, z_{s+1}, \ldots, z_m)'/z_s,$$

e_i — единичный m-вектор с единицей на i-м месте.

Переходим к следующей итерации, исходя из новых плана $\bar{x}$, базиса $\bar{J}_Б$ и матрицы $\bar{B}$, обратной к новой базисной матрице $\bar{A}_Б$.

Замечания.

1. На шаге 3 выбирается индекс $j_0 \in J_н$, для которого $\Delta_{j_0} < 0$. В общем случае существуют несколько индексов, удовлетворяющих этому условию. При численной реализации симплекс-метода для выбора j_0 можно использовать дополнительные "уточняющие"правила, например, следующие:

   а) $\Delta_{j_0} = \min \Delta_j,\ j \in J_н,\ \Delta_j < 0,$ либо

   б) $j_0 = \min\{j \in J_н : \Delta_j < 0\}$.

2. На шаге 4 выбирается индекс $j_s \in J_Б$ или $1 \le s \le m$, для которого $z_s > 0, x_{j_s}/z_s = \theta_0$. В общем случае этот выбор может оказаться неоднозначным. Можно использовать дополнительные уточняющие правила, например, следующие:

а) $s = \min\{i \in \{1, \ldots, m\} : z_i > 0, x_{j_i}/z_i = \theta_0\}$ либо

б) $j_s = \min\{j_i \in J_Б : z_i > 0, x_{j_i}/z_i = \theta_0\}$.

3. В современных версиях симплекс-методах для нахождения вектора потенциалов u (см. шаг 1) и вектора z (см. шаг 3) решаются системы $u'A_Б = c_Б{}'$ и $A_Б z = A_{j_0}$, соответственно. Для эффективного решения последних используется LU-разложение базисной матрицы $A_Б$.

4. Легко подсчитать, что приращение целевой функции при переходе от начального базисного плана x к новому базисному плану $\bar{x}$ равно: $c'\bar{x} - c'x = \theta_0|\Delta_{j_0}|$. По построению, $\Delta_{j_0} < 0, \theta_0 \geq 0$. Следовательно, при $\theta_0 > 0$ происходит "строгое улучшение"плана: $c'\bar{x} > c'x$. Из описания шага 4 видно, что в случае невырожденности начального базисного плана $\{x, J_Б\}$ всегда верно неравенство $\theta_0 > 0$.

В случае, когда базисный план x (с базисом $J_Б$) является вырожденным, может реализоваться ситуация: $\theta_0 = 0$. При этом мы получаем $\bar{x} = x, \bar{J}_Б \neq J_Б$. В этом случае не происходит улучшения целевой функции, но итерация может оказаться полезной, так как изменяется базис и новый базис может быть ближе к оптимальному базису, чем старый.

§2. Конечность симплекс-метода

Алгоритм называется конечным, если его реализация позволяет за конечное число операций (конечное время) построить оптимальный план.

Каждая итерация симплекс метода содержит конечное число операций. Следовательно, для доказательства конечности симплекс-метода достаточно показать конечность числа его итераций.

1. Невырожденные задачи

Каноническую задачу линейного программирования назовём невырожденной, если невырождены все её базисные планы.

Теорема 3 *Пусть задача линейного программирования имеет решение и является невыпожденной. Тогда симплекс-метод находит оптимальный план за конечное число итераций, начиная процесс решения с любого начального базисного плана.*

Итерация $\{x, J_{\text{Б}}\} \to \{\bar{x}, \bar{J}_{\text{Б}}\}$ симплекс-метода называется невырожденной, если $c'\bar{x} > c'x$.

Теорема 4 *Симплекс-метод решает задачу линейного программирования за конечное число итераций, если в процессе его реализации не встречается вырожденных итераций.*

Теоремы 3 и 4 доказываются одинаково. Докажем их.

Из определения базисного плана следует, что базисный план $\{x, J_{\text{Б}}\}$ однозначно задаётся базисным множеством $J_{\text{Б}}$. Действительно, зная $J_{\text{Б}}$, соответствующий план $x = (x_{\text{Б}}, x_{\text{Н}})$ строится по правилу: $x_{\text{Н}} = 0, x_{\text{Б}} = A_{\text{Б}}^{-1}b$.

Как следует из алгоритма симплекс-метода, его итерации ведутся на базисных планах

$$\{x^{(1)}, J_{\text{Б}}^{(1)}\}, \{x^{(2)}, J_{\text{Б}}^{(2)}\}, \dots, \{x^{(k)}, J_{\text{Б}}^{(k)}\}, \dots \tag{1.20}$$

Кроме того,по условиям теорем имеют место неравенства

$$c'x^{(1)} < c'x^{(2)} < \dots < c'x^{(k)} < \dots \tag{1.21}$$

Из приведённых соотношений следует, что ни один базис в последовательности (1.20) не может повториться дважды. Действительно, допустим, что $J_{\text{Б}}^{(i)} = J_{\text{Б}}^{(j)}$, где $i <$

j. Тогда $x^{(i)} = x^{(j)}$ и значит, $c'x^{(i)} = c'x^{(j)}$, где $i < j$. Однако, это противоречит (1.21). Значит в последовательности (1.20) все базисы $J_Б^{(i)}$ разные. По построению, $J_Б^{(i)} \subset \{1, 2, \dots, n\}$, $|J_Б^{(i)}| = m \leq n$, следовательно, различных базисов может быть только конечное число. Значит возможно только конечное число итераций. Теорема доказана.

При наличии вырожденных итераций при реализации симплекс-метода может возникнуть такая ситуация, когда начиная с некоторого вырожденного базисного плана $\{x,\ J_Б\}$ мы осуществляем последовательность вырожденных итераций

$$\{x,\ J_Б\} \to \{x,\ J_Б^{(1)}\} \to \{x,\ J_Б^{(2)}\} \to ...\{x,\ J_Б^{(s)}\}, \tag{1.22}$$

в которых меняются только базисы $J_Б^{(i)}$ базисного плана x, причем при некотором конечном $s > 1$ имеет место равенство $J_Б^{(s)} = J_Б$.

Очевидно, что если мы продолжим операции симплекс-метода исходя из $\{x,\ J_Б^{(s)}\}$, то опять получим ту же последовательность итерации (1.22) и т. д. Такое явление получило название зацикливания. Первоначально считалось, что зацикливание — крайне редкое явление. Однако позже было замечено, что вероятность возникновения зацикливания увеличивается с ростом размеров задачи. Кроме того, зацикливание является типичным явлением для задач ЛП, возникающих при аппроксимации задач целочисленного программирования. В связи с этим возникла необходимость в разработке специальных приемов борьбы с зацикливанием. К настоящему времени известно много таких приемов. Опишем два из них.

2. Стратегия возмущения (метод Чарнса).

Предположим, что в задаче (1.10) мы возмутили вектор правых частей b, заменив его на $b(\varepsilon)$:

$$b(\varepsilon) = b + C \begin{pmatrix} \varepsilon \\ \varepsilon^2 \\ \dots \\ \varepsilon^m \end{pmatrix},$$

где $C = (C_1, C_2, ..., C_m)$ - некоторая невырожденная $m \times m$ - матрица со столбцами C_i; $\varepsilon > 0$ — достаточно малое число. Возмущения в векторе $b(\varepsilon)$ вызовут возмущения в базисных компонентах $x_Б = (x_j, j \in J_Б)$ каждого базисного плана следующим образом:

$$x_Б(\varepsilon) = x_Б \ + \ A_Б^{-1} C \begin{pmatrix} \varepsilon \\ \varepsilon^2 \\ \ldots \\ \varepsilon^m \end{pmatrix} \ = \ x_Б + \sum_{i\,=\,1}^{m} \big(A_Б^{-1} C_i\big) \varepsilon^i \tag{1.23}$$

Матрица C должна быть такой, что на первой итерации вектор $x_Б(\varepsilon)$ должен быть строго положительным при достаточно малых $\varepsilon > 0$. Для этого достаточно положить $C = A_Б^{(0)}$, где $A_Б^{(0)}$ – базисная матрица, соответствующая начальному базису $J_Б^{(0)}$. Тогда для начального базисного плана имеем

$$x_Б(\varepsilon) \ = \ x_Б \ + \begin{pmatrix} \varepsilon \\ \varepsilon^2 \\ \ldots \\ \varepsilon^m \end{pmatrix} \ > \ 0\,.$$

Предположим, что на текущей итерации имеется базисный план $\{x_Б\,(\varepsilon)\,, J_Б\}$ возмущенной задачи, для которого справедливы неравенства $x_Б\,(\varepsilon) > 0$ при достаточно малых $\varepsilon > 0$. Согласно данным подразд. 1.4, новый базисный план $\{\bar{x}_Б\,(\varepsilon)\,, \bar{J}_Б\}$, построенный в результате одной итерации симплекс-метода из $\{x_Б\,(\varepsilon)\,, J_Б\}$, будет также невырожденным, если не существует таких двух базисных индексов $j_s,\ j_k \in J_Б$, что $z_s > 0, z_k > 0$ и

$$x_{j_s}(\varepsilon)/z_s = x_{j_k}(\varepsilon)/z_k \tag{1.24}$$

для всех достаточно малых $\varepsilon > 0$. Последние соотношения могут иметь место только в том случае, если k-я и s-я строки матрицы $A_Б^{-1}C$ линейно зависимы. Однако это противоречит тому, что $\det\, A_Б \neq 0,\ \det\, C \neq 0$. Значит, соотношения (1.24) не могут иметь места и, следовательно, новый базисный план $\{\bar{x}_Б\,(\varepsilon)\,, \bar{J}_Б\}$ возмущенной задачи будет невырожденным при достаточно малых $\varepsilon > 0$. Таким образом, мы показали, что при достаточно малых $\varepsilon > 0$ все итерации симплекс-метода для задачи с возмущенным вектором $b(\varepsilon)$ будут невырожденными и, следовательно, для возмущенной задачи симплекс- метод будет конечным. Кроме того справедлива следующая теорема.

Теорема 5 *Существует такое $\varepsilon_0 > 0$, что для любого $\varepsilon \in [0, \varepsilon_0]$ каждому базисному плану возмущенной задачи, порожденному базисом $J_Б$, соответствует базисный план (невозмущенной) исходной задачи, порожденный тем же базисом $J_Б$, а из оптимальности базиса $J_Б^0$ в возмущенной задаче следует его оптимальность в исходной задаче.*

Покажем, как описанные выше результаты можно использовать для предотвращения зацикливания в исходной задаче (1.10). Недостатком описанных выше рассуждений является то, что они верны только при достаточно малых $\varepsilon > 0$ и заранее невозможно указать конкретное значение $\varepsilon_0 > 0$. Однако существуют правила (лексикографическая стратегия), которые позволяют "мысленно"осуществить процедуру решения возмущенной задачи без явного выбора значения параметра ε. Пусть в начале текущей итерации для исходной задачи есть базисный план $x = (x_Б, x_Н = 0)$, базис $J_Б = \{j_1, j_2, ..., j_m\}$ и соответствующая ему базисная матрица $A_Б$. Очевидно, что при достаточно малых $\varepsilon > 0$ базис $J_Б$ порождает базисный план $x(\varepsilon) = (x_Б(\varepsilon), x_Н(\varepsilon) = 0)$, $x_Б(\varepsilon)$ имеет вид (1.23), возмущенной задачи. Предположим, что мы осуществляем итерацию симплекс-метода для исходной и возмущенной задач, исходя из базисных планов $\{x, J_Б(\varepsilon)\}$ и $\{x\left(\varepsilon\right), J_Б(\varepsilon)\}$. Ясно, что для обеих задач шаги 1-3 итерации будут полностью совпадать. Значит, совпадут и индексы $j_0 \in J_Н(\varepsilon)$, подлежащие вводу в базис. Перейдем к шагу 4, на котором определяется индекс j_s , подлежащий выводу из базиса $J_Б(\varepsilon)$. Для возмущенной задачи индекс j_s однозначно определяется из соотношений

$$\Theta_0\left(\varepsilon\right) \;=\; \min_{\substack{i\,=\,\overline{1,\,m}\\ z_i\,>\,0}} \; x_{j_i}\left(\varepsilon\right)/z_i \;=\; x_{j_s}\left(\varepsilon\right)/z_s, \tag{1.25}$$

где $\varepsilon > 0$ — достаточно малое число. Легко проверить, что найти единственный индекс $j_s \in J_Б(\varepsilon)$, определяемый условием (1.25), можно по следующему правилу.

Положим

$$\Delta I^0 \;=\; \{s \;\in\; \{1,\; 2,\; ...,\; m\}:\; z_s > 0, x_{j_s}/z_s \;=\; \min_{\substack{i\,=\,\overline{1,\,m},\\ z_i\,>\,0}} \; x_{j_i}/z_i\} \tag{1.26}$$

и построим множества $\Delta I^l,\; l \;=\; \overline{1,\; m}$, по рекуррентным правилам

$$\Delta I^l = \{s \in \Delta I^{l-1} : \tilde{c}_{sl}/z_s = \min_{i \in \Delta I^{l-1}} \tilde{c}_{il}/z_i\},\ l = 1,\ 2,\ ...,\ m, \tag{1.27}$$

где $(\tilde{c}_{ij}, i = \overline{1,m}\ , j = \overline{1,m}\ ) = A_{\text{Б}}^{-1}(\varepsilon)C$. По построению,

$$\Delta I^l \subset \Delta I^{l-1},\ l = \overline{1,\ m};\ |\Delta I^m| = 1. \tag{1.28}$$

Обозначим через l_0 наименьший индекс $0 \leq l \leq m$, при котором $\left|\Delta I^{l_0}\right| = 1:$

$$l_0 = \min\ l,\ l \in \{i \in \{0,\ 1, 2,\ ...,\ m\} : \left|\Delta I^i\right| = 1\}. \tag{1.29}$$

Пусть $\Delta I^{l_0} = \{s\}$. Тогда индекс $j_s \in J_{\text{Б}}$ совпадает с индексом, определяемым соотношениями (1.25). Ясно, что для исходной задачи любой индекс вида j_i, где $i \in \Delta I^0$, можно взять в качестве индекса, выводимого из базиса. Согласно (1.28) $s \in \Delta I^0$, следовательно, индекс j_s можно выводить из базиса в исходной задаче. Таким образом, в качестве нового базиса для исходной задачи можно взять базис

$$\bar{J}_{\text{Б}} = (J_{\text{Б}} \backslash j_s) \cup j_0, \tag{1.30}$$

где индекс j_s однозначно определяется по правилам (1.26)-(1.29). (Для возмущенной задачи новый базис (1.30) является единственно возможным). Из сказанного выше следует, что симплекс-метод будет конечным и для исходной задачи (1.10), если на шаге 4 индекс j_s, подлежащий выводу из базиса, определять по правилам (1.26), (1.27), (1.29).

3. Правило Бленда

В 1977 г. Р. Блэнд предложил очень простое правило предотвращения зацикливания в симплекс-методе. Это правило сводится к следующему.

При реализации итерации симплекс-метода на шаге 3 индекс $j_0 \in J_н$, подлежащий вводу в новый базис, однозначно определяется по правилу

$$j_0 = \min\ j,\ j \in J_н,\ \Delta_j < 0, \tag{1.31}$$

а на шаге 4 индекс $j_s \in J_Б$, подлежащий выводу из базиса, однозначно определяется соотношениями

$$j_s = \min_{i \in \Delta I} j_i, \quad \text{где } \Delta I := \{i \in \{1, ..., m\} : z_i > 0, x_{j_i}/z_i = \theta_0\}. \tag{1.32}$$

Теорема 6 *При использовании дополнительных правил (1.31), (1.32) симплекс-метод для любой задачи линейного программирования (с заданным начальным базисным планом) за конечное число итераций строит оптимальный базисный план либо обнаруживает неограниченность сверху целевой функции на множестве планов.*

Доказательство.

Предположим, что при использовании модификации Бленда в последовательности (1.20) возник цикл, т.е. после конечного числа итераций с нулевым шагом повторился базисный план.

Пусть $I_{\text{Б}} \subset J$ — множество индексов, которые в течении этого цикла являются базисными постоянно; $I_{\text{Н}} \subset J$ — множество индексов, которые в течении цикла являются не базисными постоянно; $I_0 \subset J$ — множество индексов, которые в течении цикла являются то базисными, то небазисными; t — максимальный индекс из множества I_0.

Рассмотрим $k-$ю итерацию. Для этой итерации обозначим через Δ^k вектор оценок, $j_0^k \in J_{\text{Н}}^{(k)}$ — небазисный индекс, который выбирается по правилу (1.31), l^k $(\Delta x^k = \theta_0^k l^k)$ — направление изменения плана, $j_s^k \in J_{\text{Б}}^{(k)}$ — базисный индекс, который выбирается по правилу (1.32).

Обозначим через p номер итерации, когда имеет место равенство $j_0^p = t$, т.е. в конце p-ой итерации индекс t становится базисным индексом. Тогда по построению имеем

$$\Delta_{j_0}^p = \Delta_t^p < 0. \tag{1.33}$$

Обозначим через q номер итерации, когда имеет место равенство $j_s^q = t$, т.е. в конце этой итерации индекс t становится небазисным. Для этой итерации по построению имеем

$$l_{j_0^q}^q = 1, \Delta_{j_0^q}^q < 0, l_{j_s^q}^q = l_t^q < 0. \tag{1.34}$$

Из определения множеств $I_{\text{Б}}$, $I_{\text{Н}}$ следует, что

$$\Delta^p(I_{\text{Б}}) = 0, \;\; l^q(I_{\text{Н}}) = 0. \tag{1.35}$$

Поскольку t — максимальный индекс из I_0 и $j_0^p = t$ и при выборе этого индекса использовалось правило (1.31), то

$$\Delta^p(I_0 \backslash t) \geq 0. \tag{1.36}$$

В течении рассматриваемого цикла (1.22) выполняются равенства

$$x_j = 0, j \in I_0,$$

так как все итерации цикла идут с нулевым шагом и в течении цикла все индексы из I_0 были (на некоторых итерациях) небазисными.

Из условия $j_s^q = t$, правила (1.32) и предположения о том, что t — максимальный индекс из I_0 следует, что

$$l^q(I_0 \backslash t) \geq 0, \;\; l_t^q < 0. \tag{1.37}$$

По построению для лубого k имеют место равенства

$$\Delta^k = A'u^k - c, \;\; Al^k = 0.$$

С учетом последних равенств получаем

$$\Delta^{p\prime} l^q = \Delta^{q\prime} l^q = \Delta^q_{j_0^q} l^q_{j_0^q} < 0. \tag{1.38}$$

С другой стороны, с учетом (1.33) – (1.37) имеем:

$$\Delta^{p\prime} l^q = \Delta^{p\prime}(I_{\text{Б}}) l^q(I_{\text{Б}}) + \Delta_t^p l_t^q + \Delta^{p\prime}(I_{\text{Н}}) l^q(I_{\text{Н}}) + \Delta^{p\prime}(I_0 \backslash t) l^q(I_0 \backslash t) =$$

$$= \Delta_t^p l_t^q + \Delta^{p\prime}(I_0 \backslash t) l^q(I_0 \backslash t) > 0.$$

Однако, это противоречит (1.38). Следовательно, наше предположение о том, что существует цикл, также противоречиво. Значит при использовании правила Бленда цикл возникнуть не может. Учитывая это и то, что различных базисных множеств $J_{\text{Б}}$ конечное число, заключаем, что при использовании правила Бленда симплекс-метод решает задачу линейного программирования за конечное число итераций.

§3. Первая фаза симплекс-метода

Рассмотрим исходную задачу линейного программирования в канонической форме

$$c'x \to \max, Ax = b, x \geq 0. \tag{1.39}$$

Все предыдущие утверждения и операции симплекс-метода были справедливы в предположении, что для задачи (1.39) выполнены условия

$$A) \qquad \text{rank } A = m \leq n,$$

$$B) \quad \text{ограничения задачи совместны,}$$

$$C) \quad \text{известен начальный базисный план.}$$

Понятно, что для произвольной задачи (1.39) эти условия могут нарушаться. Поэтому, прежде чем применять описанный симплекс-метод необходимо исходную задачу (1.39) привести к такому виду, для которого выполняются условия A)-C).

Проблема нахождения такой начальной информации может оказаться нетривиальной. Действительно, в общем случае по трудоемкости это проблема эквивалентна построению решения некоторой задачи линейного программирования. Для преодоления указанной трудности используется двухфазный симплекс-метод, общая схема которого состоит в следующем.

На первой фазе формируется вспомогательная задача ЛП, которая отличается от исходной задачи (1.39). Вспомогательная задача строится таким образом, что для нее выполняются условия A)-C). Анализ решения вспомогательной задачи позволяет:

1) определить, совместны ли ограничения исходной задачи (1.39);

2) проверить, выполняется ли условие A) для исходной задачи (1.39) и в случае его нарушения обнаружить линейно зависимые основные ограничения и удалить их из условий задачи;

3) в случае совместности ограничений задачи (1.39) построить для нее начальный базисный план x и базис $J_Б$.

Если анализ решения вспомогательной задачи закончился построением начального базисного плана для исходной задачи, то переходим ко второй фазе алгоритма. Вторая фаза состоит в решении исходной задачи описанным выше симплекс-методом, при этом в качестве начальной используется информация, полученная на первой фазе. Опишем подробнее первую фазу алгоритма.

1. Формулировка задача первой фазы

Без ограничения общности будем считать, что в задаче (1.39) вектор b удовлетворяет неравенству $b \geq 0$. (Если это условие нарушается, то для того, чтобы добиться выполнения этого условия, достаточно умножить правую и левую части равенств с $b_i < 0$ на -1.)

Тогда задачу первой фазы можно записать в виде

$$-e'x_u \to \max_{x,x_u}, \; Ax + Ex_u = b, \; x \geq 0, \; x_u \geq 0, \tag{1.40}$$

где $x = (x_j, \; j \in J)$ — исходные переменные, $J = \{1, 2, \ldots, n\}$, $x_u = x(J_u)$, $J_u = \{n+1, n+2, \ldots, n+m\}$ — m-вектор искусственных переменных, $E \in R^{m \times m}$ — единичная матрица, $e = (1, 1, 1, \ldots, 1) \in R^m$.

Легко проверить, что вектор (x, x_u) с компонентами

$$x = x(J) = 0, \quad x_u = x(J_u) = b \geq 0 \tag{1.41}$$

является базисным планом задачи (1.40) с базисом

$$J_{\text{Б}} = J_u \subset J \cup J_u.$$

и базисной матрицей $A_{\text{Б}} = E$.

Решим задачу (1.40) симплекс-методом, описанным выше, используя в качестве начального базисный план (1.41). Поскольку целевая функция задачи (1.40) ограничена сверху на множестве ее планов ($-e'x_u \leq 0$), то через конечное число итераций будет построен оптимальный базисный план (x^*, x_u^*) задачи (1.40) и соответствующий ему базис $J_{\text{Б}}^* \subset J \cup J_u, \quad |J_{\text{Б}}^*| = m$. Справедлива следующая лемма.

Лемма 1 *Ограничения исходной задачи (1.39) совместны тогда и только тогда, когда компонента x_u^* оптимального базисного плана задачи (1.40) равна нулю.*

Доказательство. **Необходимость.** Если x^* — план задачи (1.39), то $(x^*, x_u^* = 0)$ — решение задачи (1.40), так как этот вектор является планом задачи (1.40) и на нём значение целевой функции достигает максимального значения: оно равно нулю, а для любых других планов имеем $-e'x_u \leq 0$.

Достаточность. Очевидно, если $(x^*, x_u^* = 0)$ — решение задачи (1.40), то x^* — план задачи (1.39). Лемма доказана.

Решение задачи (1.40) симплекс-методом называется **первой фазой решения** задачи (1.39). После решения задачи (1.40) будут построены оптимальный базисный план (x^*, x_u^*) и базисная матрица $A_Б^*$ задачи (1.40).

2. Анализ решения задачи первой фазы

Пусть задача (1.40) решена и в результате ее решения построены оптимальный базисный план (x^*, x_u^*) и базисная матрица $A_Б^*$ задачи (1.40).

Поведем анализ решения задачи первой фазы. Возможны следующие ситуации

а) $x_u^* \neq 0$;

б) $x_u^* = 0,\ J_Б^* \cap J_u = \varnothing$;

в) $x_u^* = 0,\ J_Б^* \cap J_u \neq \varnothing$.

В случае а) из леммы 1 следует, что исходная задача (1.39) не имеет решения, так как множество ее допустимых планов пусто.

В случае б) вектор x^* — базисный план с базисом $J_Б^* \subset J$ и базисной матрицей $A_Б^* = (A_j, j \in J_Б^*)$. Переходим ко второй фазе симплекс-метода, т.е к решению задачи (1.39) симплекс-методом, исходя из x^*, $J_Б^*$.

Исследуем случай в). Ясно, что вектор x^* является планом задачи (1.39), но множество $J_Б^*$ не является его базисом для задачи (1.39).

Попытаемся построить базис для плана x^* в задаче (1.39).

Пусть $J_Б^* = \{j_1, j_2, \ldots, j_k, \ldots, j_m\}$ и $j_k \in J_u$. Обозначим

$$A_j := e_{j-n},\ j \in J_u; \quad A_Б^* = (A_j,\ j \in J_Б^*);\ \ \alpha_j = e_k'(A_Б^*)^{-1}A_j, \quad j \in J \backslash J_Б^*.$$

Здесь e_i — единичный m-вектор с единицей на i-ом месте.

Возможны подслучаи:

1) $\exists j_0 \in J \backslash J_Б^*,\ \alpha_{j_0} \neq 0$;

2) $\alpha_j = 0,\ j \in J \backslash J_Б^*$.

Если реализовался подслучай 1), то оптимальному плану (x^*, x_u^*) задачи (1.40) припишем новый базис

$$\bar{J}_Б^* = (J_Б^* \backslash j_k) \cup j_0. \tag{1.42}$$

Очевидно, что по построению, $\left|\bar{J}^*_{\text{Б}} \cap J_u\right| = |J^*_{\text{Б}} \cap J_u| - 1$. Снова проверяем, какая из ситуаций, б) или в), имеет место для оптимального плана (x^*, x^*_u) задачи (1.40) и нового базиса (1.42).

Подслучай 2) означает, что в исходной задаче (1.39) основное ограничение с индексом $i_0 := j_k - n \in I = \{1, ..., m\}$ является линейно зависимым с остальными основными ограничениями этой задачи. Множество планов задачи (1.39) не изменится, если мы удалим это ограничение, т.е. заменим множество индексов I на $\tilde{I} = I \backslash i_0$.

Для преобразованной задачи (1.39) вспомогательная задача первой фазы получится из задачи (1.40) после удаления ограничения с индексом $i_0 \in I$ и искусственной переменной $x^*_{j_k} = 0, \quad j_k \in J_u$, т.е. заменой множеств I и J_u на $\tilde{I} = I \backslash i_0$ и $\tilde{J}_u = J_u \backslash j_k$. Очевидно, что для новой задачи первой фазы вектор $(x^*_j,\ j \in J \cup \tilde{J}_u)$ является оптимальным базисным планом с базисом $\tilde{J}^*_{\text{Б}} = J^*_{\text{Б}} \backslash j_k$, для которого

$$\left|\tilde{J}^*_{\text{Б}}\right| = \left|\tilde{I}\right| = m - 1,\ \left|\tilde{J}^*_{\text{Б}} \cap \tilde{J}_u\right| = |J^*_{\text{Б}} \cap J_u| - 1.$$

Пусть $B := (A^*_{\text{Б}})^{-1}$ и $\bar{B} := (\bar{A}^*_{\text{Б}})^{-1}$, где $\bar{A}^*_{\text{Б}}$ — новая базисная матрица, соответствующая новым множествам $\tilde{I}, \tilde{J}^*_{\text{Б}}$. Она получается из $A^*_{\text{Б}}$ удалением строки с номером $(j_k - n)$ и k-го столбца. Тогда матрица $\bar{B}$ получается из известной матрицы B удалением из B k-ой строки и $(j_k - n)$-го столбца.

Для плана $(x^*_j,\ j \in J \cup \tilde{J}_u)$ и новых множеств $\tilde{I},\ \tilde{J}^*_{\text{Б}}, \tilde{J}_u$ вновь проверяем, какая из ситуаций, 2 или 3, имеет место. Ясно, что не более чем через m шагов из основных ограничений задачи (1.39) будут исключены все линейно зависимые ограничения и она будет сведена к задаче

$$\begin{aligned} &c'x \to \max, \\ &\bar{A}x = \bar{b}, \quad x \geq 0, \end{aligned} \tag{1.43}$$

где $\bar{A} = (a_{ij},\ i \in \bar{I}, j \in J), \quad \bar{b} = (b_i,\ i \in \bar{I}), \quad \bar{I} \subset I, \quad rank\overline{A} = \bar{m} \leq |J|\,, \quad \bar{m} = \left|\bar{I}\right|.$

При этом для задачи (1.43) будет построен базисный план x^* с некоторым базисом $J_{\text{Б}} \subset J, \quad |J_{\text{Б}}| = \bar{m}$.

Дальнейшее решение задачи (1.43) осуществляем симплекс-методом, описанным в параграфе 1, используя в качестве начальных построений базисный план x^* с базисом $J_{\text{Б}}$.

Суммируя приведенные выше результаты, заключаем, что две фазы симплекс-метода позволяют за конечное число итераций:

1. обнаружить противоречивость ограничений;
2. исключить линейно зависимые равенства в ограничениях;
3. установить неограниченность целевой функции;
4. построить оптимальный базисный план;

Таким образом, доказана следующая теорема.

Теорема 7 *Если задача (1.39) имеет оптимальный план, то она имеет и оптимальный базисный план.*

Множество, образованное пересечением конечной совокупности полупространств и гиперплоскостей, называется многогранником.

Следовательно,множество планов задачи линейного программирования — многогранник. Базисному плану в многогранном множестве соответствует крайняя (угловая) точка (или вершина), т.е. такая точка, которую нельзя включить в середину ненулевого прямолинейного отрезка, целиком принадлежащего многогранному множеству.

Симплексная итерация составляет переход от одной крайней точки к такой соседней крайней точке вдоль ребра, соединяющего эти точки, в котором значение целевой функции не меньше, чем в старой точке.

Симплекс метод — это направленное движение вдоль рёбер множества планов, при котором исключаются те ребра, вдоль которых целевая функция не возрастает.

Их теоремы 7 следует справедливость утверждения

Теорема 8 *Среди оптимальных планов задачи линейного программирования всегда есть оптимальный базисный план.*

Следовательно, поиск решения задачи линейного программирования среди базисных планов оправдан.

Замечание 2 *Задача линейного программирования может иметь несколько оптимальных планов. Если задача линейного программирования имеет более двух оптимальных планов, то среди ее оптимальных планов есть базисные и не базисные.*

§4. Теория двойственности

Теорией двойственности линейного программирования называется раздел, в котором задачи линейного программирования исследуются с помощью вспомогательных, тесно связанных с исходными, **двойственных задач**.

1. Двойственная задача

Рассмотрим каноническую задачу

$$c'x \to \max, Ax = b, x \geq 0, \tag{1.44}$$

которую будем называть **прямой** канонической задачей линейного программирования. Здесь

$$c \in \mathbb{R}^n,\ b \in \mathbb{R}^m, A \in \mathbb{R}^{m\times n}$$

— заданные векторы и матрица. Из результатов параграфа 3 следует, что без ограничения общности можно считать, что

$$\operatorname{rank} A = m \leq n.$$

Согласно симплекс-методу каждому оптимальному базисному плану x^0 с базисом $J_{\text{Б}} \subset J = \{1, ..., n\}$ и базисной матрицей $A_{\text{Б}} = (A_j, j \in J_{\text{Б}})$ соответствует m-вектор потенциал $u \in \mathbb{R}^m$, удовлетворяющий соотношениям:

$$u'A_{\text{Б}} = c'_{\text{Б}},\ u'A_{\text{Н}} \geq c'_{\text{Н}}. \tag{1.45}$$

Напомним, что по построению выполняются соотношения

$$x^0 = (x^0_{\text{Б}}, x^0_{\text{Н}}), \quad \text{где } x^0_{\text{Б}} = A^{-1}_{\text{Б}}b,\ x^0_{\text{Н}} = 0.$$

Подсчитаем значение линейной формы $b'u$ на векторе u (1.45):

$$b'u = u'b = c'_{\text{Б}}A^{-1}_{\text{Б}}b = c'_{\text{Б}}x^0_{\text{Б}} = c'x^0. \tag{1.46}$$

Пусть y — произвольный m-вектор, удовлетворяющий неравенству

$$A'y \geq c, \tag{1.47}$$

которое является обобщением соотношений (1.45). Для y имеем

$$b'y = y'b = y'Ax^0 \geq c'x^0. \tag{1.48}$$

Здесь мы учли неравенства (1.47) и $x^0 \geq 0$.

Из (1.45) – (1.48) следует, что вектор u является решением задачи

$$b'y \to \min, \quad A'y \geq c. \tag{1.49}$$

Задача (1.49) является задачей линейного программирования, она называется задачей, двойственной к задаче (1.44).

Если задачу (1.49) свести к канонической форме, а затем к ней по описанным выше правилам записать двойственную, то получим задачу (1.44).

Формально правила перехода от (1.44) к (1.49) можно представить следующим образом

$$x \to y, \quad \max \to \min, \quad A \to A', \quad = \to \geq 0, \quad x \geq 0 \to y \in \mathbb{R}^m. \tag{1.50}$$

Рассмотрим задачу линейного программирования, заданную в произвольной форме записи:

$$\sum_{j \in J} c_j x_j \to \max,$$

$$\sum_{j \in J} a_{ij} x_j \leq b_i, \; i \in K, \tag{1.51}$$

$$\sum_{j \in J} a_{ij} x_j = b_i, \; i \in I;$$

$$x_j \geq 0, \; j \in J_* \subset J,$$

где $J_*, \; J, \; K, \; I$ - некоторые заданные конечные множества индексов.

Задачу линейного программирования

$$\sum_{i\in K\cup I} b_i y_i \to \min,$$

$$\sum_{i\in K\cup I} a_{ij} y_i \geq c_j,\ j\in J_* \tag{1.52}$$

$$\sum_{i\in K\cup I} a_{ij} y_i = c_j,\ j\in J\backslash J_*,$$

$$y_i \geq 0,\ i\in K,$$

назовем двойственной по отношению к задаче (1.51). При этом задача (1.51) считается прямой. Из приведенного определения следует, что для задачи линейного программирования в нормальной форме

$$c'x \to \max,\ Ax \leq b,\ x\geq 0, \tag{1.53}$$

двойственной является задача

$$b'y \to \min,\ A'y \geq c,\ y\geq 0, \tag{1.54}$$

а для задачи в канонической форме

$$c'x \to \max,\ Ax = b,\ x\geq 0, \tag{1.55}$$

двойственная задача имеет вид

$$b'y \to \min,\ A'y \geq b. \tag{1.56}$$

Легко проверить, что если в качестве исходной взять задачу линейного программирования (1.52), то двойственной к ней будет задача (1.51).

Сравнивая пары двойственных задач (1.51) и (1.52), приходим к следующему правилу построения двойственной задачи:

1. Двойственная задача является задачей минимизации линейной формы, коэффициенты которой совпадают с коэффициентами вектора условий прямой задачи.

2. Каждому i-му основному ограничению прямой задачи соответствует i-я переменная y_i двойственной задачи. При этом для i-й переменной y_i двойственной задачи выполняются соотношения:

а) она не имеет ограничений на знак, если i-е основное ограничение прямой задачи имело вид равенства;

б) $y_i \geq 0$ ($y_i \leq 0$), если i-е основное ограничение прямой задачи имело вид неравенства „$\leq$“ („$\geq$“).

3. Каждой j-й переменной x_j прямой задачи соответствует j-е основное ограничение двойственной задачи. При этом вид j-го основного ограничения двойственной задачи определяется условиями:

а) оно имеет вид равенства, если переменная x_j в прямой задаче не имеет ограничений на знак;

б) оно имеет вид неравенства „$\geq$“ („$\leq$“), если в прямой задаче переменная x_j имела ограничение $x_j \geq 0$ ($x_j \leq 0$) на знак.

Рассмотрим общую задачу линейного программирования (1.51) и двойственную к ней задачу (1.52). Назовем эти задачи парой двойственных задач.

2. Соотношения двойственности

Основу теории двойственности составляют теорема существования и теорема двойственности, а также вытекающие из них соотношения двойственности.

Теорема 9 *(Теорема существования) Для существования решения задачи линейного программирования необходимо и достаточно, чтобы не были пустыми множества её прямых и двойственных планов.*

Доказательство. Необходимость. В силу общности канонической задачи достаточно рассмотреть только задачу (1.44). Пусть задача (1.44) имеет решение. Выше мы отмечали (см. теорему 7), что в этом случае существует оптимальный базисный план x^0 с некоторой базисной матрицей $A_{\text{Б}}$. В п.1 было показано, что вектор потенциалов $u' = c'_{\text{Б}} A_{\text{Б}}^{-1}$ является оптимальным двойственным планом, из чего следует, что множество двойственных планов непусто.

Достаточность. Пусть множества X и Y прямых и двойственных планов задачи (1.44) не являются пустыми. Покажем, что имеет место неравенство

$$c'x \leq b'y,\ \forall x \in X,\ \forall y \in Y. \tag{1.57}$$

Действительно, в силу соотношений

$$Ax = b,\ x \geq 0,\ \ A'y - c \geq 0,$$

имеем $x'(A'y - c) \geq 0$ и, следовательно,

$$b'y = y'b = y'Ax = (y'Ax - c'x) + c'x = x'(A'y - c) + c'x \geq c'x.$$

Соотношение (1.57) доказано.

Из (1.57) следует, что целевая функция задачи (1.44) ограничена сверху, и множество ее планов X непусто, значит эта задача имеет решение.

Теорема 10 *(Теорема двойственности) Для существования решения прямой задачи линейного программирования необходимо и достаточно существование решения y^0 двойственной ей задачи. На решениях x^0, y^0 значения прямой и двойственной целевых функций равны:*

$$c'x^0 = b'y^0;$$

Доказательство. Необходимость. Совпадает с доказательством необходимости в теореме 8.

Достаточность. Пусть y^0 — решение двойственной задачи (1.49). Рассматриваем её как прямую и используем первую часть утверждения теоремы 9, учитывая, что исходная задача станет двойственной к задаче (1.49).

Следствие 1 *На каждой паре (x, y) из прямого плана x и двойственного плана y выполняется неравенство*

$$c'x \leq b'y. \tag{1.58}$$

Доказательсто. См. доказательство неравенства (1.57).

Следствие 2 *(достаточное условие несовместимости ограничений.) Если вдоль некоторой последовательности y^k, $k = 1, 2, \ldots$, двойственных планов двойственная целевая функция неограниченно убывает:*

$$b'y^k \to -\infty, k \to \infty, \tag{1.59}$$

то прямая задача (1.44) не имеет планов.

Доказательство. Если допустить существование плана x, то в силу (1.59) найдётся такое k_0, что $c'x > b'y^{k_0}$, однако это противоречит (1.58).

Следствие 3 *(достаточное условие оптимальности.) Если на некоторых прямом x^* и двойственном y^* планах выполняется равенство*

$$c'x^* = b'y^*, \tag{1.60}$$

то x^*, y^* — решения задач (1.44), (1.49).

Доказательство. Поскольку $c'x \leq b'y$, $\forall x \in X$, $\forall y \in Y$, то из (1.60) следует оптимальность плана x^* в задаче (1.44), и оптимальность плана y^* в задаче (1.49).

§5. Двойственный симплекс-метод

Метод решения канонической задачи линейного программирования, рассмотренный выше, впредь будем называть **прямым симплекс-методом**.

Двойственный симплекс-метод — это метод построения оптимальных планов прямой канонической задачи с помощью реализации прямого симплекс-метода на двойственной канонической задаче.

При исследовании практических задач линейного программирования оба симплекс-метода — прямой и двойственный — играют важную роль, дополняя друг друга.

1. Базисный двойственный план. Коплан. Псевдоплан. Формула приращения

Двойственный симплекс-метод представляет специальный метод решения канонической задачи линейного программирования:

$$c'x \to \max, Ax = b, x \geq 0,\ rankA = m,\ A \in \mathbb{R}^{m\times n}, \tag{1.61}$$

посредством преобразования планов двойственной к (1.61) задачи:

$$b'y \to \min, A'y \geq c. \tag{1.62}$$

В §4 при формировании двойственной задачи (1.62) было показано, что оптимальному в задаче (1.61) плану x^0 с базисной матрицей $A_{\text{Б}} = (A_j, j \in J_{\text{Б}})$ соответствует решение $y^0 = u$ двойственной задачи (1.62), удовлетворяющее соотношениям

$$y^{0\prime}A_{\text{Б}} = c'_{\text{Б}},\ \ y^{0\prime}A_{\text{Н}} \geq c_{\text{Н}}. \tag{1.63}$$

С другой сторны, имея двойственный план y^0, на котором выполняется (1.63), можно построить оптимальный базисный план $x^0 = (x_{\text{Б}} = A_{\text{Б}}^{-1}b, x_{\text{Н}} = 0)$ задачи (1.61), если $A_{\text{Б}}^{-1}b \geq 0$.

Этот анализ показывает, что решение двойственной задачи (1.62), позволяющее построить оптимальный план исходной задачи (1.61), достаточно искать среди специальных двойственных планов, которые называются **базисными двойственными планами**.

Определение 3 *Двойственный план y называется* **двойственным базисным планом**, *если найдется такое множество $J_Б \subset J$, что выполняются следующие соотношения*

1) $|J_Б| = m$,

2) $\det A_Б \neq 0$, *где* $A_Б = (A_j, j \in J_Б)$,

3) на векторе y ограничения двойственной задачи выполняются в следующем виде:

$$A'_j y = c_j,\ \ j \in J_Б,\ A'_j y \geq c_j,\ \ j \in J_Н,\ где\ J_Н = J \backslash J_Б.$$

Определение 4 *Двойственный базисный план называется* **невырожденным**, *если $A'_j y > c_j,\ \ j \in J_Н$.*

В последующих вычислениях часто используются два вектора, связанные с базисным двойственным планом.

Определение 5 *Вектор $\delta = \delta(J) = (\delta_j, j \in J)$:*

$$\delta = A'y - c\ или\ \ \delta_j = A_j{}'y - c_j,\ \ j \in J,$$

подсчитанный на двойственном плане y (т.е. при $\delta \geq 0$) называется **копланом** *задачи (1.61).* **Базисный коплан** *— это коплан, соответствующий двойственному базисному плану. С некоторым множеством $J_Б$ он удовлетворяет соотношением*

$$\delta(J_Б) = 0,\ |J_Б| = m,\ \det A_Б \neq 0,\ \ A_Б = (A_j, j \in J_Б).$$

Определение 6 *Вектор $\varkappa = \varkappa(J)$ с компонентами $\varkappa_Б = \varkappa(J_Б) = A_Б^{-1}b, \varkappa_Н = \varkappa(J_Н) = 0$, подсчитанный по двойственному базисному плану $\{y, J_Б\}$, называется базисным* **псевдопланом** *задачи (1.61), соответствующим двойственному базисному плану.*

2. Формула приращения целевой функции в двойственной задаче

Пусть $\{y, J_Б\}$ – двойственный базисный план с базисной матрицей $A_Б$. Пусть $\bar{y} = y + \Delta y$ – другой двойственный план. Выведем формулу для приращения

$$b'\bar{y} - b'y = b'\Delta y$$

двойственной целевой функции.

Из определений коплана и двойственного базисного плана имеем

$$\Delta\delta'(J_Б) = \bar{\delta}'(J_Б) - \delta'(J_Б) = (\bar{y}'A_Б - c'(J_Б)) - (y'A_Б - c'(J_Б)) = \Delta y' A_Б,$$

где, как и раньше, $\Delta\delta'(J_Б) = (\Delta\delta_j,\ j \in J_Б)$, $c(J_Б) = (c_j,\ j \in J_Б)$, и т.д. С учетом этого и определения псевдоплана $\varkappa$ имеем

$$b'\Delta y = \Delta y' b = \Delta y' A_Б \varkappa_Б = \Delta\delta'_Б \varkappa_Б = \sum_{j \in J_Б} \varkappa_j \Delta\delta_j. \tag{1.64}$$

Формула (1.64) — формула приращения двойственной целевой функции.

3. Критерий оптимальности. Достаточное условие отсутствия прямых планов

Пусть $\{y, J_Б\}$ – двойственный базисный план с базисной матрицей $A_Б$, $\varkappa = \{\varkappa_Б = A_Б^{-1}b, \varkappa_Н = 0\}$ — соответствующий базисный псевдоплан.

Теорема 11 *(Критерий оптимальности). Неравенство*

$$\varkappa_Б \geq 0 \tag{1.65}$$

достаточно, а в случае невырожденности и необходимо для оптимальности двойственного базисного плана y. Базисный псевдоплан $\varkappa$, соответствующий y, является оптимальным прямым планом.

Доказательство. **Достаточность.** Пусть неравенства (1.65) имеют место. Следовательно, вектор $\varkappa = (\varkappa_Б = A_Б^{-1}b, \varkappa_Н = 0)$ является планом прямой задачи. Докажем, что $y'b = c'\varkappa$. Действительно, по построению имеем

$$y'b = y'A_Б\varkappa_Б = c'_Б\varkappa_Б = c'\varkappa.$$

Согласно следствию 3 из предыдущего параграфа, векторы $\varkappa$ и y — оптимальные планы прямой и двойственной задач соответственно.

Необходимость. Пусть $\{y, J_Б\}$ — оптимальный невырожденный базисный двойственный план, следовательно

$$\delta_j = y'A_j - c_j > 0,\ j \in J_Н. \tag{1.66}$$

Пусть $J_Б = \{j_1, j_2, \ldots, j_m\}$. Предположим,что (1.65) нарушается, т.е. $\exists k, 1 \leq k \leq m$, что для $j_k \in J_Б$ верно неравенство

$$\varkappa_{j_k} < 0. \tag{1.67}$$

Построим новый двойственный план $\bar{y} = y + \Delta y$ следующим образом. Положим

$$\Delta y' = \sigma e_k' A_Б^{-1}, \tag{1.68}$$

где $\sigma > 0$ – некоторый параметр. Этот параметр должен быть таким, что выполняются соотношения

$$\bar{y}'A_j - c_j = y'A_j - c_j + \Delta y'A_j = \delta_j + \Delta y'A_j \geq 0,\ j \in J. \tag{1.69}$$

С учетом (1.68) имеем

$$\bar{\delta}_j = \delta_j + \Delta y'A_j = \begin{cases} 0,\ j \in J_Б \setminus j_k, \\ 0 + \sigma,\ j = j_k, \\ \delta_j + \sigma\mu_j \geq 0,\ j \in J_Н, \end{cases} \tag{1.70}$$

где $\mu_Н' = (\mu_j, j \in J_Н) = e_k' A_Б^{-1} A_Н$, т.е.

$$\mu_j = e_k' A_Б^{-1} A_j,\ j \in J_Н.$$

Поскольку по предположению, $\delta_j > 0,\ j \in J_Н$, то из (1.70) следует, что всегда существует $\sigma > 0$, при котором имеют место соотношения (1.69).

Напомним, что по построению, $\Delta\delta_Б' = \bar{\delta}_Б' - \delta_Б' = \Delta y'A_Б = \sigma e_k'$. С учетом этого формула приращения (1.64) принимает вид

$$b'\bar{y} - b'y = \Delta\delta_Б'\varkappa_Б = \sigma\varkappa_{j_k} < 0. \tag{1.71}$$

Следовательно, $b'\overline{y} < b'y$. Однако это противоречит предположению об оптимальности плана y в двойственной задаче. ⊠

Рассмотрим случай, когда критерий оптимальности (1.65) не выполняется и некоторой отрицательной компоненте $\varkappa_{j_k} < 0$ псевдоплана (1.68) соответствуют неотрицательные числа

$$\mu_j = e_k' A_{\text{Б}}^{-1} A_j \geq 0, j \in J_{\text{н}}. \tag{1.72}$$

Тогда из (1.70), (1.72) следует, что при $\forall \sigma \geq 0$ вектор $\bar{\delta} = \delta + \Delta\delta$ будет копланом задачи (1.61). Из (1.71) также следует, что при $\sigma \to \infty$ двойственная целевая функция $b'\overline{y} \to -\infty$.

Согласно следствию 2 из §4 это означает, что ограничения прямой задачи несовместны. Таким образом мы доказали теорему:

Теорема 12 *(Достаточное условие отсутствия прямых планов). Если при некотором $j_k \in J_Б$ выполняются неравенства (1.67), (1.72), т.е.*

$$\varkappa_{j_k} < 0, \mu_j \geq 0, j \in J_Н,$$

то ограничения прямой задачи (1.61) противоречивы.

4. Итерация

Осталось рассмотреть случай, когда для каждой отрицательной компоненты $\varkappa_{j_k}$ (1.67) псевдоплана неравенства (1.72) не выполняются, т.е. $\mu_j < 0$ при некотором $j \in J_{\text{н}}$.

В этом случае при увеличении σ компоненты $\bar{\delta}_j = \delta_j + \sigma\mu_j$, где $\mu_j < 0$, при

$$\sigma = \sigma_j = -\delta_j/\mu_j$$

обращаются в ноль, а при $\sigma > \sigma_j$ становятся отрицательными. Отсюда следует, что максимально допустимое значение σ_0, при котором $\bar{\delta}$ остается копланом, равно

$$\sigma_0 = \sigma_{j_0} = \min \sigma_j, j \in J_{\text{н}}, \text{ где } \sigma_j = \begin{cases} -\delta_j/\mu_j \text{ при } \mu_j < 0, \\ \infty \text{ при } \mu_j \geq 0. \end{cases}$$

Для этого значения $\sigma = \sigma_0$ двойственная целевая функция при замене $y \to \bar{y}$ согласно (1.71) уменьшилась на максимальную величину $|\sigma_0 \varkappa_{j_k}|$, которая будет положительной, если y — невырожденный базисный план.

Докажем, что двойственный план $\bar{y} = y + \Delta y$, соответствующий построенному коплану $\bar{\delta}$, является базисным. У коплана $\bar{\delta}$ согласно (1.68), (1.70) только одна компонента $\bar{\delta}_{j_k}$ из компонент $\bar{\delta}_j, j \in J_{\text{Б}}$, может оказаться (при $\sigma = \sigma_0 \geq 0$) отличной от нуля. Но вместо нее среди компонент $\bar{\delta}_j, j \in J_{\text{Н}}$, обязательно появится нулевая компонента $\bar{\delta}_{j_0} = 0$. Следовательно, имеем

$$\bar{\delta}(\bar{J}_{\text{Б}}) = 0, \ \bar{J}_{\text{Б}} = (J_{\text{Б}} \backslash j_k) \cup j_0, \tag{1.73}$$

$$\bar{\delta}(\bar{J}_{\text{Н}}) \geq 0, \ \bar{J}_{\text{Н}} = J \backslash \bar{J}_{\text{Б}} = (J_{\text{Н}} \backslash j_0) \cup j_k. \tag{1.74}$$

Покажем, что $\det \bar{A}_{\text{Б}} \neq 0, \bar{A}_{\text{Б}} = (A_j, j \in \bar{J}_{\text{Б}})$. Действительно, матрица $\bar{A}_{\text{Б}}$ получается из $A_{\text{Б}}$ заменой столбца A_{j_k} на столбец A_{j_0}. По построению, $\det A_{\text{Б}} \neq 0$. Раньше мы показываем, что при этих условиях (т.е. при $\det A_{\text{Б}} \neq 0$ и условии, что $A_{\text{Б}}$ и $\bar{A}_{\text{Б}}$ отличаются только одним столбцом) матрица $\bar{A}_{\text{Б}}$ будет невырожденной тогда и только тогда, когда

$$\alpha = e_k' A_{\text{Б}}^{-1} A_{j_0} \neq 0.$$

По построению, имеем

$$\alpha = e_k' A_{\text{Б}}^{-1} A_{j_0} = \mu_{j_0} < 0.$$

Следовательно, $\det \bar{A}_{\text{Б}} \neq 0$, из чего следует (с учетом (1.74)), что $\bar{\delta}$ — базисный коплан с базисом $\bar{J}_{\text{Б}}$ и базисной матрицей $\bar{A}_{\text{Б}}, \det \bar{A}_{\text{Б}} \neq 0$.

Для подсчета матрицы $\bar{A}_{\text{Б}}^{-1}$, как и раньше, имеем формулу

$$\bar{A}_{\text{Б}}^{-1} = A_{\text{Б}}^{-1} - \frac{A_{\text{Б}}^{-1}(A_{j_0} - A_{j_k}) e_k' A_{\text{Б}}^{-1}}{\alpha}. \tag{1.75}$$

Переход от старого базисного коплана $\{\delta, J_{\text{Б}}\}$ к новому базисному коплану $\{\bar{\delta}, \bar{J}_{\text{Б}}\}$ по описанным выше правилам называется итерацией двойственного симплекс-метода.

Замечание 3 *Правила выбора индекса j_k и правила выбора индекса j_0 при неоднозначном их определении аналогичным правилам, которые используются в прямом симплекс-методе.*

Замечание 4 *Двойственный симплекс-метод эквивалентен прямому симплекс-методу, примененному к двойственной задаче с учетом ее специфики.*

5. Алгоритм

Вначале итерации известны:

базисное множество $J_{\text{Б}} = \{j_1, j_2, \ldots, j_m\}$,

матрица $B = (A_{\text{Б}}^{-1})$, где $A_{\text{Б}} = (A_j, j \in J_{\text{Б}})$ и

небазисные компоненты коплана

$$\delta_{\text{Н}} = (\delta_j, j \in J_{\text{Н}}) := c'_{\text{Б}} A_{\text{Б}}^{-1} A_{\text{Н}} - c'_{\text{Н}} \geq 0.$$

Шаг 1. Вычисляем

$$\varkappa_{\text{Б}} = Bb.$$

Шаг 2. Если $\varkappa_{\text{Б}} \geq 0$, то STOP: вектор $\varkappa = (\varkappa_{\text{Б}} = Bb, \varkappa_{\text{Н}} = 0)$ — оптимальный план прямой задачи. В противном случае переходим к шагу 3.

Шаг 3. Находим базисный индекс $j_k \in J_{\text{Б}}$, для которого $\varkappa_{j_k} < 0$. Находим числа

$$\mu_j = B'_k A_j, j \in J_{\text{Н}},$$

где B'_k — k-я строка матрицы B, A_j — j-ый столбец матрицы условий A.

Шаг 4. Вычисляем шаги

$$\sigma_j = \begin{cases} -\delta_j/\mu_j, \text{ если } \mu_j < 0, \\ \infty, \text{ если } \mu_j \geq 0, \end{cases} \quad j \in J_{\text{Н}};$$

$$\sigma_0 = \min \sigma_j, j \in J_{\text{Н}}.$$

Шаг 5. Если $\sigma_0 = \infty$, то STOP: ограничения прямой задачи несовместны. В противном случае перейдем к шагу 6.

Шаг 6. Найдем такой небазисный индекс $j_0 \in J_{\text{Н}}$, что $\sigma_0 = \sigma_{j_0}$.

Шаг 7. Строим новое базисное множество

$$\bar{J}_{\text{Б}} = (J_{\text{Б}} \backslash j_k) \cup j_0 = \{j_1, \ldots, j_{k-1}, j_0, j_{k+1}, \ldots, j_m\}.$$

<u>Шаг 8.</u> Строим небазисные компоненты $\bar{\delta}_j$, $j \in \bar{J}_{\text{н}}$, нового базисного коплана $\bar{\delta}$:

$$\bar{\delta}_j = \delta_j + \sigma_0 \mu_j, \ j \in \bar{J}_{\text{н}} \backslash j_k, \ \bar{\delta}_{j_k} = \sigma_0.$$

Здесь $\bar{J}_{\text{н}} = J \backslash \bar{J}_{\text{Б}} = (J_{\text{н}} \backslash j_0) \cup j_k$.

<u>Шаг 9.</u> Строим матрицу $\bar{B} := \bar{A}_{\text{Б}}^{-1}$, где $\bar{A}_{\text{Б}} = (A_j, \ j \in \bar{J}_{\text{Б}})$, по тем же правилам, что и в прямом симплекс-методе (см. также формулу (1.75)).

Переходим в новой итерации, исходя из $\bar{J}_{\text{Б}}$, $\bar{B}$, $\bar{\delta}_{\text{н}}$.

§6. Анализ чувствительности

При решении прикладных задач часто представляет интерес исследование зависимости решения задачи от параметров A, b, c задачи. Так на практике многие экономические параметры (цены на продукцию и сырье, запасы сырья, спрос на рынке, заработная плата и т.д.) с течением времени меняют свои значения. Поэтому оптимальное решение задачи ЛП, полученное для конкретной экономической ситуации, после ее изменения может оказаться непригодным или неоптимальным. В связи с этим становиться актуальным исследование следующих вопросов

• как возможные изменения параметров исходной модели повлияют на полученное ранее оптимальное решение задача,

• как быстро найти решение задачи, полученной в результате изменения исходных данных, при условии что решение исходной (невозмущенной) задачи уже известно.

Решениям этих вопросов занимает специальный раздел теории оптимизации, называемый анализом чувствительности. Здесь важную роль играют теория двойственности и двойственный симплекс-метод.

1. Решения двойственной задачи как оценки влияния

Покажем, что при некоторых изменениях компонент вектора ограничений b в задаче (1.61) компоненты решения y^0 двойственной задачи (1.62) могут быть интерпретированы как оценки влияния этих изменений на величину максимума целевой функции задачи (1.61).

Рассмотрим задачу

$$P(z): \qquad c'x \to \max, \ \ x \in X(z) = \{x \in \mathbb{R}^n : Ax = z, \ x \geq 0\}$$

и двойственную к ней

$$D(z): \quad z'y \to \min, \ \ y \in Q = \{y \in \mathbb{R}^m : A'y \geq c\}.$$

В предположении, что задача $(P(z))$ разрешима при некотором $z \in \mathbb{R}^m$, обозначим через $f^0(z)$ оптимальное значение целевой функции этой задачи:

$$f^0(z) := \max_{x \in X(z)} c'x.$$

Пусть при $z = z^*$ существует решение x^* задачи $(P(z^*))$ и, следовательно, существует $y^* \in Q$ — решение задачи $(Q(z^*))$. Из Теоремы 8 следует, что без ограничения общности можно считать, оптимальный план x^* является базисныс с базисом $J^*_{\text{Б}}$.

Теорема 13 *Если задача $(P(z^*))$ имеет невырожденный оптимальный базисный план, то существует такая окрестность $U(z^*)$ точки z^*, что для всех $z \in U(z^*)$ будет*

$$f^0(z) = z'y^*,$$

где y^* — решение задачи $(Q(z^*))$.

Доказательство. Пусть $\{x^*, J^*_{\text{Б}}\}$ — оптимальный невырожденный план задачи $(P(z^*))$ при $z = z^*$. Тогда по построению имеем

$$x^* = (x^*_{\text{Б}}, x^*_{\text{Н}}), \text{ где } x^*_{\text{Б}} = (x^*_j, j \in J^*_{\text{Б}}) > 0,$$

$$x^*_{\text{Н}} = (x^*_j, j \in J^*_{\text{Н}}) = 0, J^*_{\text{Н}} = J \setminus J^*_{\text{Б}}),$$

$$x^*_{\text{Б}} = B^* z^*, \ \ B^* = (A^*_{\text{Б}})^{-1}, A^*_{\text{Б}} = (A_j, j \in J^*_{\text{Б}}).$$

Обозначим

$$\alpha = \min_{j \in J^*_{\text{Б}}} x^*_j.$$

Выберем окрестность

$$U(z^*) = \{z \in \mathbb{R}^m : ||z - z^*|| \leq \alpha / ||B^*||\},$$

где нормы $||z - z^*||$ и $||B^*||$ выбраны следующим образом

$$||z - z^*|| = \max_{i = \overline{1,m}} |z^*_i - z_i|, \ ||B^*|| := \max\{||B^* z|| : z \in \mathbb{R}^m, ||z|| = 1\}.$$

Покажем, что вектор

$$x(z) = (x_{\text{Б}}(z), x_{\text{Н}}(z)),$$

где

$$x_{\text{Б}}(z) = (x_j(z), j \in J^*_{\text{Б}}) = B^* z, \;\; x_{\text{Н}}(z) = (x_j(z), j \in J^*_{\text{Н}}) = 0,$$

будет оптимальным планом задачи $(P(z))$ при любом $z \in U(z^*)$.

Действительно,

$$||x^*_{\text{Б}} - x_{\text{Б}}(z)|| = ||B^*(z^* - z)|| \leq ||B^*||||z^* - z|| \leq \alpha,$$

где

$$||x^*_{\text{Б}} - x_{\text{Б}}(z)|| := \max_{j \in J^*_{\text{Б}}} |x^*_j - x_j(z)|.$$

Получаем, что

$$x^*_j - x_j(z) \leq \alpha, \; j \in J^*_{\text{Б}},$$

т.е.

$$x_j(z) \geq x^*_j - \alpha = x^*_j - \min_{i \in J^*_{\text{Б}}} x^*_i \geq 0, \; j \in J^*_{\text{Б}}.$$

Таким образом, $x(z) \geq 0$. Кроме того, по построению $Ax(z) = z$, следовательно, $x(z) \in X(z)$ (т.е. $x(z)$ — допустимый план задачи $(P(z))$).

Докажем теперь, что $x(z)$ — оптимальный план задачи $(P(z))$. Из условия оптимальности невырожденного базисного плана $\{x^*, J^*_{\text{Б}}\}$ следует, что на решении y^* задачи $(Q(z^*))$ имеют место равенства

$$A'_j y^* = c_j, \; j \in J^*_{\text{Б}}.$$

Тогда

$$c'x(z) = \sum_{j \in J^*_{\text{Б}}} c_j x_j(z) = \sum_{j \in J^*_{\text{Б}}} (A'_j y^*) x_j(z) = (y^*)' \sum_{j \in J^*_{\text{Б}}} A_j x_j(z) =$$
$$= (y^*)' A^*_{\text{Б}} x_{\text{Б}}(z) = (y^*)' z = z' y.$$

Выше было показано, что $x(z) \in X(z)$, и по построению $y^* \in Q$. Значит, согласно Следствию 3, вектор $x(z) \in X(z)$ является оптимальным планом задачи $(P(z))$ и $f^0(z) = z' y^*$. Теорема доказана.

Очевидно, что

$$\frac{\partial f^0(z)}{\partial z_i} = y^*_i, \;\; i = \overline{1, m},$$

поэтому увеличение компоненты z_i вектора условий приводит к увеличению или уменьшению величины оптимального значения целевой функции $f^0(z)$ в задаче $(P(z))$ в зависимости от знака y_i^*. При этом скорость изменения значения целевой функции $f^0(z)$ определяется величиной $|y_i^*|$.

2. О применимости двойственного симплекс-метода для построения решений возмущенных задач

2.1. Изменение объемов ресурсов

Рассмотрим и изложим метод коррекции оптимальных планов при вариациях (возмущениях) объемов ресурсов (вектора b) в производственной задаче.

Рассмотрим задачу

$$c'x \to \max,\ Ax = b,\ x \geq 0, \tag{1.76}$$

которую будем называть невозмущенной.

Пусть $\{x^0, J_{\text{Б}}^0\}$ — оптимальный базисный план задачи (1.76). Предполагается, что он известен. Следовательно, известен и соответствующий ему вектор потенциалов u^0

$$(u^0)' = (c_j, j \in J_{\text{Б}}^0)(A_j, j \in J_{\text{Б}}^0)^{-1}. \tag{1.77}$$

Предположим теперь, что в задаче (1.76) вектор ресурсов b изменился $b \to \bar{b} = b + \Delta b$, где $\|\Delta b\|$ — малое число. Мы получим новую задачу

$$c'x \to \max,\ Ax = \bar{b},\ x \geq 0, \tag{1.78}$$

которую будем называть возмущенной.

Требуется найти решение возмущенной задачи.

Очевидно, что в общем случае вектор x^0 не являлся даже планом задачи (1.78). Следовательно, если мы будем следовать общим правилам, то решение задачи (1.78) надо начинать с первой фазы симплекс-метода, т.е. "с нуля никак не используя дополнительную информацию о том, что мы знаем решения x^0 невозмущенной задачи (1.76), "близкой"к задаче (1.78). Очевидно, что, как правило, такой путь будет неэффективным.

В этом случае гораздо эффективнее использовать двойственный симплекс-метод. Ясно, что оптимальный в задаче (1.76) вектор оценок

$$\Delta^{0\prime} = u^{0\prime}A - c,\ u^{0\prime} = c'(J_{\text{Б}}^0)(A_{\text{Б}}^0)^{-1},\ A_{\text{Б}}^0 = (A_j, j \in J_{\text{Б}}^0),$$

является базисным копланом $\delta = \Delta^0$ с базисом $J^0_{\text{Б}}$, соответствующим базисному двойственному плану $y = u^0$ задачи (1.78).

Значит оптимальный вектор оценок Δ^0 задачи (1.76) можно использовать в качестве начального базисного коплана $\{\delta, J^0_{\text{Б}}\}$ в задаче (1.78). Так как задачи (1.76) и (1.78) являются "близкими"(их параметры отличаются незначительно), то в общем случае и их решения (прямые и двойственные) отличаются незначительно. Следовательно, есть надежда, что решая задачу (1.78) двойственным симплекс-методом, начиная процесс решения с **известного** базисного коплана $\{\delta, J^0_{\text{Б}}\}$, через несколько итераций мы построим оптимальный базисный коплан задачи (1.78), а значит и оптимальный план задачи (1.78).

2.2 Изменение размеров задачи

Размеры задачи линейного программирования характеризуются двумя параметрами m и n. В приложениях нередко возникают ситуации, когда надо решить задачи, размеры которых мало отличаются от размеров задачи с известным оптимальным планом. Умелое применение прямого и двойственного симплекс-метода позволяет эффективно использовать дополнительную информацию.

Остановимся подробнее на случае увеличения числа основных ограничений. Пусть x^0 — оптимальный план задачи (1.76), $A_{\text{Б}} = (A_j, j \in J^0_{\text{Б}})$, $J^0_{\text{Б}}$ — соответствующий базис, $u^0 \in \mathbb{R}^m$ — оптимальный вектор потенциалов (1.77). Предположим, что к основным ограничениям добавлено ограничение

$$a'x \leq \beta. \tag{1.79}$$

Если $a'x^0 \leq \beta$, то очевидно, что вектор x^0 — оптимальный план новой задачи (1.76), (1.79).

Пусть $a'x^0 > \beta$. В этом случае x^0 не является даже планом задачи (1.76), (1.79) и использование прямого симплекс-метода не эффективно по тем же самым соображениям, о которых говорилось выше (теряем дополнительную информацию).

Попробуем применить двойственный симплекс-метод. Сведем задачу (1.76), (1.79) к канонической форме

$$c'x \to \max,$$

$$Ax = b, \tag{1.80}$$

$$a'x + x_{n+1} = \beta,\ x \geq 0,\ x_{n+1} \geq 0.$$

Обозначим $\bar{J} = \{1, ..., n, n+1\}$, $\bar{x} = (x, x_{n+1})$, $\bar{A} = \begin{pmatrix} A & 0 \\ a' & 1 \end{pmatrix}$, $\bar{b} = \begin{pmatrix} b \\ \beta \end{pmatrix}$, $\bar{c}' = (c', 0)$. Тогда задачу (1.80) можно записать в виде

$$\bar{c}'\bar{x} \to \max,\ \bar{A}\bar{x} = \bar{b}, \bar{x} \geq 0. \tag{1.81}$$

Нетрудно проверить, что в задаче (1.81) $(m+1)$-вектор $y = (u^0, 0)$ является двойственным базисным планом с базисом $J_Б^0 \cup \{n+1\}$ и базисной матрицей

$$\bar{A}_Б = \begin{pmatrix} A_Б & 0 \\ a'(J_Б^0) & 1 \end{pmatrix} \in R^{(m+1)\times(m+1)}.$$

Вектору y соответствует коплан $\delta(\bar{J}) = (\Delta^0(J), 0)$, где $\Delta^0(J)$ оптимальный вектор оценок в задаче (1.76). Следовательно, у нас есть вся необходимая информация для того, чтобы начать решение задачи (1.81) двойственным симплекс-методом. Отметим, что, зная $A_Б^{-1}$, легко найти матрицу, обратную к $\bar{A}_Б$ по правилу

$$\bar{A}_Б^{-1} = \begin{pmatrix} A_Б^{-1} & 0 \\ a'(J_Б^0)A_Б^{-1} & 1 \end{pmatrix}.$$

Очевидно, что все, сказанное выше, остается в силе, если к ограничениям невозмущенной задачи (1.76), добавляется не одно, а несколько новых ограничений-неравенств.

Глава 2

Транспортные задачи

§1. Сетевая транспортная задача

Транспортными задачами линейного программирования называются математические модели разнообразных прикладных задач по оптимизации перевозок. К ним сводятся многочисленные задачи, имеющие другую физическую природу.

В данном параграфе для решения транспортных задач в сетевой и матричной формах строится метод потенциалов как реализация прямого симплекс-метода, детально учитывающая специфику новых задач.

1. Сеть. Поток. Сетевая транспортная задача

Рассмотрим множество $I = \{1, 2, \ldots, n\}$ элементов, которое называется узлами.

Предположим, что некоторые пары узлов $i \in I$, $j \in I$ упорядочены. Такую пару обозначим символом (i, j) и назовем дугой с началом i и концом j.

Множество дуг, определенных на множестве $I \times I$, обозначим через U:

$$U \subset \{(i, j) : \ i \in I, j \in I\}.$$

Определение 7 *Совокупность $S = \{I, U\}$ называется (ориентированной) сетью.*

На рисунке узлы будут обозначаться точками, дуги (i, j) — линиями со стрелкой из i в j.

Каждому узлу i припишем число a_i — интенсивность узла. При $a_i > 0$ узел i будем называть узлом производства (источником); при $a_i < 0$ узел i будем называть узлом потребления (стоком); при $a_i = 0$ узел i — транзитный узел (нейтральный).

При изображении сети на рисунке возле узла i указывается величина $|a_i|$ со стрелкой: стрелка, входящая в узел, указывает на узел производства, а стрелка, выходящая из узла, — на узел потребления.

Каждой дуге $(i, j) \in U$ припишем неотрицательное число x_{ij} — дуговой поток (поток по дуге (i, j)).

Обозначим через

$$I_i^+ = I_i^+(U) = \{j \in I : \ \exists (i, j) \in U\}$$

$$I_i^- = I_i^-(U) = \{j \in I : \ \exists (j, i) \in U\}$$

множество узлов, которые соединены с узлом i дугами из U, начинающимися в i (это узлы I_i^+) или оканчивающимися в i (это узлы I_i^-).

Говорят, что в узле i выполнено условие баланса, если имеет место равенство

$$\sum_{j \in I_i^+} x_{ij} - \sum_{j \in I_i^-} x_{ji} = a_i, \tag{2.1}$$

т.е. количество (продукта) потока, поступающее в узел i и произведенного в нем, равно количеству (продукта) потока, выходящего из узла i и потребленного в нем.

Определение 8 *Совокупность $x = \{x_{ij}; \ (i, j) \in U\}$ дуговых потоков называется потоком на сети $S = \{I, U\}$, если она удовлетворяет условиям баланса (2.1) в каждом узле $i \in I$; и $x_{ij} \geq 0$, $(i, j) \in U$.*

Число $\sum\limits_{(i,j) \in U} c_{ij} x_{ij}$, где $c_{ij}, \ (i, j) \in U$, — заданные числа, называется стоимостью потока x.

Сетевая транспортная задача (транспортная задача в сетевой форме), именуемая также задачей о потоке минимальной стоимости, состоит в поиске оптимального потока $x^0 = \{x_{ij}^0; \ (i, j) \in U\}$ (потока минимальной стоимости), который доставляет решение задаче

$$\sum_{(i,j) \in U} c_{ij} x_{ij} \to \min \tag{2.2}$$

$$\sum_{j \in I_i^+} x_{ij} - \sum_{j \in I_i^-} x_{ji} = a_i, \ i \in I; \ x_{ij} \geq 0, \ (i, j) \in U.$$

Целевая функция задачи (2.2) и все функции ограничений линейны относительно переменных x_{ij}, $(i,j) \in U$, следовательно, задача (2.2) является задачей линейного программирования. Но задача (2.2) — специальная задача линейного программирования: при ее сведении к канонической форме получается большая матрица условий, которая состоит из ± 1 и большого количества нулей. Поэтому сводить задачу (2.2) к общей задаче линейного программирования в канонической форме и решать ее симплекс-методом не целесообразно. Мы предложим другой метод — метод потенциалов, который являясь методом симплексного типа, максимально учитывает специфику задачи.

Задачу (2.2) можно записать в виде

$$\bar{c}'x \to \max,$$

$$Ax = b,\ x \geq 0,$$

где $x = (x_{ij},\ (i,j) \in U)$, $\bar{c} = -(c_{ij},\ (i,j) \in U)$, $b = (b_i = a_i,\ i \in I)$, $A \in \mathbb{R}^{n \times |U|}$, $A = (A_{ij}, (i,j) \in U)$, $A_{ij} \in \mathbb{R}^n$, $n = |I|$, $A_{ij} = (a_{ij}(k),\ k \in I)$, $a_{ij}(i) = 1$, $a_{ij}(j) = -1$, $a_{ij}(k) = 0$, $k \in I \backslash \{i,j\}$.

2. Базисный поток

Введем необходимые понятия из теории сетей.

Дугу (i,j) без ориентации назовем ребром с *граничными узлами* i, j и будем обозначать следующим образом $\{i,j\}$.

Узел сети называется *висячим*, если он граничный узел для единственного (висячего) ребра.

Последовательность различных ребер

$$\{i_1, i_2\},\ \{i_2, i_3\}, \ldots, \{i_{k-1}, i_k\}, \tag{2.3}$$

в которой соседние ребра имеют общие граничные узлы, называется (простой) *цепью*, соединяющей узлы i_1 и i_k.

Если в последовательности узлов $\{i_1, i_2, \ldots, i_k\}$ этой цепи нет одинаковых, то цепь будем называть элементарной (простой).

Выберем направление движения вдоль цепи. Если это направление совпадает с направлением $i \to j$ дуги (i,j), соответствующей ребру $\{i,j\}$, то дуга (i,j) называется *прямой*. Дуга с противоположным направлением называется *обратной*.

Сеть S называется *связной*, если любые два ее узла можно соединить цепью.

В дальнейшем будем предполагать, что сеть S связная, ибо в противном случае исходная транспортная задача (2.2) распадается на две независимые транспортные задачи меньшей размерности.

Цепь (2.3) с совпадающими узлами i_1 и i_k называется *циклом.*

Лемма 2 *Связанная цепь $S = \{I, U\}$ без циклов либо содержит висячее ребро, либо $|I| = 1$, $|U| = 0$.*

Лемма 3 *Пусть сеть $S = \{I, U\}$ связная. Удаление висячего ребра вместе с висячим узлом или ребра из цикла не нарушает связности сети.*

Определение 9 *Связная сеть $S = \{I, U\}$ называется деревом, если $|I| = |U| + 1$.*

Лемма 4 *Связная сеть S является деревом тогда и только тогда, когда она не содержит циклов.*

Лемма 5 *Каждая пара узлов дерева связана единственной цепью.*

Определение 10 *Для сети $S = \{I, U\}$ сеть $S^* = \{I, U^*\}$, где $U^* \subset U$, называется частичной сетью.*

Определение 11 *Частичная сеть, являющаяся деревом, называется деревом сети.*

Лемма 6 *Пусть $S^* = \{I, U^*\}$ — дерево сети. При любой дуге $(i, j) \in U \backslash U^*$ частичная сеть $S_1 = \{I, U^* \cup (i, j)\}$ содержит ровно один цикл.*

Перейдем к основному вопросу пункта. В симплекс-методе в качестве базиса используется полная система линейно независимых векторов из множества векторов $A_j \in \mathbb{R}^m, j \in J = \{1, ..., n\}$.

Напомним: в симплекс-методе, при условии, что $\operatorname{rank} A = m$, множество $J_Б \subset J$, $|J_Б| = m$, является базисным, если $\det A_Б \neq 0$, $A_Б = (A_j, j \in J_Б)$, т.е. векторы условий $A_j, j \in J_Б$, образуют полную систему линейно независимых векторов в системе векторов $A_j, j \in J$. По определению это означает следующее:

1. уравнение $\sum\limits_{j \in J_Б} A_j x_j = 0$ имеет только нулевое решение $x_j = 0, j \in J_Б$,

2. при $\forall j_0 \in J \setminus J_Б$, система $\sum\limits_{j \in J_Б \cup j_0} A_j x_j = 0$ имеет ненулевое решение $x_j \not\equiv 0,\ j \in J_Б \cup j_0$.

При определении базисного множества $J_Б$ первым способом (т.е. с помощью условия $\det A_Б \neq 0$) мы обязательно должны предположить, что $rank A = m$. При определении базиса вторым способом (т.е. через полную систему линейно независимых векторов) нам не надо предполагать, что $rank A = m$. При использовании этого способа может быть, что $rank A < m$. Последний случай реализуется в транспортной задаче.

Конечно, можно перейти к эквивалентной задаче с $\bar{A}x = \bar{b}$, $\bar{A} \in \mathbb{R}^{\bar{m} \times n}$, $\bar{m} < m$, $rank \bar{A} = \bar{m}$. Но для транспортной задачи этот путь нецелесообразен, так как нарушается специальная структура матрицы A. Поэтому в транспортной задаче при введении базиса воспользуемся вторым способом определения базиса.

Определение 12 *Множество дуг $U_Б \subset U$ сети $S = \{I, U\}$ называется полным, если система*

$$\sum_{j \in I_i^+(U_Б)} x_{ij} - \sum_{j \in I_i^-(U_Б)} x_{ji} = 0,\ i \in I; \tag{2.4}$$

имеет только нулевое решение $x_{ij} = 0,\ j \in U_Б$, а для любой дуги $(i_0, j_0) \in U \setminus U_Б$ система

$$\sum_{j \in I_i^+(U_Б \cup (i_0, j_0))} x_{ij} - \sum_{j \in I_i^-(U_Б \cup (i_0, j_0))} x_{ji} = 0,\ i \in I, \tag{2.5}$$

имеет ненулевое решение $x_{ij} \not\equiv 0$, $(i,j) \in U_Б \cup (i_0, j_0)$.

Введем необходимые понятия.

Совокупность $x = \{x_{ij}, (i,j) \in U\}$, для которой в любом узле $i \in I$ выполняются условия баланса (2.1) назовем псевдопотоком.

Лемма 7 *Сеть с нулевыми интенсивностями узлов ($a_i = 0,\ i \in I$) допускает бесконечное число псевдопотоков, если в ней имеется цикл.*

Доказательство. Положим $x_{ij} = 0$, если ребро $\{i, j\}$ не входит в цикл. В цикле выберем направление обхода по некоторой дуге (i_0, j_0) и положим $x_{ij} = \theta$, если (i, j) — прямая дуга цикла, $x_{ij} = -\theta$, если (i, j) обратная дуга цикла. При любом θ построенная совокупность удовлетворяет системе (2.1) с $a_i = 0,\ i \in I$.

Псевдопоток, построенный при доказательстве леммы 7, называется (i_0, j_0) - *циркуляцией* со значением θ.

Теорема 14 *(Критерий полноты множества дуг). В сети $S = \{I, U\}$ множество $U_Б \subset U$ является полным тогда и только тогда, когда $S_Б = \{I, U_Б\}$ — дерево сети.*

Доказательство. **Достаточность.** Пусть $S_Б$ — дерево. Покажем, что система (2.4) имеет только нулевое решение. В сети $S_Б$ найдем висячий узел (он существует в силу леммы 2) и соответствующее висячее ребро. Из условия баланса (2.4) следует, что псевдопоток по этому ребру может быть только **нулевым**. Удалим этот висячий узел и соответсвующее ребро и рассмотрим оставшуюся сеть, которая согласно **лемме 3** будет опять связной и не содержать циклов. Поступим с ней как с исходной. Через $|I| - 1$ шагов убеждаемся, что псевдопоток на сети $S_Б$ может быть только нулевым.

Если к $S_Б$ добавить дугу $(i_0, j_0) \in U \backslash U_Б$, то согласно **лемме 6** в сети $S_2 = \{I, U_Б \cup (i_0, j_0)\}$ будет цикл, содержащий дугу (i_0, j_0). Из **леммы 7** следует, что существует ненулевое решение системы (2.5). Следовательно, согласно определению, множество дуг $U_Б$ — полная система.

Необходимость. Пусть $U_Б$ — полная система множества дуг сети $S = \{I, U\}$. Совокупность $S_Б = \{I, U_Б\}$ не может содержать циклов, ибо в противном случае, согласно **лемме 7**, система (2.4) имела бы ненулевое решение.

Сеть $S_Б$ — связная, ибо в противном случае существовала бы такая дуга $(i_0, j_0) \in U \setminus U_Б$, что сеть $S_2 = \{I, U_Б \cup (i_0, j_0)\}$ не содержит циклов и следовательно для нее система (2.5) может иметь только нулевое решение, что противоречит полноте $U_Б$. Значит, $S_Б$ — связная сеть без циклов. Следовательно, она является деревом.

Определение 13 *Поток $x = \{x_{ij}, (i, j) \in U\}$ называется базисным с базисом $U_Б \subset U$, если $x_{ij} = 0,\ (i, j) \in U_Н,\ U_Н = U \backslash U_Б$, и $U_Б$ — полное множество дуг сети $S = \{I, U\}$.*

Множество $U_Б$ *называется множество базисных дуг, множество* $U_Н = U \setminus U_Б$ *— множеством небазисных дуг.*

Дуговые потоки x_{ij}, $(i,j) \in U_Б$ *назовем базисными дуговыми потоками, а дуговые потоки* x_{ij}, $(i,j) \in U_Н$ *— небазисными дуговыми потоками.*

Определение 14 *Базисный поток* $\{x, U_Б\}$ *называется невырожденным, если*

$$x_{ij} > 0,\ (i,j) \in U_Б.$$

3. Формула приращения стоимости потока

Пусть $\{x, U_\text{Б}\}$ — базисный поток, x — поток, $U_\text{Б} \subset U$ — множество базисных дуг. Рассмотрим другой (необязательно базисный поток) $\overline{x} = x + \Delta x$. Найдем формулу для приращения

$$\sum_{(i,j)\in U} c_{ij}\overline{x}_{ij} - \sum_{(i,j)\in U} c_{ij}x_{ij} = \sum_{(i,j)\in U} c_{ij}\Delta x_{ij} \tag{2.6}$$

стоимости потока.

Каждому узлу $i \in I$ сети $S = \{I, U\}$ применим число u_i (потенциал узла), так чтобы совокупность u_i, $i \in I$, удовлетворяла системе уравнений

$$0 = u_i - u_j - c_{ij}, \qquad (i,j) \in U_\text{Б}. \tag{2.7}$$

Покажем, что искомые числа существуют. Выберем любой узел $i_1 \in I$, положим $u_{i_1} = 0$. Согласно **лемме 5** каждый узел $i \in I$ можно соединить с i_1 единственной цепью $\{i_1, i_2, \ldots, i_s, i\}$ из дуг дерева $S_\text{Б} = \{I, U_\text{Б}\}$. Рассматривая уравнение (2.7) вдоль (от i_1 до i) дуг этой цепи, определим потенциал u_i узла i по рекуррентным правилам: Например, зная u_{i_k}, находим $u_{i_{k+1}}$ следующим образом

$$u_{i_{k+1}} = \begin{cases} u_{i_k} - c_{i_k i_{k+1}}, & \text{если } (i_k, i_{k+1}) \text{ — прямая дуга,} \\ u_{i_k} + c_{i_{k+1} i_k}, & \text{если } (i_{k+1}, i_k) \text{ — обратная дуга,} \end{cases}$$

$$k = 1, 2, ..., s, s+1,\ i_{s+1} = i.$$

Имея потенциалы узлов, найдем оценки небазисных дуг по правилу

$$\Delta_{ij} = u_i - u_j - c_{ij},\ \ (i,j) \in U_\text{Н}. \tag{2.8}$$

Так как по предположению x и $\overline{x}$ — потоки, то

$$\sum_{j\in I_i^+}\Delta x_{ij}-\sum_{j\in I_i^-}\Delta x_{ij}=0,\quad i\in I. \tag{2.9}$$

Умножим обе части равенств (2.7) , (2.8) на соответствующие Δx_{ij} и просуммируем по $(i,j)\in U$. В результате получим

$$\sum_{(i,j)\in U}c_{ij}\Delta x_{ij}=-\sum_{(i,j)\in U_{\text{H}}}\Delta_{ij}\Delta x_{ij}+\sum_{(i,j)\in U}(u_i-u_j)\Delta x_{ij}=$$

$$=-\sum_{(i,j)\in U_{\text{H}}}\Delta_{ij}\Delta x_{ij}+\underbrace{\sum_{i\in I}u_i(\sum_{j\in I_i^+}\Delta x_{ij}-\sum_{j\in I_i^-}\Delta x_{ij})}_{=0\text{ в силу }(2.9)}=$$

$$=-\sum_{(i,j)\in U_{\text{H}}}\Delta_{ij}\Delta x_{ij}.$$

Таким образом, мы получили следующую формулу приращения:

$$\sum_{(i,j)\in U}c_{ij}\Delta x_{ij}=-\sum_{(i,j)\in U_{\text{H}}}\Delta_{ij}\Delta x_{ij}. \tag{2.10}$$

4. Критерий оптимальности базисного потока

Пусть x — базисный поток, $U_{\text{Б}}$ — соответствующее базисное множество дуг, а u_i, $i\in I$ — соответствующие потенциалы (2.7) и Δ_{ij}, $(i,j)\in U_{\text{H}}$ — соответствующие оценки (2.8).

Теорема 15 *(Критерий оптимальности) Неравенства*

$$\Delta_{ij}\le 0,\qquad (i,j)\in U_{н}, \tag{2.11}$$

достаточны, а в случае невырожденности необходимы для оптимальности базисного потока x.

Доказательство. **Достаточность.** Наряду с заданным базисным потоком $(x_{ij},(i,j)\in U)$ рассмотрим любой другой допустимый поток $(\bar{x}_{ij},(i,j)\in U)$. Поскольку по построению $x_{ij}\equiv 0$, $j\in U_{\text{H}}$; $\overline{x}_{ij}\ge 0$, $(i,j)\in U_{\text{H}}$, то

$$\Delta x_{ij}=\overline{x}_{ij}-x_{ij}\ge 0,\qquad (i,j)\in U_{\text{H}}. \tag{2.12}$$

Из соотношений (2.11), (2.12) в (2.10) следует, что

$$\sum_{(i,j)\in U} c_{ij}\Delta x_{ij} \geq 0.$$

Отсюда с учетом (2.6) получаем неравенство

$$\sum_{(i,j)\in U} c_{ij}\overline{x}_{ij} \geq \sum_{(i,j)\in U} c_{ij}x_{ij},$$

справедливое для любого допустимого потока $\bar{x}$. Следовательно, x — оптимальный поток.

Необходимость. Пусть x — оптимальный невырожденный базисный поток, т.е.

$$x_{ij} > 0, \qquad (i,j) \in U_{\text{Б}}. \tag{2.13}$$

Предположим, что соотношения (2.11) нарушаются, т. е. существует дуга $(i_0, j_0) \in U_{\text{Н}}$ такая, что

$$\Delta_{i_0 j_0} > 0 \tag{2.14}$$

Построим специальное приращение потока $\Delta x = (\Delta x_{ij}, (i,j) \in U)$ следующим образом.

Определим небазисные приращения дуговых потоков по правилу

$$\Delta x_{ij} = \begin{cases} \theta \geq 0, & \text{если } (i,j) = (i_0, j_0)\ , \\ \quad 0, & \text{если } (i,j) \neq (i_0, j_0)\ , \end{cases} \quad (i,j) \in U_{\text{Н}}. \tag{2.15}$$

Остальные компоненты Δx_{ij} , $(i,j) \in U_{\text{Б}}$, найдем из условий

$$\sum_{j\in I_i^+(U)} \Delta x_{ij} - \sum_{j\in I_i^-(U)} \Delta x_{ji} = 0, \qquad i \in I;$$

которые с учетом (2.15) принимают вид

$$\sum_{j\in I_i^+(U_{\text{Б}})} \Delta x_{ij} - \sum_{j\in I_i^-(U_{\text{Б}})} \Delta x_{ji} = \begin{cases} -\theta, & \text{если } i = i_0\ , \\ \quad\theta, & \text{если } i = j_0\ , \\ \quad 0, & \text{если } i \neq j_0, i_0\ , \end{cases} \quad i \in I. \tag{2.16}$$

Найдем решение системы (2.16) , которая определена на сети $S_1 = \{I, U_{\text{Б}} \cup (i_0, j_0)\}$. Добавление дуги (i_0, j_0) к дереву $S_{\text{Б}} = \{I, U_{\text{Б}}\}$ приводит к появлению единственного цикла (**лемма 6**), содержащего дугу (i_0, j_0). Значение дугового потока $\Delta x_{i_0 j_0}$ на дуге (i_0, j_0) задано (см. (2.15)) : $\Delta x_{i_0 j_0} = \theta$. Тогда для выполнения (2.16) необходимо и достаточно, чтобы:

1. $\Delta x_{ij} = \theta$, если (i, j) — прямая дуга, принадлежащая циклу;

   $\Delta x_{ij} = -\theta$, если (i, j) — обратная дуга, принадлежащая циклу.

   В цикле направление обхода определяется дугой (i_0, j_0).

2. $\Delta x_{ij} = 0$, если $(i, j) \in U_{\text{Б}}$, но (i, j) не принадлежит циклу.

Следовательно, мы построили (i_0, j_0)-циркуляцию со значением θ.

Наложим построенную циркуляцию на базисный поток x. В результате получим новый поток с дуговыми потоками:

$$\overline{x}_{i_0 j_0} = x_{i_0 j_0} + \Delta x_{i_0 j_0} = \theta \geq 0,$$

$$\overline{x}_{ij} = x_{ij}, \qquad (i, j) \in U_{\text{Н}} \backslash (i_0, j_0);$$

$$\overline{x}_{ij} = x_{ij} + \Delta x_{ij} = \begin{cases} x_{ij}, & \text{если } (i, j) \in U_{\text{Б}}\ ,\ (i, j) \notin \text{циклу}, \\ x_{ij} \pm \theta, & \text{если } (i, j) \in U_{\text{Б}}\ ,\ (i, j) \in \text{циклу}. \end{cases}$$

Из (2.13) следует, что при достаточно малых $\theta > 0$ верно неравенство: $\overline{x}_{ij} = x_{ij} + \Delta x_{ij} \geq 0$, $(i, j) \in U_{\text{Б}}$. Следовательно, $\overline{x}$ — поток на сети S при достаточно малых $\theta > 0$. Из формулы приращения получаем

$$\sum_{(i,j) \in U} c_{ij} \overline{x}_{ij} - \sum_{(i,j) \in U} c_{ij} x_{ij} = - \sum_{(i,j) \in U_{\text{Н}}} \Delta_{ij} \Delta x_{ij} = -\Delta_{i_0 j_0} \theta < 0.$$

Однако это противоречит оптимальности потока x. Следовательно, неравенства (2.11) имеют место. Теорема доказана.

Таким образом, проверка критерия оптимальности сводится к следующим операциям:

- по правилам (2.7), используя множество $U_{\text{Б}}$, находим потенциалы узлов $u_i, i \in I$;

- зная потенциалы узлов, находим оценки Δ_{ij}, $(i, j) \in U_{\text{Н}}$, небазисных дуг согласно (2.8);

- проверяем неравенства (2.11) .

5. Достаточное условие существования потока с неограниченной снизу стоимостью

Предположим, что имеет место неравенство:

$$\Delta_{i_0 j_0} > 0, \qquad (i_0, j_0) \in U_\text{н}$$

и все дуги в цикле сети $S_1 = \{I, U_\text{Б} \cup (i_0, j_0)\}$ имеют то же направление, что и дуга (i_0, j_0), т. е. являются прямыми. Из доказательства критерия оптимальности следует, что в этом случае имеют место неравенства

$$\Delta x_{ij} \geq 0, \qquad (i, j) \in U.$$

Следовательно, при лубом $\theta \geq 0$ совокупность $\overline{x} = x + \Delta x$ будет потоком на сети S и

$$\sum_{(i,j)\in U} c_{ij}\overline{x}_{ij} - \sum_{(i,j)\in U} c_{ij}x_{ij} = -\Delta_{i_0 j_0}\theta < 0.$$

Следовательно, целевая функция (стоимость потока $\overline{x}$) неограниченно убывает при $\theta \to \infty$. Мы доказали следующую теорему.

Теорема 16 *Если существует дуга $(i_0, j_0) \in U_\text{н}$ такая, что $\Delta_{i_0 j_0} > 0$ и все дуги цикла сети $S_1 = \{I, U_\text{Б} \cup (i_0, j_0)\}$ являются прямыми, то существует поток со сколь угодно малой стоимостью (т. е. задача (2.2) не имеет решения).*

6. Итерация

Рассмотрим последний случай, когда соотношения (2.11) нарушаются, т. е. имеет место неравенство (2.14), но в цикле сети $S_1 = \{I, U_\text{Б} \cup (i_0, j_0)\}$ не все дуги прямые. Ясно, что в этом случае на поток x можно накладывать только циркуляцию с конечным значением θ, чтобы $\overline{x}$ было потоком.

Поскольку с увеличением θ стоимость потока $\overline{x}$ уменьшается, то мы заинтересованы выбрать максимальное значение $\theta \geq 0$, при котором совокупность $\overline{x}$ является потоком.

Из формулы

$$\overline{x}_{ij} = x_{ij} + \Delta x_{ij} = x_{ij} - \theta,$$

которая верна для обратных дуг цикла, следует, что максимально допустимое значение параметра θ равно

$$\theta_0 = \theta_{i_* j_*} = x_{i_* j_*} = \min x_{ij}, \qquad (i,j) \in U^-_{\text{цикл}},$$

где $U^-_{\text{цикл}}$– обратные дуги цикла сети $S_1 = \{I, U_{\text{Б}} \cup (i_0, j_0)\}$.

Поток x заменим на поток $\overline{x}$, что согласно описанным действиям сводится к следующему: потоки на прямых дугах цикла сети S_1 увеличиваем на θ_0, потоки на обратных дугах цикла сети S_1 уменьшаем на величину θ_0. Потоки на остальных дугах оставляем без изменения. При этом стоимость потока уменьшается на величину $\theta_0 \Delta_{i_0 j_0}$. Очевидно, что $\theta_0 > 0$, если x — невырожденный базисный поток.

Покажем, что поток $\overline{x}$ является базисным. Для построения нового базисного множества удалим дугу $(i_*, j_*) \in U_{\text{Б}}$ из $U_{\text{Б}}$ (по построению $\overline{x}_{i_* j_*} = 0$), дугу $(i_0, j_0) \in U_{\text{Н}}$, (где $\overline{x}_{i_0 j_0} = \theta_0$) добавляем ко множеству $U_{\text{Б}}$. Покажем, что сеть $\overline{S}_{\text{Б}} = \{I, \overline{U}_{\text{Б}}\}$, где $\overline{U}_{\text{Б}} = (U_{\text{Б}} \backslash (i_*, j_*)) \cup (i_0, j_0)$, — дерево. Действительно, сеть $S_1 = \{I, U_{\text{Б}} \cup (i_0, j_0)\}$ содержит единственный цикл, кроме того $|I| = |U_{\text{Б}}| + 1$. По построению, дуга $(i_*, j_*) \in U_{\text{Б}}$, $(i_*, j_*) \neq (i_0, j_0)$ и (i_*, j_*) принадлежит циклу. Следовательно, удаление дуги (i_*, j_*) разрушает цикл, но не разрушает связности сети, и мы приходим к сети $\overline{S}_{\text{Б}} = \{I, \overline{U}_{\text{Б}}\}$, $\overline{U}_{\text{Б}} = (U_{\text{Б}} \backslash (i_0, j_0)) \cup (i_*, j_*)$, которая является связной и $|I| = |\overline{U}_{\text{Б}}| + 1$. По определению, $\overline{S}_{\text{Б}}$ — дерево. Следовательно, $\{\overline{x}, \overline{U}_{\text{Б}}\}$ — базисный поток.

Переход $\{x, U_{\text{Б}}\} \to \{\overline{x}, \overline{U}_{\text{Б}}\}$ называется итерацией метода потенциалов.

Метод потенциалов для невырожденных транспортных задач (все базисные потоки — невырожденные) является конечным.

7. Первая фаза метода потенциалов

Для построения начального базисного потока $\{x, U_{\text{Б}}\}$ поступаем следующим образом. К сети $S = \{I, U\}$ добавляем искусственный узел $n+1$ с интенсивностью

$$a_{n+1} = \sum_{i \in I} a_i = 0.$$

Отметим, что условие

$$\sum_{i \in I} a_i = 0$$

является необходимым условием для существования потока на сети $S = \{I, U\}$.

Ко множеству дуг U добавим n искусственных дуг вида:

$$(i, n+1), \quad \text{если } i \text{ — источник или нейтральный узел } (a_i \geq 0)\ , \quad i \in I;$$
$$(n+1, i), \quad \text{если } i \text{ — сток } (a_i < 0)\ , \quad i \in I.$$

Множество добавленных искусственных дуг обозначим через $U_{\text{И}}$.

Рассмотрим расширенную сеть $\bar{S} = \{\bar{I}, \bar{U}\}$, $\bar{I} = I \cup \{n+1\}$, $\bar{U} = U \bigcup U_{\text{И}}$.

В расширенной сети $\bar{S} = \{\bar{I}, \bar{U}\}$ множество искусственных дуг $U_{\text{И}}$ является базисным. Определим соответствующий ему базисный поток по правилу:
дуговые потоки вдоль базисных (искусственных) дуг положим равными абсолютным значениям интенсивностей узлов $i \in I$ исходной сети, которым эти дуги инцидентны, остальные дуговые потоки положим равными $x_{ij} = 0, (i, j) \in \bar{U} \setminus U_{\text{И}} = U$.

На расширенной сети $\bar{S} = \{\bar{I}, \bar{U}\}$ рассмотрим следующую задачу о потоке минимальной стоимости

$$\sum_{(i,j)\in U_{\text{И}}} x_{ij} \to \min,$$

$$\sum_{j\in I_i^+(\bar{U})} x_{ij} - \sum_{j\in I_i^-(\bar{U})} x_{ji} = a_i,\ i \in \bar{I}, \tag{2.17}$$

$$x_{ij} \geq 0,\ (i, j) \in \bar{U}.$$

Задача (2.17) и есть задача первой фазы.

Для задачи первой фазы начальный базисный поток уже построен. Целевая функция ограничена снизу:

$$\sum_{(i,j)\in U_{\text{И}}} x_{ij} \geq 0.$$

Следовательно, задача первой фазы имеет решение. Это решение можно найти методом потенциалов.

Пусть $x^* = \{x^*_{ij}, (i, j) \in U \cup U_{\text{И}}\}$ оптимальный базисный поток для задачи первой фазы с базисом $U^*_{\text{Б}} \subset U \cup U_{\text{И}}$. Проанализируем полученное решение.

1. Если $x^*_{ij} \not\equiv 0, (i, j) \in U_{\text{И}}$, то исходная задача не имеет решения.

2. Пусть $x^*_{ij} \equiv 0, (i, j) \in U_{\text{И}}$, и базисное множество дуг $U^*_{\text{Б}}$ содержит единственную искусственную дугу. Удалив эту дугу из $U^*_{\text{Б}}$, получим базисное множество $U_{\text{Б}}$ и соответствующий базисный поток $x = \{x^*_{ij}, (i, j) \in U\}$ для исходной сети.

Процесс решения задачи (2.2) продолжаем стандартным методом потенциалов (вторая фаза), исходя из построенного начального базисного потока.

3. Пусть $x^*_{ij} \equiv 0$, $(i,j) \in U_{\text{И}}$, но базисное множество дуг $U^*_{\text{Б}}$ содержит более одной искусственной дуги. Тогда среди небазисных дуг $(i,j) \in U \backslash U^*_{\text{Б}}$ всегда найдется такая дуга (i_*, j_*), что цикл, построенный из базисных дуг $U^*_{\text{Б}}$ и дуги (i_*, j_*), содержит две искусственных дуги (предполагается, что сеть $S = \{I, U\}$– связная). Одну из этих искусственных дуг выводим из множества базисных дуг и вместо нее вводим дугу (i_*, j_*). Через конечное число шагов получаем базис, содержащий только одну искусственную дугу, т. е. приходим к случаю 2).

Важное свойство. Если все интенсивности a_i , $i \in I$ — целые числа, и стоимости потоков ограничены снизу, то среди оптимальных потоков существует целочисленный поток.

§2. Матричная транспортная задача

1. Постановка задачи. Базисный план перевозок

Рассмотрим задачу о потоке минимальной стоимости на простой сети $S = \{I, U\}$, в которой множество узлов I состоит из двух непересекающихся подмножеств I_1 (источников) и I_2 (стоков): $I = I_1 \cup I_2$, $I_1 \cap I_2 = \emptyset$, а множество дуг U — из всевозможных дуг вида (i,j), $i \in I_1$, $j \in I_2$.

Если $|I_1|$, $|I_2|$ — большие, то пользоваться сетевым представлением неудобно. Используют матричную запись. Обозначим $I_1 = \{1, \ldots, m\}$, $I_2 = \{1, \ldots, n\}$. Пусть a_i, $i \in I_1$ — объемы производства, b_i, $i \in I_2$ — объемы потребления, c_{ij} — стоимость перевозки из i в j единицы продукта, x_{ij} — объем перевозки по дуге (i,j).

Условия баланса в узлах $i \in I = I_1 \cup I_2$ принимают вид

$$\sum_{j=1}^{n} x_{ij} = a_i,\ i = \overline{1,m}; \qquad \sum_{i=1}^{m} x_{ij} = b_j,\ j = \overline{1,n}. \tag{2.18}$$

Совокупность чисел $x = (x_{ij}, i = \overline{1,m}, j = \overline{1,n})$, удовлетворяющих (2.18) и неравенствам

$$x_{ij} \geq 0, \qquad i = \overline{1,m}, \qquad j = \overline{1,n}, \tag{2.19}$$

назовем планом перевозок.

Стоимость плана перевозок равна

$$\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}c_{ij} \to \min \tag{2.20}$$

Задача (2.18) - (2.20) — матричная транспортная задача.

Цель данного пункта — перенести методов потенциалов на матричную транспортную задачу. Вначале сформулируем теорему.

Теорема 17 *Для существования плана перевозок в матричной транспортной задаче необходимо и достаточно, чтобы выполнялось условие общего баланса*

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j,\ a_i \geq 0, i = \overline{1,m};\ \ b_j \geq 0,\ \ j = \overline{1,n}. \tag{2.21}$$

Поскольку для любого плана перегрузок x выполняются соотношения

$$0 \leq x_{ij} \leq \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij} = \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j < \infty,\ \ i = \overline{1,m},\ \ j = \overline{1,n},$$

то множество планов перевозок компактно, следовательно, каждая линейная форма ограничена на этом множестве. Следовательно, условия (2.21) являются критерием существования оптимального плана перевозок.

Перейдем к введению базисного множества клеток и базисного плана перевозок. Рассмотрим $m \times n$-матрицу с элементами-клетками

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

$(1,1)$	$(1,2)$	$\dots$	$(1,n)$
$(2,1)$	$(2,2)$	$\dots$	$(2,n)$
$\dots$	$\dots$	$\dots$	$\dots$
$(m,1)$	$(m,2)$	$\dots$	(m,n)

Рис.

Цепью (простой, элементарной), соединяющей клетку (i_1, j_1) с клеткой (i_k, j_k) называются последовательности вида:

$$\{(i_1, j_1),\ \ (i_1, j_2),\ \ (i_2, j_2), ..., (i_k, j_k)\}$$

или

$$\{(i_1, j_1),\ (i_2, j_1),\ (i_2, j_2), ..., (i_k, j_k)\},$$

в которых каждые соседние две клетки лежат в одной строке или в одном столбце, но ни в одной строке и ни в одном столбце нет трех последовательных клеток.

Цикл — это цепь, крайние клетки которой лежат в одной строке или в одном столбце. Если в таблице соседние клетки цепи соединить прямыми линиями (звеньями), то соседние звенья цепи всегда будут перпендикулярными.

Определение 15 *Множество клеток $U_Б \subset U$ называется* **полным***, если $|U_Б| = n + m - 1$ и из его элементов невозможно составить ни одного цикла.*

Определение 16 *План перевозок $x = (x_{ij}, i = \overline{i, m}, j = \overline{1, n})$ называется* **базисным***, если $\exists\ U_Б \subset U$ такое, что*

1) $U_Б$ — полное множество,

2) $x_{ij} = 0,\ (i, j) \in U_Н = U \backslash U_Б$.

Множество $U_Б$ называемся множеством базисных клеток, множество $U_Н$ — множеством небазисных клеток.

Определение 17 *Базисный план перевозок называется* **невырожденным***, если*

$$x_{ij} > 0,\ (i, j) \in U_Б.$$

Свойства базисного множества клеток.

Рассмотрим $m \times n-$матрицу (см. Рис.) и базисное множество клеток $U_Б$. Имеют место следующие свойства.

1. В каждом столбце и в каждой строке есть базисная клетка.

2. Существует строка или столбец, в которой лежит только одна базисная клетка.

3. Добавление к базисному множеству $U_Б$ любой клетки $(i_0, j_0) \in U_Н$ создает единственный цикл во множестве $U_Б \cup (i_0, j_0)$.

4. Каждую пару из строк и столбцов можно соединить единственной цепью из элементов $U_Б$.

5. Если $U_{\text{Б}}$ — базисное множество, то базисный поток восстанавливается следующим образом:

   (a) полагаем $x_{ij} = 0,\ (i, j) \in U_{\text{Н}}$.

   (b) ищем базисную клетку (i_1, j_1), единственную в строке i_1 (в столбце j_1).

   (c) полагаем $x_{i_1 j_1} = a_{i_1}$ $(x_{i_1 j_1} = b_{j_1})$.

   (d) обьем b_{j_1} заменяем на $b_{j_1} - x_{i_1 j_1}$ (a_{i_1} на $a_{i_1} - x_{i_1 j_1}$) и строку i_1 (столбец j_1) исключаем из рассмотрения.

   (e) с уменьшенной таблицей повторяем операции (5a)-(5d).

2. Метод минимального элемента

Для матричной транспортной задачи существует несколько способов построения начального базисного плана перевозок. Приведем один из них, известный как метод минимального элемента

Среди чисел $c_{ij},\ i = \overline{1, m},\ j = \overline{1, n}$, найдем минимальное $c_{i_1 j_1}$. Положим

$$x_{i_1 j_1} = \min\{a_{i_1}, b_{j_1}\}.$$

Если $x_{i_1 j_1} = a_{i_1}$ $(x_{i_1 j_1} = b_{i_1})$, то исключаем из рассмотрения строку i_1 (столбец j_1), а число b_{j_1} (a_{i_1}) заменяем на $b_{j_1} - x_{i_1 j_1}$ $(a_{j_1} - x_{i_1 j_1})$. С уменьшенной таблицей повторяем описанные выше операции.

Через конечное число $p,\ p \leq n + m - 1$, шагов будут вычеркнуты все строки и столбцы нашей матрицы и найдены числа $x_{i_k j_k},\ k = \overline{1, p}$, и множество клеток $U_* := \{(i_k, j_k), k = \overline{1, p}\}$.

Для остальных клеток положим $x_{ij} = 0,\ (i, j) \in U \backslash U_*$. Легко убедиться, что построенная совокупность $x = (x_{ij}, i = \overline{1, m}, j = \overline{1, n})$ — план перевозок, а клетки множества U_* не содержат циклов.

Если $p = m + n - 1$, то множество U_* будет полным. Следовательно, $\{x, U_{\text{Б}}\}$, где $U_{\text{Б}} = U_*$, — базисный план перевозок.

Если $p < m + n - 1$, то множество U_* необходимо дополнить $p_0 := m + n - 1 - p$ клетками до полного множества. Это можно сделать следующим образом.

Положим $U_*^0 = U_*$.

Пусть для $k, 0 \le k < p_0$, найдено множество U_*^k, в котором $p+k$ клеток, которые не содержат цикла. Найдем клетку (она обязательно найдется) $(i_*, j_*) \in U \setminus U_*^k$, такую, что множество $U_*^k \cup (i_*, j_*)$ не содержит циклов. Полагаем $U_*^{k+1} = U_*^k \cup (i_*, j_*)$, заменяем k на $k+1$ и повторяем описанные выше операции.

Очевидно, что через p_0 шагов будет построено множество $U_*^{p_0}$, в котором $m+n-1$ клеток и эти клетки не образуют циклов. Следовательно, $\{x, U_{\text{Б}}\}$, где $U_{\text{Б}} = U_*^{p_0}$, — базисный план перевозок.

3. Метод потенциалов для матричной транспортной задачи

Пусть $\{x,\, U_{\text{Б}}\}$ — базисный план перевозок. Общая итерация метода потенциалов для матричной транспортной задачи состоит из следующих шагов.

1. Некоторой произвольной строке i_1 (столбцу j_1) припишем потенциал u_{i_1} (v_{j_1}).

   Нетрудно показать, что уравнение потенциалов

   $$u_i + v_j = c_{ij},\ (i,j) \in U_{\text{Б}},$$

   позволяет найти (однозначно) остальные потенциалы, $u_i,\ i = \overline{1,m},\ v_j,\ j = \overline{1,n}$.

2. Зная потенциалы $u_i,\ i = \overline{1,m},\ v_j,\ j = \overline{1,n}$, вычислим оценки

   $$\Delta_{ij} = u_i + v_j - c_{ij},\ (i,j) \in U_{\text{Н}}.$$

3. Неравенства

   $$\Delta_{ij} \le 0,\ (i,j) \in U_{\text{Н}}, \tag{2.22}$$

   достаточны, а в случае невырожденности и необходимы для оптимальности базисного плана перевозок $\{x, U_{\text{Б}}\}$.

   Если (2.22) выполняется, то $STOP : x^0 = x$ — оптимальный план перевозок.

4. Предположим, что (2.22) не выполняется. Найдем $(i_0, j_0) \in U_{\text{Н}}$, такую что

   $$\Delta_{i_0 j_0} > 0.$$

   Поскольку в матричной транспортной задаче целевая функция ограничена снизу, то в матричной транспортной задаче не может реализоваться ситуация, когда целевая функция $\to$ -∞.

5. С помощью клетки $(i_0 j_0)$ и $U_{\text{Б}}$ строим единственный цикл. Обозначим через $U_{\text{цик}}$ клетки этого цикла. Обойдем этот цикл, начиная горизонтальное движение с клетки (i_0, j_0).

   Среди перевозок, лежащих на концах горизонтальных звеньев этого цикла (аналог обратных дуг) выберем наименьшую:

   $$\theta^0 = \theta_{i_* j_*} = x_{i_* j_*} = \min x_{ij}, \ (i,j) \in U_{\text{цик}}^{\text{гр}}.$$

   Здесь $U_{\text{цик}}^{\text{гр}}$ — клетки из $U_{\text{цик}}$, лежащие на концах горизантальных звеньев этого цикла.

   Для невырожденного базисного плана перевозок $\theta^0 > 0$.

6. Строим новый план перевозок по следующим правилам.

   (a) К перевозкам, лежащим на концах вертикальных звеньев из $U_{\text{цикл}}$ добавляем число θ^0;

   (b) от перевозок, лежащих на концах горизонтальных звеньев , вычитаем число θ^0;

   (c) остальные перевозки не меняем.

7. Новому плану перевозок приписываем новый базис

   $$\overline{U}_{\text{Б}} = (U_{\text{Б}} \backslash (i_*, j_*)) \cup (i_0, j_0).$$

Новый план перевозок и множество $\overline{U}_{\text{Б}}$ образуют новый базисный план перевозок. Исходя из него переходим к новой итерации.

Глава 3

Задача квадратичного программирования

В данной главе рассматривается задача квадратичного программирования (ЛП). Интерес к таким задачам можно объяснить, по крайней мере, двумя факторами. Во-первых, квадратичные модели широко распространены в приложениях и представляют самостоятельный интерес. Во-вторых, во многих методах нелинейного программирования направление спуска на каждой итерации определяется как решение задачи квадратичного программирования.

§1. Постановка задачи. Условия оптимальности

1. Постановка задачи. Определения

Задача квадратичного программирования в канонической форме состоит в минимизации квадратичной функции

$$f(x) := c'x + \frac{1}{2}x'Dx \to \min \tag{3.1}$$

при линейных ограничениях

$$Ax = b, \; x \geq 0 \tag{3.2}$$

Здесь $A = (A_j, j \in J)$ — $m \times n$-матрица со столбцами A_j, $j \in J = \{1, 2, \dots, n\}$, x и c — n-векторы, b — m-вектор, D — симметричная ($D = D'$) положительно полуопределенная ($D \geq 0$) $n \times n$-матрица.

Напомним, что матрица $D \in \mathbb{R}^{n \times n}$ называется **положительно полуопределенной**, если $l'Dl \geq 0$ для любого $l \in \mathbb{R}^n$.

В дальнейшем элементы матрицы D будем обозначать через d_{ij}, т.е.

$$D = \begin{pmatrix} d_{ij}, j \in J \\ i \in J \end{pmatrix}.$$

В задачах квадратичного программирования, как и задачах линейного программирования, вместо линейных ограничений в канонической форме (3.2) можно рассматривать ограничения более общего вида. При этом, используя приемы, описанные при рассмотрении задач линейного программирования, эти ограничения всегда можно свести к ограничениям в канонической форме (3.2).

Без ограничения общности будем считать, что $rank\ A = m \leq n$.

Вектор $x = (x_j, j \in J)$, удовлетворяющий всем ограничениям задачи (3.1), (3.2), назовем **планом**.

Множество индексов $J_{\text{оп}} \subset J$ назовем **опорой ограничений**, если

$$|J_{\text{оп}}| = m, \ det A_{\text{оп}} \neq 0, \ \text{где } A_{\text{оп}} = (A_j, j \in J_{\text{оп}}). \tag{3.3}$$

Пара $\{x, J_{\text{оп}}\}$ из плана и опоры ограничений называется **опорным планом**.

Опорный план $\{x, J_{\text{оп}}\}$ называется невырожденным, если $x_j > 0, \ j \in J_{\text{оп}}$.

Замечание 5 *Если $x_j = 0, \ j \in J \backslash J_{\text{оп}}$, то опорный план является базисным, а невырожденный опорный план — невырожденным базисным планом.*

Рассмотрим опорный план $\{x, J_{\text{оп}}\}$. Обозначим

$$\overline{c}(x) = (\overline{c}_j(x), \ j \in J) = c + Dx.$$

Найдем m-вектор потенциалов $u(x)$

$$u'(x) = -\overline{c}'_{\text{оп}}(x) A^{-1}_{\text{оп}}, \ \text{где } \overline{c}_{\text{оп}}(x) = (\overline{c}_j(x), j \in J_{\text{оп}}), \tag{3.4}$$

и оценки

$$\Delta_j(x) = u'(x) A_j + \overline{c}_j(x), \ j \in J.$$

Отметим, что, по построению, $\Delta_j(x) = 0, \ j \in J_{\text{оп}}$.

2. Формула приращения

Пусть $\{x, J_{\text{оп}}\}$ — опорный план. Рассмотрим произвольный план $\overline{x} = x + \Delta x$. Очевидно, что

$$A\Delta x = 0. \tag{3.5}$$

Подсчитаем приращение

$$f(x+\Delta x) - f(x) = c'(x+\Delta x) + \frac{1}{2}(x+\Delta x)'D(x+\Delta x) - c'x - \frac{1}{2}x'Dx =$$

$$= \overline{c}'(x)\Delta x + \frac{1}{2}\Delta x' D\Delta x = \overline{c}'(x)\Delta x + u'(x)A\Delta x + \frac{1}{2}\Delta x' D\Delta x = \tag{3.6}$$

$$= \sum_{j\in J}\Delta_j(x)\Delta x_j + \frac{1}{2}\Delta x' D\Delta x = \sum_{j\in J\setminus J_{\text{оп}}}\Delta_j(x)\Delta x_j + \frac{1}{2}\Delta x' D\Delta x.$$

3. Критерий оптимальности

Теорема 18 *Соотношения*

$$\Delta_j(x) = 0 \textit{ при } x_j > 0,\ \Delta_j(x) \geq 0 \textit{ при } x_j = 0,\ j \in J\setminus J_{\textit{оп}} \tag{3.7}$$

достаточны, а в случае невырожденности и необходимы для оптимальности опорного плана $\{x, J_{\textit{оп}}\}$.

Доказательство. **Достаточность.** Пусть соотношения (3.7) выполняются. Рассмотрим любой допустимый план $\overline{x} = x + \Delta x$. Из (3.7) и допустимости плана $\bar{x}$ следует, что

$$\Delta_j(x)\Delta x_j \geq 0, j \in J \setminus J_{\text{оп}}.$$

Из условия $D\dot{\geq}0$ следует, что $\Delta x' D\Delta x \geq 0$. С учетом последних соотношений из (3.6) получаем:

$$f(x+\Delta x) - f(x) \geq 0.$$

Следовательно, x — оптимальный план.

Необходимость. Предположим противное: $\{x, J_{\text{оп}}\}$ — невырожденный оптимальный опорный план, но соотношения (3.7) нарушаются. Из этого следует, что существует такой индекс $j_0 \in J\setminus J_{\text{оп}}$, что

$$\Delta_{j_0} < 0 \text{ либо } \Delta_{j_0} > 0 \text{ и } x_{j_0} > 0.$$

Построим направление $l = (l_j, j \in J)$ по правилам:

$$l_{j_0} = -\text{sign}\Delta_{j_0},\ l_j = 0,\ j \in J \backslash (J_{\text{оп}} \cup j_0); \tag{3.8}$$

$$l_{\text{оп}} = -A_{\text{оп}}^{-1} A_{j_0} l_{j_0}.$$

Новый план строим в виде $\overline{x} = x + \theta l$, где $\theta \geq 0$ — некоторый параметр.

Согласно формулам приращения имеем:

$$f(x + \theta l) - f(x) = \theta \Delta_{j_0} l_{j_0} + \frac{1}{2}\theta^2 l' D l = -\theta |\Delta_{j_0}| + \frac{1}{2}\theta^2 l' D l. \tag{3.9}$$

Ясно, что при достаточно малых $\theta > 0$ имеет место неравенство:

$$f(x + \theta l) - f(x) < 0. \tag{3.10}$$

При выборе $\theta > 0$ мы ещё должны учесть прямые ограничения:

$$x_j + \theta l_j \geq 0,\ j \in J. \tag{3.11}$$

Из условий $x_j > 0,\ j \in J_{\text{оп}}$, и условий $x_{j_0} > 0$ при $l_{j_0} < 0$ следует, что найдется такое число $\theta_0 > 0$, что соотношения (3.11) верны при $\forall \theta \in (0, \theta_0)$.

Из (3.8), (3.10), (3.11) следует, что при достаточно малых $\theta > 0$ вектор $x + \theta l$ является планом и на нем значение целевой функции лучше, чем на оптимальном плане x. Это противоречит оптимальности плана x. Необходимость доказана.

4. Достаточное условие неограниченности снизу целевой функции

Пусть для опорного плана $\{x, J_{\text{оп}}\}$ существует индекс $j_0 \in J \backslash J_{\text{оп}}$ такой, что $\Delta_{j_0} < 0$ и для вектора l, построенного по правилам (3.8), выполняются соотношения

$$l \geq 0,\ l' D l = 0.$$

Тогда целевая функция задачи (3.1) неограниченно убывает на множестве планов вида $\bar{x} = x + \theta l$, $\theta \to \infty$.

В линейном программировании (ЛП) далее шло утверждение о том, что если критерий оптимальности и достаточное условие неограниченности не выполняется для данного базисного плана, то мы можем заменить этот план на другой базисный план с нехудшим значением целевой функции.

Для квадратичного программирования можно доказать аналогичное утверждение, заменяя базисный план на опорный. Это утверждение очевидным образом следует из критерия оптимальности, т.к. в невырожденном случае, согласно определению, пара $\{x + \theta l, J_{\text{оп}}\}$ является опорным планом при достаточно малых $\theta > 0$.

Однако, даже если начальный план $\{x, J_{\text{оп}}\}$ был базисным $\{x, J_{\text{оп}}\} = \{x, J_{\text{Б}}\}$, то и в этом случае мы не сможем доказать, что базисный план $\{x, J_{\text{Б}}\}$ можно заменить на новый **базисный** план с нехудшим значением целевой функции!

Это вызвано тем, что среди оптимальных планов задачи квадратичного программирования может не существовать базисных.

Покажем из-за чего возникает такая ситуация.

Предположим, что есть начальный базисный план $\{x, J_{\text{Б}}\}$, $J_{\text{Б}} = J_{on}$, критерий оптимальности и достаточное условие неограниченности целевой функции не выполняется для индекса j_0. Отметим, что в данном случае (т.е. в случае, когда опорный план является базисным) обязательно будет $\Delta_{j_0} < 0$. Построим направление l по правилам (8) и новый план $\bar{x}$ будем искать в виде $\bar{x} = x + \theta l$, где $\theta \geq 0$. Как и в линейном программировании, шаг θ должен быть допустимым по прямым ограничениям, т.е. должны выполняться неравенства $x + \theta l \geq 0$. Из этого следует, что

$$\theta \leq \theta_0 = \min_{j \in J_{\text{Б}} \cup j_0} \theta_j = \theta_{j_*} \tag{3.12}$$

где

$$\theta_j = \begin{cases} \infty, \text{ если } l_j \geq 0, \\ -x_j/l_j, \text{ если } l_j < 0, \end{cases} \quad j \in J_{\text{Б}} \cup j_0.$$

Проанализируем целевую функцию

$$f(x + \theta l) = f(x) + \theta\alpha + \frac{1}{2}\theta^2\delta,$$

$$\alpha = -|\Delta_{j_0}|, \quad \delta = l'Dl \geq 0.$$

Подсчитаем число

$$\theta_\delta = \begin{cases} \infty, & \text{если } l'Dl = 0, \\ |\Delta_{j_0}|/l'Dl, & \text{если } l'Dl > 0. \end{cases}$$

Очевидно, что при $\theta \in [0, \theta_\delta)$ функция $f(x+\theta l)$ убывает, а при $\theta > \theta_\delta$ эта функция начинает возрастать (см. Рис....).

Следовательно, максимально допустимый шаг вдоль l нужно выбрать по правилу

$$\theta_* = \min\{\theta_0, \theta_\delta\}. \tag{3.13}$$

Если $\theta_* = \theta_0 = \theta_{j_*}$, где $j_* \in J_Б$, то мы можем начальный базисный план $\{x, J_Б\}$ заменить на новый базисный план $\{\bar{x}, \bar{J}_Б\}$, где $\bar{x} = x + \theta_* l$, $\bar{J}_Б = (J_Б \setminus j_*) \cup j_0$.

Если $\theta_0 = \theta_\delta$, то мы не можем начальный базисный план $\{x, J_Б\}$ заменить на лучший базисный план, так как при любом $\theta \in (0, \theta_*]$ у плана $\bar{x} = x + \theta l$ число ненулевых компонент равно $m+1 > m$ и, следовательно, ему нельзя приписать никакой базис. Таким образом, наши надежды на то, что задачу квадратичного программирования можно решить, используя только базисные планы, оказались необоснованными.

§2. Алгоритмы решения общей задачи квадратичного программирования

1. "Примитивный"алгоритм квадратичного программирования

Из описанных выше рассуждений можно получить "примитивный"алгоритм квадратичного программирования, который является достаточно простым и позволяет заменять текущий опорный план на новый опорный план с нехудшим значением целевой функции. Однако этом алгоритм не во всех случаях позволет получить решение исходной задачи. Генерируемые им планы могут даже в пределе не сходиться к оптимальному плану. "Примитивный"алгоритм можно использовать в качестве предварительного этапа с целью получения хорошего начального приближения.

Опишем этот алгоритм по шагам.

1. Есть начальный опорный план $\{x, J_{оп}\}$. Строим потенциалы $u(x)$ и оценки $\Delta_j(x)$, $j \in J \setminus J_{оп}$, по правилам (3.3).

2. Проверяем критерий оптимальности. Если он выполняется, то STOP: x — оптимальный план. В противном случае находим $j_0 \in J \setminus J_{оп}$, где нарушается критерий оптимальности.

3. Строим направление l по правилам (3.8) и находим шаг θ_* по правилам (3.12), (3.13). Если $\theta_* = \infty$, то STOP: целевая функция неограниченна на множестве планов. Пусть $\theta_* < \infty$.

4. Строим новый план $\bar{x} = x + \theta_* l$. Строим новую опору

$$\bar{J}_{\text{оп}} = J_{\text{оп}}, \text{ если } \theta_* = \theta_\delta \text{ либо } \theta_* = \theta_{j_0} < \theta_\delta,$$

$$\bar{J}_{\text{оп}} = (J_{\text{оп}} \backslash j_*) \cup j_0, \text{ если } \theta_* = \theta_{j_*}, j_* \in J_{\text{оп}}.$$

Идем на шаг 1, используя построенный опорный план $\{\bar{x}, \bar{J}_{\text{оп}}\}$.

Недостатком данного алгоритма является то, что он может оказаться не сходящимся. Это происходит из-за того, что возникает последовательность итераций вида

$$\{x^1, J_{\text{ОП}}\}, \{x^2, J_{\text{оп}}\}, \ldots, \{x^n, J_{\text{оп}}\}, \ldots$$

где $x_j^i = x_j^1$, $j \in J \backslash (J_{\text{оп}} \cup j_1 \cup j_2)$, $f(x^{i+1}) < f(x^i)$, $f(x^{i+1}) - f(x^i) \to 0$ при $i \to \infty$, однако $x^i \not\to x^0$ при $i \to \infty$. Чтобы этого не происходило, описанный выше "примитивный"алгоритм дополняют специальными операциям. Эти операции описаны в следующем пункте.

2. Конечный метод решения задачи квадратичного программирования

Для описания метода нам понадобится понятие правильного опорного плана.

Определение 18 *Опорный план $\{x, J_{on}\}$ называется правильным, если существует такое подмножество $J_* \subset J$, что*

$$J_{on} \subset J_* \subset \{j \in J : \Delta_j(x) = 0\},$$
$$x_j = 0, j \in J_H = J \backslash J_*,$$
$$\det H_* \neq 0,$$
$$H_* = \begin{pmatrix} D_*, & A'_* \\ A_*, & 0 \end{pmatrix}.$$

Здесь $A_* = (A_j, j \in J_*)$,

$$D_* = \begin{pmatrix} d_{ij}, \ j \in J_* \\ i \in J_* \end{pmatrix}.$$

Оказывается, что среди оптимальных планов задачи квадратичного программирования всегда существует оптимальный правильный опорный план.

Замечание 6 *Базисный план* $\{x, J_Б\}$ *— частный случай правильного опорного плана, где* $J_Б = J_{оп} = J_*$.

Опишем конечный алгоритм по шагам.

1. Задан начальный правильный опорный план $\{x, J_{\text{оп}}, J_*\}$. Строим потенциалы $u'(x) = -\bar{c}'_{\text{оп}}(x)A^{-1}_{\text{оп}}$, где $\bar{c}(x) = c + Dx$.

   Вычисляем оценки:

$$\Delta_j = \Delta_j(x) = u'(x)A_j + \bar{c}_j(x), j \in J_{\text{н}} = J \backslash J_*;$$

2. Проверяем критерий оптимальности:

$$\Delta_j \geq 0, j \in J \backslash J_*.$$

   Если он выполняется, то STOP: x — оптимальный план. В противном случае, идем на шаг 3, используя индекс $j_0 \in J \backslash J_*$, для которого $\Delta_{j_0} < 0$.

3. Построение направления l изменения плана. Положим

$$l_j = 0, j \in J_{\text{н}} \backslash j_0,\ l_{j_0} = 1. \tag{3.14}$$

   Остальные компоненты $l_* = (l_j, j \in J_*)$ выберем некоторым образом (пока не уточняем каким) так, чтобы $Al = 0$. Последнее равенство с учётом (3.14) принимает вид:

$$A_* l_* + A_{j_0} = 0. \tag{3.15}$$

   При этом согласно формуле приращения имеем

$$f(x + \theta l) = f(x) - \theta|\Delta_{j_0}| + \frac{1}{2}\underbrace{l'Dl}_{\delta}\theta^2$$

   где число $\alpha = -|\Delta_{j_0}|$ не зависит от выбора l_*, а число $\delta = l'Dl$ зависит от выбора l_*. Нам выгоднее, чтобы δ было по возможности минимальным. Т.е. при заданных компонентах $l_j, j \in J_{\text{н}}$, (см. (3.14)) мы должны найти компоненты l_* таким образом, чтобы имело место равенство (3.15) и

$$l'Dl = l'_* D_* l_* + d_{j_0 j_0} + 2D'_{*j_0} l_* \to \min_{l_*}. \tag{3.16}$$

Здесь D_{*j_0} – $|J_*|$-вектор с компонентами

$$D_{*j_0} = (d_{jj_0}, j \in J_*)'.$$

Нетрудно показать, что если мы решим систему

$$\begin{cases} D_* l_* + A'_* y + D_{*j_0} = 0, \\ A_* l_* \qquad\qquad + A_{j_0} = 0, \end{cases} \tag{3.17}$$

относительно неизвестных $(l_*, y), y \in \mathbb{R}^m$, то l_* и будет искомым решением задачи (3.16), (3.15). Отметим, что система (3.17) имеет единственное решение в силу условия $\det H_* \neq 0$.

Таким образом, мы находим l_* из системы (3.17) и направление l полностью построено.

4. Подсчитаем шаги $\theta_j, j \in J_* \cup j_0$, по правилам

$$\theta_j = \begin{cases} \infty, \text{ если } \; l_j \geq 0, \; j \in J_*; \\ -x_j/l_j, \text{ если } \; l_j < 0, \; j \in J_*; \end{cases}$$

$$\theta_{j_0} = \theta_\delta = \begin{cases} \infty, \text{ если } \; \delta = 0; \\ |\Delta_{j_0}| \backslash \delta, \text{ если } \; \delta > 0; \end{cases} \text{ где } \delta = l'Dl = D'_{*j_0} l_* + A'_{j_0} y + d_{j_0 j_0}.$$

Находим $\theta_0 = \min \theta_j, j \in J_* \cup j_0$. Обоснование этих правил выбора шага такое же как и в „примитивном“ алгоритме.

Если $\theta_0 = \infty$, то STOP: целевая функция неограничена снизу на множестве планов. Пусть $\theta_0 = \theta_{j_*} < \infty$, где $j_* \in J_* \cup j_0$.

5. Строим новый план $\bar{x} = x + \theta_0 l$.

6. Строим новые множества $\bar{J}_{\text{оп}}$ и $\bar{J}_*$. Предполагается, что множество индексов $J_{\text{оп}}$ имеет вид $J_{\text{оп}} = \{j_1, j_2, ..., j_s, ..., j_m\}$. Возможны случаи:

   (a) $j_* = j_0$;

   (b) $j_* \in J_* \backslash J_{\text{оп}}$;

   (c) $j_* = j_s \in J_{\text{оп}}$ и существует такой индекс $j_+ \in J_* \backslash J_{\text{оп}}$, что $e'_s A_{\text{оп}}^{-1} A_{j_+} \neq 0$;

   (d) $j_* = j_s \in J_{\text{оп}}$ и $e'_s A_{\text{оп}}^{-1} A_j = 0$, $j \in J_* \backslash J_{\text{оп}}$, либо $J_* = J_{\text{оп}}$.

В случае (6a) полагаем

$$\bar{J}_{\text{оп}} = J_{\text{оп}}, \quad \bar{J}_* = J_* \cup j_0$$

и переходим к новой итерации, т.е. к шагу 1, используя найденные данные $\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*$.

В случае (6b) полагаем

$$\bar{J}_{\text{оп}} = J_{\text{оп}}, \bar{J}_* = J_* \setminus j_*, \bar{j}_0 = j_0, \bar{\Delta}_{j_0} = \Delta_{j_0} + \theta_0 \delta$$

и переходим к шагу 3, используя далее данные $\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*, \bar{j}_0, \bar{\Delta}_{\bar{j}_0}$ вместо $x, J_{\text{оп}}, J_*, j_0, \Delta_{j_0}$.

В случае (6c) полагаем

$$\bar{J}_{\text{оп}} = (J_{\text{оп}} \setminus j_*) \cup j_+, \bar{J}_* = J_* \setminus j_*, \bar{j}_0 = j_0, \bar{\Delta}_{j_0} = \Delta_{j_0} + \theta_0 \delta$$

и переходим к шагу 3, используя обновлённые данные $\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*, \bar{j}_0, \bar{\Delta}_{\bar{j}_0}$ вместо $x, J_{\text{оп}}, J_*, j_0, \Delta_{j_0}$.

В случае (6d) полагаем

$$\bar{J}_{\text{оп}} = (J_{\text{оп}} \setminus j_*) \cup j_0, \bar{J}_* = (J_* \setminus j_*) \cup j_0$$

и переходим к новой итерации (т.е. к шагу 1), используя новые данные $\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*$.

Замечание 7 *Построение матрицы $\bar{A}_{on}^{-1}$, обратной к новой матрице $\bar{A}_{on} = (A_j, j \in \bar{J}_{on})$, осуществляется по правилам, описанным на шаге 6 симплекс-метода, с использованием матрицы $\bar{A}_{on}^{-1}$.*

По аналогии с ЛП решение системы (2.8) можно строить с помощью матрицы H_*^{-1}, обратной к матрице

$$H_* = \begin{pmatrix} D_* & A'_* \\ A_* & 0 \end{pmatrix} \in \mathbb{R}^{(k+m)*(k+m)},$$

по правилу

$$\begin{pmatrix} l_* \\ y \end{pmatrix} = -H_*^{-1} h_{j_0},$$

где $h_{j_0} = \begin{pmatrix} D_{*j_0} \\ A_{j_0} \end{pmatrix} \in \mathbb{R}^{k+m}, k = |J_*|$.

При этом нетрудно получить простые формулы построения новой матрицы $\bar{H}_*^{-1}$, зная H_*^{-1}. Действительно, матрица H_* полностью задается множеством индексов J_*. В результате итерации $\{x, J_{\text{оп}}, J_*\} \to \{\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*\}$ множество J_* может измениться по одному из следующих правил:

a $\bar{J}_* = J_* \cup j_0$;

b $\bar{J}_* = J_* \backslash j_*$;

c $\bar{J}_* = (J_* \backslash j_*) \cup j_0$.

Пусть для определённости $J_* = \{j_1, j_2, \ldots, j_k\}, k \geq m, j_* = j_s, s \leq k$. Тогда матрица $\bar{H}_*^{-1}$, обратная к матрице

$$\bar{H}_* = \begin{pmatrix} \bar{D}_* & \bar{A}_* \\ \bar{A}_* & 0 \end{pmatrix},$$

где $\bar{A}_* = (A_j, j \in \bar{J}_*)$, $\bar{D}_* = (D_{ij}, j \in \bar{J}_*, i \in \bar{J}_*)$, строиться следующим образом.

В случае **a** полагаем

$$\bar{H}_*^{-1} = \begin{pmatrix} \tilde{H}_* & \bar{h}_*/\delta \\ \bar{h}'_*/\delta & 1/\delta \end{pmatrix},$$

где $h'_* = (l'_*, y'), \tilde{H}_* = H_*^{-1} + h_* h'_* \backslash \delta$.

В случае **b** матрица $\bar{H}_*^{-1}$ строится по формуле

$$\bar{H}_*^{-1} = \breve{H}_* - \breve{h}_* \breve{h}'_* / \breve{h}_{ss},$$

где матрица $\breve{H}_*$ получается из матрицы $\bar{H}_*^{-1}$ удалением s-ой строки и s-ого столбца; вектор $\breve{h}_*$ получается из s-го столбца матрицы $\bar{H}_*^{-1}$ удалением s-го элемента; $\breve{h}_{ss}$ — (s, s)-й элемент матрицы $\bar{H}_*^{-1}$.

В случай **c** матрица $\bar{H}_*^{-1}$ находится по формуле

$$\bar{H}_*^{-1} = M H_*^{-1} \bar{M}',$$

где $M = (e_1, \ldots, e_{s-1}, d_s, e_{s+1}, \ldots, e_{k+m})$; $\bar{M} = (e_1, \ldots, e_{s-1}, \bar{d}_s, e_{s+1}, \ldots, e_{k+m})$; $e_i \in \mathbb{R}^{k+m}$ — единичный вектор с единицей на i-ом месте; $d_s = e_s + (e_s - h_*)/\alpha$, $\bar{d}_s = e_s + (e_s - M h_*)/\bar{a}$, $\alpha = e'_s h_*$, $\bar{a} = e'_s M h_*$.

Итерацию $\{x, J_{\text{оп}}, J_*\} \to \{\bar{x}, \bar{J}_{\text{оп}}, \bar{J}_*\}$ назовём невырожденной, если $\theta_0 > 0$.

Теорема 19 *Описанный выше алгоритм решает задачу(3.1), (3.2) за конечное число итераций, если в процессе его реализации встречается конечное число вырожденных итераций.*

При аналогии с подразделом 1.5 можно ввести в алгоритм дополнительные правила определения индексов $j_0 \in J_{\text{н}}, j_* \in J_* \cup j_0$, гарантирующие конечность описанного алгоритма без требования невырожденности итераций.

Как видно из описания алгоритма, для начала его работы необходимо знать начальный правильный опорный план. Выше отмечалось, что любой базисный план является правильным опорным планом. Следовательно, для построения информации, необходимой для начала работы данного алгоритма, можно воспользоваться первой фазой симплекс-метода, описанной в подразделе 1.6. При этом, как и в задаче линейного программирования, можно отказаться от предположения о том, что $rankA = m \leq n$, ибо при реализации первой фазы будут обнаружены и удалены все линейно зависимые основные ограничения.

Замечание 8 *При* $D = 0$ *задача квадратичного программирования (3.1), (3.2) совпадает с задачей линейного программирования (3.6) из раздела 1.*

Глава 4

Выпуклое программирование

Выпуклым программированием (ВП) называется раздел математики, в котором исследуется задача

$$f(x) \to \min, x \in X, \tag{4.1}$$

где $f(x)$ выпуклая функция, заданная на выпуклом множестве $X \subset \mathbb{R}^n$.

При формировании основной задачи выпуклого программирования (4.1) принято множество X задавать в форме

$$X = \{x : g(x) \leq 0, x \in Q\},$$

где $g(x) = (g_i(x), i = \overline{1,m}) \in \mathbb{R}^m$, $g_i(x), i = \overline{1,m}$, — выпуклые функции, Q — выпуклое множество постой структуры.

§1. Выпуклые функции и множества

Определение 19 *Множество $X \subset \mathbb{R}^n$ называется выпуклым, если для $\forall x_1 \in X$, $\forall x_2 \in X$ имеет место включение*

$$x(\lambda) = \lambda x_1 + (1-\lambda)x_2 \in X \text{ при } \forall \lambda \in [0,1].$$

Пример: выпуклые множества из $\mathbb{R}^2$ приведены на Рис. ...., невыпуклые множества из $\mathbb{R}^2$ приведены на Рис.....

Определение 20 *Функция $f(x)$, определённая и конечная на выпуклом множестве X, называется выпуклой, если для $\forall x_1, x_2 \in X$ и при всех $\lambda \in [0,1]$ выполняются неравенства*

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2).$$

1. Если $f(x) \in C^{(1)}$ (т.е. $f(x)$ — гладкая), то $f(x), x \in \mathbb{R}^n$, выпукла тогда и только тогда, когда при всех $x, x_* \in \mathbb{R}^n$

$$f(x) - f(x^*) \geq (x - x^*)' \partial f(x^*)/\partial x. \tag{4.2}$$

2. Если $f(x) \in C^{(2)}, x \in \mathbb{R}^n$, то $f(x), x \in \mathbb{R}^n$ выпукла тогда и только тогда, когда матрица $\partial^2 f(x)/\partial x^2$ неотрицательно определенная на $x \in \mathbb{R}^n$.

Напомним, что симметричная $n \times n$-матрица A называется неотрицательной (положительной), если

$$x'Ax \geq 0 \ \ (x'Ax > 0) \quad \text{при } \forall x \in \mathbb{R}^n, x \neq 0.$$

§2. Теорема Куна-Таккера

Теоремой Куна-Таккера в выпуклом программировании называется основной результат — критерий оптимальности, сформулированный в терминах седловой точки функции Лагранжа.

1. Седловая точка функции Лагранжа и решение основной задачи выпуклого программирования

Основная задача выпуклого программирования имеет вид

$$f(x) \to \min, \ g(x) \leq 0, x \in Q, \tag{4.3}$$

где $Q \subset \mathbb{R}^n$ — выпуклое множество, часто $Q = \{x \in \mathbb{R}^n : x \geq 0\}, \ g(x) = (g_1(x), \ldots, g_m(x))$, $f(x), g_1(x), \ldots, g_m(x)$ — выпуклые функции.

Каждый n-вектор x, удовлетворяющий ограничениям задачи (4.3), называется **планом** (допустимым планом, допустимой точкой). Решение x^0 задачи (4.3)

$$f(x^0) = \min f(x), g(x) \leq 0, x \in Q$$

называется **оптимальным планом**.

Из свойств выпуклых множеств и функций следует, что как множество планов

$$X = \{x \in \mathbb{R}^n : g(x) \leq 0, x \in Q\},$$

так и множество оптимальных планов

$$X^0 = \{x : f(x) = f(x^0), x \in X\}$$

являются выпуклыми.

По элементам $f(x)$, $g(x)$ задачи (4.3) составим **функцию Лагранжа**

$$F(x, \lambda) = f(x) + \lambda' g(x) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) \tag{4.4}$$

и рассмотрим ее при

$$x \in Q, \lambda \geq 0, \lambda = (\lambda_i, i = \overline{1, m}) \in \mathbb{R}^m.$$

Определение 21 *Говорят, что пара* $\{x^*, \lambda^*\}$, $\lambda^* \geq 0$, $x^* \in Q$, *—* **седловая точка** *функции Лагранжа (4.4), если для всех* $x \in Q$, $\lambda \geq 0$ *выполняются неравенства*

$$F(x^*, \lambda) \leq F(x^*, \lambda^*) \leq F(x, \lambda^*). \tag{4.5}$$

Теорема 20 *Если* $\{x^*, \lambda^*\}$, $x^* \in Q$, $\lambda^* \geq 0$, *— седловая точка функции Лагранжа (4.5), то* x^* *— оптимальный план задачи (4.3) и выполняются* **условия дополняющей нежесткости**

$$g'(x^*)\lambda^* = 0 \tag{4.6}$$

$$(\Rightarrow g_i(x^*)\lambda_i^* = 0, i = \overline{1, m}).$$

Доказательство. Запишем (4.5) в исходных функциях

$$f(x^*) + \lambda' g(x^*) \leq f(x^*) + g'(x^*)\lambda^* \leq f(x) + g'(x)\lambda^*, x \in Q, \lambda \geq 0. \tag{4.7}$$

Из левого неравенства в (4.7) следует, что

$$\lambda' g(x^*) \leq {\lambda^*}' g(x^*), \forall \lambda \geq 0. \tag{4.8}$$

Для выполнения (4.8) необходимо выполнение неравенства

$$g(x^*) \leq 0,$$

ибо в противном случае (если $\exists \tau_0 \in \{1, \ldots, m\} : g_{i_0}(x^*) > 0$), выбирая λ_{i_0} сколь угодно большим, можно сделать левую часть (4.8) сколь угодно большой. Таким образом, мы показали, что x^* — план задачи (4.3).

Из (4.8) следуют условия дополняющей нежесткости (4.6). Действительно, если допустить, что $g'(x^*)\lambda^* = \alpha < 0$, то, положив $\lambda = \lambda^*/2$, в (4.8) получим противоречие: $\alpha/2 \leq \alpha < 0$. (Допустить, что $g'(x^*)\lambda^* > 0$ мы не можем, так как $\lambda^* \geq 0, g(x^*) \leq 0$.)

С учетом (4.6) правое неравенство в (4.7) переходит в неравенство

$$f(x^*) \leq f(x) + g'(x)\lambda^*. \tag{4.9}$$

Так как $\lambda^* \geq 0$ и для любого **плана** задачи (4.3) верно неравенство $g(x) \leq 0$, то (4.9) можно переписать в виде

$$f(x^*) \leq f(x) \text{ для любого плана } x \text{ задачи (4.3)} .$$

Следовательно, x^* — оптимальный план задачи (4.3). ⊠

Согласно Теореме 20 для построения оптимального плана задачи (4.3) **достаточно** найди седловую точку функции Лагранжа. Однако не для каждой задачи (4.3) функция Лагранжа имеет седловую точку.

Первые результаты, эквивалентные теореме существования седловой точки функции Лагранжа для гладких задач были получены Куном Г. и Таккером А.

2. Гладкие задачи

Рассмотрим **гладкую** задачу выпуклого программирования, под которой понимается задача (4.3), в которой функции $f(x)$ и $g(x)$ — гладкие и множество Q имеет вид

$$Q = \{x \in \mathbb{R}^n : x \geq 0\}.$$

Определение 22 *Говорят, что множество планов X (ограничения) основной задачи выпуклого программирования* **регулярно** *(удовлетворяет* **условию Слейтера***), если существует такой план $\overline{x}$, что*

$$g(\overline{x}) < 0.$$

Пусть $I = \{1, \ldots, m\}$, $I_0(x) = \{i \in I : g_i(x) = 0\}$ — множество индексов **ограничений**, **активных** на плане x, $J = \{1, \ldots, n\}$, $J_0(x) = \{j \in J : x_j = 0\}$, $J_+(x) = J \backslash J_0(x)$ — множества индексов нулевых и положительных компонент плана x.

Лемма 8 *Предположим, что множество X планов гладкой задачи (4.3) удовлетворяет условию Слейтера, т.е. существует такой план $\bar{x} \in X$, что*

$$g(\bar{x}) < 0. \tag{4.10}$$

Тогда для любого плана $x^0 \in X$ и любого вектора $l^* \in \mathbb{R}^n$, удовлетворяющих системе

$$l'^* \partial g_i(x^0)/\partial x \leq 0,\ i \in I_0(x^0),\ l_j^* \geq 0,\ j \in J_0(x^0), \tag{4.11}$$

существует такое число $\alpha_0 > 0$, что для любого $\alpha \in (0, \alpha_0]$ найдется такое число $t_0(\alpha) > 0$, что при всех $t \in [0, t_0(\alpha)]$ вектор

$$x(t) = x^0 + l^* t + \alpha(\bar{x} - x^0)t \tag{4.12}$$

является планом задачи (4.3).

Теорема 21 (Необходимое условие оптимальности в прямой форме) *Пусть x^0 — оптимальный план гладкой задачи (4.3) с регулярным множеством планов. Тогда для каждого вектора l^*, удовлетворяющего системе (4.11), выполняется неравенство*

$$l^{*\prime} \partial f(x^0)/\partial x \geq 0. \tag{4.13}$$

Доказательство. Предположим, что вектор l^* удовлетворяет (4.11), но

$$l^{*\prime} \partial f(x^0)/\partial x < 0. \tag{4.14}$$

Согласно лемме 8 функция (4.12) при $t \in [0, t_0],\ t_0 > 0$, удовлетворяет ограничениям задачи (4.3). При достаточно малых $\alpha > 0$ в силу (4.14) выполняется неравенство

$$\frac{df(x(t))}{dt}\,|_{t=0} = \frac{\partial f'(x(t))}{\partial x}\frac{dx}{dt}\,|_{t=0} = (l^* + \alpha(\bar{x} - x^0))' \frac{\partial f(x^0)}{\partial x} < 0,$$

следовательно, при достаточно малых $\alpha > 0$ верно неравенство $f(x(t)) < f(x^0)$, которое противоречит оптимальности x^0. Следовательно, наше предположение было неверным. Теорема доказана.

Теорема 21 содержит прямое необходимое условие оптимальности в том смысле, что позволяет в случае неоптимальности плана x^0 "улучшить"его, т.е. построить такой план $\tilde{x}$, что $f(\tilde{x}) < f(x^0)$.

Действительно, проверка условий теоремы 21 сводится к решению задачи линейного программирования вида

$$l'\partial f(x^0)/\partial x \to \min_{l},$$

$$l'\partial g_i(x^0)/\partial x \leq 0,\ i \in I_0(x^0), \tag{4.15}$$

$$l_j \geq 0,\ j \in J_0(x^0),$$

с каким-либо нормировочным условием, например,

$$|l_j| \leq 1,\ j \in J, \tag{4.16}$$

для исключения неограниченных решений l^*.

Пусть l^* — решение задачи (4.15), (4.16).

Если $l^{*\prime}\partial f(x^0)/\partial x = 0$, то необходимое условие оптимальности (4.13) выполняется.

Если $l^{*\prime}\partial f(x^0)/\partial x < 0$, то по формуле (4.12) строится план $\tilde{x} = x(t)$, для которого $f(\tilde{x}) < f(x^0)$.

Таким образом мы показали, что оптимальность плана $l^* = 0$ в задаче (4.15), (4.16) является необходимым условием оптимальности плана x^0 в задаче (4.3).

Записывая условия оптимальности плана $l^* = 0$ в задаче линейного программирования (4.15), (4.16), мы приходим к следующей форме необходимых условий оптимальности (двойственные необходимые условия оптимальности) плана x^0 в гладкой задаче (4.3).

Теорема 22 (Необходимое условие оптимальности в двойственной форме) *Для оптимальности плана x^0 в гладкой задаче (4.3) с регулярным множеством планов необходимо существование такого неотрицательного m-вектора $\lambda^0 \geq 0$, что выполняются условия*

1. *стационарности:*

$$\frac{\partial F(\lambda^0, x^0)}{\partial x} = \frac{\partial f(x^0)}{\partial x} + \sum_{i=1}^{m} \lambda_i^0 \frac{\partial g_i(x^0)}{\partial x} =: \Delta^0 \geq 0, \tag{4.17}$$

2. дополняющей нежесткости:

$$g'(x^0)\lambda^0 = 0,\ \Delta^{0'} x^0 = 0. \tag{4.18}$$

Лемма 9 *Из соотношений (4.17), (4.18) следует, что пара* $\{\lambda^0, x^0\}$ *— седловая точка функции Лагранжа.*

Доказательство. Нам надо показать, что верны соотношения

$$F(\lambda, x^0) \leq F(\lambda^0, x^0) \leq F(\lambda^0, x) \tag{4.19}$$

при любых $\lambda \geq 0$, $x \geq 0$.

В силу (4.18) соотношения (4.19) принимают вид

$$f(x^0) + \sum_{i=1}^{m} \lambda_i g_i(x^0) \leq f(x^0) \leq f(x) + \sum_{i=1}^{m} \lambda_i^0 g_i(x)$$

По построению,

$$\sum_{i=1}^{m} \lambda_i g_i(x^0) \leq 0,\ \text{следовательно},\ F(\lambda, x^0) \leq F(\lambda^0, x^0).$$

Рассмотрим функцию $F(\lambda^0, x)$, $x \geq 0$. Эта функция является выпуклой, следовательно, для всех $\bar{x} \geq 0$ согласно (4.2) верно

$$F(\lambda^0, \bar{x}) - F(\lambda^0, x^0) \geq (\bar{x} - x^0)\frac{\partial F(\lambda^0, x^0)}{\partial x}. \tag{4.20}$$

Предположим, что второе неравенство в (4.19) не выполняется, т.е. существует $\bar{x} \geq 0$ такой, что

$$F(\lambda^0, \bar{x}) < F(\lambda^0, x^0). \tag{4.21}$$

Из (4.20) и (4.21) получаем

$$\begin{gathered} 0 > F(\lambda^0, \bar{x}) - F(\lambda^0, x^0) \geq (\bar{x} - x^0)' \tfrac{\partial F(\lambda^0, x^0)}{\partial x} = \\ (\bar{x} - x^0)'\Delta^0 = \text{с учетом (4.18)} = \bar{x}'\Delta^0. \end{gathered} \tag{4.22}$$

С другой стороны $\Delta^0 \geq 0$ и $\bar{x} \geq 0$, следовательно, $\bar{x}'\Delta^0 \geq 0$.

Получили противоречие. Следовательно, не существует такого $\bar{x} \geq 0$, что верно (4.21). Значит верно $F(\lambda^0, x) \geq F(\lambda^0, x^0)$ для любого $x \geq 0$.

Таким образом мы показали, что в случае гладкости и регулярности задачи (4.3) для существования решения задачи (4.3) необходимо существование седловой точки Лагранжа.

Объединяя лемму 9 с теоремой 20, приходим к следующей теореме.

Теорема 23 *(Кун-Таккер). Для оптимальности плана $x^0 \in X$ основной (гладкой) задачи выпуклого программирования с регулярным множеством планов необходимо и достаточно существование такого неотрицательного m-вектора $\lambda^0 \geq 0$, что пара $\{x^0, \lambda^0\}$ является седловой точкой функции Лагранжа.*

§3. Теория двойственности

Теорема Куна-Таккера позволяет естественным образом ввести **двойственную** задачу и **доказать соотношение двойственности** для выпуклых задач.

1. Двойственная задача, соотношение двойственности

Рассмотрим задачу выпуклого программирования:

$$f(x) \to \min, g(x) \leq 0, x \in Q \tag{4.23}$$

с **регулярным** множеством планов, то есть $\exists\, x^* \in Q : g(x^*) < 0$.

По элементам $f(x)$ и $g(x)$ сформируем функцию Лагранжа:

$$F(x, \lambda) = f(x) + \lambda' g(x), x \in Q, \lambda \geq 0.$$

Введём **прямую** $\varphi(x)$, $x \in Q$, и **двойственную** $\psi(\lambda)$, $\lambda \geq 0$ функции:

$$\begin{aligned} \varphi(x) &= \sup_{\lambda} F(x, \lambda), \lambda \geq 0; \\ \psi(\lambda) &= \inf_{x} F(x, \lambda), x \in Q. \end{aligned} \tag{4.24}$$

Задачу

$$\varphi(x) \to \min, x \in Q, \tag{4.25}$$

назовём **прямой задачей**. Задачу

$$\psi(\lambda) \to \max, \lambda \geq 0, \tag{4.26}$$

назовём **двойственной задачей** выпуклого программирования.

Множество $\{x \in Q : \varphi(x) < \infty\}$ назовём **множеством прямых планов**, множество $\bigwedge = \{\lambda \geq 0 : \psi(\lambda) > -\infty\}$ — множеством **двойственных планов**.

Можно показать, что $\varphi(x) = f(x)$, если $x \in X$, где $X = \{x \in \mathbb{R}^n : x \in Q, g(x) \leq 0\}$ и $\varphi(x) = \infty$, если $x \in Q$, но $x \notin X$. Из сказанного следует, что множество прямых планов совпадает с множеством планов (4.23), а задача (4.25) — с задачей (4.23)!

В силу этого и задачу (4.23) будем называть прямой задачей выпуклого программирования.

Нетрудно проверить, что в случае линейности функций $f(x)$, $g(x)$ задача (4.26) совпадает с двойственной задачей линейного программирования, которую мы рассматривали ранее.

Решения x^0, λ^0 задач (4.25), (4.26) удовлетворяют следующим **соотношениям двойственности**, которые выражают тесную связь между прямой и двойственной задачами:

1. Для существования решения x^0 прямой задачи необходимо существование решения λ^0 двойственной задачи.

2. На решениях x^0, λ^0 задач (4.25), (4.26) выполняется равенство
$$\varphi(x^0) = \psi(\lambda^0).$$

3. Для любой пары $\{x, \lambda\}$ из прямого и двойственного планов выполняется неравенство
$$\varphi(x) \geq \psi(\lambda).$$

4. Если на некоторой паре $\{x^*, \lambda^*\}$ из прямого и двойственного планов выполняется равенство
$$\varphi(x^*) = \psi(\lambda^*),$$
то x^*, λ^* — решения задач (4.25) и (4.26).

5. Если вдоль некоторой последовательности $\lambda^k, k = 1, 2, \dots$ ($x^k, k = 1, 2, \dots$) двойственных (прямых) планов целевая функция двойственной (прямой) задачи неограниченно возрастает (убывает), то пусто множество прямых (двойственных) планов.

6. На решениях x^0, λ^0 задач (4.25), (4.26) выполняются условия дополняющей нежёсткости:
$$\lambda^{0\prime} g(x^0) = 0.$$

7. Решение λ^0 двойственной задачи (4.26) представляет собой вектор Лагранжа, соответствующий оптимальному плану x^0 задачи (4.23).

8. Для того, чтобы векторы x^0, λ^0 были решениями задач (4.25) и (4.26), необходимо и достаточно, чтобы они составляли седловую точку $\{x^0, \lambda^0\}$ функции Лагранжа.

9. Справедлива теорема о минимаксе:
$$\min_{x \in Q} \max_{\lambda \geq 0} F(x, \lambda) = \max_{\lambda \geq 0} \min_{x \in Q} F(x, \lambda).$$

Идеи и результаты теории двойственности находят разнообразное применение в теории экстремальных задач и в известном смысле характеризуют современный уровень вычислительных методов.

Глава 5

Нелинейное программирование

Нелинейное программирование — это раздел математики, в котором исследуются задачи оптимизации на множествах **конечномерного** пространства. В данной главе мы рассмотрим теоретические вопросы из нелинейного программирования. В следующей главе рассмотрим схемы вычислительных методов нелинейного программирования.

В общей постановке задача нелинейного программирования имеет чрезвычайно простую формулу:

$$f(x) \to \min, x \in X, \tag{5.1}$$

где X — некоторое множество из конечномерного пространства $\mathbb{R}^n$, $f(x)$ — скалярная или векторная функция, определённая на X.

При такой общей постановке задачи не удаётся получить какие-либо интересные и полезные сведения о её оптимальных планах. Поэтому задача (5.1) исследуется при дополнительных предположениях относительно X и $f(x)$. Это порождает разнообразные классы задач, распространённые в приложениях, и допускает соответствующее классу более детальное описание свойств оптимальных планов.

Выделяют следующие классы:

1. Линейное программирование.

2. Квадратичное программирование.

3. Выпуклое программирование.

4. Задачи нелинейного программирования на безусловный экстремум.

5. Задачи нелинейного программирования на условный экстремум с ограничениями типа равенств.

6. Задачи нелинейного программирования на условный экстремум с ограничениями типа неравенств.

7. Задачи векторной оптимизации.

8. Задачи целочисленного программирования.

§1. Задачи на безусловный минимум

Для общей постановки задачи нелинейного программирования универсальный метод поиска оптимального плана состоит в **переборе** значений целевой функции на множестве планов. Однако это метод редко используется даже при наличии современных мощных вычислительных средств (слишком велик перебор). Поэтому экстремальные задачи подвергаются, как правило, предварительному математическому исследованию.

Исследование задач направлено

1. на выявление свойств (необходимых условий оптимальности), которыми обладают оптимальные планы и которые позволяют уменьшить объем перебора;

2. на формирование соотношений (достаточных условий оптимальности), выполнение которых гарантирует оптимальность рассматриваемого плана.

Необходимое условие считается тем сильнее, чем меньше неоптимальных планов ему удовлетворяет.

Достаточное условие тем сильнее, чем шире класс задач, оптимальные планы которых ему удовлетворяют. Самые сильные условия необходимые условия оптимальности это те, которые совпадают с достаточными, образуя **критерий оптимальности**.

В оценке необходимых и достаточных условий ведущую роль играет их **конструктивность**, то есть насколько легко эти условия могут быть проверены.

Исследования по необходимым и достаточным условиям оптимальности развиваются в направлении углубления условий обоих типов с целью их сближения. При этом условию приписывают **порядок** k, если в его формулировке участвуют производные до k-го порядка от функций, входящих в формулировку задачи.

1. Необходимое условие первого порядка. Стационарные точки

Рассмотрим задачу на безусловный минимум

$$f(x) \to \min, x \in \mathbb{R}^n, \tag{5.2}$$

где $f(x)$ — определена и непрерывна на $\mathbb{R}^n$.

Определение 23 *Точка $x^0 \in \mathbb{R}^n$ называется* **оптимальным планом** *(точкой абсолютного или глобального минимума), если*

$$f(x^0) = \min f(x), x \in \mathbb{R}^n.$$

Определение 24 *Точка $x^0 \in \mathbb{R}^n$ называется локально оптимальным планом (точкой относительного или локального минимума), если при некотором $\varepsilon > 0$ выполняются соотношения*

$$f(x^0) = \min f(x), \|x - x^0\| \leq \varepsilon, x \in \mathbb{R}^n.$$

Каждый оптимальный план является локально оптимальным, но не наоборот!

Заметим, что в задачах выпуклого (и, следовательно, линейного) программирования каждый локально оптимальный план является оптимальным.

Обозначим через $\partial f(x)/\partial x$ вектор частных производных:

$$\frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}\right) = \mathrm{grad} f(x).$$

Теорема 24 *На каждом локально оптимальном плане x^0 задачи (5.2) выполняется* **условие стационарности***:*

$$\partial f(x^0)/\partial x = 0, \text{\textit{то есть}}\ \mathrm{grad}\ f(x^0) = 0; \tag{5.3}$$

Доказательство. Предположим противное: $\partial f(x^0)/\partial x \neq 0$. Положим $l = -\frac{\partial f(x^0)}{\partial x} \neq 0$ и рассмотрим планы вида $x(t) = x^0 + t\ell$ при $t \geq 0$. Подсчитаем

$$\left.\frac{df(x(t))}{dt}\right|_{t=0} = \frac{\partial f'(x(t))}{\partial x} \cdot \left.\frac{dx(t)}{dt}\right|_{t=0} = -\frac{\partial' f(x^0)}{\partial x} \cdot \frac{\partial f(x^0)}{\partial x} = -\|\frac{\partial f(x^0)}{\partial x}\|^2 < 0.$$

По построению, $f(x(t))|_{t=0} = f(x^0)$. Из этого следует, что при достаточно малых $t > 0$ имеем $f(x(t)) < f(x^0)$, что противоречит оптимальности x^0.

Приведённое доказательство **конструктивно**, ибо легко проверяется и в случае невыполнения условий (5.3) позволяет перейти к плану с меньшим значением целевой функции.

Это необходимое условие оптимальности положено в основу многих численных методов.

Определение 25 *Решения x уравнения*

$$\mathrm{grad} f(x) = 0 \tag{5.4}$$

называются **стационарными точками** *функции $f(x)$.*

Из этого следует, что оптимальные планы задачи (5.1) находятся среди стационарных точек. Значит, в случае существования решения задачи (5.1) для его построения достаточно найти все стационарные точки и проверить (перебрать) на них значения целевой функции.

Теорема 24 осуществляет редукцию построения локально оптимальных планов задачи (5.1) к решению уравнения (5.4). Этот шаг также часто используется при численном решении.

2. Необходимое условие минимума второго порядка

Пусть теперь $f(x) \in C^{(2)}$. Найдём условие, позволяющее среди стационарных точек функции $f(x)$ отбросить те, которые не могут быть оптимальными планами, тем самым мы сократим перебор.

Теорема 25 *На каждом оптимальном плане x^0 задачи (5.2) матрица вторых производных целевой функции неотрицательно определена:*

$$\frac{\partial^2 f(x^0)}{\partial x^2} \dot{\geq} 0. \tag{5.5}$$

Доказательство. Предположим противное: существует такой $l \in \mathbb{R}^n$, $l \neq 0$, что $l' \frac{\partial^2 f(x^0)}{\partial x^2} l <\ 0$. Рассмотрим множество планов $x(t) = x^0 + tl$ при $t \geq 0$.

Подсчитаем

1. $f(x(t))|_{t=0} = f(x^0)$;
2. $\frac{d}{dt} f(x(t))\Big|_{t=0} = \frac{\partial' f(x^0)}{\partial x} \cdot l\Big|_{t=0} = 0$;
3. $\frac{d^2}{dt^2} f(x(t))\Big|_{t=0} = \frac{d}{dt}\left(\frac{\partial f(x(t))}{\partial x} \cdot \frac{dx(t)}{dt}\right)\Big|_{t=0} = l' \frac{\partial^2 f(x^0)}{\partial x^2} l < 0.$

Из перечисленных пунктов следует, что при достаточно малых $t > 0$ имеет место неравенство $f(x(t)) < f(x^0)$, которое противоречит оптимальности x^0, а значит, верно (5.5).

3. Достаточное условие локального минимума

Идея усиления необходимых условий минимума за счёт привлечения производных высокого порядка ($k > 2$) не получила развития при $n \geq 2$ по двум причинам:

1. резко усложняется проверка условия;
2. существуют задачи, в которых привлечение любой высоты порядка не позволяет удалить из множества стационарных точек все неоптимальные.

Между тем небольшое изменение условий теоремы 25 приводит к **достаточному условию минимума**.

Теорема 26 *Стационарная точка x^* является* **локально** *оптимальной, если матрица вторых производных $\frac{\partial^2 f(x^*)}{\partial x^2}$ является положительно определенной:*

$$\frac{\partial^2 f(x^*)}{\partial x^2} \dot{>} 0.$$

§2. Задача на условный минимум. Ограничения типа равенств

1. Обобщённое правило множителей Лагранжа

Рассмотрим задачу на условный минимум:

$$f(x) \to \min,\ g(x) = 0, \tag{5.6}$$

где $f(x) \in C^{(1)}$, $g(x) = (g_1(x), \ldots, g_m(x))$, $g_i(x) \in C^{(1)}, i = 1, ..., m$.

Определение 26 *Вектор x —план задачи (5.6), если $g(x) = 0$. План x^0 называется* **оптимальным планом** *(точкой условно глобального минимума), если*

$$f(x^0) = \min_x f(x) \text{ при условии, что } g(x) = 0.$$

План x^0 называется **локально оптимальным** *(точкой условно локального минимума), если при некотором $\varepsilon > 0$ верно:*

$$f(x^0) = \min_x f(x) \text{ при условии, что } g(x) = 0, \|x^0 - x\| \leq \varepsilon.$$

Классическим методом исследования задач на условный минимум является **метод множителей Лагранжа**, который состоит в следующем.

По элементам задачи (5.6) составим **обобщённую функцию Лагранжа**

$$F(x, \overline{\lambda}) = \lambda_0 f(x) + \lambda' g(x),$$

где λ_0 — скаляр, $\lambda \in \mathbb{R}^m$, $\overline{\lambda} = (\lambda_0, \lambda) \in \mathbb{R}^{m+1}$ — **обобщённый вектор Лагранжа**.

Теорема 27 *(Обобщённое правило множителей Лагранжа.) Для любого локально оптимального плана x^0 задачи (5.6) существует ненулевой обобщённый вектор Лагранжа $\bar{\lambda}^0 = (\lambda_0^0, \lambda^0) \neq 0$, такой что*

$$\frac{\partial F(x^0, \bar{\lambda}^0)}{\partial x} = 0, \tag{5.7}$$

т.е. x^0 — стационарная точка функции Лагранжа при $\bar{\lambda} = \bar{\lambda}^0$.

Доказательство. Условие (5.7) перепишем в виде

$$\lambda_0^0 \frac{\partial f(x^0)}{\partial x} + \sum_{i=1}^{m} \lambda_i^0 \frac{\partial y_i(x^0)}{\partial x} = 0, \ (\lambda_0^0, \lambda_1^0, ..., \lambda_m^0) \neq 0. \tag{5.8}$$

Условие (5.8) эквивалентно условию о том, что векторы

$$\frac{\partial f(x^0)}{\partial x}, \ \frac{\partial g_i(x^0)}{\partial x}, \ i = \overline{1, m}, \tag{5.9}$$

линейно зависимы.

Если $m \geq n$, то очевидно, что векторы (5.9) линейно зависимы, следовательно, верно (5.7).

Пусть $m < n$. Предположим, что векторы (5.9) линейно независимы. Рассмотрим систему уравнений

$$f(x) = f(x^0) + \beta,\ \ g(x) \equiv 0$$

относительно неизвестных x. В силу того, что векторы (5.9) линейно независимы, из теоремы о неявной функции следует, что существует такая n-вектор-функция $x(\beta)$, $\beta \in [-\beta_0, \beta_0]$, что

$$f(x(\beta)) \equiv f(x^0) + \beta,\ \ g(x(\beta)) \equiv 0 \text{ при } \forall \beta \in [-\beta_0, \beta_0],$$

где $\beta_0 > 0$ — некоторое число.

Значит при $\beta < 0$ имеем

$$f(x(\beta)) < f(x^0),\ \ g(x(\beta)) = 0,$$

что противоречит оптимальности плана x^0. Полученное противоречие доказывает теорему.

При исследовании задач на условный минимум долгое время (со времён Лагранжа) используется классическая функция Лагранжа:

$$F(x, \lambda) = f(x) + \lambda' g(x). \tag{5.10}$$

Для функции Лагранжа (5.10) правило множителей Лагранжа, вообще говоря, неверно.

2. Классическое правило множителей Лагранжа

Выясним, когда при исследовании задачи (5.6) можно использовать классическую функцию Лагранжа (5.10).

Определение 27 *План x^0 называется обыкновенным, если на нем линейно независимы векторы*

$$\frac{\partial g_1(x^0)}{\partial x}, \ldots, \frac{\partial g_m(x^0)}{\partial x}. \tag{5.11}$$

Теорема 28 *(Правило множителей Лагранжа) Если на локально оптимальном плане x^0 задачи (5.6) векторы (5.11) линейно независимы, то найдётся такой (единственный) вектор Лагранжа λ^0, что на паре $\{x^0, \lambda^0\}$ выполняются следующие условия-равенства (условия стационарности функции Лагранжа):*

$$\frac{\partial F(x^0, \lambda^0)}{\partial x} = 0,\ \ \frac{\partial F(x^0, \lambda^0)}{\partial \lambda} = 0. \tag{5.12}$$

Определение 28 *Точку x^* называют условно-стационарной, если существует такой m-вектор λ^*, что пара $\{x^*, \lambda^*\}$ — стационарная точка функции Лагранжа, т.е. имеют место равентсва*

$$\frac{\partial F(x^* \lambda^*)}{\partial x} = 0, \ \frac{\partial F(x^* \lambda^*)}{\partial \lambda} = 0. \tag{5.13}$$

Поиск условно-стационарных точек сводится к решению системы (5.13) из $m+n$ уравнений с $m+n$ неизвестными λ, x.

Таким образом правило множителей Лагранжа сводит решение задачи (5.6) к перебору условно-стационарных точек.

3. Необходимое условие минимума второго порядка. Достаточное условие оптимальности

Рассмотрим задачу (5.6), предполагая, что $f(x), \ g(x) \in C^{(2)}$ и что на оптимальном плане x^0 векторы (5.11) линейно независимы.

Теорема 29 *Пусть x^0 — локально оптимальный план задачи (5.6) и векторы (5.11) линейно независимы, λ^0 — соответствующий плану x^0 вектор множителей Лагранжа, на котором выполняются соотношения (5.12). Тогда квадратичная форма*

$$l' \frac{\partial^2 F(x^0, \lambda^0)}{\partial x^2} l \geq 0$$

неотрицательна на гиперплоскости, заданной уравнениями

$$l' \frac{\partial g_i(x^0)}{\partial x} = 0, \ i = \overline{1, m}.$$

Теорема 30 *Пусть $f(x), \ g(x) \in C^{(2)}$. Для локальной оптимальности в задаче (5.6) условно стационарной точки x^* достаточно, чтобы при соответствующем ей векторе Лагранжа λ^* была положительна квадратичная форма*

$$l' \frac{\partial^2 F(x^*, \lambda^*)}{\partial x^2} l > 0$$

на гиперплоскости

$$l' \frac{\partial g_i(x^*)}{\partial x} = 0, \ i = \overline{1, m}.$$

§3. Минимизация функций при ограничениях типа неравенств

Нелинейное программирование, как современный этап развития теории классических задач на максимум и минимум, оформилось во второй половине XX века в связи с систематическим исследованием экстремальных задач с ограничениями типа неравенств.

1. Необходимые условия оптимальности первого порядка

Под задачей минимизации с ограничениями типа неравенств в данном пункте будем понимать задачу

$$f(x) \to \min,\ g(x) \le 0, \tag{5.14}$$

где $f(x),\ g(x) = (g_1(x), \ldots, g_m(x)) \in C^{(1)}$.

Определение 29 *Вектор x^0 — план, если $g(x^0) \le 0$.*

План x^0 — (глобально) оптимальный, если

$$f(x^0) = \min_x f(x) \text{ при условии, что } g(x) \le 0.$$

План x^0 — локально оптимальный, если существует $\varepsilon > 0$ такое, что $f(x^0) = \min\limits_x f(x)$ при условии, что $g(x) \le 0$, $||x - x^0|| \le \varepsilon$.

Говорят, что на плане x^* ограничение $g_i(x^*) \le 0$ активно (пассивно), если $g_i(x^*) = 0$ $(g_i(x^*) < 0)$.

Обозначим

$$I_a(x^*) = \{i \in \{1, \ldots, m\} :\ g_i(x^*) = 0\}.$$

Для задачи (5.14) обобщённая функция Лагранжа имеет вид

$$F(x, \bar\lambda) = \lambda_0 f(x) + \lambda' g(x),$$

где $\bar\lambda = (\lambda_0, \lambda) \in \mathbb{R}^{m+1}$.

Теорема 31 *(Обобщённое правило множителей Лагранжа.) Для любого локально оптимального плана x^0 задачи (5.14) существует такой ненулевой вектор Лагранжа $\bar\lambda^0 = \{\lambda_0^0, \lambda^0\}$, что выполняются условия:*

1. *неотрицательности:* $\bar{\lambda}^0 \geq 0$;

2. *стационарности:* $\partial F(x^0, \bar{\lambda}^0)/\partial x = 0$;

3. *дополняющей нежесткости:* $g'(x^0)\lambda^0 = 0$.

Определение 30 *Будем говорить, что план x^* обыкновенный, если линейно независимы векторы*

$$\frac{\partial g_i(x^*)}{\partial x_i},\ i \in I_a(x^*).$$

Если план x^0 обыкновенный, то вместо обобщённой функции Лагранжа и обобщённого правила множителей Лагранжа, можно использовать классическую функцию Лагранжа и классическое правило.

Теорема 32 *(Классическое правило множителей Лагранжа.) Для любого обыкновенного локально оптимального плана x^0 задачи (5.14) существует единственный вектор Лагранжа λ^0, на котором выполняются условия:*

1. *неотрицательности:* $\lambda^0 \geq 0$;

2. *стационарности:* $\partial F(x^0, \lambda^0)/\partial x = 0$;

3. *дополняющей нежесткости:* $g'(x^0)\lambda^0 = 0$.

2. Необходимое условие оптимальности второго порядка

В этом пункте предположим, что $f(x),\ g(x) \in C^{(2)}$. Пусть x^0 — обыкновенный оптимальный план. Ограничение $g_i(x) \leq 0$, активное на плане x^0, назовём жёстким (мягким), если $\lambda_i^0 > 0$ ($\lambda_i^0 = 0$). (Здесь мы предполагаем, что x^0 — обыкновенный оптимальный план и, следовательно, ему соответствует единственный вектор Лагранжа λ^0.)

Обозначим

$$I_a^+(x^0) = \{i \in \{1, \dots, m\} :\ g_i(x^0) = 0,\ \lambda_i^0 > 0\};$$

$$I_a^0(x^0) = \{i \in \{1, \dots, m\} :\ g_i(x^0) = 0,\ \lambda_i^0 = 0\}.$$

Теорема 33 *Пусть x^0 — обыкновенный локально оптимальный план задачи (5.14), λ^0 — соответствующий ему вектор Лагранжа. Тогда на множестве векторов l, удовлетворяющих системе*

$$l'\frac{\partial g_i(x^0)}{\partial x} = 0,\ i \in I_a^+(x^0);$$

$$l'\frac{\partial g_i(x^0)}{\partial x} \leq 0,\ i \in I_a^0(x^0),$$

неотрицательна квадратичная форма

$$l'\frac{\partial^2 F(x^0, \lambda^0)}{\partial x^2} l \geq 0.$$

3. Достаточное условие локальной оптимальности

Пусть в задаче (5.6) $f(x)$ и $g(x) \in C^{(2)}$.

Определение 31 *План x^* задачи (5.14) называется условно-стационарным, если для него существует такой m-вектор λ^*, что выполняются соотношения*

$$\partial F(x^*, \lambda^*)/\partial x = 0,\ g'(x^*)\lambda^* = 0,\ \lambda^* \geq 0.$$

Теорема 34 *Для локальной оптимальности условно-стационарного плана x^* в задаче (5.14) достаточно, чтобы неравенство*

$$l'\frac{\partial^2 F(x^*, \lambda^*)}{\partial x^2} l > 0$$

выполнялось на каждом векторе $l \neq 0$, удовлетворяющем системе

$$l'\frac{\partial g_i(x^*)}{\partial x} = 0,\ i \in I_a^+(x^*);$$

$$l'\frac{\partial g_i(x^*)}{\partial x} \leq 0,\ i \in I_a^0(x^*).$$

Глава 6

Вычислительные методы нелинейного программирования

Теория нелинейного программирования (НЛП), занимаясь исследованием важнейших характеристик решений экстремальных задач, позволяющих во многих конкретных случаях получить достаточно полные сведения о результате, служит и фундаментом для построения разнообразных вычислительных методов.

Вычислительные методы НЛП делятся на прямые и непрямые.

Как правило, прямые методы генерируют последовательность векторов

$$x^1, x^2, \dots, x^k, \dots,$$

состоящую из последовательных приближений к решению x^0.

Итерация $x^k \to x^{k+1}$ строится по схеме

$$x^{k+1} = x^k + \theta_k l^k,$$

где l^k — направление, $\theta_k \geq 0$ — шаг.

Методы отличаются друг от друга способами выбора l^k и θ_k.

Методы делятся на точные и приближенные.

В точных методах на каждой итерации план преобразуется в новый план. Приближенные методы преобразуют одну оценку (приближение) плана в другую.

Методы делятся на конечные и итерационные.

В конечных методах — задача решается за конечное число итераций, в итерационных — за бесконечное число итераций.

Итерационный метод называется сходящимся, если генерирует последовательность векторов, которая сходится к оптимальному плану.

Иногда рассматривают сходимость по целевой функции $f(x^k) \to f(x^0)$ или сходимость к условно стационарной точки $x^k \to x^*$.

Качество итераций в итерационном методе оценивается по скорости сходимости. Если верны неравенства

$$\|x^{k+1} - x^0\| \leq q\|x^k - x^0\|, \qquad 0 < q < 1, \qquad k \geq k_0,$$

то говорят о линейной скорости сходимости.

Если $\|x^{k+1} - x^0\| \leq q\|x^k - x^0\|^\alpha, \qquad k \geq k_0$, то говорят, что метод имеет сверхлинейную скорость, если $1 < \alpha < 2$, и квадратичную, если $\alpha = 2$.

Другие важные характеристики методов:

- требуемый объем оперативной памяти;
- устойчивость к ошибкам округления.

§1. Методы безусловной минимизации

1. Аппроксимация функций

Основная идея решения нелинейных задач состоит в аппроксимации задачи, которая основана на аппроксимации элементов этой задачи.

В задаче на безусловный минимум

$$f(x) \to \min, x \in \mathbb{R}^n \tag{6.1}$$

единственным элементом является целевая функция $f(x)$.

Пусть $f(x) \in C^{(1)}$. Функция

$$f_1(x, x^*) = f(x^*) + (x - x^*)' \frac{\partial f(x^*)}{\partial x} \tag{6.2}$$

в окрестности точки $x = x^*$ совпадает с $f(x)$ с точностью до $o(\|x - x^*\|)$, что служит основанием назвать её аппроксимацией первого порядка (линейной аппроксимацией) функции $f(x)$ в окрестности точки x^*.

Если $f(x) \in C^{(2)}$, то функция

$$f_2(x, x^*) = f(x^*) + (x - x^*)' \frac{\partial f(x^*)}{\partial x} + \frac{1}{2}(x - x^*)' \frac{\partial^2 f(x)}{\partial x^2}(x - x^*) \tag{6.3}$$

называется аппроксимацией второго порядка (квадратичной аппроксимацией) функции $f(x)$ в в окрестности точки x^*, так как

$$|f(x) - f_2(x, x^*)| \leq o(\|x - x^*\|^2).$$

2. Методы первого порядка

В большинстве методов первого порядка последовательность $x^1, x^2, \ldots$ приближений к оптимальному плану генерируется по правилу

$$x^{k+1} = x^k + \theta_k l^k,$$

где l^k — направление в точке x^k, $\theta_k \geq 0$ — шаг вдоль l^k.

Направление l^k выбирается из решения задачи

$$\bar{f}_1(l^k, x^k) = \min_l \bar{f}_1(l, x^k), \qquad S_k(l, x^k) \leq 1, \tag{6.4}$$

где $\bar{f}_1(l, x^k) = f_1(x^k + l, x^k) - f(x^k) = l' \frac{\partial f(x^k)}{\partial x}$, $S_k(l, x^k) \leq 1$ — некоторое нормировочное условие. Предполагается, что нормировочное условие таково, что задача (6.4) имеет решение.

Конкретные правила выбора нормировочного условия $S_k(l, x^k)$ и шага θ_k и определяЮт конкретный метод первого порядка.

Например, можно положить

$$S_k(l, x^k) = l' D l + d' l, \qquad D = D(x^k) \dot{>} 0, \tag{6.5}$$

где D – заданная положительно определенная матрица, d — заданный вектор. При $D = E$, $d = 0$ использование в (6.4) нормировочного условия (6.5) приводит к направлению

$$l^k = -\alpha \frac{\partial f(x^k)}{\partial x}, \alpha = \|\frac{\partial f(x^k)}{\partial x}\| > 0,$$

направлению антиградиента в точке x^k.

Численные методы, основанные на таких направлениях, называются **градиентными**.

Можно использовать и другие нормированные условия:

$$S_k(l, x^k) = \sum_{i=1}^{n} |\sum_{j=1}^{n} d_{ij} l_j - d_i|,$$

$$S_k(l, x^k) = \max_{1 \le i \le n} |\sum_{j=1}^{n} d_{ij} l_j - d_i|.$$

Здесь d_{ij} – коэффициенты матрицы D, d_i — коэффициенты вектора d.

Перейдём ко второй задаче, возникающей на итерации $x^k \to x^{k+1}$: к проблеме выбора шага θ_k.

При решении этой проблемы используются три основных способа построения шага θ_k :

1. $\theta_k \equiv \theta > 0$,

2. $f(x^k + \theta_k l^k) = \min_{\theta \ge 0} f(x^k + \theta l^k)$,

3. $f(x^k + \theta_k l^k) - f(x^k) \le \varepsilon \theta_k \frac{\partial f'(x^k)}{\partial x} l^k$,

где ε — заданное число, $0 < \theta < 1$. Первый способ наиболее простой. В совокупности с $l^k = -gradf(x^k)$ он образует градиентный метод минимизации функции $f(x)$.

Второй способ в точности почти не осуществим, так как даже задача одномерной минимизации всегда решается приближённо. Если $l^k = -gradf(x^k)$, то мы получаем градиентный метод наискорейшего спуска.

В третьем способе выбор шага θ_k достигается, например, последовательным дроблением произвольного числа $\theta > 0$.

3. Методы второго порядка

В методах второго порядка для построения направления l^k используется задача

$$\bar{f}_2(l^k, x^k) = \min_{l} \bar{f}_2(l, x^k), \qquad S(l, x^k) \le 1, \tag{6.6}$$

где $\bar{f}_2(l, x^k) = f_2(x^k + l, x^k) - f(x^k) = l' \frac{\partial f(x^k)}{\partial x} + \frac{1}{2} l' [\frac{\partial^2 f(x^k)}{\partial x^2}] l$, $\qquad S(l, x^k)$ — нормировочные условия.

Если $\frac{\partial^2 f(x)}{\partial x^2} \dot{>} 0$, то задача (6.6) имеет решения и без нормировочного условия.

Далее мы будим рассматривать только этот случай: $\frac{\partial^2 f(x)}{\partial x^2} \dot{>} 0$.

Можно проверить,что решение l^k задачи

$$2l' \frac{\partial f(x^k)}{\partial x} + l' [\frac{\partial^2 f(x^k)}{\partial x^2}] l \to \min, \qquad l \in \mathbb{R}^n$$

имеет вид $l^k = -A_k^{-1} b_k$, где $A_k = \frac{\partial^2 f(x^k)}{\partial x^2}$, $b_k = \frac{\partial f(x^k)}{\partial x}$.

Вектор $l^k = -A_k^{-1} b_k$ называется **направлением Ньютона**.

При вычислении шага θ_k используют три способа, рассмотренные в пункте 2.

Если $l^k = -A_k^{-1} b_k$, и используем первый способ построения $\theta_k \equiv 1$, то получаем метод Ньютона.

Если направление l^k строится по правилам

$$l^k = -\widetilde{A}_k^{-1} \widetilde{b}_k,$$

где $\widetilde{A}_k$, $\widetilde{b}_k$ — аппроксимации матрицы A_k и вектора b_k, основанные на значениях $f(x)$, $\frac{\partial f(x)}{\partial x}$ из малой окрестности точки x^k, то методы называются методами Ньютоновского типа.

Если $\widetilde{A}_k, \widetilde{b}_k$ строятся только по значениям $f(x)$ и $\frac{\partial f(x)}{\partial x}$ на предыдущих итерациях $x^{k-s}, x^{k-s+1}, \ldots, x^k$, то методы называются **квазиньютоновскими**.

Распространены следующие типа построения аппроксимаций A_k, b_k в квазиньютоновских методах:

- $$\widetilde{A}_k = \frac{\partial^2 f(x^1)}{\partial x^2}, \qquad \widetilde{b}_k = b_k,$$
- матрица $\widetilde{A}_k$ аппроксимируется выражениями, составленными с помощью значений $f(x)$, $\frac{\partial f(x)}{\partial x}$; $\widetilde{b}_k = b_k$ (квазиньютоновские методы первого порядка),
- матрица $\widetilde{A}_k$ и вектор $\widetilde{b}_k$ аппроксимируются только с помощью значений функции $f(x)$ (квазиньютоновские методы нулевого порядка).

4. Метод сопряжённых градиентов

Особую популярность среди квазиньютоновских методов приобрел метод сопряжённых градиентов. Этот метод основан на специальном методе решения уравнений:

$$b_k + A_k l = 0 \tag{6.7}$$

в котором матрица A_k^{-1} явно не используется.

В вычислительных методах линейной алгебры показывается, что решение l^k уравнения (6.7) совпадает с вектором m^n, построенным по рекуррентным формулам:

$$p^0 = -\nabla f_2(m^0, x^k), \qquad m^0 \in \mathbb{R}^n,$$

$$m^{s+1} = m^s + \alpha_s p^s;$$

где

$$p^s = \beta_s p^{s-1} - \nabla f_2(m^s, x^k),$$
$$\alpha_s = -\frac{p^{s\prime}\nabla f_2(m^s, x^k)}{p^{s\prime} A_k p^s}, \quad \beta_s = -\frac{||\nabla f_2(m^s, x^k)||^2}{||\nabla f_2(m^{s-1}, x^k)||^2},$$
$$s = 0, 1, \dots, n-1.$$

Здесь $\nabla f_2(l, x^k) = \partial f_2(l, x^k)/\partial l$.

Своё название эти методы получили по основному свойству градиентов p^s, $s = 0, 1, \dots, n-1$:

$$p^{s\prime} A_k p^t = 0, \qquad 0 \le t \le s-1.$$

Это свойство послужило основой для построения многочисленных методов сопряжённых (двойственных) направлений.

§2. Методы условной минимизации

1. Задачи с линейными ограничениями. Точные методы

Рассмотрим задачу

$$f(x) \to \min, Ax = b, x \in Q \tag{6.8}$$

где $A \in R^{m\times n}, m \le n, rankA = m$.

Последовательные приближения к x^0 строятся по правилу

$$x^{k+1} = x^k + \theta_k l^k, k = 1, 2, \dots \tag{6.9}$$

где l^k — решение задачи:

$$\bar{f}_1(l^k, x^k) = \min \bar{f}_1(l, x^k), Al = 0, S_k(l, x^k) \le 1. \tag{6.10}$$

Здесь $S_k(l, x^k) \le 1$ — нормированное условие.

Если множество планов $X = \{x : Ax = b, x \in Q\}$ — компакт, то в качестве нормировочного условия можно взять условие:

$$x^k + l \in Q, \tag{6.11}$$

полученное из прямого ограничения задачи (6.8).

Решение задачи (6.10), (6.11) называется **условным антиградиентом** в точке x^k.

Методы, в которых в качестве l^k используется условные (анти)градиенты, а шаг θ_k строится по одному из указанных выше правил и с учётом условия $x^{k+1} = x^k + \theta_k l^k \in Q$ называются **методами условного градиента**.

Рассмотрим задачу (6.10) для случая, когда нормировачное условие имеет вид:

$$l'l \leq \alpha, \qquad l_j \geq 0, \text{ если } x_j^k = 0, \qquad j = \overline{1,n}. \tag{6.12}$$

(Здесь мы предполагаем, что $Q = \{x : x \geq 0\}$.)

Решение l^k задачи (6.10), (6.12) с достаточно малым $\alpha > 0$ называется проекцией антиградиента $-gradf(x^k)$ на множество допустимых планов. Исходя из (6.10), (6.12) легко получить формулы для вектора l^k. Вектор l^k по направлению совпадает с решением квадратичной задачи:

$$\bar{f}_1(l, x^k) + l'l/2 \to \min, \ Al = 0, \ l_j \geq 0, \text{ если } x_j^k = 0, j = \overline{1,n}.$$

Методы, в которых в качестве l^k используется проекция антиградиента, а шаг θ_k выбирается одним из указанных выше способов с учётом дополнительного требования:

$$x^{k+1} \in Q$$

называются **методами проекции градиента**.

2. Случай нелинейных ограничений

Определение 32 *<u>Линейной аппроксимацией</u> в точке $x = x^*, -\alpha \leq g(x^*) \leq 0, \alpha = \alpha(x^*) \geq 0$, неравенства*

$$g(x) \leq 0 \tag{6.13}$$

называется неравенство

$$g_1(x, x^*) = g(x^*) + (x - x^*)'\partial g(x^*)/\partial x \leq \beta(x^*), \ \beta(x^*) \geq 0.$$

Если $g(x^*) < -\alpha(x^*)$, то линейной аппроксимацией неравенств (6.13) в точке $x*$ будет "отсутствие всякого ограничения".

Если функцию $g_1(x,x^*)$ заменить на

$$g_2(x,x^*) = g_1(x,x^*) + \frac{1}{2}(x-x^*)'\frac{\partial^2 f(x^*)}{\partial x^2}(x-x^*),$$

то получим квадратичную аппроксимацию неравенства. Но такие аппроксимации используются в вычислительной практике крайне редко.

Определение 33 *Линейной аппроксимацией в точке $x = x^*, |h(x^*)| \le \alpha, \alpha = \alpha(x^*) \ge 0$, равенства*

$$h(x) = 0 \tag{6.14}$$

называются неравенства

$$-\beta_*(x^*) \le h_1(x,x^*) := h(x^*) + (x-x^*)'\partial h(x^*)/\partial x \le \beta^*(x^*),\ \beta(x^*),\ \beta_*(x^*) \ge 0.$$

Определение 34 Линейной аппроксимацией задачи

$$f(x) \to \min,\ g(x) \le 0,\ h(x) = 0, \tag{6.15}$$

где $f(x), g(x), h(x) \in C^{(1)}$, в точке x^k называется задача:

$$\bar f_1(l,x^k) \to \min_l,\ \bar g_1(l,x^k) \le \beta(x^k), \tag{6.16}$$

$$\beta_*(x^k) \le \bar h_1(l,x^k) \le \beta^*(x^k), S_k(l,x^k) \le 1,$$

полученная в результате линейной аппроксимации целевой функции и функций ограничений исходной задачи.

Здесь $\bar g_1(l,x^k) = g_1(x^k+,l,x^k) - g(x^k) = l'\partial g(x^k)/\partial x$, $\bar h_1(l,x^k) = h_1(x^k+,l,x^k) - h(x^k) = l'\partial h(x^k)/\partial x$. При квадратичной аппроксимации задачи (6.15) с $f(x) \in C^{(2)}$ меняется **только** целевая функция $\bar f_1(l,x^k)$ на $\bar f_2(l,x^k)$. Ограничения остаются прежними.

Из решения задачи (6.16) находим направление l^k. Шаг θ_k определяется по описанным выше правилам и дополнительным требованиям на точность выполнения ограничений.

Метод возможных направлений

Продолжим рассмотрение задачи минимизации гладкой функции $f(x)$ на заданном множестве $X \subset \mathbb{R}^n$. Направление $e \ne 0$ назовем возможным в точке $x \in X$, если $x + te \in X$ при всех t, $0 \le t \le t^0$, где t^0 — положительное число, зависящее от точки x, направления e и от структуры множества X.

Определение 35 *Направление $e \neq 0$ назовем возможным направлением убывания функции $f(x)$ в точке x на множестве X, если e — возможное направление в точке x и $f(x + \alpha e) < f(x)$при всех α, $0 < \alpha < \beta$, где $0 < \beta < t^Q$.*

Метод возможных направлений основан на следующей естественной и прозрачной идее: на каждой итерации этого метода определяется возможное направление убывания функции, и по этому направлению осуществляется спуск с некоторым положительным шагом. Собственно говоря, эта идея для нас уже не новая — именно на ней были основаны многие варианты изложенных выше методов. В самом деле, если $X = \mathbb{R}^n$, $\partial f(x)/\partial x \neq 0$, то возможное направление убывания функции легко находится — это направление антиградиента $e = -\partial f(x)/\partial x$. Более трудным был выбор возможного направления убывания при $X \neq \mathbb{R}^n$. Понятно, что если задача выбора возможного направления убывания на каждой итерации слишком сложна и требует решения вспомогательных задач минимизации, сравнимых по трудности, быть может, с исходной задачей, то такой метод минимизации будет малоэффективным. Возникает вопрос: нельзя ли указать простые и достаточно удобные для реализации на ЭВМ способы выбора возможных направлений убывания? Оказывается, для достаточно широких классов гладких задач такие способы существуют. Покажем это на примере следующей задачи:

$$f(x) \to \min, \quad x \in X = \{x \in \mathbb{R}^n : g_i(x) \leq 0, i = 1, ..., m\}, \tag{6.17}$$

где функции $f(x), g_i(x), i = 1, ..., m$, определены на всем пространстве $\mathbb{R}^n$, и $f(x),\ g_i(x) \in C^1(X)$.

Чтобы проще было пояснить суть метода возможных направлении для задачи (6.17), опишем более простой вариант этого метода.

Пусть $x^1 \in X$ — некоторое начальное приближение. Пусть известно k-е приближение $x^k \in X$, ≥ 1. Введем множество номеров

$$I_k = \{i : 1 \leq i \leq m,\ g_i(x^k) = 0\}.$$

Возможно, что $I_k = \emptyset$ — это будет означать, что $g_i(x^k) < 0$ при всех $i = 1, ..., m$, т. е. $x^k \in int\ X$ — такая возможность ниже не исключается. В пространстве переменных

$$z = (e, \sigma) = (e_1, ..., e_n, \sigma) \in \mathbb{R}^{n+1}$$

рассмотрим вспомогательную задачу

$$\sigma \to \inf,\ z = (e, \sigma) \in W_k, \tag{6.18}$$

$$W_k = \{(e,\sigma) : \frac{\partial f'(x^k)}{\partial x} e \le \sigma, \frac{\partial g_i'(x^k)}{\partial x} e \le \sigma,\ i \in I_k, |e_j| \le 1, j = 1, ...n\}.$$

Заметим, что задача (6.18) является задачей линейного программирования, причем минимизируемая функция $c'z = 0'e + \sigma,\ c = (0, 1) \in \mathbb{R}^{n+1}$, явно не зависит от переменных $e = (e^1, ..., e^n)$. Далее, ясно, что точка $z = (e = 0, \sigma = 0) = (0, 0) \in W_k$, так что $W_k \ne \emptyset$ и $\inf\limits_{W_k} \sigma = \sigma_k \le 0$ при всех $k = 0, 1, ...$ Очевидно, множество W_k замкнуто. Наконец, условия $|e_j| \le 1, j = 1, ...n$, называемые условиями нормировки, гарантируют ограниченность множества W_k. Тогда ясно, что задача (6.18) имеет хотя бы одно решение. Для получения решения задачи (6.18) могут быть использованы известные конечные методы линейного программирования (например, симплекс-метод, описанный в гл. 1).

Предположим, что задача (6.18) решена и найдены $(e^k, \sigma_k) \in W_k$ такие, что $\inf\limits_{W_k} \sigma = \sigma_k$. Выше было замечено, что $\sigma_k \le 0$.

Сначала рассмотрим случай $\sigma_k < 0$. Оказывается, в этом случае направление e^k, полученное из задачи(6.18), является возможным направлением убывания функции $f(x)$ в точке x^k. В самом деле, из условия $(e^k, \sigma_k) \in W_k$ следует, что

$$\frac{\partial f'(x^k)}{\partial x} e^k \le \sigma_k < 0, \frac{\partial g_i'(x^k)}{\partial x} e^k \le \sigma_k < 0,\ i \in I_k.$$

Отсюда ясно, что $e^k \ne 0$. Кроме того, для любого номера $i \in I_k$ и достаточно малых $\alpha > 0$ имеем

$$g_i(x^k + \alpha e^k) = g_i(x^k + \alpha e^k) - g_i(x^k) = \alpha \frac{\partial g_i'(x^k)}{\partial x} e^k + o(\alpha) \le \alpha\sigma_k + o(\alpha) < 0.$$

Если $i \notin I_k$, т. е. $g_i(x^k) < 0$, то в силу непрерывности функции $g_i(x)$ неравенство $g_i(x^k + \alpha e^k) < 0$ сохранится при всех достаточно малых $\alpha > 0$.

Для $i = 1, ..., m$, обозначим через $\alpha_i > 0$ максимально возможное число, при котором $g_i(x^k + \alpha e^k) \le 0,\ \alpha \in [0, \alpha_i]$. Положим $\alpha_0 = \min\{\alpha_1, ..., \alpha_m\} > 0$. Тогда

$$g_i(x^k + \alpha e^k) \le 0,\quad \alpha \in [0, \alpha_0],\ i = 1, ..., m,$$

т. е. e^k — возможное направление в точке x^k на множестве X.

Далее, взяв при необходимости число $\alpha_0 > 0$ еще меньшим, можно добиться выполнения неравенства

$$f(x^k + \alpha e^k) - f(x^k) = \alpha \frac{\partial f'(x^k)}{\partial x} e^k + o(\alpha) \leq \alpha\sigma_k + o(\alpha) < 0, \alpha \in [0, \alpha_0].$$

Тем самым показано, что если (e^k, σ_k) — решение задачи (6.18), причем $\sigma_k < 0$, то e^k — возможное направление убывания функции $f(x)$ в точке x^k на множестве X.

Используя найденное таким образом направление e^k, следующее $(k+1)$-е приближение определим так:

$$x^{k+1} = x^k + \alpha_k e^k, \;\; 0 < \alpha_k \leq \beta_k, \tag{6.19}$$

где

$$\beta_k = \sup\{\alpha : \; x^k + te^k \in X, \; 0 \leq t \leq \alpha\}. \tag{6.20}$$

Выбирая α_k в (6.19) различными способами, будем получать различные варианты метода возможных направлений. Перечислим некоторые способы выбора α_k.

1) Величина α_k может выбираться из условий

$$0 < \alpha_k \leq \beta_k, \;\; \bar{g}_k(\alpha_k) = \inf_{0<\alpha\leq\beta_k} \bar{g}_k(\alpha) = \bar{g}_k^*, \; \bar{g}_k(\alpha) := f(x^k + \alpha e^k). \tag{6.21}$$

Для минимизации функции $\bar{g}_k(\alpha)$ могут быть использованы известные методы. Точное решение задачи (6.21) удается найти лишь в редких случаях; возможно также, что на некоторых направлениях e^k величина $\beta_k = \infty$ и нижняя грань функции $\bar{g}_k(\alpha)$ при $\alpha >$ не достигается. Поэтому вместо (6.21) на практике целесообразно употреблять такой способ выбора α_k:

$$0 < \alpha_k \leq \beta_k, \; \bar{g}_k(\alpha_k) \leq \bar{g}_k^* + \delta_k, \; \delta_k \geq 0, \; \sum_{k=1}^{\infty} \delta_k = \delta < \infty, \tag{6.22}$$

или

$$f(x^k + \alpha_k e^k) \leq (1 - \lambda_k) f(x^k) + \lambda_k \bar{g}_k^*, \; 0 < \lambda \leq \lambda_k \leq 1.$$

2) Возможен выбор величин α_k из следующих условий:

$$f(x^k) - f(x^k + \alpha_k e^k) \geq \epsilon\alpha_k |\sigma_k|, 0 < \alpha_k \leq \beta_k, 0 < \epsilon < 0.5.$$

Для определения такого α_k сначала можно положить $\alpha_k = \beta_k$, а затем при необходимости дробить эту величину.

3) В тех случаях, когда трудно оценить величину β_k из (6.20), приходится довольствоваться нахождением какого-либо $\alpha_k > 0$, обеспечивающего включение $x^k + \alpha_k e^k \in X$ и условие монотонности $f(x^k + \alpha_k e^k) < f(x^k)$. Для этого обычно выбирают какую-либо постоянную $\alpha > 0$, полагают $\alpha_k = \alpha$ и проверяют условие монотонности и принадлежность точки x^{k+1} множеству X; при необходимости дробят величину $\alpha_k = \alpha$, добиваясь выполнения упомянутых условий.

Один шаг метода возможных направлений для задачи (6.17) в случае $\sigma_k < 0$ описан. Попутно выяснен смысл вспомогательной задачи (6.18): минимизируя σ, мы добиваемся того, чтобы направление e^k было как можно ближе к направлению антиградиента (это обеспечивается условием $\frac{\partial f'(x^k)}{\partial x} e^k \leq \sigma_k$) и в то же время оставалось возможным направлением для множества X в точке x^k (это обеспечивается условиями $\frac{\partial g_i'(x^k)}{\partial x} e^k \leq \sigma_k, i \in I_k$), причем, чем меньше σ_k, тем ярче выражены указанные свойства направления e^k.

Теперь рассмотрим случай, когда в решении (e^k, σ_k) задачи (6.18) координата $\sigma_k = 0$. Как видно из (6.18), это может случиться, например, при $\frac{\partial f(x^k)}{\partial x} = 0$ или $\frac{\partial g_i(x^k)}{\partial x} = 0$ для некоторого номера $i \in I_k$. При $\sigma_k = 0$ уже нельзя гарантировать, что e^k будет возможным направлением убывания. В этом случае итерационный процесс (6.18)-(6.20) прекращается. Оказывается, при $\sigma_k = 0$ в точке x^k выполняются необходимые условия минимума первого порядка. Для выпуклой задачи (6.17) со свойством

$$\exists \bar{x} \in X : \quad g_i(\bar{x}) < 0, \ i = 1, ..., m, \tag{6.23}$$

условие $\sigma_k = 0$ гарантирует, что $x^k = x^0$ — решение задачи (6.17). А именно справедлива

Теорема 35 *Пусть функции $f(x), g_i(x), i = 1, ..., m$, определены на $\mathbb{R}^n$, $f(x), g_i(x) \in C^1(X)$, где множество X задано условиями (6.17), и пусть задача (6.17) имеет решение. Тогда для любого решения x^0 задачи (6.17) задача*

$$\sigma \to \inf, \ z = (e, \sigma) \in W_0, \tag{6.24}$$

$$W_0 = \{(e, \sigma) : \frac{\partial f'(x^0)}{\partial x} e \leq \sigma, \frac{\partial g_i'(x^0)}{\partial x} e \leq \sigma, \ i \in I_*, |e_j| \leq 1, j = 1, ...n\}.$$

где

$$I_0 = \{i : 1 \leq i \leq m, g_i(x^0) = 0\},$$

необходимо имеет решение (e^0, σ_0) *с* $\sigma_0 = \min\limits_{\sigma \in W_0} \sigma = 0$. *Если, кроме того,* $f(x), g_i(x), i = 1, ..., m$, *выпуклы на* $\mathbb{R}^n$, *и выполнено условие Слейтера 6.23), то всякая точка* $x^0 \in X$, *для которой задача (6.24) определяет величину* $\sigma_0 = \min\limits_{\sigma \in W_*} \sigma = 0$, *является решением задачи (6.17).*

В невыпуклых задачах условие $\sigma_k = 0$ не является достаточным для оптимальности точки x^0.

3. Методы штрафных функций

Пусть $X \in \mathbb{R}^n$ — множество планов некоторой задачи НЛП.

Определение 36 *Функцию* $\psi(x, r), x \in \mathbb{R}^n, r \in \mathbb{R}^1$ *назовём* **внешней штрафной функцией** *множества* X, *если:*

$$\psi(x, r) = \begin{cases} 0, \text{ если } x \in X, \\ \to \infty \text{ при } r \to \infty, \text{ если } x \notin X. \end{cases}$$

Простейшие примеры штрафных функций:

$$\psi(x, r) = r \sum_{i=1}^{m} g_i^2(x) \text{ для } X = \{x : g_i(x) = 0, i = \overline{1, m}\};$$

$$\psi(x, r) = r \sum_{i=1}^{m} \max\{g_i(x), 0\} \text{ для } X = \{x : g_i(x) \leq 0, i = \overline{1, m}\}.$$

Определение 37 **Внутренней штрафной(барьерной)** *функцией множества* X *называется функция:*

$$\phi(x, r) = \begin{cases} \infty, \text{ если } x \notin X, \\ \to 0, \text{ если } x \in X, \ x \notin \partial X, \ r \to 0, \\ \to \infty, \text{ если } x \in X, \ x \to \partial X. \end{cases}$$

где ∂X *— граница множества* X

Простейшие примеры барьерной функции множества $X = \{x : g_i(x) \leq 0, i = \overline{1, m}\}$:

$$\phi(x, r) = \begin{cases} r \sum_{i=1}^{m} 1/g_i(x), \text{ если } g_i(x) \leq 0, i = \overline{1, m}, \\ \infty, \text{ если } \exists \ g_{i_0}(x) > 0. \end{cases}$$

Физическая интерпретация штрафных функций: $\psi(x, r)$ — величина штрафа за удаление от множества X, $\phi(x, r)$ — величина штрафа за приближение к границе множества X, r — параметр, характеризующий относительную величину штрафа.

Метод (внешних) штрафных функций решение задачи на условный минимум

$$f(x) \to \min, \qquad x \in X \tag{6.25}$$

сводит к решению последовательности (при $r \to \infty$) задач безусловной минимизации

$$f(x_\psi(r)) + \psi(x_\psi(r), r) = \min\{f(x) + \psi(x, r)\}, x \in \mathbb{R}^n.$$

Метод барьерных (внутренних штрафных) функций задачу (6.25) на условный минимум сводит к последовательности задач (при $r \to 0$) на безусловный минимум

$$f(x_\phi(r)) + \phi(x_\phi(r), r) = \min\{f(x) + \phi(x, r)\}, x \in \mathbb{R}^n.$$

При весьма слабых предположениях справедливы соотношения:

1. $f(x_\psi(r_2)) \leq f(x_\psi(r_1))$, если $r_2 > r_1$; $f(x_\phi(r_2)) \leq f(x_\phi(r_1))$, если $r_2 < r_1$;
2. $f(x_\phi(r)) \to f(x^0)$ при $r \to \infty$; $f(x_\phi(r)) \to f(x^0)$ при $r \to 0$;
3. $\psi(x_\phi(r), r) \to 0$ при $r \to \infty$; $\phi(x_\phi(r), r) \to 0$ при $r \to 0$.

Эти свойства лежат в основе обоснования методов штрафных функций.

Достоинство методов штрафных функций: задача на условный минимум приближается задачами на безусловный минимум.

Недостаток: при добавлении штрафной функции ухудшается структура целевой функции, которая теперь приобретает овражную структуру, что отрицательно сказывается на характеристиках методов безусловной минимизации.

4. Регуляризация некорректных задач минимизации

Пусть $f(x) \geq \beta, x \in X \subset \mathbb{R}^n, \beta > -\infty$, и ищется оптимальный план задачи:

$$f(x) \to \min, x \in X$$

с непустым множеством решений X^0.

Определение 38 *Последовательность* $x^k \in X, k = 1, 2, \ldots$ *называется* **минимизирующей**, *если* $f(x^k) \to f^0 = \inf\limits_{x \in X} f(x)$, *при* $k \to \infty$.

Определение 39 *Говорят, что задача (6.25) корректна относительно* X^0, *если каждая минимизирующая последовательность* $x^k \in X, k = 1, 2, \ldots$ *сходится к множеству* X^0:

$$\rho(x^k, X^0) \to 0, k \to \infty, \tag{6.26}$$

где $\rho(x^k, X) = \inf ||x^k - x||, x \in X,$ — *расстояние от точки* x^k *до множества* X.

Существуют задачи некорректные, минимизирующие последовательности которых не удовлетворяют свойству (6.26).

Пример некорректной задачи:

$$f(x) = x/(1 + x^2) \to \min, x \geq 0.$$

Эта задача имеет единственное решение $x^0 = 0$.

Ясно, что последовательность $x^k = k, k = 1, 2, \ldots,$ является минимизирующей. Однако

$$x^k = k \not\to 0 \text{ при } k \to \infty.$$

Основное достоинство (свойство) корректных задач, важное с точки зрения вычислительных методов, состоит в том, что для них элементы минимизирующей последовательности можно брать в качестве приближения к решению.

Справедлива следующая теорема, дающая достаточное условие корректности задачи.

Теорема 36 *Пусть* $f(x) \in \mathbb{C}, x \in X \subset R^n$. *Если при некотором* $c, -\infty < c < \infty$, *множество* $X^c = \{x \in X : f(x) \leq c\}$ *компактно, то задача (6.25) корректна относительно* X^0.

Доказательство. Пусть $x^k \in X^c$, $k = 1, 2, ...,$ — минимизирующая последовательность. Обозначим через X^* множество предельных точек этой последовательности. Из компактности множества X^c следует, что X^* — непустое множество. Пусть

$x^* \in X^*$. Так как функция $f(x)$ непрерывна, $f^0 \leq f(x^*) = \lim\limits_{k\to\infty} f(x^k) = f^0$, т.е. $x^* \in X^0$. Следовательно свойство (6.26) имеет место. Теорема доказана.

Наряду с задачей (6.25) рассмотрим последовательность **регуляризованных** задач:

$$f(x,\alpha_k) = f(x) + \alpha_k\Omega(x) \to \min, x \in X, k = 1,2,\ldots \tag{6.27}$$

где $\alpha_k > 0, \Omega(x) \geq 0$ — функция, обладающая следующими свойствами:

1. $\Omega(x) \geq 0, \forall x \in X$
2. множество $\Omega_c = \{x : x \in X; \Omega(x) \leq c\}$ является компактным, т.е. из каждой последовательности $\{x_k\} \in \Omega_c$ можно выбрать подпоследовательность $\{x_{k_n}\}$, сходящуюся к некоторой точке $z \in \Omega_c$.

Функция $\Omega(x)$, обладающая этими свойствами, называется **стабилизатором**.

Можно показать, что задача (6.27) является корректной.

Обозначим через $f^0(\alpha_k)$ — минимальное значение целевой функции задачи (6.27), через $x(\alpha_k,\epsilon_k)$ — её ϵ_k оптимальный план:

$$f^0(\alpha_k) \leq f(x(\alpha_k,\epsilon_k)) + \alpha_k\Omega(x(\alpha_k,\epsilon_k)) \leq f^0(\alpha_k) + \epsilon_k.$$

Теорема 37 *Если $\alpha_k \to 0, \epsilon_k/\alpha_k \to a < \infty$ при $k \to \infty$,*
то $\rho(x(\alpha_k,\epsilon_k), X^0) \to 0$ при $k \to \infty$.

Доказательство. Обозначим $x^k = x(\alpha_k,\epsilon_k)$. Следующие неравенства очевидны

$$f^0 = f(x^0) \leq f(x^k) \leq f(x^k) + \alpha_k\Omega(x^k) \leq f^0(\alpha_k) + \epsilon_k \leq$$

$$\leq f(x^0) + \alpha_k\Omega(x^0) + \epsilon_k \leq f(x^k) + \alpha_k\Omega(x^0) + \epsilon_k, \tag{6.28}$$

где $x^0 \in X^0$.

Значит $\Omega(x^k) \leq \Omega(x^0) + \epsilon_k/\alpha_k$. По условию, $\epsilon_k/\alpha_k \to a < \infty$ при $k \to \infty$. Следовательно, все элементы последовательности $x^k, k \geq k^0$, $k^0 < \infty$, принадлежат некоторому множеству уровня $\{x \in X : \Omega(x) \leq \bar{a}\}$ функции $\Omega(x)$, которое по условию компактно. Из (6.28) имеем $f^0 \leq f(x^k) \leq f^0 + \alpha_k\Omega(x^0) + \epsilon_k$. Отсюда с учетом того, что $\alpha_k \to 0$, $\epsilon_k \to 0$ при $k \to \infty$, получаем, что $f(x^k) \to f^0$ при $k \to \infty$, т.е. последовательность $x^k, k = 1,2,...$, является минимизирующей.

Выше показано, что

$$x^k \in \{x \in X : \Omega(x) \leq \bar{a}\},\ k \geq k^0,\ k^0 < \infty.$$

Обозначим через X^* множество предельных точек этой последовательности. Из компактности множества $\{x \in X : \Omega(x) \leq \bar{a}\}$ следует, что X^* — непустое множество. Пусть $x^* \in X^*$. Так как функция $f(x)$ непрерывна, $f^0 \leq f(x^*) = \lim\limits_{k\to\infty} f(x^k) = f^0$, т.е. $x^* \in X^0$. Следовательно $\rho(x^k, X^0) \to 0$ при $k \to \infty$.. Теорема доказана.

Отсюда делаем вывод, что в качестве приближенного решения некорректной задачи (6.25) можно взять приближенное решение задачи (6.27), где α_k — достаточно мало.

Глава 7

Задачи оптимального управления

7.1 Введение

В этой главе рассматриваются задачи оптимального управления процессами, описываемыми системами обыкновенных дифференциальных уравнений. Этот класс экстремальных задач существенно отличается от рассмотренных: если в задачах минимизации функции конечного числа переменных искомая точка минимума являлась точкой n-мерного пространства, то в задачах оптимального управления искомая точка минимума представляет собой функцию, принадлежащую некоторому бесконечномерному функциональному пространству. Такие задачи имеют многочисленные приложения в механике космического полета, в вопросах управления электроприводами, химическими или ядерными реакторами, виброзащиты и т. д. Эффективным средством исследования задач оптимального управления является принцип максимума Понтрягина [587], представляющий собой необходимое условие оптимальности в таких задачах. Принцип максимума, открытый коллективом российских математиков во главе с академиком Л. С. Понтрягиным, наряду с методом динамического программирования представляют собой одно из крупных достижений современной математики и являются краеугольным камнем современной математической теории оптимального управления. Появление этих результатов стимулировало последующее бурное развитие теории экстремальных задач и методов их решения.

Сейчас теория оптимального управления переживает период бурного развития как в связи с наличием трудных и интересных математических проблем, так и в связи с обилием приложений, в том числе и в таких областях, как экономика, биология, медицина, ядерная энергетика и др.

7.2 Примеры задач оптимального управления

Приведем несколько конкретных задач оптимального управления.

Пример 1. Рассмотрим прямолинейное движение тела постоянной массы M под действием управляющего воздействия u, создаваемого установленным на теле двигателем. Обозначим через x координату центра масс тела и предположим, что никакие другие силы на тело не действуют. Тогда в соответствии со вторым законом Ньютона уравнение движения тела имеет вид

$$M\ddot{x}(t) = u(t).$$

Последнее уравнение эквивалентно системе двух уравнений первого порядка

$$\dot{x}_1(t) = x_2(t), \quad M\dot{x}_2(t) = u(t).$$

Пример 2. Движение плоского маятника, подвешенного к точке опоры при помощи жесткого невесомого стержня (рис. 7.1), как известно, описывается уравнением

$$I\ddot{\theta} + b\dot{\theta} + mgl\sin\theta = M(\tau),$$

где l — длина жесткого стержня маятника, m — масса, сосредоточенная в конце стержня, $I = ml^2$ — момент инерции, g — гравитационная постоянная (ускорение силы тяжести), $b > 0$ — коэффициент демпфирования, τ — время, $M(\tau)$ — внешний управляющий момент, $\theta = \theta(\tau)$ — угол отклонения стержня от точки устойчивого равновесия. Если сделать замену переменной $t = \tau\sqrt{mgl/I}$, то это уравнение можно привести к виду

$$\ddot{\varphi} + \beta\dot{\varphi} + \sin\varphi = u(t), \tag{7.1}$$

где

$$\varphi = \varphi(t) = \theta(t\sqrt{I/(mgl)}),\ \beta = b/\sqrt{Imgl},\ u(t) = M(t\sqrt{I/(mgl)})/(mgl).$$

Рис. 7.1:

Обозначим $x^1(t) = \varphi(t)$ (угол отклонения маятника), $x^2(t) = \dot{\varphi}(t)$ (скорость маятника). Тогда уравнение (7.1) запишется в виде системы двух уравнений первого порядка:

$$\dot{x}^1(t) = x^2(t),\ \dot{x}^2(t) = -\beta x^2(t) - \sin x^1(t) + u(t). \tag{7.2}$$

Пусть в начальный момент $t = 0$ маятник отклонился на угол $x^1(0) = x^1_0$ и имеет начальную скорость $x^2(0) = x^2_0$. Будем также считать, что функция $u(t)$ — управляющий момент (выбор которого может влиять на движение маятника) — удовлетворяет ограничению

$$|u(t)| \leq \gamma,\ \ \gamma = const > 0,\ t \geq 0. \tag{7.3}$$

Здесь возможны следующие постановки задач оптимального управления: выбрать управление $u(t)$, удовлетворяющее условиям (7.3) так, чтобы:

1) за минимальное время T остановить маятник в одной из точек устойчивого равновесия, т. е. добиться выполнения условий

$$x^1(T) = 2\pi k,\ \ x^2(T) = 0 \tag{7.4}$$

при некотором $k, = 0, \pm 1, \pm 2, ...$ (задача быстродействия);

2) за минимальное время T добиться выполнения условия

$$(x^1(T))^2 + (x^2(T))^2 \leq \epsilon,$$

где $\epsilon > 0$ — заданное число;

3) к заданному моменту времени T величина $(x^1(T))^2 + (x^2(T))^2$, или $\int\limits_0^T (x^1(t))^2 dt$, или $\int\limits_0^T ((x^1(t))^2 + (x^2(t))^2) dt$, или $\max\limits_{0 \leq T} |x^1(t)|$, или $\max\limits_{0 \leq T} \max(|x^1(t)|, |x^2(t)|)$ принимала минимально возможное значение, или

4) в заданный момент T выполнялось равенство $x^2(T) = 0$, а величина $x^1(T) = 0$ была максимально возможной (задача о накоплении возмущений), или

5) к заданному моменту T добиться выполнения условий (7.4) и минимизировать величину

$\int\limits_0^T u^2(t)dt$ (условие (7.3) здесь может быть опущено).

Если колебание маятника ограничено какими-либо упорами, то в перечисленных задачах нужно еще требовать выполнения условия вида

$$|x^1(t)| \leq \mu, \ \ \mu = const > 0.$$

На управление $u(t)$ вместо условия (7.3) (или наряду с условием (7.3)) могут накладываться ограничения вида

$$\int\limits_0^T u^2(t)dt \leq R,$$

где $R = const > 0$.

При изучении малых колебаний маятника часто полагают $\sin\varphi \approx \varphi$, sin (pnstp, и тогда уравнение (7.1) и эквивалентная ему система становятся линейными и будут иметь вид

$$\ddot{\varphi} + \beta\dot{\varphi} + \varphi = u(t)$$

и соответственно

$$\dot{x}^1(t) = x^2(t), \ \ \dot{x}^2(t) = -\beta x^2(t) - x^1(t) + u(t).$$

Пример 3. Как известно, движение центра масс космического аппарата и расход массы описывается системой дифференциальных уравнений

$$\dot{r} = v, \ \dot{v} = gp/G + F, \ \dot{G} = -gq, \ 0 \leq t \leq T, \qquad (7.5)$$

где t — время, $r = r(t) = (r_1(t), r_2(t), r_3(t))$ — радиус-вектор центра масс аппарата, $v = v(t) = (v_1(t), v_2(t), v_3(t))$— скорость центра масс, $G == G(t)$ — текущий вес аппарата, g — коэффициент пропорциональности между массой и весом, $p = p(t) = (p_1(t), p_2(t), p_3(t))$ — вектор тяги двигателя, $q = q(t)$ — расход рабочего вещества, $F = F(r,t) = (F_1, F_2, F_3)$ — вектор ускорения от гравитационных сил.

В каждый момент времени t движение космического аппарата характеризуется величинами $r(t), v(t), G(t)$, называемыми фазовыми координатами.

Пусть в начальный момент $t = 0$ фазовые координаты аппарата известны:

$$r(0) = r_0, \ \ v(0) = v_0, \ \ G(0) = G_0. \qquad (7.6)$$

Величины $q = q(t)$, $p = p(t)$ являются управлением — задавая их по- разному, можно получить различные фазовые траектории (решения) задачи (7.5), (7.6). Конструктивные возможности аппарата, ограниченность ресурсов рабочего вещества накладывают на управление $q(t), p(t)$ ограничения, например, вида

$$p_{min} \leq |p(t)| \leq p_{max},\ q_{min} \leq |q(t)| \leq q_{max},\ 0 \leq t \leq T,$$

или $\int\limits_0^T q^2(t)dt \leq R$, R = const > 0. Кроме того, на фазовые траектории задачи (7.5), (7.6) могут накладываться некоторые ограничения, вытекающие, например, из условий того, чтобы вес аппарата был не меньше определенной величины или траектория полета проходила вне определенных областей космического пространства (областей повышенной радиации) и др. Здесь возникают задачи выбора управлений $q(t), p(t)$ так, чтобы управления и соответствующие им траектории задачи (7.5), (7.6)удовлетворяли всем наложенным ограничениям, и кроме того, достигалась та или иная цель. Например, здесь возможны следующие задачи:

1) попасть в заданную точку или область космического пространства за минимальное время;

2) к заданному моменту времени попасть в заданную область пространства с заданной скоростью (совершить мягкую посадку, например) и с максимальным весом аппарата или с минимальной затратой энергии;

3) достичь определенной скорости за минимальное время и т. п.

Существует большое число прикладных задач оптимального управления, которые связаны с механикой полета летательных аппаратов в космосе и атмосфере, с работой электроприводов, химических и ядерных реакторов, с вопросами виброзащиты и амортизации, с математической экономикой и т. д.,

7.3 Постановка задачи оптимального управления

Постановка любой конкретной задачи оптимального управления включает в себя ряд факторов: математическую модель управляемого объекта, цель управления (именуемую иногда критерием качества), различного рода ограничения на траекторию системы, управляющее воздействие, длительность процесса управления, класс допустимых управлений и т.д. Остановимся на этих факторах подробнее.

7.3.1 Модели объекта

В зависимости от вида рассматриваемого явления и желаемой степени детализации его изучения могут быть использованы различные типы уравнений: обыкновенные дифференциальные уравнения, уравнения с последействием, стохастические уравнения, уравнения в частных производных и т.д. Предположим ради определенности, что эволюция объекта описывается системой обыкновенных дифференциальных уравнений

$$\dot{x}(t) = f(t, x(t), u(t)), \quad \dot{x}(t) = \frac{dx(t)}{dt}, \quad t_0 \leq t \leq t_*. \tag{7.7}$$

Здесь $u \in \mathbb{R}^m$ — управление, $x \in \mathbb{R}^n$ — фазовый вектор (или вектор состояния) системы, $f \in \mathbb{R}^n$ — заданная функция, $\mathbb{R}^n$ — евклидово пространство размерности n. Придавая управлению $u(t) = (u_1(t), ..., u_m(t))$, $t \in [t_0, t_*]$, различные возможные значения, получаем различные состояния $x(t)$, $t \in [t_0, t_*]$, объекта, среди которых и выбирается оптимальное (то есть наилучшее) в том или ином смысле.

Пример. Рассмотрим прямолинейное движение тела постоянной массы M. Координаты тела будем обозначать через x. При движении тела его координата x изменяется во времени. Производная $\dot{x}$ представляет собой скорость движения тела, а $\ddot{x}$ – ускорение.

Будем предполагать, что на тело действую две внешние силы:

сила трения — $-s\dot{x}$ и

сила упругости — $-kx$.

Кроме того, будем предполпгать, что тела снабжено двигателем. Развиваемую двигателем силу воздейстрия на тело обозначим через u. Таким образом, согласно второму закону Ньютона двидение тела с течением времени будет описываться дифференциальным уравнением

$$M\ddot{x} = -b\dot{x} - kx + u.$$

Как и в предыдущем примере, нетрудно показать, что это уравнение эквивалентно ситеме

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= -\frac{k}{M}x_1(t) - \frac{s}{M}x_2(t) + u(t). \end{aligned} \tag{7.8}$$

Значит в данном примере динамика системы описывается системой (7.8).

Данную систему можно представить в виде (7.7), полижив

$$x(t) = (x_1(t), x_2(t)),\ f(t,x,u) = Ax + bu,\ x \in \mathbb{R}^2,\ u \in \mathbb{R}^1,$$

где

$$A = \begin{pmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{s}{M} \end{pmatrix},\ b = \begin{pmatrix} 0 \\ \frac{1}{M} \end{pmatrix}.$$

7.3.2 Критерий качества (минимизируемый функционал)

Управление системой (7.7) осуществляется для достижения некоторых целей, которые формально записываются в терминах минимизации по u функционалов J, определяемых управлением u и траекторией x, где

$$J = \int\limits_{t_0}^{t_*} F(t, x(t), u(t))dt + \varphi(t_*, x(t_*)) \to \min. \tag{7.9}$$

Здесь F и φ — заданные скалярные функции. Задача (7.7), (7.9) именуется задачей О. Больца; если $F \equiv 0$, то задачей А. Майера и, наконец, задачей Лагранжа при $\varphi \equiv 0$.

7.3.3 Ограничения на траекторию

В некоторых реальных ситуациях траектория системы не может принадлежать тем или иным частям пространства $\mathbb{R}^n$. Указанное обстоятельство находит отражение в ограничении вида $x(t) \in G(t)$, где $G(t)$ — заданная область в $\mathbb{R}^n$. В зависимости от конкретного типа этих ограничений выделяют различные классы задач управления.

В задачах с фиксированными концами начальное состояние $x(t_0)$ и конечное состояние $x(t_*)$ заданы. Если же $x(t_0)$ (или $x(t_*)$) не задано, то получаем задачу со свободным левым (правым) концом. Задача с подвижными концами — это задача, в которой моменты t_0 и t_* фиксированы, а векторы $x(t_0)$ и $x(t_*)$ принадлежат соответственно областям $G(t_0)$ и $G(t_*)$. В ряде случаев ограничения носят интегральный характер и имеют вид

$$\int\limits_{t_0}^{t_*} F_i(t, x(t), u(t))dt \le 0,\ i = 1, ..., p.$$

Если в задаче (7.7), (7.9) начальное положение $x(t_0)$ и конечное $x(t_*)$ заданы, моменты начала движения t_0 и окончания t_* свободны, функция $\varphi \equiv 0$ и $F \equiv 1$, то получаем задачу о переводе системы (7.7) из заданного положения $x(t_0)$ в заданное положение $x(t_*)$ за минимально возможное время. Подобного рода задачи именуются задачами оптимальными по быстродействию.

7.3.4 Ограничения на траекторию

В некоторых реальных ситуациях траектория системы не может принадлежать тем или иным частям пространства $\mathbb{R}^n$. Указанное обстоятельство находит отражение в ограничении вида $x(t) \in G(t)$, где $G(t)$ — заданная область в $\mathbb{R}^n$. В зависимости от конкретного типа этих ограничений выделяют различные классы задач управления.

В задачах с фиксированными концами начальное состояние $x(t_0)$ и конечное состояние $x(t_*)$ заданы. Если же $x(t_0)$ (или $x(t_*)$) не задано, то получаем задачу со свободным левым (правым) концом. Задача с подвижными концами — это задача, в которой моменты t_0 и t_* фиксированы, а векторы $x(t_0)$ и $x(t_*)$ принадлежат соответственно областям $G(t_0)$ и $G(t_*)$. В ряде случаев ограничения носят интегральный характер и имеют вид

$$\int\limits_{t_0}^{t_*} F_i(t, x(t), u(t))dt \leq 0,\ i = 1, ..., p.$$

Если в задаче (7.7), (7.9) начальное положение $x(t_0)$ и конечное $x(t_*)$ заданы, моменты начала движения t_0 и окончания t_* свободны, функция $\varphi \equiv 0$ и $F \equiv 1$, то получаем задачу о переводе системы (7.7) из заданного положения $x(t_0)$ в заданное положение $x(t_*)$ за минимально возможное время. Подобного рода задачи именуются задачами оптимальными по быстродействию.

7.3.5 Ограничения на управление

Информационные ограничения на управление зависят от того, какая именно информация о системе (7.7) доступна при выработке управляющего воздействия. Если вектор $x(t)$ недоступен измерению, то оптимальное управление ищется в классе функций $u(t)$, зависящих только от t. В этом случае оптимальное управление именуется программным. Если же вектор $x(t)$ известен точно при $t_0 \le t \le t_*$, то оптимальное управление ищется в виде $u(t, x(t))$ и называется синтезом оптимального управления (или управлением по принципу обратной связи). Отметим, что принцип обратной связи является одним из центральных принципов кибернетики [2]. Техническим примером реализации принципа обратной связи являются центробежные регуляторы И.И. Ползунова (1766) и Дж. Уатта (1784), авиационный автопилот Д. Ольховского (1912) и братьев Сперри (1914), гидравлический усилитель Л. Фарко (1873).

Кроме информационных ограничений возможен и другой тип ограничений, обусловленный ограниченностью ресурсов управления, имеющих вид $u(t) \in U(t)$, где $U(t) \subset \mathbb{R}^m$ — заданное множество.

Подчеркнем, что для детерминированных задач (то есть задач, в которых уравнения движения, критерий качества и ограничения известны точно) оптимальное значение критерия качества (7.9), реализуемое в классе программных управлений и управлений по принципу обратной связи, одно и то же.

7.4 Условия оптимальности

7.4.1 Принцип максимума [3]

Принцип максимума соответствует принципу Вейерштрасса и методу канонических уравнений Гамильтона в классическом вариационном исчислении.

Сформулируем необходимые условия оптимальности в форме принципа максимума для терминальной задачи оптимального управления вмда

$$\dot{x}(t) = f(x(t), u(t)), \quad t_0 \le t \le t_*; \quad x(t_0) = x_0, \quad g(x(t_*)) = 0, \tag{7.10}$$

$$u(t) \in U, \quad \varphi(x(t_*)) \to \min .$$

Здесь $U \subset R^r$ — заданное множество, x_0 — заданное начальное положения системы, t_0 — заданный момент времени, $g(x), \varphi(x)$ — заданные $m-$вектор функция и скалярная функция. Требуется найти управление $u(t), t \in [t_0, t_*]$, которое переводит систему из заданного начального состояния x_0 на множество $X_* = \{x \in \mathbb{R}^n : g(x) = 0\}$ в конечный (терминальный) момент времени t_* и при этом функция $\varphi(x(t_*))$ принимает минимальное значение.

Введем в рассмотрение скалярную функцию H

$$H(x(t), u(t), \psi(t)) = \psi'(t) f(x(t), u(t)),$$

где штрих $'$ — знак транспонирования, и с ее помощью запишем следующую систему дифференциальных уравнений для вспомогательной вектор-функции $\psi(t)$

$$\dot{\psi}(t) = \frac{\partial H(x(t), u(t), \psi(t))}{\partial x}. \tag{7.11}$$

Система (7.11) называется сопряженной системой, а вектор-функция $\psi(t)$ — решением сопряженной системы.

Теорема 38 *Предположим, что $u(t)$, $t \in [t_0, t_*]$, — оптимальное управление, а $x(t)$,$t \in [t_0, t_*]$, — соответствующая траектория ему траектория. Пусть* $\operatorname{rank} \frac{\partial g(x(t_*))}{\partial x} = m$. *Тогда существует такое $m-$вектор y, что вдоль решения $\psi(t)$,$t \in [t_0, t_*]$, сопряженной системы (7.11) с терминальным условие*

$$\psi(t_*) = -\frac{\partial \varphi(x(t_*))}{\partial x} - \frac{\partial g(x(t_*))}{\partial x} y, \tag{7.12}$$

$H(x(t), u, \psi(t))$ достигает своего максимума по $u \in U$ в точке $u(t)$

$$H(x(t), u(t), \psi(t)) = \max_{u \in U} H(x(t), u, \psi(t)) \quad \text{для почти всех } t \in [t_0, t_*]. \tag{7.13}$$

Найдем из соотношения (7.13) зависимость u от ψ, x, то есть

$$u = \tilde{u}(x(t), \psi(t)). \tag{7.14}$$

Далее подставим (7.14) в (7.10), (7.11). В результате получим краевую задачу для системы обыкновенных дифференциальных уравнений относительно функций $x(t)$ и $\psi(t)$, и вектора y, среди решений которой только и может находиться оптимальная траектория:

$$\dot{x}(t) = f(x(t), \tilde{u}(x(t), \psi(t))), \quad t_0 \le t \le t_*; \quad x(t_0) = x_0, \quad g(x(t_*)) = 0,$$

$$\dot{\psi}(t) = \frac{\partial H(x(t), \tilde{u}(x(t), \psi(t)), \psi(t))}{\partial x}, \quad \psi(t_*) = -\frac{\partial \varphi(x(t_*))}{\partial x} - \frac{\partial g(x(t_*))}{\partial x} y.$$

Если оптимальная траектория $x(t)$ и вектор $\psi(t)$ найдены, то оптимальное управление дается выражением (7.14). Отметим, однако, что поскольку соотношения (7.10)-(7.13) суть лишь необходимые условия оптимальности, то необходимо дополнительное обоснование оптимальности траектории и управления, найденных из соотношений (7.10)-(7.13).

7.4.2 Метод динамического программирования [4]

Метод динамического программирования является аналогом метода Гамильтона-Якоби, в котором изучается все поле оптимальных траекторий.

Опишем применение метода динамического программирования для задачи

$$\varphi(x(t_*)) \to \min, \tag{7.15}$$

$$\dot{x}(t) = f(x(t), u(t)), \quad t_0 \le t \le t_*; \quad x(t_0) = x_0, \quad u(t) \in U.$$

Выберем и зафиксируем произвольный момент времени $t \in [t_0, t_*]$ и рассмотрим вспомогательную задачу управления (7.10) на отрезке $[t, t_*]$. Обозначим через $V(t, x)$ минимальное значение критерия качества во вспомогательной задаче при начальном условии $x(t) = x$, где x — произвольный вектор из $\mathbb{R}^n$. При некоторых предположениях функция $V(t, x)$ удовлетворяет соотношениям

$$\frac{\partial V(t,x)}{\partial t} + \min_{u \in U} f(t,x,u) \frac{\partial V(t,x)}{\partial x} = 0.$$

$$t_0 \le t \le t_*, \ x \in \mathbb{R}^n, \tag{7.16}$$

$$V(t_*, x) = \varphi(x).$$

Решив задачу (7.16) и определив функцию $V(t, x)$, можно затем найти синтез оптимального управления $u(t, x)$ из соотношения

$$\min_{u \in U} f(t,x,u) \frac{\partial V(t,x)}{\partial x} = f(t,x,u(t,x)) \frac{\partial V(t,x)}{\partial x}. \tag{7.17}$$

Возможность определять именно синтез оптимального управления является характерной чертой метода динамического программирования, особенно существенной в задачах управления при неполной информации. Использование метода динамического программирования при решении конкретных задач сопряжено с необходимостью решать нелинейное уравнение в частных производных (7.16), а также с необходимостью дополнительного исследования оптимальности управления, полученного из (7.17).

Выше была описана универсальная техника решения задач оптимального управления, основанная на применении принципа максимума или метода динамического программирования. Наряду с этими методами при исследовании специальных классов задач оптимального управления могут быть использованы специальные приемы, эффективные именно для данного класса задач. Преимущество этих специальных приемов состоит в том, что с их помощью иногда можно проще осуществить аналогичное исследование рассматриваемой конкретной задачи. Один из таких подходов, опирающийся на теорию двойственности выпуклых функций [5], эффективен для задач, линейных по фазовым координатам. При этом вычисление минимума критерия качества в исходной задаче сводится к вычислению максимума некоторого вспомогательного функционала. Это, в частности, дает возможность изучать задачи, в которых оптимального управления не существует. Другой подход, эффективный для линейных задач (то есть задач линейных и по фазовым координатам, и по управлениям), основан на использовании классической проблемы моментов [6]. Отмеченные выше трудности использования общих необходимых условий оптимальности также приводят к необходимости выделения конкретных классов задач, для которых оказывается возможным эффективное построение решения.

7.4.3 Задача об успокоении твердого тела

Одним из хорошо изученных классов задач оптимального управления являются задачи линейного оптимального быстродействия [3], а также некоторые задачи успокоения движения твердого тела (возникающие, например, в теории управления летательными аппаратами). Рассмотрим в качестве примера одну из них, именно задачу об успокоении твердого тела, вращающегося относительно центра масс O. Обозначим: Ox_i, $i = 1, 2, 3,$ — главные центральные оси инерции тела, x_i — проекция вектора кинетического момента x на ось Ox_i и A, B, C — главные центральные моменты инерции твердого тела. Управление u движением твердого тела осуществляется парой поворотных двигателей, которые можно расположить под любым углом относительно тела. Движение тела относительно центра масс описывается уравнениями Эйлера

$$\dot{x}_1(t) + \frac{C-B}{CB} x_2(t) x_3(t) = u_1(t), \ t \geq 0,$$

$$\dot{x}_2(t) + \frac{A-C}{AC} x_1(t) x_3(t) = u_2(t), \ t \geq 0, \tag{7.18}$$

$$\dot{x}_3(t) + \frac{B-A}{BA} x_1(t)x_2(t) = u_3(t),\ t \geq 0.$$

Тело считается успокоенным, если величина модуля кинетического момента не превосходит заданной величины $R \geq 0$. Предположим, что в начальный момент времени $t_0 = 0$ значение кинетического момента $x(0) = (x_1(0), x_2(0), x_3(0))$ задано, причем

$$||x(0)|| = \sqrt{x_1^2(0) + x_2^2(0) + x_3^3(0)} \geq R.$$

Пусть далее $T \geq 0$ — первый момент времени, когда $||x(T)|| = R$. Управляющий момент u, подчиненный ограничению $||u(t)|| \leq b$, требуется выбрать так, чтобы минимизировать время T успокоения твердого тела.

Используем для решения этой задачи метод динамического программирования. Для рассматриваемой задачи быстродействия с учетом стационагности системы (7.18) (в ней функция $f(t,x,u)$ явно не зависит от t, т.е. $f(t,x,u) \equiv f(x,u)$) получаем, что $V(t,x) = V(x)$ и соответствующее уравнение в частных производных относительно функции $V(x)$ имеет вид

$$\min_{||u|| \leq b} \left[\frac{\partial V}{\partial x_1} \left(u_1 - \frac{C-B}{CB} x_2 x_3 \right) + \frac{\partial V}{\partial x_2} \left(u_2 - \frac{A-C}{AC} x_1 x_3 \right) + \right.$$

$$\left. \frac{\partial V}{\partial x_3} \left(u_3 - \frac{B-A}{AB} x_1 x_2 \right) \right] = -1,\ ||x|| \geq R, \tag{7.19}$$

$$V(x) = 0 \ \text{ при } \ ||x|| = R.$$

Вычисляя управление u, доставляющее минимум выражения в квадратных скобках (7.19), получим

$$u_i = -b \frac{\partial V}{\partial x_i} \left(\sum_{k=1}^{3} \left(\frac{\partial V}{\partial x_k} \right)^2 \right)^{-1/2}, \ \ ||x|| \geq R. \tag{7.20}$$

Подставляя (7.20) в (7.19), получим нелинейное уравнение в частных производных относительно функции V

$$\frac{\partial V}{\partial x_1} \left(- \frac{C-B}{CB} x_2 x_3 \right) + \frac{\partial V}{\partial x_2} \left(- \frac{A-C}{AC} x_1 x_3 \right) +$$

$$\frac{\partial V}{\partial x_3} \left(- \frac{B-A}{AB} x_1 x_2 \right) - b \left(\sum_{k=1}^{3} \left(\frac{\partial V}{\partial x_k} \right)^2 \right)^{1/2} = -1, \ \ ||x|| \geq R, \tag{7.21}$$

$$V(x) = 0 \ \text{ при } \ ||x|| = R.$$

Будем искать решение задачи (7.21) в виде

$$V(x) = \frac{1}{b} w(||x||),$$

где w - скалярная функция скалярного аргумента $r \geq R$, подлежащая определению. Подставляя это выражение для V в (7.21), получим соотношения, определяющие функцию w :

$$\frac{dw(r)}{dr} = 1, \ \ w(R) = 0, \ \ r \geq R.$$

Отсюда вытекает, что $w(||x||) = ||x|| - R$. Значит, оптимальное управление u, успокаивающее тело и минимальное время успокоения $V(x)$ равны

$$u_i(t) = -bx_i(t)/||x(t)||, \quad V(x(t)) = \frac{1}{b}(||x(t)|| - R), \ \ \text{при} \ \ ||x(t)|| \geq R.$$

Эти выражения показывают, что $||u(t)|| = b$, то есть вектор оптимального управления всегда максимален по величине и направлен против вектора кинетического момента.

7.4.4 Некоторые замечания о вычислительных и приближенных методах

В связи со значительными трудностями построения аналитического решения задач оптимального управления исключительное значение приобрели различные приближенные и численные методы их исследования [7, 8, 9]. В зависимости от алгоритмической основы метода он может быть отнесен к той или иной группе. Охарактеризуем некоторые из них. В основе первой группы методов лежит возможность сведения задачи оптимального управления к краевой задаче для системы дифференциальных уравнений с помощью принципа максимума с последующим использованием различных алгоритмов задания недостающих начальных условий. Ко второй группе численных методов относятся те, в которых ищется оптимальное управление с помощью итерационных процедур в пространстве управлений. При этом используются формулы для приращений критерия качества при вариации управления, приводящие либо к методам градиентного типа в пространстве управлений, либо к процедурам последовательных приближений. Третью группу составляют численные процедуры, основанные на переборе в пространстве траекторий или методе динамического программирования. Кроме того, имеются методы, эффективные для конкретных классов систем, например для линейных, а также методы, основанные на сведении исходной задачи оптимального управления к задаче математического программирования.Ниже описаны некоторые из упомянутых методов.

1. Опишем подробнее метод последовательных приближений, основанный на использовании метода динамического программирования для задачи (7.15). Зададимся произвольным допустимым управлением $u_0(t,x)$ (нулевым приближением к оптимальному), то есть управлением, при котором существует решение уравнения (7.10) и выполнено ограничение $u_0(t,x) \in U$. Решим далее линейное уравнение частных производных

$\mathbb{R}^n$. При некоторых предположениях функция $V(t,x)$ удовлетворяет соотношениям

$$\frac{\partial V_0(t,x)}{\partial t} + \min_{u\in U} f(t,x,u_0(t,x))\frac{\partial V_0(t,x)}{\partial x} = 0,$$

$$t_0 \le t \le t_*,\ x \in \mathbb{R}^n,$$

$$V_0(t_*,x) = \varphi(x).$$

Функция $V_0(t,x)$ представляет собой значение критерия качества при управлении u_0 и начальном условии $x(t) = x$. После того как функция V_0 определена, найдем следующее приближение $u_1(t,x)$ к оптимальному управлению из соотношения

$$\min_{u\in U} f(t,x,u)\frac{\partial V_0(t,x)}{\partial x} = f(t,x,u_1(t,x))\frac{\partial V_0(t,x)}{\partial x}.$$

Продолжая указанный процесс, можно построить последовательность управлений $u_k(t,x)$ и функций $V_k(t,x)$, которые в некоторых случаях (например, в линейно-квадратическом) сходятся к решению исходной задачи.

Описанная процедура естественным образом может быть использована и при применении принципа максимума к задаче (7.10). В этом случае необходимые условия оптимальности имеют вид краевой задачи (7.10)-(7.13). Зададимся опять некоторым допустимым управлением $u_0(t)$. Далее найдем при этом управлении соответствующую ему траекторию $x_0(t)$, решая уравнение (7.10) при $u(t) = u_0(t)$. Наконец, подставим $u_0(t)$ и x_0 в (7.11) и определим $\psi_0(t)$. Зная функции $x_0(t)$ и $\psi_0(t)$, определим управление $u_1(t)$ из соотношения

$$H(t, x_0(t), u_1(t), \psi_0(t)) = \max_{u \in U} H(t, x_0(t), u, \psi_0(t)).$$

Продолжая итерации, построим последовательности $x_k(t)$, $u_k(t)$, которые в некоторых ситуациях сходятся к решению исходной задачи оптимального управления (7.10).

2. Опишем метод решения задач оптимального управления с линейной динамикой, основанный на сведении исходной задачи оптимального управления к задаче математического программирования.

Рассмотрим задачу оптимального управления вида (7.10), в которой функция $f(x, u), x \in \mathbb{R}^n, u \in \mathbb{R}$, является линейной

$$\dot{x}(t) = Ax(t) + bu(t),\ u(t) \in [-1, 1],\ t_0 \le t \le t_*;\ \ x(t_0) = x_*, \tag{7.22}$$

$$g(x(t_*)) = 0,\ \ \Phi(x(t_*)) \to \min. \tag{7.23}$$

Из теории обыкновенных дифференциальных уравнений известно, что решение $x(t)$ системы (7.22) представимо в виде

$$x(t) = F(t, t_0)x_* + \int\limits_{t_0}^{t} F(t, \tau)bu(\tau)d\tau,\ t \in [t_0, t_*], \tag{7.24}$$

где $F(t, \tau)$ — фундаментальная матрица решений системы $\dot{x}(t) = Ax(t)$, которая удовлетворяет соотношениям

$$F(t, \tau) = F(t)F^{-1}(\tau),\ \ \dot{F}(t) = AF(t), F(t_0) = E. \tag{7.25}$$

Выберем параметр $N > 0$ и разобьем отрезок управления $[t_0, t_*]$ точками

$$t_j = t_0 + jh, i = 1, ..., N,\ h = (t_* - t_0)/N,$$

на N отрезков $[t_{j-1}, t_j], j = 1, ..., N$. Сузим класс допустимых управлений в задаче (7.22), а именно, будем искать решение задачи в классе кусочно-постоянных управлений с допустимыми точками разрыва в фиксированные моменты $t_j, j = 1, ..., N-1$, т.е.

$$u(t) = u_j,\ t \in [t_{j-1}, t_j), j = 1, ..., N;\ |u_j| \le 1, j = 1, ..., N. \tag{7.26}$$

Из (7.25), (7.26) получаем выражение для конечного состояния $x(t_*)$ нашей системы

$$x(t_*) = F(t_*, t_0)x_* + \int\limits_{t_0}^{t_*} F(t_*, \tau)bu(\tau)d\tau =$$

$$= B_0 + \sum_{j=1}^{N} \int\limits_{t_{j-1}}^{t_j} F(t_*, \tau)bu_j d\tau = B_0 + \sum_{j=1}^{N} B_j u_j = B_0 + BU, \tag{7.27}$$

где

$$B_0 = F(t_*, t_0)x_*,\ B_j = \int\limits_{t_{j-1}}^{t_j} F(t_*, \tau)bd\tau,\ j = 1, ..., N, \tag{7.28}$$

$$U = (u_j, j = 1, ..., N),\ \ B = (B_j, j = 1, ..., N). \tag{7.29}$$

Подставляя найденное представление $x(t_*) = B_0 + BU$ в (7.23), получаем следующую задачу нелинейного программирования: найти N-вектор $U = (u_j, j = 1, ..., N)$, который удовлетворяет ограничениям

$$\bar{g}(U) = 0, |u_j| \le 1, j = 1, ..., N, \tag{7.30}$$

и минимизирует целевую функцию

$$\bar{\Phi}(U) \to \min, \tag{7.31}$$

где

$$\bar{g}(U) = g(B_0 + BU),\ \bar{\Phi}(U) = \Phi(B_0 + BU).$$

Отметим, что для построения векторов B_0, B_j, $j = 1, ..., N$, матрицы B достаточно осуществить следующие построения.

A) Проинтегрировать систему дифференциальных уравнений

$$\dot{w}(t) = Aw(t),\ w(t_0) = x_*, t \in [t_0, t_N], \tag{7.32}$$

и найти вектор $w(t_N)$.

B) Проинтегрировать систему дифференциальных уравнений

$$\dot{y}(t) = Ay(t) + b,\ y(t_0) = 0, t \in [t_0, t_1], \tag{7.33}$$

и найти вектор $y(t_1)$.

C) Используя $y(t_1)$, проинтегрировать систему дифференциальных уравнений

$$\dot{z}(t) = Az(t),\ z(t_1) = y(t_1), t \in [t_1, t_N], \tag{7.34}$$

и найти значения $z(t_j)$, $j = 1, ..., N$, решения данной системы в точках t_j, $j = 1, ..., N$.

D) Положить

$$B_0 = w(t_N), \quad B_j = z(t_{N-j+1}),\ j = 1, ..., N.$$

Системы (7.32), (7.33) и (7.34) можно проинтегрировать с помощью стандартных математических пакетов.

Для решения задачи математического программирования можно использовать результаты, отписанные в предыдущих главах.

Отметим, также, что если функции $g(x)$ и $\Phi(x)$ являются линейными, т.е. $g(x) = Hx + h$, $\Phi(x) = c'x$, то задача (7.30), (7.31) становится задачей линейного программирования. В случае $g(x) = Hx + h$, $\Phi(x) = c'x + 0.5x'Dx$ задача (7.30), (7.31) является задачей квадратичного программирования.

Пример. Рассмотрим прямолинейное движение тела постоянной массы M. Координаты тела будем обозначать через x. При движении тела его координата x изменяется во времени. Производная $\dot{x}$ представляет собой скорость движения тела, а $\ddot{x}$ – ускорение.

Будем предполагать, что на тело действую две внешние силы:

сила трения — $-s\dot{x}$ и

сила упругости — $-kx$.

Кроме того, будем предполагать, что тела снабжено двигателем. Развиваемую двигателем силу воздействия на тело обозначим через u. Таким образом, согласно второму закону Ньютона движение тела с течением времени будет описываться дифференциальным уравнением

$$M\ddot{x} = -b\dot{x} - kx + u.$$

Как и в предыдущем примере, нетрудно показать, что это уравнение эквивалентно системе

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= -\frac{k}{M}x_1(t) - \frac{s}{M}x_2(t) + u(t). \end{aligned} \tag{7.35}$$

Значит в данном примере динамика системы описывается системой (7.8).

Предположим, что мы рассматриваем процесс на отрезке времени $t \in [0, t_*]$, где $t_* > 0$ — некоторый фиксированный момент времени.

Также предположим, что известно начальное состояние системы в момент времени $t = 0$: $x_1(0) = x_1^*$, $x_2(0) = x_2^*$, и есть ограничение на силу двигателя $|u(t)| \leq 1,\ t \in [0, t_*]$.

Требуется так управлять двигателем, чтобы в конечный момент времени $t = t_*$ наше тело находилось в состоянии, удовлетворяющем условию $h_1x_1(t_*)+h_2x_2(t_*) = h_0$ и терминальный критерий качества $c_1x_1(t_*) + c_2x_2(t_*)$ принимал минимальное значение. Таким образом получаем задачу оптимального управления следующего вида

$$\Phi(x_1(t_*), x_2(t_*)) \to \min,$$

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = A \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + bu(u),$$

$$x_1(0) = x_1^*,\ \ x_2(0) = x_2^*,\ \ g(x_1(t_*), x_2(t_*)) = 0,\ \ |u(t)| \leq 1,\ t \in [0, t_*],$$

где

$$\Phi(x_1(t_*), x_2(t_*)) = c_1x_1(t_*) + c_2x_2(t_*)),\ \ g(x_1(t_*), x_2(t_*)) = h_1x_1(t_*) + h_2x_2(t_*) - h_0,$$

$$A = \begin{pmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{s}{M} \end{pmatrix},\ b = \begin{pmatrix} 0 \\ \frac{1}{M} \end{pmatrix}.$$

Используя описанный выше подход сведения задачи оптимального управления к задаче нелинейного программирования, решим сформулированную сформулированную задачу при различных значениях исходных данных

$$k,\ s, M, t_*, x_1^*,\ x_2^*,\ g,\ c_1,\ c_2,\ h_1,\ h_2 = 0. \tag{7.36}$$

Отметим, что в данной задаче функции $g(x)$ и $\Phi(x)$ являются линейными, следовательно, задачи вида (7.30), (7.31) будут задачами линейного программирования.

Отметим также, что в рассматриваемой задаче для матрицы A фундаментальную матрицу $F(t,\tau)$ решений системы $\dot{x} = Ax$ можно найти в явном виде

$$F(t,\tau) = \begin{pmatrix} e^{-t+\tau} + (t-\tau)e^{-t+\tau} & (t-\tau)e^{-t+\tau} \\ -(t-\tau)e^{-t+\tau} & e^{-t+\tau} - (t-\tau)e^{-t+\tau} \end{pmatrix},$$

a) b)

Рис. 7.2:

Поэтому векторы $B_j, j = 0, 1, ..., N$, легдо находятся по формулам (7.28) без интегрирования систем (7.32), (7.33), (7.34).

I) Определим данные (7.36) следующим обзаром

$$k = 1,\ s = 2, M = 1, t_* = 8, x_1^* = x_2^* = g = 0, c_1 = 0, c_2 = 1, h_1 = 1, h_2 = 0, \tag{7.37}$$

и решим полученную задачу оптимального управления методом сведения к задаче линейного программирования. В результате получим оптимальное управление вида

$$u(t) = 1, t \in [0, \bar{t});\ \ u(t) = -1, t \in [\bar{t}, t_*),$$

где $\bar{t} = 6.3$.

II) Для данных

$$\begin{gathered} k = 1,\ s = 1, M = 1, t_* = 8, x_1^* = 0, \\ x_2^* = 3, g = 1.3, c_1 = 0, c_2 = 1, h_1 = 1, h_2 = 0 \end{gathered} \tag{7.38}$$

оптимальное управление (красная линия) приведено на Рис. 7.2, часть a). Оптимальное значение критерия качества равно $x_2^0(t_*) = -0.6441$.

III) Для данных

$$\begin{gathered} k = 1,\ s = 1, M = 1, t_* = 8, x_1^* = 0, \\ x_2^* = 3, g = 1.3, c_1 = 1, c_2 = 0, h_1 = 0, h_2 = 1 \end{gathered} \tag{7.39}$$

оптимальное управление (красная линия) приведено на Рис. 7.2, часть b). Оптимальное значение критерия качества равно $x_1^0(t_*) = -0.7079$.

На Рис. 7.2 голубой пунктирной линией приведена функция $\psi'(t)b,\ t \in [t_0, t_*]$, где $\psi(t), t \in [t_0, t_*]$, — решение сопряженной системы (7.11), (7.12) с соответствующим вектором y.

Список использованных источников к главе 7

[1] Колмановский В.Б. Задачи оптимального управления. Соросовский образовательный журнал. № 6, 1997.

[2] Винер Н. Кибернетика и общество. М.: Изд-во иностр. лит., 1958.

[3] Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов. М.: Наука, 1969.

[4] Беллман Р. Динамическое программирование. М.: Изд-во иностр. лит., 1960.

[5] Алексеев В.М., Тихомиров В.М., Фомин С.В. Оптимальное управление. М.: Наука, 1979.

[6] Красовский Н.Н. Теория управления движением. М.: Наука, 1968.

[7] Черноусько Ф.Л., Колмановский В.Б. Вычислительные и приближенные методы оптимального управления // Итоги науки и техники. Мат. анализ. 1977. Т. 14.

[8] Afanasiev V.N., Kolmanovskii V.B., Nosov V.R. Mathematical Theory of Control Systems Design. Dordrecht: Kluwer, 1996.

[9] Беллман Р. Методы вычислений // Автоматика и телемеханика. 1993. № 8. С. 10.

[10] Swan G.W. Application of Control Theory in Medicine. N.Y.: Dekker, 1984.

[11] Рудик А.П. Ядерные реакторы и принцип максимума Понтрягина. М.: Атомиздат, 1970.

[12] Ройтенберг Я.Н. Гироскопы. М.: Наука, 1975.

[13] Брокате М. Оптимальное управление системами гистерезисного типа // Автоматика и телемеханика. 1991. № 12; 1992. № 1.

СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ

1. Габасов Р., Кириллова Ф.М. Методы оптимизации: Учеб. пособие. Мн.: Изд-во БГУ, 1981.

2. Гороховик В.В. Конечномерные задачи оптимизации: Учеб. пособие. Мн.: Изд-во „Издательский центр БГУ“ , 2007.

3. Моисеев Н.Н., Иванилов Ю.П., Столяров Е.М. Методы оптимизации: Учеб. пособие. М.: Наука, 1978.

4. Ашманов С.А. Линейное программирование: Учеб. пособие. М. : Наука, 1981.

5. Карманов В.Г. Математическое программирование: Учеб. пособие. М.: Наука, 1980.

6. Васильев Ф.П. Численные методы решения экстремальных задач: Учеб. пособие. М.: Наука, 1980.

7. Nocedal J., Wright S.J. Numerical Optimization. Spriger-Verlag New York, 1999. 636 p.

8. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985.

9. Химмельблау Д. Прикладное нелинейное программирование. М.: Мир, 1975.

10. Полак Э. Численные методы оптимизации. М.: Мир, 1974.