15. МЕТОДИ ЛІНІЙНОГО ПРОГРАМУВАННЯ

15.1. Формулювання та види запису задач лінійного програмування

Задача лінійного програмування в **загальній формі** записується у вигляді:

$$f(x)=\sum_{j=1}^{n}c_jx_j \to \min_{x\in X} \tag{15.1}$$

при обмеженнях

$$X=\{x:\ \sum_{j=1}^{n}a_{ij}x_j=b_i,\ i=\overline{1,m_1},\ \sum_{j=1}^{n}a_{ij}x_j\le b_i,\ i=\overline{m_1+1,m},\ x_j\ge 0,\ j=\overline{1,n}\} \tag{15.2}$$

,

де $c_j\in R^1, j=\overline{1,n},\ a_{ij}\in R^1, i=\overline{1,m}, j=\overline{1,n};\ b_i\in R^1,\ i=\overline{1,m}$ – задані.

Зазначимо, що задача лінійного програмування найчастіше формулюється на пошук максимального значення цільової функції. Це пояснюється тим, що спочатку така постановка була використана для розв'язання економічних задач, де цільовою функцією зазвичай виступає прибуток.

Якщо в (15.2) $m_1=0$, тобто обмеження типу рівності відсутні

$$X=\{x:\ \sum_{j=1}^{n}a_{ij}x_j\le b_i,\ i=\overline{1,m},\ x_j\ge 0,\ j=\overline{1,n}\}, \tag{15.3}$$

то таке формулювання задачі лінійного програмування називають **стандартною формою.**

Якщо ж в (15.2) $m_1=m$, тобто відсутні обмеження типу нерівностей

$$X=\{x:\ \sum_{j=1}^{n}a_{ij}x_j=b_i,\ i=\overline{1,m},\ x_j\ge 0,\ j=\overline{1,n}\}, \tag{15.4}$$

то таке формулювання задачі лінійного програмування називають **канонічною формою**.

Якщо для канонічної форми задачі лінійного програмування (15.1), (15.4) ввести позначення

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix}, \; b = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_m \end{pmatrix}, \; c = \begin{pmatrix} c_1 \\ c_2 \\ \cdots \\ c_n \end{pmatrix}, \; x = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix},$$

де A – матриця розмірності $m \times n$, b – вектор-стовпець розмірності m ($b \in R^m$); c, x – вектори-стовпці розмірності n ($c, x \in R^n$), то її можна записати в матричному вигляді

$$f(x) = \langle c, x \rangle \to \min_{x \in X}; \tag{15.5}$$

$$X = \{x : Ax = b, x \geq 0\}. \tag{15.6}$$

У принципі будь-яка форма запису задачі лінійного програмування може бути приведена до канонічної [28].

15.2. Симплекс-метод

Основним чисельним методом розв'язання задач лінійного програмування є так званий **симплекс-метод**. Його теоретичні і обчислювальні аспекти добре розроблені [18; 24; 28]. Є безліч стандартних програм, що реалізовують симплекс-метод.

Слід зазначити, що термін "симплекс-метод" не відображає суті обчислювальної процедури. Цей термін пов'язаний з тією теоретичною обставиною, що спочатку метод був розроблений стосовно задачі лінійного програмування з допустимою множиною

$$X = \{ x : \sum_{i=1}^{n} x_i = 1, x \geq 0 \},$$

яка називається **стандартним симплексом**.

Симплекс-метод призначений для розв'язання задачі лінійного програмування в канонічній формі (15.5) – (15.6). Як вже наголошувалося раніше, будь-яка форма запису задачі лінійного програмування може бути приведена до канонічної.

Позначимо через $a^1, a^2, \ldots a^n$ стовпці матриці A, що визначає допустиму множину (15.5), і для будь-якої точки $x \in R^n$ введемо множину

$$J(x) = \{j : x_j > 0,\ 1 \le j \le n\},$$

тобто множину цілих чисел від 1 до n, що відповідають строго додатнім елементам вектора x.

Зазначимо, що якщо $x \in X$, то

$$b = Ax = \sum_{j=1}^{n} a^j x_j = \sum_{j \in J(x)} a^j x_j ,$$

тобто вектор b є додатньою ($x_j > 0$) лінійною комбінацією стовпців a^j, $j \in J(x)$.

Визначення. Точка $x \in X$ називається **опорною точкою допустимої множини** X, якщо стовпці a^j, $j \in J(x)$ лінійно незалежні.

Зазначимо, що допустима множина X для канонічної форми задачі лінійного програмування в просторі R^n є перетином гіперплощин, обмежених від'ємними значеннями змінних x_j. Таким чином, опорні точки – це крайні (кутові) точки цієї множини.

Справедливі наступні твердження [16; 18; 24; 28].

Твердження 1. Якщо допустима множина X задачі (15.5) – (15.6) не пуста, то вона містить опорні точки та їх число скінченне.

Твердження 2. Якщо множина розв'язків задачі (15.5) – (15.6) не пуста, то вона містить хоча б одну опорну точку допустимої множини X.

З цих тверджень виходить, що для пошуку розв'язків задачі (15.5) – (15.6) достатньо перебрати лише опорні точки допустимої множини X. Та з них, у якій цільова функція $f(x) = \langle c, x \rangle$ прийме максимальне значення і буде одним з розв'язків задачі (15.5) – (15.6). Проте, число опорних точок хоча і скінченне, але при великих n і m достатньо велике, тобто простий перебір опорних точок вимагає величезної обчислю-вальної роботи.

Таким чином, такий метод практичної цінності не має, але він приводить до основної ідеї симплекс-методу: повний перебір опорних точок слід замінити впорядкованим, розумним перебором. Бо, якщо вже є деяка опорна точка, то немає необхідності розглядати опорні точки, в яких значення цільової функції менше, ніж в тій, що є, оскільки вони вже не можуть бути розв'язками задачі (15.5) – (15.6).

Загальна схема симплекс-методу має наступний вигляд. Генерується послідовність $\{x^k\}$, $k = 0,1,2,\ldots$ опорних точок множини X і на черговому k-му кроці робиться вибір:

1. x^k є розв'язком задачі (15.5) – (15.6);
2. задача (15.5) – (15.6) не має розв'язку (тобто цільова функція не обмежена на множині X);
3. вказується "більш краща" опорна точка x^{k+1}, тобто $\langle c, x^{k+1} \rangle \le \langle c, x^k \rangle$.

Оскільки число опорних точок скінченне і серед них обов'язково є розв'язок задачі (15.5) – (15.6), то розв'язок (якщо він є) досягається за скінченне число кроків. Зазвичай потрібно від m до $2m$ кроків методу.

Надалі припускатимемо, що всі рядки матриці A лінійно незалежні, тобто лінійно залежні рядки виключені. Дане припущення, насправді, не важливе на практиці, оскільки лінійно залежні рядки виключаються в процесі обчислення початкової опорної точки x^0. Зазначимо також що, оскільки всі рядки матриці A лінійно незалежні, то число обмежень повинне бути менше, ніж число невідомих (тобто $m < n$). При $m = n$ допустима множина X складатиметься тільки з однієї точки, а значить, оптимізація цільової функції втрачає сенс.

Визначення. **Базисом опорної точки** x множини X називається довільна лінійно незалежна система з m стовпців матриці A, що включає всі стовпці a^j, відповідні додатнім координатам x_j.

Визначення. Опорна точка x множини X називається **невиродженою**, якщо вона має рівно m додатних координат x_j.

Визначення. Задача (15.5) – (15.6) називається невиродженою, якщо будь-яка опорна точка множини X не вироджена. В противному випадку, задача (15.5) – (15.6) вироджена.

Очевидно, що невироджена опорна точка має єдиний базис, який утворюють стовпці a^j, $j \in J(x)$. У виродженої опорної точки базисів багато.

15.3. Алгоритм симплекс-методу

Нехай вже отримана послідовність $x^0, x^1, \ldots, x^k$ опорних точок множини X. Скорочено покладемо $x^k = x$.

Нехай a^j, $j \in J(x)$ – деякий базис точки x. Оскільки це базис у просторі R^n, то всі стовпці матриці A можна представити як лінійну комбінацію стовпців a^j, $j \in J(x)$, тобто

$$a^p = \sum_{j \in J(x)} a^j \lambda_{jp}\ ,\ p = \overline{1,n}, \qquad (15.7)$$

де λ_{jp} – деякі числа. Покладемо

$$\Delta_p = \sum_{j \in J(x)} c_j \lambda_{jp} - c_p,\ p = \overline{1,n}. \qquad (15.8)$$

Очевидно, що для будь-якого $p \in J(x)$ $\lambda_{pp} = 1$, $\lambda_{jp} = 0$ при $j \in J(x)$ і $j \neq p$, а значить $\Delta_p = 0$.

Залежно від знаків чисел λ_{jp}, Δ_p при $j \in J(x)$, $j \neq p$, виконується одна з трьох умов:

I. Для всіх індексів $p \notin J(x)$ справедлива нерівність $\Delta_p \leq 0$.

II. Знайдеться $s \notin J(x)$ такий, що $\Delta_s > 0$ і $\lambda_{js} \leq 0$ для всіх $j \in J(x)$.

III. Знайдеться $s \notin J(x)$ такий, що для нього існує $\lambda_{js} > 0$, де $j \in J(x)$.

Для індексу $s \notin J(x)$, вказаного в умовах II та III, проведемо деякі побудови. Використовуючи формулу (15.7) при $p = s$, а також, враховуючи, що $x_j = 0$ при $j \notin J(x)$, для будь-якого $\alpha \in R^1$ маємо

$$b = \sum_{j \in J(x)} a^j x_j = \sum_{j \in J(x)} a^j x_j + \alpha(a^s - \sum_{j \in J(x)} a^j \lambda_{js}) =$$
$$= \sum_{j \in J(x)} a^j (x_j - \alpha\lambda_{js}) + \alpha a^s.$$

Розглянемо тепер точку $x' \in R^n$ з координатами

$$x'_j = \begin{cases} x_j - \alpha\lambda_{js},\ j \in J(x) \\ \alpha, \quad\quad j = s \\ 0,\ \ j \notin J(x), j \neq s \end{cases}. \qquad (15.9)$$

Тоді $b = \sum_{j \in J(x)} a^j x'_j = Ax'$. При цьому

$$\sum_{j=1}^{n} c_j x'_j = \sum_{j \in J(x)} c_j (x_j - \alpha\lambda_{js}) + c_s\alpha = \sum_{j \in J(x)} c_j x_j - \alpha(\sum_{j \in J(x)} c_j \lambda_{js} - c_s),$$

тобто

$$\langle c, x' \rangle = \langle c, x \rangle - \alpha\Delta_s. \qquad (15.10)$$

Справедливі наступні твердження [16; 28].

Твердження 1. Якщо виконується умова II, то задача (15.5) – (15.6) не має розв'язку.

Таким чином, при виконанні умов I і II робота симплекс-методу закінчується. Далі покладемо

$$\alpha = \min_{j \in J(x), \lambda_{js} > 0} \frac{x_j}{\lambda_{js}},$$

де s узято з умови III. Нехай мінімум досягається при $j = r$, тобто

$$\alpha = \frac{x_r}{\lambda_{rs}}, \quad r \in J(x), \lambda_{rs} > 0. \qquad (15.11)$$

Твердження 2. Якщо виконується умова III, то точка x' з (15.5), (15.7) є опорною точкою множини X, причому стовпці a^j, $j \in J'(x')$, де $J'(x') = (J(x) \setminus \{r\}) \cup \{s\}$, утворюють базис.

Таким чином, на $(k+1)$-й ітерації симплекс-методу в якості x^{k+1} приймається опорна точка x' з базисом a^j, $j \in J'(x')$. При цьому говорять, що стовпець a^r виводиться з базису, а стовпець a^s вводиться в базис. Елемент λ_{rs} називається **ведучим**. Отже, на наступній ітерації описана процедура повторюється вже для побудованої опорної точки x' з базисом a^j, $j \in J'(x')$. В першу чергу стовпці матриці A виражаються через a^j, $j \in J'(x')$:

$$a^p = \sum_{j \in J(x)} a^j \lambda'_{jp}, \; p = \overline{1,n},$$

а потім обчислюються значення

$$\Delta'_p = \sum_{j \in J'(x)} c_j \lambda'_{jp} - c_p, \; p = \overline{1,n}.$$

Числа λ'_{jp}, Δ'_p можуть бути визначені через вже відомі λ_{jp}, Δ_p.

Твердження 3. Для будь-яких $p = \overline{1,n}$ виконуються співвідношення

$$\lambda'_{jp} = \begin{cases} \lambda_{jp} - \dfrac{\lambda_{js}}{\lambda_{rs}} \lambda_{rp}, \; j \in J(x) \setminus \{r\}, \\ \dfrac{\lambda_{rp}}{\lambda_{rs}}, \quad j = s, \end{cases};$$

$$\Delta'_p = \Delta_p - \frac{\Delta_s}{\lambda_{rs}} \lambda_{rp}.$$

Корисно зазначити, що з урахуванням (15.11) формули (15.9) – (15.10) можна записати у вигляді

$$x'_j = \begin{cases} x_j - \dfrac{\lambda_{js}}{\lambda_{rs}} x_r, \; j \in J(x) \setminus \{r\}, \\ \dfrac{x_r}{\lambda_{rs}}, \quad j = s, \end{cases};$$

$$\langle c, x' \rangle = \langle c, x \rangle - \frac{\Delta_s}{\lambda_{rs}} x_r .$$

Таким чином, обчислення за симплекс-методом зводяться до достатньо простих рекурентних дій.

15.4. Метод штучного базису для пошуку початкової опорної точки

Описаний вище алгоритм розв'язання задачі (15.5) – (15.6) припускає наявність початкової опорної точки x^0. Її можна знайти **методом штучного базису**. Без обмеження спільності вважатимемо $b \geq 0$. Розглянемо допоміжну задачу лінійного програмування

$$\sum_{i=1}^{m} u_i \to \min_{\binom{x}{u} \in \hat{X}} ; \tag{15.12}$$

$$\hat{X} = \{ \begin{pmatrix} x \\ u \end{pmatrix} \in R^{n+m} : Ax + u = b, x \geq 0, u \geq 0 \}, \tag{15.13}$$

де $u \in R^m$ додаткові змінні. Ясно, що точка $\begin{pmatrix} x \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}$ є опорною точкою множини $\hat{X}$. Тоді, застосовуючи процедуру симплекс-методу до задачі (15.12) – (15.13), знайдемо розв'язок $\begin{pmatrix} x^* \\ u^* \end{pmatrix}$. Нехай $f^* = \sum_{i=1}^{m} u_i^*$.

Справедливо наступне твердження [4; 33].

Твердження. Якщо $f^* = 0$, то x^* – опорна точка множини X . Якщо $f^* > 0$, то задача (15.5) – (15.6) не має допустимих точок, тобто $X = \varnothing$.

15.5. Висновки

1. Одним з найефективніших методів розв'язання задач лінійного програмування є симплекс-метод.
2. Для застосування симплекс-методу необхідно знайти початкову опорну точку.

15.6. Контрольні запитання та завдання

1. Сформулюйте постановку задачі лінійного програмування в загальному вигляді.
2. Що називається канонічним видом задачі лінійного програмування?
3. Що називається стандартним видом задачі лінійного програмування?
4. Що називається опорною точкою?
5. У чому полягає ідея симплекс-методу?
6. Яким чином можна знайти початкову опорну точку?