Линейная оптимизация и метод генерации столбцов

Определение 1. *Задачей линейного программирования в канонической форме называется задача поиска максимума линейной функции*

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

заданной на множестве, описываемом системой линейных уравнений

$$\sum_{j=1}^{n} A_{ij} x_j = b_i, \qquad i = 1, \ldots, m, \tag{2}$$

$$x_j \geq 0, \qquad j = 1, \ldots, n, \tag{3}$$

где

A *— прямоугольная числовая матрица размером* m *на* n*,*

b *— числовой вектор из* m *компонент, называемый вектором правых ограничений,*

c *— числовой вектор из* n *компонент, называемый вектором коэффициентов целевой функции.*

$x \in \mathbf{R}^n$ *— вектор неизвестных задачи.*

Функция (1) называется *целевой функцией* задачи, множество переменных x, удовлетворяющих условиям (2)–(3) называется *допустимым множеством* задачи.

Для удобства изложения будем считать, что ранг матрицы A равен m.

Задачей, *двойственной* к задаче (1)–(3), является задача

$$\sum_{i=1}^{m} b_i y_i \longrightarrow \min, \tag{4}$$

$$\sum_{i=1}^{m} A_{ij} y_i \geq c_j, \qquad i = 1, \ldots, n, \tag{5}$$

где $y \in \mathrm{R}^n$ — вектор двойственных переменных.

Определение 2. *Решение* x' *системы (2)–(3) называется допустимым базисным решением, если столбцы матрицы* A*, соответствующие ненулевым элементам вектора* x'*, линейно независимы.*

Определение 3. *Пусть* x' *— допустимое базисное решение. Индексное множество* $I \subset \{1, \ldots, n\}$ *называется базисным, если* $|I| = m$ *и номера* j *всех положительных значений* x'_j *входят в* I $(x'_j > 0 \Rightarrow j \in I)$*.*

Определение 4. *Матрица $B[m][m]$, составленная из столбцов матрицы A с номерами из I, называется базисной матрицей, соответствующей базису I.*

Любой базисной матрице однозначно соответствует допустимое базисное решение.

Переменные допустимого базисного решения x' с номерами из соответствующего базисного множества I называются базисными. Остальные переменные называются небазисными.

Для реализации базисного множества I удобнее всего применять массив $t[m]$, содержащий номера индексов из I.

Для решения задачи (1)–(3) можно применить процедуру, известную как модифицированный симплекс-метод.

Опишем алгоритм применения симплекс-метода.

Шаг 0. Начальный шаг алгоритма состоит в поиске любого начального допустимого базисного решения x'. Пусть массив $t[m]$ содержит номера базисных переменных этого решения, $v[m]$ — массив значений базисных переменных, то есть $v[i] = x'_{t[i]}$, B — соответствующая обратная базисная матрица.

Шаг 1. Поиск переменной для ввода в базис.

Вычисляются значения двойственных переменных

$$y = c_B^T B,$$

где c_B — вектор коэффициентов целевой функции, соответствующих базисным переменным, т. е. $c_B[i] = c_{t[i]}$.

Затем подсчитываются оценки переменных

$$\Delta_j = \langle y, A_j \rangle - c_j,$$

где A_j — j-ый столбец матрицы A.

Отметим, что оценки базисных переменных равны нулю.

Пусть

$$\Delta_k = \min\{\Delta_j | j = 1, \ldots, n\}.$$

Если $\Delta_k \geq 0$, то следует перейти на шаг 4, иначе переход на шаг 2.

Шаг 2. Поиск переменной для вывода из базиса.

Вычисляется столбец

$$z = BA_k.$$

Если $z \leq 0$, то целевая функция задачи бесконечно возрастает на допустимом множестве. На этом процесс решения прекращается.

В противном случае обозначим

$$s = \operatorname{argmin} \left\{ \frac{v[i]}{z[i]} \middle| \, z[i] > 0, i = 1, \ldots, m \right\}.$$

и переходим на шаг 3.

Шаг 3. Переход к новому базису.

На этом шаге в базис вводится переменная с номером k взамен переменной с номером $t[s]$. Для этого выполним следующие действия.

1. $$t[s] = k.$$

2. $$v[s] = \frac{v[s]}{z[s]}.$$

3. Разделим строку s матрицы B на $z[s]$.

$$B[s][j] = \frac{B[s][j]}{z[s]}, \qquad j = 1, \ldots, m.$$

4. Вычтем из каждой строки i матрицы B строку s, умноженную на $z[i]$:

$$B[i][j] = B[i][j] - B[s][j] \cdot z[i], \qquad j = 1, \ldots, m, \; i = 1, \ldots, m.$$

5. Вычтем из каждого элемента $v[i]$ элемент $v[s]$, умноженный на $z[i]$:

$$v[i] = v[i] - v[s] \cdot z[i], \qquad i = 1, \ldots, m.$$

Переходим на шаг 1.

Шаг 4. Формирование ответа.

1. Изначально присвоим всем переменным нулевое значение $x = 0$.
2. Базисным переменным присвоим их значения:

$$x_{t[i]} = v[i], \qquad i = 1, \ldots, m.$$

Метод генерации столбцов

Симплексный метод — простое и удобное средство решения задач линейного программирования небольшой размерности, однако, если число столбцов матрицы ограничений задачи планирования раскроев велико, вычисления становятся малоэффективными. Основной недостаток симплексного метода — необходимость хранить и пересчитывать симплексную таблицу или обратную базисную матрицу, число столбцов которой соответствует огромному количеству планов раскроя. Множество переменных, которые при поиске оптимального плана будут хотя бы раз базисными, составляют малую часть по сравнению со всеми возможными планами раскроя, хранимыми и пересчитываемыми напрасно. Этот недостаток устраняется применением *метода генерации столбцов*.

Введем обозначения. Пусть:

m — количество заказов;

l_i — длина заготовки i-го заказа, $i = 1, \ldots, m$;

b_i — требуемое количество i-го заказа, $i = 1, \ldots, m$;

L — длина раскраиваемого стержня.

Обозначим через S множество всевозможных планов раскроя стержня на заготовки длиной $l_1, \ldots, l_m$. Каждый раскрой s можно описать вектором $A[s]$, в котором A_{is} — количество заготовок заказа i в раскрое. Обозначим через $A'_{is} = A_{is}\frac{l_i}{L}$ долю длины стержня, соответствующую заказчику i при способе раскроя s.

Доля отхода материала при использовании схемы s равна:

$$c_s = \frac{L - \sum_{i=1}^{m} A_{is} l_i}{L} = 1 - \sum_{i=1}^{m} A_{is}\frac{l_i}{L}, \qquad s \in S.$$

Неизвестные x_s в этой задаче — количество применений плана раскроя $s \in S$.

Математическая модель в случае критерия минимума отходов приобретает вид:

$$\sum_{s \in S} c_s x_s \to \min, \tag{6}$$

$$\sum_{s \in S} A'_{is} x_s \geq b_i, \qquad i = 1, \ldots, m, \tag{7}$$

$$x_s \geq 0, \qquad i = 1, \ldots, m. \tag{8}$$

Для проверки оптимальности допустимого базисного решения необходимо вычисление оценок Δ_s переменных x_s по формулам:

$$\Delta_s = \langle y, A'[s] \rangle - c_s.$$

Используя определение A'_{is} и c_s получим:

$$\Delta_s = \sum_{i=1}^{m} y_i A_{is}\frac{l_i}{L} - 1 + \sum_{i=1}^{m} A_{is}\frac{l_i}{L} = \sum_{i=1}^{m} (y_i + 1) A_{is}\frac{l_i}{L} - 1.$$

Для удобства обозначим $v_i = (y_i + 1)\frac{l_i}{L}$. Тогда

$$\Delta_s = \sum_{i=1}^{m} v_i A_{is} - 1.$$

Поиск максимальной оценки Δ_s эквивалентен поиску максимума линейной функции

$$\sum_{i=1}^{m} v_i \lambda_i,$$

где λ_i — переменная величина.

Для проверки оптимальности текущего базисного плана осуществляется решение задачи, называемой *простейшей задачей линейного раскроя*:

$L[m, v]$:

$$Z^* = \sum_{i=1}^{m} v_i \lambda_i \to \max, \tag{9}$$

$$\sum_{i=1}^{m} l_i \lambda_i \leq L, \tag{10}$$

$$\lambda_i \geq 0, \qquad \lambda_i \text{ — целые}, \qquad i = 1, \ldots m. \tag{11}$$

Оптимальное решение этой задачи Λ^* представляет собой раскрой с наибольшей оценкой для улучшения текущего базисного плана. Имеет место теорема.

Теорема 1. *Если выполняется соотношение*

$$Z^* \leq 1, \tag{12}$$

то текущий базисный план задачи (6)–(8) оптимален. Если

$$Z^* > 1,$$

то раскрой Λ^* является подходящим новым базисным элементом текущего решения задачи (6)–(8).

Доказательство очевидным образом вытекает из критерия оптимальности задачи линейного программирования.

Теорема 1 представляет собой обоснование алгоритма решения задачи (6)–(8):

Шаг 1. Построение S' — начального базисного плана (7)–(8).

Шаг 2. Решение задачи (9)–(11).

Шаг 3. Если условие (12) выполнено, то решение найдено. Иначе — ввод Λ^* в S' (шаги 2 и 3 симплекс-метода), переход на Шаг 2.