南京大學 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

线性规划中的高级主题

1. 修正单纯形法
2. 列生成算法

Ref.《Operations Research: Applications and Algorithms》Chapter 10

下料问题

某工厂生产一型号的机床，每台机床上分别需用 2.9、2.1、1.5米长的轴1根、2根和1根，这些轴需用同一种圆钢制作，圆钢的长度为7.4米。如需要生产100台机床，问应如何安排下料，才能使用料最省？试建立其线性规划模型。

	B1	B2	B3	B4	B5	B6	B7	B8	需要量
2.9m	1	2	0	1	0	1	0	0	100
2.1m	0	0	2	2	1	1	3	0	200
1.5m	3	1	2	0	3	1	0	4	100
余料	0	0.1	0.2	0.3	0.8	0.9	1.1	1.4	

x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8

下料问题

解：设 x_i 为按第 i 种方案切割的原材料根数

$$\min \sum_{i=1}^{8} x_i$$

$$s.t.\begin{cases} x_1 + 2x_2 + x_4 + x_6 \geq 100 \\ 2x_3 + 2x_4 + x_5 + x_6 + 3x_7 \geq 200 \\ 3x_1 + x_2 + 2x_3 + 3x_5 + x_6 + 4x_8 \geq 100 \\ x_i \geq 0, x_i\text{是整数} \end{cases}$$

Cutting Stock Problem

下料问题

解：设 x_i 为按第 i 种方案切割的原材料根数

$$\min \sum_{i=1}^{n} x_i$$

$$s.t. \begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \geq 100 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \geq 200 \\ a_{31}x_1 + a_{32}x_2 + \cdots + a_{3n}x_n \geq 100 \\ x_i \geq 0, \cancel{x_i \text{是整数}} \end{cases}$$

松弛问题

Cutting Stock Problem

线性规划中的高级主题

1. 修正单纯形法

(The Revised Simplex Algorithm)

Ref.《Operations Research: Applications and Algorithms》Chapter 10

1.1 符号

- BV = any set of basic variables (基变量)
- $\boldsymbol{b}$ = right-hand-side vector of the original tableau's constraints
- $\boldsymbol{a}_j$ = column for x_j in the constraints of the **original problem**
- $B = m \times m$ matrix whose j-th column is the column for BV_j in the **original constraints**
- c_j = coefficient of x_j in the objective function
- $\boldsymbol{c}_{BV} = 1 \times m$ row vector whose j-th element is the objective function coefficient for BV_j
- $\boldsymbol{u}_i = m \times 1$ column vector with i-th element 1 and all other elements equal to zero.

- In the tableau for any set of basic variables BV:
  - $B^{-1}\boldsymbol{a}_j$ = column for x_j in BV tableau
  - $c_j - \boldsymbol{c}_{BV}B^{-1}\boldsymbol{a}_j$ = coefficient of x_j in row 0
  - $B^{-1}\boldsymbol{b}$ = right-hand side of constraints in BV tableau
  - $\boldsymbol{c}_{BV}B^{-1}\boldsymbol{u}_i$ = coefficient of slack variable s_i in BV in row 0
  - $\boldsymbol{c}_{BV}B^{-1}\boldsymbol{b}$ = right-hand side of BV row 0
- If we know BV, B^{-1}, and the original tableau, the above formulas enable us to compute any part of the simplex tableau for any set of basic variables.

1.2 例子

$$\begin{aligned}\max\ \ z &= 60x_1 + 30x_2 + 20x_3\\ \text{s.t.}\quad 8x_1 + 6x_2 + x_3 &\le 48\\ 4x_1 + 2x_2 + 1.5x_3 &\le 20\\ 2x_1 + 1.5x_2 + 0.5x_3 &\le 8\end{aligned}$$

$$\Downarrow$$

$$\begin{aligned}\max\ \ z &= 60x_1 + 30x_2 + 20x_3\\ \text{s.t.}\quad 8x_1 + 6x_2 + x_3 + s_1 &= 48\\ 4x_1 + 2x_2 + 1.5x_3 + s_2 &= 20\\ 2x_1 + 1.5x_2 + 0.5x_3 + s_3 &= 8\end{aligned}$$

For the original tableau:

基变量 BV(0) = $\{s_1, s_2, s_3\}$, 非基变量 NBV(0) = $\{x_1, x_2, x_3\}$

1.2 例子

- We let B_i be the columns in the original LP that correspond to the basic variables for tableau i.

$$B_0^{-1} = B_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

We can now determine which non-basic variable should enter the basis by computing the coefficient of each non-basic variable in the current row 0.

Because $\boldsymbol{c}_{BV} = [0, 0, 0]$, we have:

$$\boldsymbol{c}_{BV} B_0^{-1} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

- For each non-basic variable $\{x_1, x_2, x_3\}$, we have:

$$\bar{\boldsymbol{c}}_j = c_j - \boldsymbol{c}_{BV} B^{-1} \boldsymbol{a}_j$$

= coefficient of x_j in row 0

$$\bar{c_1} = 60 - [0 \quad 0 \quad 0] \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix} = 60$$

✓ **x_1 should enter the basis**

$$\bar{c_2} = 30 - [0 \quad 0 \quad 0] \begin{bmatrix} 6 \\ 2 \\ 1.5 \end{bmatrix} = 30$$

$$\bar{c_3} = 20 - [0 \quad 0 \quad 0] \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix} = 20$$

- Which variable (s_1, s_2, or s_3) should leave the basis?
- We compute the column for x_1 in the current tableau and the right-hand side of the current tableau:

$$\text{Column for } x_1 \text{ in current tableau} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix}$$

$$\text{Right−hand side of current tableau} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 48 \\ 20 \\ 8 \end{bmatrix} = \begin{bmatrix} 48 \\ 20 \\ 8 \end{bmatrix}$$

Thus, the new tableau (tableau 1) will have BV(1) = $\{s_1, s_2, x_1\}$ and NBV(1) = $\{s_3, x_2, x_3\}$.

$$\begin{bmatrix} 48/8 \\ 20/4 \\ 8/2 \end{bmatrix} = \begin{bmatrix} 6 \\ 5 \\ 4 \end{bmatrix}$$

1.2 例子

南京大学 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

$$B_1 = \begin{bmatrix} 1 & 0 & 8 \\ 0 & 1 & 4 \\ 0 & 0 & 2 \end{bmatrix} \qquad B_1^{-1} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}$$

$$\boldsymbol{c}_{BV} B_1^{-1} = [0 \quad 0 \quad 60] \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} = [0 \quad 0 \quad 30]$$

$$\bar{c}_2 = 30 - [0 \quad 0 \quad 30] \begin{bmatrix} 6 \\ 2 \\ 1.5 \end{bmatrix} = -15$$

$$\bar{c}_3 = 20 - [0 \quad 0 \quad 30] \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix} = 5 \checkmark$$

x_3 should enter the basis

$$\text{Coefficient of } s_3 \text{ in row } 0 = 0 - [0 \quad 0 \quad 30] \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = -30$$

1.2 例子

$$x_3 \text{ column in tableau } 1 = B_1^{-1}\boldsymbol{a}_3 = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} -1 \\ 0.5 \\ 0.25 \end{bmatrix}$$

$$\text{Right-hand side of tableau } 1 = B_1^{-1}\boldsymbol{b} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 48 \\ 20 \\ 8 \end{bmatrix} = \begin{bmatrix} 16 \\ 4 \\ 4 \end{bmatrix}$$

Thus, the new tableau (tableau 2) will have BV(2) = $\{s_1, x_3, x_1\}$ and NBV(2) = $\{s_3, x_2, s_2\}$.

$$B_2 = \begin{bmatrix} 1 & 1 & 8 \\ 0 & 1.5 & 4 \\ 0 & 0.5 & 2 \end{bmatrix} \qquad B_2^{-1} = \begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & -0.5 & 1.5 \end{bmatrix}$$

1.2 例子

$$\boldsymbol{c}_{BV}B_2^{-1} = [0 \quad 20 \quad 60]\begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & -0.5 & 1.5 \end{bmatrix} = [0 \quad 10 \quad 10]$$

$$\bar{c_2} = 30 - [0 \quad 10 \quad 10]\begin{bmatrix} 6 \\ 2 \\ 1.5 \end{bmatrix} = -5$$

$$\text{Coefficient of } s_2 \text{ in row } 0 = 0 - [0 \quad 10 \quad 10]\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = -10 \qquad all \le 0$$

$$\text{Coefficient of } s_3 \text{ in row } 0 = 0 - [0 \quad 10 \quad 10]\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = -10$$

$$\text{Optimal solution: } B_2^{-1}\boldsymbol{b} = \begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & -0.5 & 1.5 \end{bmatrix}\begin{bmatrix} 48 \\ 20 \\ 8 \end{bmatrix} = \begin{bmatrix} 24 \\ 8 \\ 2 \end{bmatrix}$$

1.3 算法步骤

A summary of the revised simplex method (for a max problem) follows:

Step 0 Note the columns from which the current B^{-1} will be read. Initially, $B^{-1} = I$.

Step 1 For the current tableau, compute $\mathbf{c}_{\mathrm{BV}}B^{-1}$.

Step 2 Price out all nonbasic variables in the current tableau. If each nonbasic variable prices out to be nonnegative, then the current basis is optimal. If the current basis is not optimal, then enter into the basis the nonbasic variable with the most negative coefficient in row 0. Call this variable x_k.

Step 3 To determine the row in which x_k enters the basis, compute x_k's column in the current tableau ($B^{-1}\mathbf{a}_k$) and compute the right-hand side of the current tableau ($B^{-1}\mathbf{b}$). Then use the ratio test to determine the row in which x_k should enter the basis. We now know the set of basic variables (BV) for the new tableau.

Step 4 Use the column for x_k in the current tableau to determine the EROs needed to enter x_k into the basis. Perform these EROs on the current B^{-1}. This will yield the new B^{-1}. Return to step 1.

(Elementary Row Operations：初等行变换)

1.4 注记

- Most linear programming computer codes use some version of the revised simplex to solve LPs.
- Knowing the current tableau's B^{-1} and the initial tableau is all that is needed to obtain the next tableau.
- The computational effort required to solve an LP by the revised simplex depends primarily on the size of B^{-1}.

南京大學 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

线性规划中的高级主题

2. 逆矩阵的乘积形式 (The Product Form of the Inverse)

2.1 计算公式

- Much of the computation in the revised simplex algorithm is concerned with updating B^{-1} from one tableau to the next.
- Suppose we are solving an LP with m constraints.
- Assume that we have found that x_k should enter the basis, in row r. Let the column for x_k in the current tableau be

$$\begin{bmatrix} \bar{a}_{1k} \\ \bar{a}_{2k} \\ \vdots \\ \bar{a}_{mk} \end{bmatrix}$$

Define the $m \times m$ matrix E:

E is simply I_m with column r replaced by the column vector:

$$
E = \begin{array}{c@{}l}
\begin{bmatrix}
1 & 0 & \cdots & -\dfrac{\bar{a}_{1k}}{\bar{a}_{rk}} & \cdots & 0 & 0 \\
0 & 1 & \cdots & -\dfrac{\bar{a}_{2k}}{\bar{a}_{rk}} & \cdots & 0 & 0 \\
\vdots & \vdots & & \vdots & & \vdots & \vdots \\
0 & 0 & \cdots & \dfrac{1}{\bar{a}_{rk}} & \cdots & 0 & 0 \\
\vdots & \vdots & & \vdots & & \vdots & \vdots \\
0 & 0 & \cdots & -\dfrac{\bar{a}_{m-1,k}}{\bar{a}_{rk}} & \cdots & 1 & 0 \\
0 & 0 & \cdots & -\dfrac{\bar{a}_{mk}}{\bar{a}_{rk}} & \cdots & 0 & 1
\end{bmatrix}
&
\begin{array}{l}
\\ \\ \\ (\text{row } r) \\ \\ \\ \\
\end{array}
\\
\text{(column } r\text{)} &
\end{array}
$$

2.1 计算公式

- Define the initial tableau to be tableau 0, and let E_i be the matrix associated with the i-th simplex tableau.
  - $B_0^{-1} = I_m$
  - $B_1^{-1} = E_0 B_0^{-1} = E_0$
  - $B_2^{-1} = E_1 B_1^{-1} = E_1 E_0$
  - In general, $B_k^{-1} = E_{k-1} E_{k-2} \dots E_1 E_0$

2.2 例子

$$\max\ z = 60x_1 + 30x_2 + 20x_3$$

$$s.t.\begin{cases} 8x_1 + 6x_2 + x_3 + s_1 & = 48 \\ 4x_1 + 2x_2 + 1.5x_3 + s_2 & = 20 \\ 2x_1 + 1.5x_2 + 0.5x_3 + s_3 & = 8 \end{cases}$$

Recall that in tableau 0, x_1 entered the basis in row 3. Thus, for tableau 0, $r = 3$ and $k = 1$.

$$\begin{bmatrix} \bar{a}_{11} \\ \bar{a}_{21} \\ \bar{a}_{31} \end{bmatrix} = \begin{bmatrix} 8 \\ 4 \\ 2 \end{bmatrix} \quad E_0 = \begin{bmatrix} 1 & 0 & -\frac{8}{2} \\ 0 & 1 & -\frac{4}{2} \\ 0 & 0 & \frac{1}{2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}$$

$$\mathrm{BV}(0) = \{s_1, s_2, s_3\}$$

$$\mathrm{BV}(1) = \{s_1, s_2, x_1\}$$

$$B_1^{-1} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}$$

2.2 例子

- As we proceeded from tableau 1 to tableau 2, x_3 entered the basis in row 2.
- Hence, in computing E_1, we set $r = 2$ and $k = 3$.
- To compute E_1, we need to find the column for the entering variable (x_3) in tableau:

$$\begin{bmatrix} \bar{a}_{13} \\ \bar{a}_{23} \\ \bar{a}_{33} \end{bmatrix} = B_1^{-1}\boldsymbol{a}_3 = \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} -1 \\ 0.5 \\ 0.25 \end{bmatrix}$$

2.2 例子

As before, x_3 enters the basis in row 2. Then

$$E_1 = \begin{bmatrix} 1 & -\left(\frac{-1}{0.5}\right) & 0 \\ 0 & \frac{1}{0.5} & 0 \\ 0 & -\frac{0.25}{0.5} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 2 & 0 \\ 0 & -0.5 & 1 \end{bmatrix}$$

$$B_2^{-1} = E_1 B_1^{-1} = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 2 & 0 \\ 0 & -0.5 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 0.5 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -8 \\ 0 & 2 & -4 \\ 0 & -0.5 & 1.5 \end{bmatrix}$$

南京大學 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

线性规划中的高级主题

3.利用列生成求解大规模线性规划

(Using Column Generation to Solve Large-Scale LPs)

- Cutting Stock Problem
- Woodco sells 3-ft, 5-ft, and 9-ft pieces of lumber.
- Woodco's customers demand 25 3-ft boards, 20 5-ft boards, and 15 9-ft boards.
- Woodco, who must meet its demands by cutting up 17-ft boards, wants to minimize the waste incurred.
- Formulate an LP to help Woodco accomplish its goal, and solve the LP by column generation.

Ways to Cut a Board in the Cutting Stock Problem

Combination	Number of 3-ft Boards	Number of 5-ft Boards	Number of 9-ft Boards	Waste (Feet)
1	5	0	0	2
2	4	1	0	0
3	2	2	0	1
4	2	0	1	2
5	1	1	1	0
6	0	3	0	2

利用列生成求解大规模线性规划

- Define: x_i= number of 17-ft boards cut according to combination i.
- Minimize the waste incurred = minimize the total number of 17-ft boards that are cut.
- Constraint 1: at least 25 3-ft boards must be cut
- Constraint 2: at least 20 5-ft boards must be cut
- Constraint 3: at least 15 9-ft boards must be cut

$$
\begin{aligned}
\min\ \ z = {} & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \\
\text{s.t.}\quad & 5x_1 + 4x_2 + 2x_3 + 2x_4 + x_5 \geq 25 && \text{(3-ft constraint)} \\
& x_2 + 2x_3 + x_5 + 3x_6 \geq 20 && \text{(5-ft constraint)} \\
& x_4 + x_5 \geq 15 && \text{(9-ft constraint)} \\
& x_1, x_2, x_3, x_4, x_5, x_6 \geq 0
\end{aligned}
$$

利用列生成求解大规模线性规划

- It is clear that the x_i should be required to assume integer values.
- In problems with large demand, a near-optimal solution can be obtained by solving the cutting stock problem as an LP and then rounding all fractional variables upward.
- The above procedure may not yield the best possible integer solution, but it usually yields a near optimal integer solution.

$$
\begin{aligned}
\min\ \ & z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \\
\text{s.t.}\ \ & \boxed{5x_1} + \boxed{4x_2} + \boxed{2x_3} + \boxed{2x_4} + \boxed{x_5} \ge 25 \quad \text{(3-ft constraint)} \\
& x_2 + 2x_3 + x_5 + 3x_6 \ge 20 \quad \text{(5-ft constraint)} \\
& x_4 + x_5 \ge 15 \quad \text{(9-ft constraint)} \\
& x_1, x_2, x_3, x_4, x_5, x_6 \ge 0
\end{aligned}
$$

利用列生成求解大规模线性规划

$$\min\ z = x_1 + \square + \square + \square + \square + x_6 + x_7 + \square + \cdots + \square$$

$$
\begin{aligned}
s.t.\ & \boxed{\begin{matrix}5x_1\\0x_1\\0x_1\end{matrix}} \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \boxed{\begin{matrix}0x_6\\3x_6\\0x_6\end{matrix}} \begin{matrix}+\\+\\+\end{matrix} \boxed{\begin{matrix}0x_7\\0x_7\\1x_7\end{matrix}} \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}+\\+\\+\end{matrix} \square \begin{matrix}\geq 25\\ \geq 20\\ \geq 15\end{matrix}
\end{aligned}
$$

Basic variables (BV) $\boldsymbol{x}_{\mathrm{BV}} = \{x_1, x_6, x_7\}$

We let the tableau for this basis be tableau 0, then we have:

$$B_0 = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad B_0^{-1} = \begin{bmatrix} \frac{1}{5} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Shadow price

$$\boldsymbol{c}_{BV} B_0^{-1} = [1 \quad 1 \quad 1] \begin{bmatrix} \frac{1}{5} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{5} & \frac{1}{3} & 1 \end{bmatrix}$$

利用列生成求解大规模线性规划

$$\boldsymbol{x}_{BV} = B_0^{-1}\boldsymbol{b}$$

$$\boldsymbol{a}_k = \left(a_k^1, a_k^2, a_k^3\right)^T$$

Reduced cost for each k

$$\bar{c}_k = c_k - \boldsymbol{c}_{BV} B_0^{-1} \boldsymbol{a}_k$$

$$1 - \boldsymbol{c}_{BV} B_0^{-1} \begin{bmatrix} a_k^1 \\ a_k^2 \\ a_k^3 \end{bmatrix} = 1 - \frac{1}{5} a_k^1 - \frac{1}{3} a_k^2 - a_k^3$$

Optimal conditions: all reduced costs are non-negative (最小化问题)

- We need to find the $\boldsymbol{a}_k$ with minimum reduced cost, that is:

$$\bar{c}^* = min_k \left\{1 - \frac{1}{5}a_k^1 - \frac{1}{3}a_k^2 - a_k^3\right\} = max_k \left\{\frac{1}{5}a_k^1 + \frac{1}{3}a_k^2 + a_k^3 - 1\right\}$$

- If the minimum reduced cost is non-negative, then the current solution is optimal.

a_k^1, a_k^2, a_k^3 must be chosen so they don't use more than 17 ft of wood.

We also know that a_k^1, a_k^2, a_k^3 must be non-negative integers.

For any combination, a_k^1, a_k^2 and a_k^3 must satisfy:

$$3a_k^1 + 5a_k^2 + 9a_k^3 \leq 17 \ (a_k^1 \geq 0,\ a_k^2 \geq 0,\ a_k^3 \geq 0 \ \text{and integer})$$

利用列生成求解大规模线性规划

- We can find the column $\boldsymbol{a}_k$ with maximum reduced cost by the following **knapsack problem**:

$$
\begin{aligned}
\max \quad & z = \frac{1}{5}a^1 + \frac{1}{3}a^2 + a^3 - 1 \\
s.t. \quad & 3a^1 + 5a^2 + 9a^3 \leq 17 \\
& a^1, a^2, a^3 \geq 0 \text{ and integer}
\end{aligned}
$$

- Note that a^1, a^2, a^3 now are decision variables in this problem.

- That is, we use this knapsack problem to check the reduced cost of all columns $\boldsymbol{a}_k$.
- The solution of this knapsack problem must correspond to a column.

利用列生成求解大规模线性规划

$$\max \quad z = \frac{1}{5}a^1 + \frac{1}{3}a^2 + a^3 - 1$$

$$s.t. \qquad 3a^1 + 5a^2 + 9a^3 \leq 17$$

$$a^1, a^2, a^3 \geq 0 \text{ and integer}$$

Optimal Solution: $z = 8/15 \geq 0$, $(a^1, a^2, a^3)^T = (1,1,1)^T$

利用列生成求解大规模线性规划

$$\min\ z = x_1 + \boxed{} + \boxed{} + \boxed{} + \boxed{} + x_6 + x_7 + \boxed{} + \cdots + \boxed{}$$

$$\begin{aligned}
s.t.\ 5x_1 + \boxed{} + \boxed{} + \boxed{} + \boxed{} + 0x_6 + 0x_7 + \boxed{} + \boxed{} + \boxed{} &\geq 25 \\
0x_1 + \boxed{} + \boxed{} + \boxed{} + \boxed{} + 3x_6 + 0x_7 + \boxed{} + \boxed{} + \boxed{} &\geq 20 \\
0x_1 + \boxed{} + \boxed{} + \boxed{} + \boxed{} + 0x_6 + 1x_7 + \boxed{} + \boxed{} + \boxed{} &\geq 15
\end{aligned}$$

利用列生成求解大规模线性规划

We assume the optimal solution of the knapsack problem corresponds to the fifth column.

$$\min\ z = x_1 + \square + \square + \square + \boxed{x_5} + x_6 + x_7 + \square + \cdots + \square$$

$$\begin{array}{lllllllllll}
s.t. & 5x_1 + & \square + & \square + & \square + & 1x_5 & + 0x_6 + 0x_7 + & \square + & \square + & \square & \geq 25 \\
 & 0x_1 + & \square + & \square + & \square + & 1x_5 & + 3x_6 + 0x_7 + & \square + & \square + & \square & \geq 20 \\
 & 0x_1 + & \square + & \square + & \square + & 1x_5 & + 0x_6 + 1x_7 + & \square + & \square + & \square & \geq 15
\end{array}$$

Will enter into the basis.

$$x_5 \text{ column in current tableau} = B_0^{-1}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}\frac{1}{5} & 0 & 0\\ 0 & \frac{1}{3} & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}\frac{1}{5}\\ \frac{1}{3}\\ 1\end{bmatrix}$$

$$\text{Right} - \text{hand side of current tablea}u \ = B_0^{-1}\boldsymbol{b} = \begin{bmatrix}\frac{1}{5} & 0 & 0\\ 0 & \frac{1}{3} & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}25\\20\\15\end{bmatrix} = \begin{bmatrix}5\\ \frac{20}{3}\\ 15\end{bmatrix}$$

Which variable should leave the basis?

利用列生成求解大规模线性规划

$$\min\ z = x_1 + \square + \square + \square + x_5 + x_6 + x_7 + \square + \square + \square$$

$$\begin{aligned} s.t.\ & \boxed{5x_1} + \square + \square + \square + \boxed{1x_5} + \boxed{0x_6} + 0x_7 + \square + \square + \square \geq 25 \\ & \boxed{0x_1} + \square + \square + \square + \boxed{1x_5} + \boxed{3x_6} + 0x_7 + \square + \square + \square \geq 20 \\ & \boxed{0x_1} + \square + \square + \square + \boxed{1x_5} + \boxed{0x_6} + 1x_7 + \square + \square + \square \geq 15 \end{aligned}$$

The new basic variables are $\{x_1, x_6, x_5\}$

$$B_1^{-1} = E_0 B_0^{-1} = \begin{bmatrix} 1 & 0 & -\frac{1}{5} \\ 0 & 1 & -\frac{1}{3} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{5} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{5} & 0 & -\frac{1}{5} \\ 0 & \frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{c}_{BV} B_1^{-1} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{5} & 0 & -\frac{1}{5} \\ 0 & \frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{5} & \frac{1}{3} & \frac{7}{15} \end{bmatrix}$$

利用列生成求解大规模线性规划

南京大学 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

$$\boldsymbol{c}_{BV}B_1^{-1}\begin{bmatrix} a_k^1 \\ a_k^2 \\ a_k^3 \end{bmatrix} - 1 = \frac{1}{5}a_k^1 + \frac{1}{3}a_k^2 + \frac{7}{15}a_k^3 - 1$$

For the current tableau, the column generation procedure yields
The following problem:

$$\begin{aligned} \max \quad & \mathrm{z} = \frac{1}{5}a^1 + \frac{1}{3}a^2 + \frac{7}{15}a^3 - 1 \\ s.t. \quad & 3a^1 + 5a^2 + 9a^3 \le 17 \\ & a^1, a^2, a^3 \ge 0 \text{ and integer} \end{aligned}$$

Optimal solution 2/15 and (a^1, a^2, a^3)= (4, 1, 0)

利用列生成求解大规模线性规划

We assume the optimal solution of the knapsack problem corresponds to the second column.

$$\min\ z = x_1 + \boxed{x_2} + \square + \square + \square + x_6 + x_7 + \square + \square + \square$$

$$\begin{aligned} s.t\quad & 5x_1 + 4x_2 + \square + \square + 1x_5 + 0x_6 + 0x_7 + \square + \square + \square \geq 25 \\ & 0x_1 + 1x_2 + \square + \square + 1x_5 + 3x_6 + 0x_7 + \square + \square + \square \geq 20 \\ & 0x_1 + 0x_2 + \square + \square + 1x_5 + 0x_6 + 1x_7 + \square + \square + \square \geq 15 \end{aligned}$$

Will enter into the basis.

The column for x_2 in the current tableau is:

$$B_1^{-1}\begin{bmatrix}4\\1\\0\end{bmatrix}=\begin{bmatrix}\frac{1}{5} & 0 & -\frac{1}{5}\\ 0 & \frac{1}{3} & -\frac{1}{3}\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}4\\1\\0\end{bmatrix}=\begin{bmatrix}\frac{4}{5}\\ \frac{1}{3}\\ 0\end{bmatrix}$$

The right-hand side of the current tableau is:

$$B_1^{-1}\boldsymbol{b}=\begin{bmatrix}\frac{1}{5} & 0 & -\frac{1}{5}\\ 0 & \frac{1}{3} & -\frac{1}{3}\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}25\\20\\15\end{bmatrix}=\begin{bmatrix}2\\ \frac{5}{3}\\ 15\end{bmatrix}$$

Which variable should leave the basis?

利用列生成求解大规模线性规划

$$\min\ z = x_1 + x_2 + \square + \square + x_5 + x_6 + x_7 + \square + \square + \square$$

$$
\begin{array}{llllllllllll}
s.t & 5x_1 + & 4x_2 + & \square + & \square + & 1x_5 + & 0x_6 + & 0x_7 + & \square + & \square + & \square & \geq 25 \\
 & 0x_1 + & 1x_2 + & \square + & \square + & 1x_5 + & 3x_6 + & 0x_7 + & \square + & \square + & \square & \geq 20 \\
 & 0x_1 + & 0x_2 + & \square + & \square + & 1x_5 + & 0x_6 + & 1x_7 + & \square + & \square + & \square & \geq 15
\end{array}
$$

The new basic variables are $\{x_2, x_6, x_5\}$

$$B_2^{-1} = E_1 B_1^{-1} = \begin{bmatrix} \frac{5}{4} & 0 & 0 \\ -\frac{5}{12} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{5} & 0 & -\frac{1}{5} \\ 0 & \frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & 0 & -\frac{1}{4} \\ -\frac{1}{12} & \frac{1}{3} & -\frac{1}{4} \\ 0 & 0 & 1 \end{bmatrix}$$

The new set of shadow prices is given by

$$\boldsymbol{c}_{BV}B_2^{-1} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}\begin{bmatrix}\frac{1}{4} & 0 & -\frac{1}{4} \\ -\frac{1}{12} & \frac{1}{3} & -\frac{1}{4} \\ 0 & 0 & 1\end{bmatrix} = \begin{bmatrix}\frac{1}{6} & \frac{1}{3} & \frac{1}{2}\end{bmatrix}$$

Thus, the column-generation procedure requires us to solve the following problem:

$$\begin{aligned} \max \quad & z = \frac{1}{6}a^1 + \frac{1}{3}a^2 + \frac{1}{2}a^3 - 1 \\ s.t. \quad & 3a^1 + 5a^2 + 9a^3 \le 17 \\ & a^1, a^2, a^3 \ge 0 \text{ and integer} \end{aligned}$$

The optimal z-value for the above model is found to be $z = 0$.

利用列生成求解大规模线性规划

南京大学 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

- The optimal z-value for the above model is found to be $z = 0$.
- The current basic solution must be an optimal solution.
- To find the values of the basic variables in the optimal solution, we find the right-hand side of the current tableau:

$$B_2^{-1}\boldsymbol{b} = \begin{bmatrix} \frac{1}{4} & 0 & -\frac{1}{4} \\ -\frac{1}{12} & \frac{1}{3} & -\frac{1}{4} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} = \begin{bmatrix} \frac{5}{2} \\ \frac{5}{6} \\ 15 \end{bmatrix}$$

- Thus, x_2= 5/2, x_6 = 5/6 and x_5=15 is the optimal solution to Woodco's cutting stock problem (relaxed version).

利用列生成求解大规模线性规划

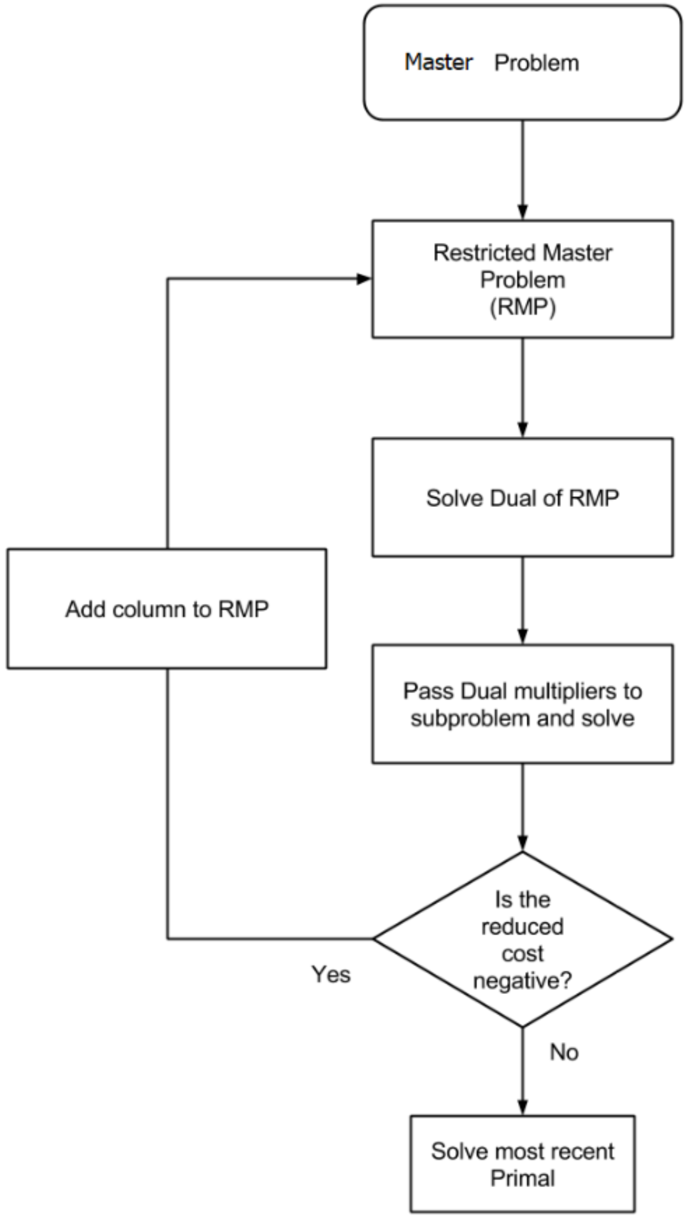
南京大学 NANJING UNIVERSITY 工程管理学院 SCHOOL OF MANAGEMENT & ENGINEERING

- We need not list all possible ways in which a board may be cut.

- At each iteration, a good combination (one that will improve the z-value when entered into the basis) is generated by solving a branch-and-bound problem.

- A cutting stock problem that was solved in Gilmore and Gomory (1961) involves more than 100 million possible combinations.

列生成算法框架



$$\boldsymbol{c}_{BV}B^{-1}$$

第三章 运输问题

- 运输问题及其数学模型
- 表上作业法
- 运输问题的进一步讨论
- 应用问题举例

第三章 运输问题

1. 运输问题及其数学模型

1.1 运输问题的提出

- 在经济建设中，经常会遇到大宗物资调拨的运输问题，如煤炭、钢铁、木材、粮食等。
- 从产地到销地之间运送货物：
  - 多个产地、多个销地
  - 每个产地的产量不同，每个销地的销量也不同
  - 各产销地之间的运价不同
- 如何组织调运，才能既满足各销地的需求，又使总的运输费用（或里程、时间等）最小。

1.1 运输问题的提出

设有同一种货物从 m 个产地 1, 2, …, m 运往 n 个销地 1, 2, …, n。第 i 个产地的供应量 (Supply) 为 $a_i(a_i \geq 0)$，第 j 个销地的需求量 (Demand) 为 $b_j(b_j \geq 0)$。每单位货物从产地 i 运到销地 j 的运价为 c_{ij}。求一个使总运费最小的运输方案。

D 销地 / O 产地	1	2	…	n	Supply 产量
1	c_{11}	c_{12}	…	c_{1n}	a_1
2	c_{21}	c_{22}	…	c_{2n}	a_2
…	…	…	…		…
m	c_{m1}	c_{m2}	…	c_{mn}	a_m
Demand 销量	b_1	b_2	…	b_n	

三类运输问题

产销平衡： $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

产大于销： $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$

产小于销： $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$

1.2 运输问题的模型

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

$$\begin{cases} \sum_{j=1}^{n} x_{ij} = a_i, & i = 1,2,\ldots,m \\ \sum_{i=1}^{m} x_{ij} = b_j, & j = 1,2,\ldots,n \\ x_{ij} \geq 0 \end{cases}$$

产销平衡条件 $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$ 自然满足！

1.3 运输问题—举例1

某饮料在国内有三个生产厂，分布在城市A_1、A_2、A_3,其一级承销商有4个，分布在城市B_1、B_2、B_3、B_4，已知各厂的产量、各承销商的销售量及从A_i到B_j的每吨饮料运费为C_{ij}，为发挥集团优势，公司要统一筹划运销问题，求运费最小的调运方案。

销地 产地	B_1	B_2	B_3	B_4	产量
A_1	6	3	2	5	5
A_2	7	5	8	4	2
A_3	3	2	9	7	3
销量	2	3	1	4	

1.3 运输问题—举例1

（1）决策变量：设从 A_i 到 B_j 的运输量为 x_{ij}

（2）目标函数：

$$\min\ z = 6x_{11} + 3x_{12} + 2x_{13} + 5x_{14} + 7x_{21} + 5x_{22} + 8x_{23}$$
$$+4x_{24} + 3x_{31} + 2x_{32} + 9x_{33} + 7x_{34}$$

（3）约束条件：产量之和等于销量之和（产销平衡）

- 供应平衡条件

$$x_{11} + x_{12} + x_{13} + x_{14} = 5$$
$$x_{21} + x_{22} + x_{23} + x_{24} = 2$$
$$x_{31} + x_{32} + x_{33} + x_{34} = 3$$

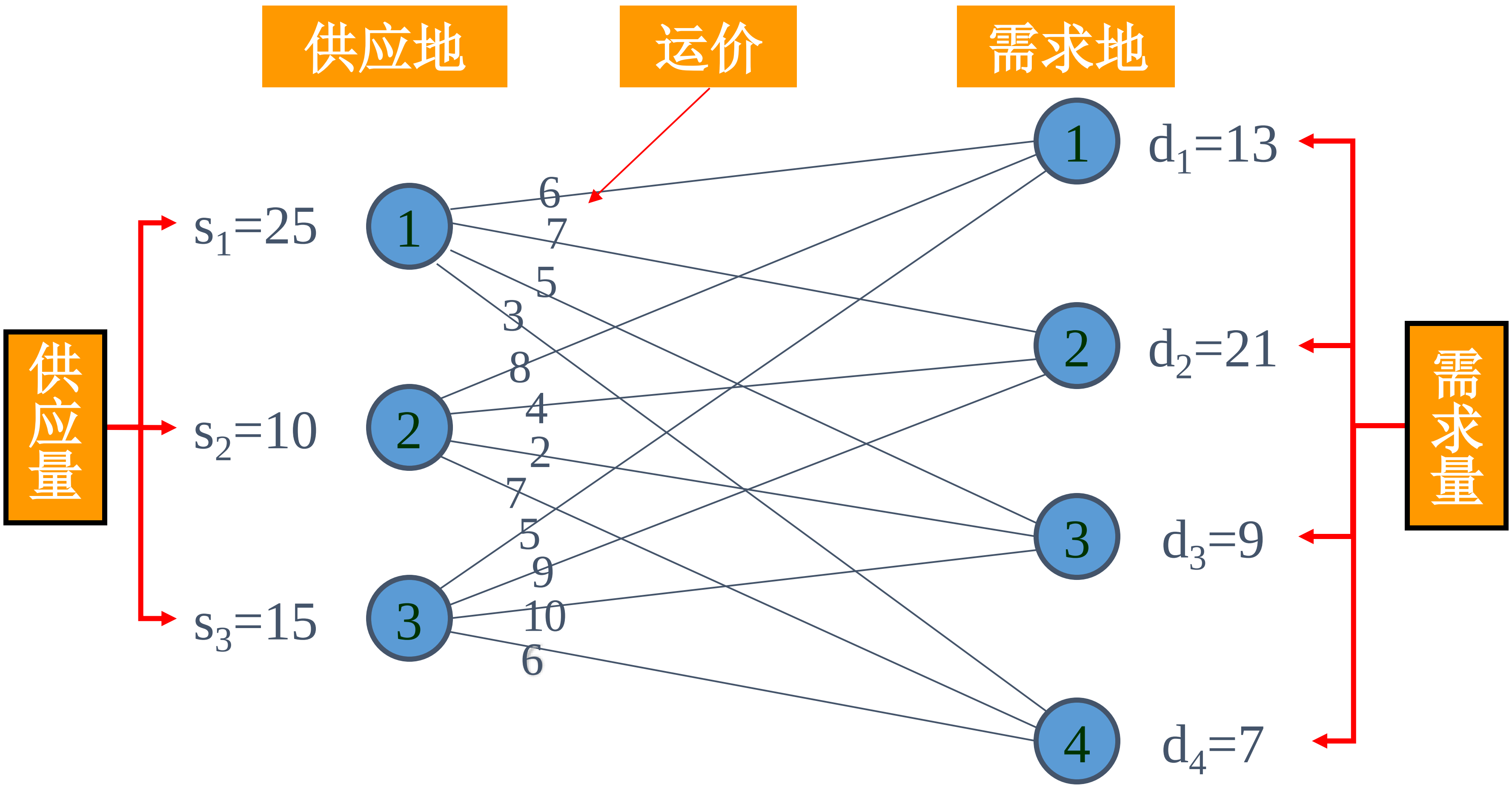
- 销售平衡条件

$$x_{11} + x_{21} + x_{31} = 2$$
$$x_{12} + x_{22} + x_{32} = 3$$
$$x_{13} + x_{23} + x_{33} = 1$$
$$x_{14} + x_{24} + x_{34} = 4$$

- 非负约束 $x_{ij} \geq 0\ (i = 1,2,3; j = 1,2,3,4)$

1.3 运输问题—举例2

运输问题网络图



1.3 运输问题—举例2

$$
\begin{array}{lrcl}
\min & z = 6x_{11}+7x_{12}+5x_{13}+3x_{14}+8x_{21}+4x_{22}+2x_{23}+7x_{24}+5x_{31}+9x_{32}+10x_{33}+6x_{34} & & \\
s.t. & x_{11}+x_{12}+x_{13}+x_{14} & = & 25 \\
 & x_{21}+x_{22}+x_{23}+x_{24} & = & 10 \\
 & x_{31}+x_{32}+x_{33}+x_{34} & = & 15 \\
 & x_{11}+x_{21}+x_{31} & = & 13 \\
 & x_{12}+x_{22}+x_{32} & = & 21 \\
 & x_{13}+x_{23}+x_{33} & = & 9 \\
 & x_{14}+x_{24}+x_{34} & = & 7 \\
 & x_{11}\ \ x_{12}\ \ x_{13}\ \ x_{14}\ \ x_{21}\ \ x_{22}\ \ x_{23}\ \ x_{24}\ \ x_{31}\ \ x_{32}\ \ x_{33}\ \ x_{34} & \geq & 0
\end{array}
$$

供应地约束

需求地约束

1. 运输问题有有限最优解

令变量 $\boxed{x_{ij}=\frac{a_i b_j}{Q}}$, $i=1,2,\cdots,m;\ j=1,2,\cdots,n$ $\left(Q=\sum_{i=1}^{m} a_i=\sum_{j=1}^{n} b_j\right)$

运输问题的一个可行解

$c_{ij}\geq 0,\ x_{ij}\geq 0$ ⟹ $z=\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}\geq 0$ 下有界

⟹ 运输问题必存在有限最优解

1.4 运输问题的特点

TABLE 8.6 Constraint coefficients for the transportation problem

Coefficient of:

$$
\begin{array}{cc}
 & \begin{array}{ccccccccccccc} x_{11} & x_{12} & \cdots & x_{1n} & x_{21} & x_{22} & \cdots & x_{2n} & \cdots & x_{m1} & x_{m2} & \cdots & x_{mn} \end{array} \\
\mathbf{A} = & \left[\begin{array}{ccccccccccccc}
1 & 1 & \cdots & 1 & & & & & & & & & \\
 & & & & 1 & 1 & \cdots & 1 & & & & & \\
 & & & & & & & & \cdots & & & & \\
 & & & & & & & & & 1 & 1 & \cdots & 1 \\
1 & & & & 1 & & & & & 1 & & & \\
 & 1 & & & & 1 & & & & & 1 & & \\
 & & \cdots & & & & \cdots & & \cdots & & & \cdots & \\
 & & & 1 & & & & 1 & & & & & 1
\end{array}\right]
\begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} \text{Supply constraints} \\ \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} \text{Demand constraints} \end{array}
\end{array}
$$

2. 运输问题约束条件的系数矩阵

1. 决策变量的个数：$m \times n$;
2. 约束方程的个数：$m + n$;
3. 独立方程的个数：$m + n - 1$;
4. 系数矩阵的特殊结构：变量 x_{ij} 的系数向量 A_{ij}

$$A_{ij} = (0,0,\dots,1,\dots,0,\ 0,0,\dots,1,\dots,0\ )^T$$

i 分量

m+j 分量

3. 运输问题求解

运输问题是一种线性规划，可用类似单纯形的迭代法进行求解，即先找到它的某一个基可行解，再进行最优性检验，若不是就进行迭代调整，直到得到最优解为止。

每步得到的解 $\vec{x} = \{x_{ij}\}$ 都必须是基可行解，即

- ✓ 满足模型中的所有约束条件
- ✓ 基变量对应的约束方程组的系数列向量线性无关
- ✓ 非零变量x_{ij}的个数不能大于$(m+n-1)$
- ✓ 基变量的个数在迭代过程中保持为$(m+n-1)$个

第三章 运输问题

2. 表上作业法求解运输问题

2.1 运输问题求解的基本思路

1. 建立运输问题表格
2. 找出初始解
   - 最小元素法 Intuitive Lowest Cost Method
   - 西北角法 Northwest Corner Method
   - 伏格尔法 Vogel Approximation Method
3. 从初始解到最优解 — 闭回路调整法
   - 判别：是否为最优解？
   - 调整、改进，直到求得最优解。

2.1 运输问题求解的基本思路

- 产销平衡的运输问题有$\boldsymbol{m}\times\boldsymbol{n}$个变量，$\boldsymbol{m}+\boldsymbol{n}-\mathbf{1}$个独立约束。
- 产销平衡的运输问题一定有解。
- 如果运输问题中的参数均是整数，则其任意基本解中各变量的取值均为整数。
- 初始解中基变量的个数应当为$\boldsymbol{m}+\boldsymbol{n}-\mathbf{1}$个，对应表中的数字格。
- 剩下的为非基变量，值均为$\mathbf{0}$，对应空格。

1. 最小元素法 Intuitive Lowest-Cost (ILC) Method

- 从单位运价表中最小的运价开始确定供销关系，然后次小……
- 即$\min\{c_{ij}\} = c_{pq}$，从A_p供应B_q
- 若$a_p > b_q$，则B_q的需求全部满足，划去q列尚有余量$a_p - b_q$供应给其他的销地
- 若$a_p < b_q$，则B_q需求不能全部满足，划去p行，不足部分$b_q - a_p$需由其他的产地供应

2.2 初始基可行解

最小元素法

lowest cost

lowest cost

lowest cost

To / From	A	B	C	Supply
D	5	4	100 3	100
E	8	200 4	100 3	300
F	300 9	7	5	300
Demand	300	200	200	700

1. 先找运价最小的单元格
2. 尽可能分配最多的运量，但不超出产量或销量的限制
3. 划去满足的行或列
4. 从剩下格子中再找最小运价
5. 重复上述1~4步

1. 最小元素法 Intuitive Lowest-Cost (ILC) Method

From \ To	A	B	C	Supply
D	(5)	(4)	100 (3)	100
E	(8)	200 (4)	100 (3)	300
F	300 (9)	(7)	(5)	300
Demand	300	200	200	700

总成本 = 100*3+200*4+100*3+300*9 = 4100