
1. Transport- und Zuordnungsprobleme

Themen:

- Analyse der Problemstruktur
- Spezielle Varianten des Simplexalgorithmus für Transport- und Zuordnungsprobleme
- Bezug zur Graphentheorie

Transportproblem

siehe Beispiel 1.4 aus OR I

Definition 1.1. Das Optimierungsproblem

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

unter den Nebenbedingungen

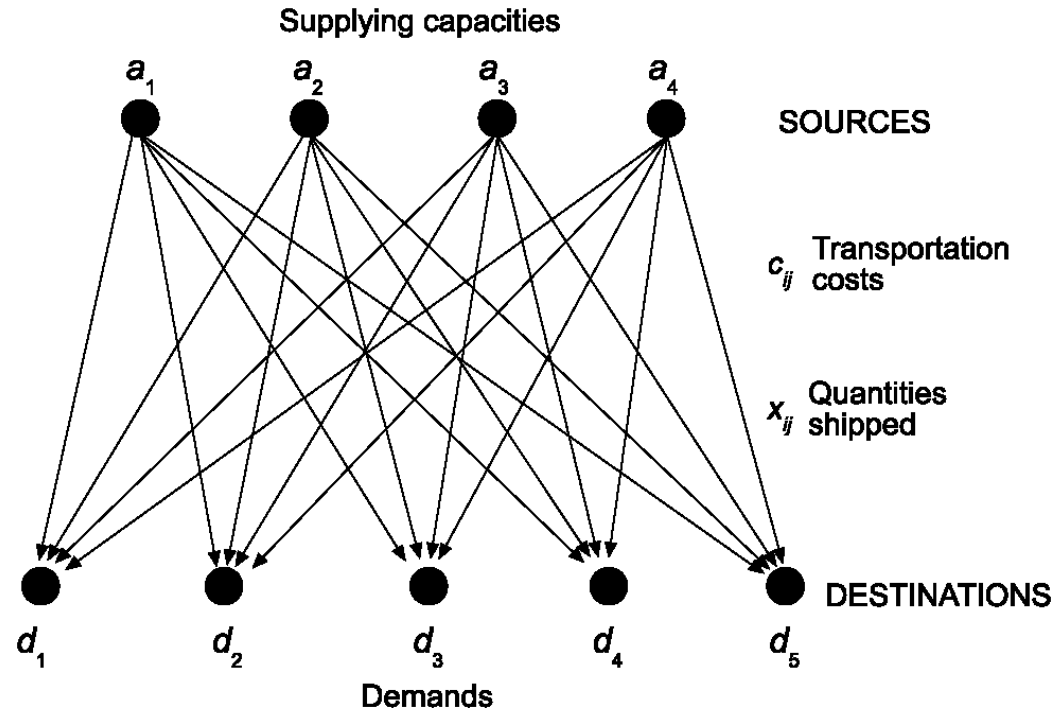
$$\sum_{j=1}^n x_{ij} = a_i \quad \text{für } i = 1, \dots, m$$

$$\sum_{i=1}^m x_{ij} = b_j \quad \text{für } j = 1, \dots, n$$

und den Vorzeichenbedingungen

$$x_{ij} \geq 0 \text{ für } i = 1, \dots, m \text{ und } j = 1, \dots, n$$

heißt *Transportproblem*.



Bemerkungen:

- Wir setzen ein *geschlossenes Transportproblem* voraus: $a_i > 0, b_j > 0$ und $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$, also **Gesamtangebot = Gesamtnachfrage**.
- Für den Fall $\sum_{i=1}^m a_i > \sum_{j=1}^n b_j$ führen wir ein **zusätzliches Warenhaus** mit $b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j$ und $c_{i,n+1} = 0$ ein.
- Für den Fall $\sum_{i=1}^m a_i < \sum_{j=1}^n b_j$ führen wir eine **zusätzliche Produktionsstätte** mit $a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i$ ein.
Die $c_{m+1,j}$ modellieren dann die **Kosten pro ME** für das mangelnde Angebot in Warenhaus j .

- Anzahl Variablen: $m \cdot n$

Beispielproblem

Beispiel 1.1. Wir gehen von folgenden Kosten, Angebot und Nachfrage aus:

	B_1	B_2	B_3	
A_1	9	1	3	50
A_2	4	5	8	70
	40	40	40	

Damit lautet das zugehörige Transportproblem

$$\min 9x_{11} + x_{12} + 3x_{13} + 4x_{21} + 5x_{22} + 8x_{23}$$

unter den Nebenbedingungen

$$\begin{array}{rcccccccl}
 x_{11} & + & x_{12} & + & x_{13} & & & = & 50 \\
 & & & & & + & x_{21} & + & x_{22} & + & x_{23} & = & 70 \\
 x_{11} & & & & & + & x_{21} & & & & & = & 40 \\
 & & x_{12} & & & & & + & x_{22} & & & = & 40 \\
 & & & & x_{13} & & & & & + & x_{23} & = & 40
 \end{array}$$

und Vorzeichenbedingungen

$$x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23} \geq 0$$

Lösbarkeit des Transportproblems

Satz 1.1. *Zu jedem Transportproblem existiert eine optimale Lösung.*

Beweis: Es sei

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

und

$$x_{ij} = \frac{a_i b_j}{G}$$

Dann gilt:

$$\sum_{j=1}^n x_{ij} = \sum_{j=1}^n \frac{a_i b_j}{G} = \frac{a_i \sum_{j=1}^n b_j}{G} = a_i \quad \text{für } i = 1, \dots, m$$

und

$$\sum_{i=1}^m x_{ij} = \sum_{i=1}^m \frac{a_i b_j}{G} = \frac{b_j \sum_{i=1}^m a_i}{G} = b_j \quad \text{für } j = 1, \dots, n$$

Damit existiert eine **zulässige Lösung**.

Wegen $0 \leq x_{ij} \leq \min\{a_i, b_j\}$ ist der **Zulässigkeitsbereich $\mathcal{X}$** darüberhinaus **beschränkt**.

Also existiert eine optimale Lösung (siehe Satz 3.8 und Folie 174, OR I).

Transportproblem in Matrixdarstellung

$$\mathbf{c} = (c_{11}, c_{12}, \dots, c_{1n}, c_{21}, \dots, c_{2n}, \dots, c_{m1}, \dots, c_{mn}) \in \mathbb{R}^{m \cdot n}$$

$$\mathbf{x} = (x_{11}, x_{12}, \dots, x_{1n}, x_{21}, \dots, x_{2n}, \dots, x_{m1}, \dots, x_{mn}) \in \mathbb{R}^{m \cdot n}$$

$$\mathbf{A} = \left(\begin{array}{cccc|cccc|c|cccc} 1 & 1 & \dots & 1 & 0 & 0 & & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & & 0 & 1 & 1 & \dots & 1 & & & 0 & 0 & 0 \\ & & & & & & & & \vdots & & & & \\ 0 & 0 & & 0 & 0 & 0 & & 0 & & & 1 & 1 & \dots & 1 \\ \hline 1 & 0 & & 0 & 1 & 0 & & 0 & & & 1 & 0 & & 0 \\ 0 & 1 & & 0 & 0 & 1 & & 0 & & & 0 & 1 & & 0 \\ & & \ddots & & & & \ddots & & \dots & & & & \ddots & \\ 0 & 0 & & 1 & 0 & 0 & & 1 & & & 0 & 0 & & 1 \end{array} \right) \in \mathbb{R}^{(m+n) \times m \cdot n}$$

genauer

$$a_{ij} = \begin{cases} 1 & \text{falls } 1 \leq i \leq m \wedge (i-1) \cdot n < j \leq i \cdot n \\ 1 & \text{falls } m < i \leq m+n \wedge j = k \cdot n + (i-m) \\ 0 & \text{sonst} \end{cases}$$

Begrenzungsvektor:

$$\mathbf{b} = (a_1, \dots, a_m, b_1, \dots, b_n) \in \mathbb{R}^{m+n}$$

Damit hat das **Transportproblem in Normalform** die Darstellung

$$\min \mathbf{c}^T \mathbf{x}$$

unter den Nebenbedingungen

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0}$$

Satz 1.2. *Die Matrix $\mathbf{A}$ des Transportproblems hat den Rang $r(\mathbf{A}) = m + n - 1$.*

Beweis: Die Summe der Zeilen 1 bis m ist gleich der Summe der Zeilen $m + 1$ bis $m + n$. Also sind die $m + n$ Zeilenvektoren linear abhängig und es folgt $r(\mathbf{A}) \leq m + n - 1$.

Andererseits sind die $m + n - 1$ Spaltenvektoren mit den Indizes

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

linear unabhängig, also $r(\mathbf{A}) \geq m + n - 1$.

Insgesamt folgt $r(\mathbf{A}) = m + n - 1$.

Eröffnungsverfahren

- Nach Satz 1.2 besteht eine **Basislösung** eines Transportproblems aus $m + n - 1$ **Basisvariablen**.
- Zur Konstruktion einer ersten Ecke benötigen wir daher eine zulässige Lösung mit $n + m - 1$ Variablen $x_{ij} > 0$ und restlichen Variablen $x_{ij} = 0$ (falls keine Entartung vorliegt).
- Wir stellen nun **zwei Verfahren zur Konstruktion einer ersten zulässigen Basislösung** bzw. Ecke vor:
 - Nordwesteckenregel
 - Minimale-Kosten-Regel

Transporttableau

	B_1	B_2	$\dots$	B_n	
A_1	c_{11} x_{11} B_{11}	c_{12} x_{12} B_{12}		c_{1n} x_{1n} B_{1n}	a_1
A_2	c_{21} x_{21} B_{21}	c_{22} x_{22} B_{22}		c_{2n} x_{2n} B_{2n}	a_2
$\vdots$					$\vdots$
A_m	c_{m1} x_{m1} B_{m1}	c_{m2} x_{m2} B_{m2}		c_{mn} x_{mn} B_{mn}	a_m
	b_1	b_2	$\dots$	b_n	z

- A_i : i -te Fabrik (bleibt unverändert)
- a_i : noch zu transportierende Menge aus Fabrik i
- B_j : j -tes Warenhaus (bleibt unverändert)
- b_j : noch zu transportierende Menge zu Warenhaus j
- c_{ij} : **Kosten** pro Einheit für den Transport von A_i zu B_j (bleibt unverändert)
- x_{ij} : **Transportmenge** von A_i nach B_j (nur für Basisvariablen gesetzt)
- B_{ij} : **Schattenpreise** (nur für Nichtbasisvariablen gesetzt)
- z : **Zielfunktionswert**

Nordwesteckenregel

Idee:

- Man transportiere über die Verbindung ganz links oben im Tableau so viel wie möglich.
- Wird dadurch das Lager erschöpft, streiche man die erste Zeile des Tableaus, ansonsten die erste Spalte, und beginne wieder mit dem ersten Schritt.

Algorithmus 1.1. [Nordwesteckenregel] $x_{ij} := 0$ für $i = 1, \dots, m, j = 1, \dots, n$ $z := 0$ $i := 1, j := 1$ **while** $i \leq m$ **and** $j \leq n$ **do** $x_{ij} := \min\{a_i, b_j\}$ $z := z + c_{ij}x_{ij}$ $a_i := a_i - x_{ij}$ $b_j := b_j - x_{ij}$ **if** $a_i = 0$ **then** $i := i + 1$ **else** $j := j + 1$ **end****end**

- Die tatsächlichen Kosten werden zur Auswahl der Basisvariablen nicht berücksichtigt, daher i.d.R. keine gute zulässige Lösung.
- Entartung, wenn in einer Iteration sowohl a_i als auch b_j gleich 0 werden.
In der nächsten Iteration wird dann $x_{i+1,j}$ Basisvariable mit $x_{i+1,j} = 0$.
- In jeder Iteration wird genau eine Zeile oder Spalte “gestrichen”, in letzter Iteration ist aber nur genau eine Spalte und genau eine Zeile übrig.
- Daher insgesamt $m + n - 1$ Iterationen mit der Auswahl von $m + n - 1$ Basisvariablen.
- Die B_{ij} betrachten wir erst später!

Beispiel 1.2. Wir gehen von Kosten, Angebot und Nachfrage gemäß Beispiel 1.1 aus.

Starttableau:

	B_1	B_2	B_3	
A_1	9 B_{11}	1 B_{12}	3 B_{13}	50
A_2	4 B_{21}	5 B_{22}	8 B_{23}	70
	40	40	40	0

1. Iteration: $x_{11} = 40, a_1 = 10, b_1 = 0, z = 360$

2. Iteration: $x_{12} = 10, a_1 = 0, b_2 = 30, z = 370$

3. Iteration: $x_{22} = 30, a_2 = 40, b_2 = 0, z = 520$

4. Iteration: $x_{23} = 40, a_2 = 0, b_3 = 0, z = 840$

Tableau nach Nordwesteckenregel:

	B_1	B_2	B_3	
A_1	9 40	1 10	3 B_{13}	0
A_2	4 B_{21}	5 30	8 40	0
	0	0	0	840

Minimale-Kosten-Regel

- Statt die erste Möglichkeit links oben im Transporttableau wählt man unter den möglichen Variablen x_{ij} diejenige mit minimalen Kosten c_{ij} .
- Ansonsten verläuft der Algorithmus analog zur Nordwesteckenregel.
- I.d.R. erhalten wir eine bessere zulässige Basis als bei der Nordwesteckenregel, dies ist aber nicht garantiert.
- Typischer Greedy-Algorithmus: Treffe die lokal beste Entscheidung!

Algorithmus 1.2. [Minimale-Kosten-Regel]

$x_{ij} := 0$ für $i = 1, \dots, m, j = 1, \dots, n$

$I := \{1, \dots, m\}; J := \{1, \dots, n\}; z := 0$

while $I \neq \emptyset$ **and** $J \neq \emptyset$ **do**

 wähle i und j so, dass $c_{ij} = \min\{c_{lk} | l \in I, k \in J\}$

$x_{ij} := \min\{a_i, b_j\}$

$z := z + c_{ij}x_{ij}$

$a_i := a_i - x_{ij}$

$b_j := b_j - x_{ij}$

if $a_i = 0$ **then**

$I := I \setminus \{i\}$

else

$J := J \setminus \{j\}$

end

end

Beispiel 1.3.

1. Iteration:

$$i = 1, j = 2, x_{12} = 40, a_1 = 10, b_2 = 0, I = \{1, 2\}, J = \{1, 3\}, z = 40$$

2. Iteration:

$$i = 1, j = 3, x_{13} = 10, a_1 = 0, b_3 = 30, I = \{2\}, J = \{1, 3\}, z = 70$$

3. Iteration:

$$i = 2, j = 1, x_{21} = 40, a_2 = 30, b_1 = 0, I = \{2\}, J = \{3\}, z = 230$$

4. Iteration:

$$i = 2, j = 3, x_{23} = 30, a_2 = 0, b_3 = 30, I = \emptyset, J = \emptyset, z = 470$$

Tableau nach Minimale-Kosten-Regel:

	B_1	B_2	B_3	
A_1	9 B_{11}	1 40	3 10	0
A_2	4 40	5 B_{22}	8 30	0
	0	0	0	470