ROZDZIAŁ 1

Zadanie transportowe

Żadna inna nauka nie umacnia tak wiary w siłę ludzkiego ducha, jak matematyka.
[H. Steinhaus]

1.1. Podstawowe pojęcia

1.1.1. Zbilansowane zadanie transportowe

Dla określonego towaru dana jest sieć m magazynów (lub dostawców) A_1,...,A_m i n sklepów (lub odbiorców) B_1,...,B_n. Zapas magazynu (lub podaż dostawcy) A_i wynosi a_i jednostek towaru, $i = 1, ..., m$. Sklep (lub odbiorca) B_j potrzebuje b_j jednostek towaru, $j = 1, ..., n$. Zakładamy, że

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{1.1}$$

(łączna podaż jest równa łącznemu popytowi). Koszty transportu jednostki towaru z magazynu A_i do sklepu B_j wynoszą c_{ij} jednostek pieniężnych, $i = 1, ..., m$, $j = 1, ..., n$. Należy określić plan transportowy o minimalnych kosztach zaspokajający zapotrzebowania wszystkich sklepów (czyli, przy podanym założeniu, wyczerpujący łączne zapasy magazynów).

Oznaczając przez x_{ij} ilość towaru transportowanego z magazynu A_i do sklepu B_j, $i = 1, ..., m$, $j = 1, ..., n$, powyższe zagadnienie można zapisać w następującej formie:

$$\begin{array}{lll} \text{minimalizować} & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} & \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} = a_i, & i = 1, ..., m, \\ & \sum_{i=1}^{m} x_{ij} = b_j, & j = 1, ..., n, \\ & x_{ij} \geq 0, & i = 1, ..., m, j = 1, ..., n. \end{array} \tag{1.2}$$

Zadanie (1.2) nazywa się *zbilansowanym zadaniem transportowym.* Macierz

$$C = \begin{bmatrix} c_{11} & ... & c_{1n} \\ ... & ... & ... \\ c_{m1} & ... & c_{mn} \end{bmatrix}$$

nazywa się *macierzą jednostkowych kosztów transportowych* lub krótko *macierzą kosztów*, zaś macierz

$$X = \begin{bmatrix} x_{11} & ... & x_{1n} \\ ... & ... & ... \\ x_{m1} & ... & x_{mn} \end{bmatrix}$$

– *macierzą zmiennych decyzyjnych* lub *planem transportowym.*

1.1.2. Niezbilansowane zadanie transportowe

Z praktycznego punktu widzenia założenie

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

wydaje się za mocne. Jeśli nie jest ono spełnione, to po odpowiedniej modyfikacji ograniczeń otrzymamy *niezbilansowane zadanie transportowe.* Jeśli zachodzi nierówność

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

(podaż jest większa od popytu), to niezbilansowane zadanie transportowe przybiera postać

$$\begin{array}{lll} \text{minimalizować} & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} & \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} \le a_i, & i = 1, ..., m, \\ & \sum_{i=1}^{m} x_{ij} = b_j, & j = 1, ..., n, \\ & x_{ij} \ge 0, & i = 1, ..., m, \ j = 1, ..., n. \end{array}$$

Nowe ograniczenia oznaczają, że nie wszystkie zapasy magazynów zostaną wyczerpane (nie cała podaż dostawców znajdzie nabywcę). W tym przypadku zadanie to można sprowadzić do zbilansowanego zadania transportowego wprowadzając fikcyjny sklep (fikcyjnego odbiorcę) B_{n+1} z zapotrzebowaniem (popytem)

$$b_{n+1} = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j$$

i zerowymi kosztami transportowymi z dowolnego magazynu (od dowolnego dostawcy):

$$c_{i,n+1} = 0, \ i = 1, ..., m.$$

Jeśli natomiast zachodzi nierówność

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

(podaż jest mniejsza od popytu), to niezbilansowane zadanie transportowe przybiera postać

$$\begin{array}{rlr} \text{minimalizować} & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} & \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} = a_i, & i = 1, ..., m \\ & \sum_{i=1}^{m} x_{ij} \le b_j, & j = 1, ..., n \\ & x_{ij} \ge 0, & i = 1, ..., m, \ j = 1, ..., n. \end{array}$$

Nowe ograniczenia oznaczają, że zapotrzebowanie (popyt) nie wszystkich sklepów (odbiorców) zostanie zaspokojone. W tym przypadku zadanie to można sprowadzić do zbilansowanego zadania transportowego wprowadzając fikcyjny magazyn (fikcyjnego dostawcę) A_{m+1} z zapasem (podażą)

$$a_{m+1} = \sum_{j=1}^{n} b_j - \sum_{i=1}^{m} a_i$$

i zerowymi kosztami transportowymi do dowolnego sklepu (odbiorcy):

$$c_{m+1,j} = 0,\ j = 1, ..., m.$$

W dalszej części będziemy rozpatrywać wyłącznie zbilansowane zadanie transportowe, gdyż – jak zauważyliśmy – nie prowadzi to do ograniczenia ogólności rozważań.

1.1.3. Macierz transportowa

Zbilansowane zadanie transportowe ma następującą postać macierzową

$$\begin{array}{ll} \text{minimalizować} & c^\top x \\ \text{przy ograniczeniach} & Ax = \begin{bmatrix} a \\ b \end{bmatrix} \\ & x \geq 0, \end{array}$$

gdzie $c = (c_{11}, ..., c_{1n}, ..., c_{m1}, ..., c_{mn})$, $x = (x_{11}, ..., x_{1n}, ..., x_{m1}, ..., x_{mn})^\top$, $a = (a_1, ..., a_m)$, $b = (b_1, ..., b_n)$ zaś macierz A ma postać

$$A = \begin{bmatrix} 1 & 1 & ... & 1 & 0 & 0 & ... & 0 & ... & 0 & 0 & 0 & 0 \\ 0 & 0 & ... & 0 & 1 & 1 & ... & 1 & ... & 0 & 0 & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & 0 & 0 & 0 & ... & 0 & ... & 1 & 1 & ... & 1 \\ 1 & 0 & ... & 0 & 1 & 0 & ... & 0 & ... & 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 & 0 & 1 & ... & 0 & ... & 0 & 1 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & 1 & 0 & 0 & ... & 1 & ... & 0 & 0 & 0 & 1 \end{bmatrix}$$

lub w zapisie blokowym

$$A = \begin{bmatrix} e^\top & 0 & ... & 0 \\ 0 & e^\top & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & e^\top \\ I_n & I_n & ... & I_n \end{bmatrix},$$

gdzie $e = (1, ..., 1) \in \mathbb{R}^n$, zaś I_n jest macierzą jednostkową typu $n \times n$. Macierz A typu $(m+n) \times (mn)$ nazywa się *macierzą transportową*.

Oznaczmy przez A^i i-ty wiersz macierzy A, $i = 1, ..., m+n$, i przez A_{ij} - jej $((i-1)n+j)$-tą kolumnę, $i = 1, ..., m$, $j = 1, ..., n$ (kolumna ta odpowiada zmiennej x_{ij}).

Twierdzenie 1.1.1. *Macierz transportowa A ma rząd* $\mathrm{r}(A) = m + n - 1$.

Dowód. Zachodzi oczywista równość

$$\sum_{i=1}^{m} A^i = \sum_{i=m+1}^{m+n} A^i.$$

Wiersze macierzy A są więc liniowo zależne, w konsekwencji $\mathrm{r}(A) < m+n$. Ponadto macierz ta zawiera $m+n-1$ liniowo niezależnych kolumn $A_{1n}, A_{2n,}..., A_{mn}$, $A_{11}, A_{12}, ..., A_{1,n-1}$, ponieważ w macierzy utworzonej z tych kolumn:

$$[A_{1n}, A_{2n,}..., A_{mn}, A_{11}, A_{12}, ..., A_{1,n-1}] =$$

$$\begin{bmatrix} 1 & 0 & ... & 0 & 1 & 1 & ... & 1 \\ 0 & 1 & ... & 0 & 0 & 0 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & 1 & 0 & 0 & ... & 0 \\ 0 & 0 & ... & 0 & 1 & 0 & ... & 0 \\ 0 & 0 & ... & 0 & 0 & 1 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & 1 \\ 1 & 1 & ... & 1 & 0 & 0 & ... & 0 \end{bmatrix}$$

pierwsze $m + n - 1$ wierszy jest liniowo niezależnych (macierz utworzona z ostatniej macierzy przez skreślenie ostatniego wiersza ma postać

$$A_B = \begin{bmatrix} I_m & R \\ 0 & I_{n-1} \end{bmatrix},$$

w konsekwencji jest ona nieosobliwa) jako macierz trójkątna o niezerowych elementach na głównej przekątnej. □

Macierz tę nieprzypadkowo oznaczyliśmy symbolem A_B. Oznaczając bowiem macierz utworzoną z pozostałych kolumn macierzy A po usunięciu z niej ostatniego wiersza symbolem A_N możemy układ $Ax = \begin{bmatrix} a \\ b \end{bmatrix}$ zapisać w postaci

$$[A_B, A_N] \begin{bmatrix} x_B \\ x_N \end{bmatrix} = \begin{bmatrix} a \\ b^- \end{bmatrix},$$

b^- jest wektorem pierwszych $n - 1$ współrzędnych wektora b (skoro usunęliśmy ostatni wiersz macierzy A, to powinniśmy usunąc również ostatnią współrzędną wektora b). Wektor x zapisujemy w postaci $x = \begin{bmatrix} x_B \\ x_N \end{bmatrix}$, przy czym x_B (zmienne bazowe) odpowiadają indeksom kolumn macierzy A, z których utworzyliśmy macierz A_B, a x_N (zmienne niebazowe) odpowiadają indeksom pozostałych kolumn macierzy A. Przyjmując $x_N = 0$ i $x_B = A_B^{-1} \begin{bmatrix} a \\ b^- \end{bmatrix}$ otrzymamy rozwiązanie bazowe x. Mając dane rozwiązanie bazowe zadanie transportowe można teoretycznie rozwiązać przy pomocy algorytmu sympleksowego. Byłoby to jednak zbyt pracochłonne – zauważmy, że już dla $m = 20$, $n = 500$ długa forma tablicy sympleksowej ma $m+n-1 = 519$ wierszy i 10000 kolumn, natomiast liczba niezerowych elementów stanowi około $0,4\%$ wszystkich elementów macierzy transportowej. Oznacza to, że przy stosowaniu metody sympleksowej najczęstszymi operacjami będzie mnożenie przez zero lub dodawanie zera. Dzięki szczególnej postaci macierzy transportowej A zadanie transportowe można rozwiązać znacznie prościej, niemniej jednak główne etapy wyznaczania rozwiązania są takie same, jak w metodzie sympleksowej: wyznaczenie bazowego rozwiązania dopuszczalnego, a następnie wyznaczenie bazowego rozwiązania optymalnego. Ponadto, jak się później przekonamy, w drugim etapie będziemy otrzymywać kolejno te same rozwiązania bazowe co w drugiej fazie metody sympleksowej, ale przy mniejszym nakładzie obliczeniowym.

Do opisu odpowiednich metod służących rozwiązywaniu zadania transportowego używa się tak zwanej *tablicy transportowej*, która ma następującą postać:

	B_1	B_2	...	B_n	
A_1	x_{11}	x_{12}	...	x_{1n}	a_1
A_2	x_{21}	x_{22}	...	x_{2n}	a_2
...	...	...	...	...	...
A_m	x_{m1}	x_{m2}	...	x_{mn}	a_m
	b_1	b_2	...	b_n	

Tablica ta odpowiada rozwiązaniu dopuszczalnemu zbilansowanego zadania transportowego dokładnie wtedy, gdy wszystkie x_{ij} są nieujemne, ostatni element dowolnego wiersza (kolumny) jest równy sumie poprzednich elementów tego wiersza (tej kolumny) i suma wszystkich elementów ostatniego wiersza równa jest sumie wszystkich elementów ostatniej kolumny.

Opiszemy teraz, kiedy tablica transportowa przedstawia rozwiązanie bazowe. Niech $I = \{1, ..., m\}$, $J = \{1, ..., n\}$, oraz K niech będzie podzbiorem zbioru $I \times J$:

$$K \subset I \times J = \{(i, j) : i = 1, ..., m, \ j = 1, ..., n\}.$$

Podzbiór $\{i\} \times J$ nazywa się *wierszem zbioru* $I \times J$, $i = 1, ..., m$, zaś podzbiór $I \times \{j\}$, – *kolumną zbioru* $I \times J$, $j = 1, ..., n$.

Zbiorowi K przyporządkowany jest wzajemnie jednoznacznie podzbiór A_K zbioru kolumn macierzy transportowej A:

$$A_K = \{A_{ij} : (i, j) \in K\}.$$

Definicja 1.1.2. Podzbiór K nazywa się *cyklem*, jeśli w każdym wierszu i w każdej kolumnie zbioru $I \times J$ znajdują się 0 lub 2 elementy zbioru K.

Przykład 1.1.3. Elementy podzbioru $K \subset I \times J$ zostały w poniższej tablicy oznaczone symbolem $\times$.

$I \backslash J$	1	2	3	4	5	6
1	$\times$			$\times$		
2						
3		$\times$				$\times$
4	$\times$	$\times$				
5						
6				$\times$		$\times$

Podzbiór K jest cyklem. Czytelnikowi pozostawiamy sprawdzenie, że jeśli $\times$ w i-tym wierszu i w j-tej kolumnie oznacza niezerowy transport z magazynu i do sklepu j, to graf ilustrujący te transporty jest cyklem.

Twierdzenie 1.1.4. *Układ kolumn A_L macierzy transportowej A, gdzie $L \subset I \times J$, jest liniowo zależny wtedy i tylko wtedy, gdy podzbiór $L \subset I \times J$ zawiera pewien cykl K.*

Dowód powyższego twierdzenia można znaleźć na przykład w książce W. Grabowskiego [Gra80].

Uwaga 1.1.5. Dana jest tablica transportowa

	B_1	B_2	...	B_n	
A_1	x_{11}	x_{12}	...	x_{1n}	a_1
A_2	x_{21}	x_{22}	...	x_{2n}	a_2
...	...	...	...	...	...
A_m	x_{m1}	x_{m2}	...	x_{mn}	a_m
	b_1	b_2	...	b_n	

,

która przedstawia rozwiązanie dopuszczalne zadania transportowego. Zgodnie z twierdzeniami 1.1.1 i 1.1.4 rozwiązanie to jest rozwiązaniem bazowym dokładnie wtedy, gdy podzbiór

$$L = \{(i, j) \in I \times J : x_{ij} \neq 0\}$$

posiada co najwyżej $m + n - 1$ elementów i nie zawiera cyklu. Jeśli ponadto zbiór L posiada dokładnie $m + n - 1$ elementów, to rozwiązanie to jest niezdegenerowane.

Przykład 1.1.6. a) Tablica transportowa

	B_1	B_2	B_3	B_4	
A_1	6	4			10
A_2		1	7		8
A_3			1	6	7
	6	5	8	6	

przedstawia niezdegenerowane dopuszczalne rozwiązanie bazowe (puste miejsca oznaczają zera).

b) Tablica transportowa

	B_1	B_2	B_3	B_4	
A_1	5			5	10
A_2			8		8
A_3	1	5		1	7
	6	5	8	6	

przedstawia rozwiązanie dopuszczalne, które jednak nie jest rozwiązaniem bazowym, bo zbiór L odpowiadający niezerowym elementom planu transportowego zawiera cykl $K = \{(1, 1), (1, 4), (3, 1), (3, 4)\}$.

c) Tablica transportowa

	B_1	B_2	B_3	B_4	
A_1	5	5			10
A_2			8		8
A_3	1		0	6	7
	6	5	8	6	

przedstawia dopuszczalne rozwiązanie bazowe. Zmienna x_{33} jest zmienną bazową, ale ponieważ przyjmuje ona wartość zero, więc rozwiązanie przedstawione za pomocą tej tablicy jest zdegenerowane.

1.1.4. Dualne zadanie transportowe

Zadanie dualne do zbilansowanego zadania transportowego ma postać

$$\begin{array}{rr} \text{maksymalizować} & \sum_{i=1}^{m} a_i u_i + \sum_{j=1}^{n} b_j v_j \\ \text{przy ograniczeniach} & u_i + v_j \leq c_{ij}, \quad i \in I, \ j \in J \\ & u_i, v_j \in \mathbb{R}, \quad i \in I, \ j \in J. \end{array} \tag{1.3}$$

Istotnie. Zbilansowane zadanie transportowe możemy zapisać w postaci standardowej

$$\begin{array}{ll} \text{maksymalizować} & -c^\top x \\ \text{przy ograniczeniach} & -Ax = -\begin{bmatrix} a \\ b \end{bmatrix} \\ & x \geq 0, \end{array}$$

Wymiar wektora zmiennych dualnych y jest równy liczbie ograniczeń zadania pierwotnego, a więc $y \in \mathbb{R}^{m+n}$. Oznaczając wektor pierwszych m zmiennych dualnych symbolem $u = (u_1, u_2, ..., u_m)$, zaś wektor pozostałych n zmiennych dualnych symbolem $v = (v_1, v_2, ..., v_n)$ mamy $y = (u, v)$. Zadaniem dualnym do zadania transportowego jest więc

$$\begin{array}{ll} \text{minimalizować} & -\begin{bmatrix} a \\ b \end{bmatrix}^\top \begin{bmatrix} u \\ v \end{bmatrix} \\ \text{przy ograniczeniach} & -A^T \begin{bmatrix} u \\ u \end{bmatrix} \geq -c, \end{array}$$

(por. [Ceg02, równość (4.5)]), czyli

$$\begin{array}{ll} \text{maksymalizować} & a^T u + b^T v \\ \text{przy ograniczeniach} & A^T \begin{bmatrix} u \\ u \end{bmatrix} \leq c, \end{array}$$

którego zapis rozwinięty na współrzędne ma postać (1.3).

Uwaga 1.1.7. Niech X będzie rozwiązaniem dopuszczalnym zadania transportowego i niech (u, v) będzie rozwiązaniem dopuszczalnym dualnego zadania transportowego. Z twierdzenia o komplementarności [Ceg02, twierdzenie 4.2.13] wynika, że X i (u, v) są rozwiązaniami optymalnymi odpowiednio zadania pierwotnego i dualnego wtedy i tylko wtedy, gdy

$$x_{ij}(c_{ij} - u_i - v_j) = 0 \text{ dla wszystkich } (i, j) \in I \times J. \tag{1.4}$$

Aby to sprawdzić wystarczy przedstawić twierdzenie o komplementarności dla zadania programowania liniowego w postaci standardowej. Własność ta oznacza, że dla rozwiązań dopuszczalnych X i (u, v), jeśli $x_{ij} > 0$, to $u_i + v_j = c_{ij}$, $(i, j) \in I \times J$. Własność ta służy wyznaczeniu zmiennych dualnych i będzie później wykorzystana przy wyznaczaniu rozwiązania optymalnego zadania transportowego.

1.2. Wyznaczanie dopuszczalnego rozwiązania bazowego

1.2.1. Opis ogólny

Opiszemy teraz, jak można otrzymać początkowe dopuszczalne rozwiązanie bazowe dla zadania transportowego (1.2).

Najpierw rozpatrzymy przypadki trywialne:

- $m = 1$

  W tym przypadku $I = \{1\}$ i wektor $x = (x_{11}, ..., x_{1n})$ o współrzędnych

  $$x_{ij} = b_j, \qquad j = 1, ..., n, \quad i \in I, \tag{1.5}$$

  jest jedynym rozwiązaniem dopuszczalnym zadania transportowego. Rozwiązanie to jest więc jednocześnie rozwiązaniem optymalnym.

- $n = 1$

  W tym przypadku $J = \{1\}$ i wektor $x = (x_{11}, ..., x_{m1})$ o współrzędnych

  $$x_{ij} = a_i, \qquad i = 1, ..., m, \quad j \in J, \tag{1.6}$$

  jest jedynym rozwiązaniem dopuszczalnym zadania transportowego. Podobnie jak w poprzednim przypadku, rozwiązanie to jest jednocześnie rozwiązaniem optymalnym.

Opiszemy teraz ogólną postać metody wyznaczania początkowego dopuszczalnego rozwiązania bazowego dla dowolnych m i n. Niech $I = \{1, ..., m\}$, $J = \{1, ..., n\}$ i niech $K = \emptyset$.

Schemat iteracyjny 1.2.1

Krok 1.

- Jeśli $\#J = 1$, to:

  1.1. położyć $K := K \cup I \times J$,

  1.2. położyć $x_{ij} := a_i$, $\quad (i, j) \in I \times J$,

  1.3. algorytm zatrzymuje się: X jest dopuszczalnym rozwiązaniem bazowym.

- Jeśli $\#I = 1$, to:

  1.4. położyć $K := K \cup I \times J$,

  1.5. położyć $x_{ij} := b_j$, $\quad (i, j) \in I \times J$,

  1.6. algorytm zatrzymuje się: X jest dopuszczalnym rozwiązaniem bazowym.

- Jeśli $\min\{\#I, \#J\} \geq 2$, to:

  1.7. wybrać parę $(p, q) \in I \times J$,

  1.8. położyć $K := K \cup \{(p, q)\}$,

  1.9. położyć $x_{pq} := \min\{a_p, b_q\}$.

Krok 2.

- Jeśli $a_p < b_q$, to:

  2.1. położyć $b_q := b_q - a_p$ i $a_p := 0$,

  2.2. położyć $I := I \setminus \{p\}$ (p-ty dostawca zostaje wyeliminowany).

- Jeśli $a_p \geq b_q$, to:

  2.3. położyć $a_p := a_p - b_q$ i $b_q := 0$,

  2.4. położyć $J := J \setminus \{q\}$ (q-ty odbiorca zostaje wyeliminowany).

- Ze zredukowanym zadaniem transportowym przejść do kroku 1.

Twierdzenie 1.2.2. *Opisana w powyższym schemacie iteracyjnym metoda pozwala na wyznaczenie dopuszczalnego rozwiązania bazowego X w co najwyżej $m + n - 2$ iteracjach.*

Dowód można znaleźć na przykład w podręczniku [Gra80].

Uwaga 1.2.3. Jeśli w kroku 1.7. za każdym razem wybiera się parę $(p, q) \in I \times J$, taką, że $\min\{a_p, b_q\} > 0$, to otrzymane rozwiązanie jest niezdegenerowane.

Przykład 1.2.4. Rozpatrzmy zadanie transportowe z 3 dostawcami i 4 odbiorcami, dla którego popyt, podaż oraz macierz kosztów podane są w poniższej tabeli:

	B_1	B_2	B_3	B_4	
A_1	7	2	4	7	10
A_2	9	5	3	3	8
A_3	7	7	6	4	7
	6	5	8	6	

Jeśli wybierzemy w kolejnych iteracjach opisanego wyżej schematu $(p, q) = (1, 3), (2, 1), (3, 2)$ w kroku 1.7, to otrzymamy następujące dopuszczalne rozwiązanie bazowe

$$X = \begin{bmatrix} 0 & 0 & 8 & 2 \\ 6 & 0 & 0 & 2 \\ 0 & 5 & 0 & 2 \end{bmatrix}.$$

Podamy teraz konkretne algorytmy pozwalające wyznaczyć dopuszczalne rozwiązanie bazowe. Są one szczególnymi przypadkami opisanego wyżej schematu iteracyjnego. Dokładniej, w każdym z podanych algorytmów, w kroku 1.7 powyższego schematu iteracyjnego para (p, q) wyznaczana jest w specjalny sposób.

1.2.2. Metoda kąta północno-zachodniego

Krok 1.7 ma w *metodzie kąta północno-zachodniego* (ang. *north-west corner method* – NWC) postać:

1.7. wybrać parę $(p, q) \in I \times J$ taką, że

$$p = \min\{i : i \in I\}, \ q = \min\{j : j \in J\}.$$

Innymi słowy, wybieramy pary magazyn-sklep w kolejności zgodnej z ich numeracją.

Przykład 1.2.5. Rozpatrzmy zadanie transportowe z podażą, popytem i macierzą kosztów podanymi w przykładzie 1.2.4 Postępując zgodnie z metodą kąta północno-zachodniego otrzymujemy następujące dopuszczalne rozwiązanie bazowe:

$$X = \begin{bmatrix} 6 & 4 & & \\ & 1 & 7 & \\ & & 1 & 6 \end{bmatrix}.$$

Funkcja celu ma dla tego rozwiązania wartość $z = 106$.

Sprawdzimy teraz, czy jest to rozwiązanie optymalne. Zgodnie z uwagą 1.1.7, X jest optymalne dokładnie wtedy, gdy

$$x_{ij}(c_{ij} - u_i - v_j) = 0 \text{ dla wszystkich } (i, j) \in I \times J,$$

gdzie (u, v) jest rozwiązaniem dopuszczalnym zadania dualnego do rozpatrywanego zadania transportowego. Dla podanych współrzędnych x_{ij} rozwiązania X zachodzić więc muszą poniższe równości

$$\begin{aligned} u_1 + v_1 &= 7 \\ u_1 + v_2 &= 2 \\ u_2 + v_2 &= 5 \\ u_2 + v_3 &= 3 \\ u_3 + v_3 &= 6 \\ u_3 + v_4 &= 4. \end{aligned}$$

Wykorzystamy posiadany „stopień swobody” ($\mathrm{r}(A) = 3 + 4 - 1 = 6$) przypisując jednej ze zmiennych dowolną wartość, na przykład $v_1 = 0$. Po prostych rachunkach otrzymamy rozwiązanie powyższego układu:

$$(u, v) = (7, 10, 13, 0, -5, -7, -9).$$

Teraz sprawdzimy, czy spełnione są pozostałe ograniczenia zadania dualnego. W tym celu musimy wyznaczyć tak zwane *ujemne koszty zredukowane*

$$\bar{c}_{ij} = c_{ij} - u_i - v_j,$$

gdzie (i, j) odpowiadają zmiennym niebazowym. Jeśli wszystkie $\bar{c}_{ij}$ są nieujemne, to rozwiązanie jest optymalne i odwrotnie. Jednak w naszym przypadku tak nie jest, bo na przykład $\bar{c}_{31} = -6$. Tak więc otrzymane rozwiązanie nie jest optymalne.

1.2.3. Metoda minimalnego elementu macierzy kosztów

Krok 1.7 ma w *metodzie minimalnego elementu macierzy kosztów* (ang. *least cost method* – LCM) postać:

1.7. wybrać parę $(p, q) \in I \times J$, taką, że

$$c_{pq} = \min\{c_{ij} : i \in I,\ j \in J\}.$$

Innymi słowy, wybieramy kolejno pary magazyn-sklep o minimalnym koszcie transportowym. Można się więc spodziewać, że otrzymane w ten sposób rozwiązanie dopuszczalne będzie lepsze niż w metodzie kąta północno-zachodniego.

Przykład 1.2.6. Rozpatrzmy zadanie transportowe z podażą, popytem i macierzą kosztów podanymi w przykładzie 1.2.4. Postępując wedle metody elementu minimalnego macierzy kosztów otrzymujemy następujące dopuszczalne rozwiązanie bazowe:

$$X = \begin{bmatrix} 5 & 5 & & \\ & & 8 & 0 \\ 1 & & & 6 \end{bmatrix}.$$

Funkcja celu ma dla tego rozwiązania wartość $z = 100$. Zmienne dualne możemy wyznaczyć podobnie jak w przykładzie 1.2.5. Otrzymamy układ równań

$$\begin{aligned} u_1 + v_1 &= 7 \\ u_1 + v_2 &= 2 \\ u_2 + v_3 &= 3 \\ u_2 + v_4 &= 3 \\ u_3 + v_1 &= 7 \\ u_3 + v_4 &= 4. \end{aligned}$$

Jeśli położymy $u_1 = 0$, to otrzymamy rozwiązanie $(u, v) = (0, -1, 0, 7, 2, 4, 4)$. Nietrudno sprawdzić, że wszystkie ujemne koszty zredukowane są w tym przypadku nieujemne, a więc X jest rozwiązaniem optymalnym.

1.2.4. Metoda aproksymacyjna Vogla

Krok 1.7 ma w metodzie *aproksymacyjnej Vogla* (ang. *Vogel's approximation method* – VAM) postać:

1.7.a) Dla każdego $i \in I$ wyznaczyć różnicę $dz_i = c_{il} - c_{ik}$ między drugim co do wielkości (licząc od najmniejszego) elementem c_{il} i najmniejszym elementem c_{ik} i-tego wiersza macierzy kosztów zadania transportowego ($k, l \in J$).

1.7.b) Dla każdego $j \in J$ wyznaczyć różnicę $ds_j = c_{lj} - c_{kj}$ między drugim co do wielkości (licząc od najmniejszego) elementem c_{lj} i najmniejszym elementem c_{kj} j-tej kolumny macierzy kosztów zadania transportowego ($k, l \in I$).

1.7.c) Wyznaczyć
$$\max\{dz_i, ds_j : i \in I,\ j \in J\}.$$

Jeśli maksimum to jest osiągnięte dla p-tego wiersza, to wybrać parę (p, q), dla której

$$c_{pq} = \min\{c_{pj} : j \in J\}$$

i przejść do kroku 1.8. Jeśli natomiast maksimum to jest osiągnięte dla q-tej kolumny, to wybrać parę (p, q), dla której
$$c_{pq} = \min\{c_{iq} : i \in I\}.$$

Przykład 1.2.7. Rozpatrzmy zadanie transportowe z podażą, popytem i macierzą kosztów podanymi w przykładzie 1.2.4. Postępując zgodnie z metodą aproksymacyjną Vogla otrzymujemy następujące dopuszczalne rozwiązanie bazowe

$$X = \begin{bmatrix} & 5 & 5 & \\ & & 3 & 5 \\ 6 & & & 1 \end{bmatrix}.$$

Funkcja celu ma dla tego rozwiązania wartość $z = 100$. Z przykładu 1.2.6 wynika, że wartość ta jest minimalna, tak więc X jest rozwiązaniem optymalnym. To samo stwierdzimy korzystając z twierdzenia o komplementarności, podobnie jak uczyniliśmy w przykładzie 1.2.6.

1.3. Algorytm transportowy

Mówi się często, że liczby rządzą światem. Pewne jest tylko: liczby pokazują, jak świat jest rządzony.

[J.W. von Goethe]

1.3.1. Podstawowe własności

Dane jest zbilansowane zadanie transportowe (1.2) i bazowe rozwiązanie dopuszczalne X. Niech

$$x_{i_1j_1}, x_{i_2j_2}, ..., x_{i_pj_p}$$

będą zmiennymi bazowymi, gdzie $p = m + n - 1$ i niech x_{ij} będzie zmienną niebazową.

Uwaga 1.3.1. Dla bazowego rozwiązania dopuszczalnego X ze zmiennymi bazowymi $x_{i_1j_1}, x_{i_2j_2}, ..., x_{i_pj_p}$ zachodzą równości

$$I = \{i_1, i_2, ..., i_p\}$$

i

$$J = \{j_1, j_2, ..., j_p\}.$$

Równości te oznaczają po prostu, że żaden z dostawców i żaden z odbiorców nie został pominięty w planie transportowym X.

Twierdzenie 1.3.2. *Zbiór*

$$\{(i, j), (i_1, j_1), (i_2, j_2), ..., (i_p, j_p)\}$$

zawiera dokładnie jeden cykl

$$L = \{(i, j), (i, l_1), (k_1, l_1), (k_1, l_2), ..., (k_r, l_r), (k_r, j)\}.$$

Dowód pomijamy.

Uwaga 1.3.3. Cykl L nazywa się *pętlą bazową.* Liczba elementów cyklu jest oczywiście parzysta. Zbiory

$$L^+ = \{(i, j), (k_1, l_1), ..., (k_r, l_r)\}$$

i

$$L^- = \{(i, l_1), (k_1, l_2), ..., (k_r, j)\}$$

nazywają się *półcyklami.* Każdy element półcyklu L^+ jest końcem pionowej krawędzi cyklu L i każdy element półcyklu L^- jest końcem poziomej krawędzi cyklu L.

Przykład 1.3.4. Rozpatrzmy zadanie transportowe z przykładu 1.2.4 i rozwiązanie bazowe

$$X = \begin{bmatrix} 6 & 4 & & \\ & 1 & 7 & \\ & & 1 & 6 \end{bmatrix}.$$

Niech $(i, j) = (3, 1)$. Oczywiście x_{31} jest zmienną niebazową. Pętla bazowa L ma postać

$$L = \{(3,1),(3,3),(2,3),(2,2),(1,2),(1,1)\},$$

natomiast półcykle mają postaci

$$L^{+} = \{(3,1),(2,3),(1,2)\}$$

i

$$L^{-} = \{(3,3),(2,2),(1,1)\}.$$

Przykład 1.3.5. Rozpatrzmy układ równań podany w przykładzie 1.2.6. Macierz współczynników tego układu ma postać

$$\begin{bmatrix} 1 & & & 1 & & & \\ 1 & & & & 1 & & \\ & 1 & & & 1 & & \\ & 1 & & & & 1 & \\ & & 1 & & & 1 & \\ & & 1 & & & & 1 \end{bmatrix}.$$

Widzimy, że wiersze tej macierzy są jednocześnie kolumnami macierzy transportowej

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & & & & & & & & \\ & & & & 1 & 1 & 1 & 1 & & & & \\ & & & & & & & & 1 & 1 & 1 & 1 \\ 1 & & & & 1 & & & & 1 & & & \\ & 1 & & & & 1 & & & & 1 & & \\ & & 1 & & & & 1 & & & & 1 & \\ & & & 1 & & & & 1 & & & & 1 \end{bmatrix}$$

odpowiadającymi zmiennym bazowym $x_{11}, x_{12}, x_{22}, x_{23}, x_{33}, x_{34}$. Nie jest to przypadek – własność ta zachodzi dla dowolnego rozwiązania bazowego zbilansowanego zadania transportowego.

Twierdzenie 1.3.6. *Niech X będzie bazowym rozwiązaniem dopuszczalnym ze zmiennymi bazowymi $x_{i_1j_1}, x_{i_2j_2}, ..., x_{i_pj_p}$. Układ równań*

$$\begin{array}{c} u_{i_1} + v_{j_1} = c_{i_1j_1} \\ u_{i_2} + v_{j_2} = c_{i_2j_2} \\ \dots \\ u_{i_p} + v_{j_p} = c_{i_pj_p} \end{array} \tag{1.7}$$

posiada nieskończenie wiele rozwiązań zależnych od jednego parametru.

Dowód. Macierzą współczynników układu równań jest $A_B^{\top}$, gdzie A_B jest podmacierzą macierzy transportowej A, której (liniowo niezależne) kolumny odpowiadają zmiennym bazowym $x_{i_1j_1}, x_{i_2j_2}, ..., x_{i_pj_p}$. Zatem

$$\mathrm{r}(A_B^{\top}) = p = m + n - 1.$$

Łatwo zauważyć, że rząd macierzy rozszerzonej układu (1.7) jest również równy p, gdyż jest to układ p równań. Na mocy twierdzenia Kroneckera–Capellego układ ten posiada nieskończenie wiele rozwiązań zależnych od $m+n-p = 1$ parametru (liczba zmiennych w układzie jest równa $m + n$ na mocy uwagi 1.3.1). □

Twierdzenie 1.3.7. *Niech X będzie bazowym rozwiązaniem dopuszczalnym ze zmiennymi bazowymi $x_{i_1 j_1}, x_{i_2 j_2}, ..., x_{i_p j_p}$. Ujemne koszty zredukowane $\bar{c}_{ij} = c_{ij} - u_i - v_j$ są niezależne od przyjętego rozwiązania układu równań* (1.7).

Dowód. Niech

$$L = \{(i,j), (i,l_1), (k_1,l_1), (k_1,l_2), ..., (k_r,l_r), (k_r,j)\}$$

będzie pętlą bazową. Zachodzą równości

$$\begin{aligned} \bar{c}_{ij} &= c_{ij} - u_i - v_j \\ &= c_{ij} - (u_i + v_{l_1}) + (u_{k_1} + v_{l_1}) - (u_{k_1} + v_{l_2}) + ... \\ &\quad +(u_{k_r} + v_{l_r}) - (u_{k_r} + v_j) \\ &= c_{ij} - c_{il_1} + c_{k_1 l_1} - c_{k_1 l_2} + ... + c_{k_r l_r} - c_{k_r j}. \end{aligned}$$

□

Uwaga 1.3.8. Zgodnie z dowodem powyższego twierdzenia, ujemne koszty zredukowane można również wyznaczyć ze wzoru

$$\bar{c}_{ij} = c_{ij} - c_{il_1} + c_{k_1 l_1} - c_{k_1 l_2} + ... + c_{k_r l_r} - c_{k_r j}.$$

1.3.2. Ogólny opis algorytmu transportowego

Startując z dopuszczalnego rozwiązania bazowego (które można otrzymać jedną z podanych uprzednio metod) algorytm transportowy pozwala na wyznaczenie rozwiązania optymalnego po wykonaniu skończenie wielu iteracji. Podobnie jak w algorytmie sympleksowym, każda iteracja polega na wymianie bazy: jedna ze zmiennych bazowych zostaje zastąpiona jedną ze zmiennych niebazowych. Aby wybrać zmienną niebazową, która zostanie wzięta do bazy, należy wyznaczyć ujemne koszty zredukowane $\bar{c}_{ij} = c_{ij} - u_i - v_j$ dla wszystkich zmiennych niebazowych. W tym celu należy najpierw wyznaczyć zmienne dualne u_i i v_j, $i = 1, ..., m$, $j = 1, ..., n$, które odpowiadają aktualnej bazie. Można je wyznaczyć korzystając z twierdzenia o komplementarności (patrz uwaga 1.1.7):

$$x_{ij} \text{ jest zmienną bazową } \Rightarrow c_{ij} = u_i + v_j.$$

Do bazy można włączyć zmienną bazową, której ujemne koszty zredukowane są ujemne lub przynajmniej niedodatnie (wybór ten odpowiada wyborowi kolumny głównej w algorytmie sympleksowym). Najczęściej wybiera się zmienną niebazową, której odpowiada najmniejszy ujemny koszt zredukowany. Następnie wybiera się zmienną bazową, która przy danej wymianie bazy opuszcza tę bazę. W tym celu wyznacza się pętlę bazową dla aktualnej bazy i dla ustalonej już zmiennej niebazowej, która ma być wzięta do bazy (patrz twierdzenie 1.3.2). Wybrać należy najmniejszą zmienną bazową x_{ij} taką, że $(i,j) \in L^-$ (odpowiada to wyborowi wiersza głównego w algorytmie sympleksowym). Przy wymianie bazy, wartości zmiennych x_{ij} odpowiadających pętli bazowej zostają wyznaczone tak, aby zmienna opuszczająca bazę przyjęła wartość 0 i aby spełnione były nadal wszystkie ograniczenia (odpowiada to piwotyzacji w algorytmie sympleksowym).

1.3.3. Algorytm transportowy

Dane jest zbilansowane zadanie transportowe z m dostawcami i n odbiorcami. Niech $a \in \mathbb{R}^m$ będzie wektorem zapasów (podaży), $b \in \mathbb{R}^n$ – wektorem zapotrzebowań (popytu) i C – macierzą kosztów.

Algorytm 1.3.9 (transportowy)

Krok 1. Wyznaczyć dopuszczalne rozwiązanie bazowe X i odpowiadającą mu bazę B (zbiór indeksów zmiennych bazowych).

Krok 2. (*wyznaczenie zmiennych dualnych*). Wyznaczyć dowolne rozwiązanie układu $m + n - 1$ równań

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

z $m + n$ niewiadomymi (na przykład nadać jednej ze zmiennych ustaloną wartość i wyznaczyć pozostałe $m + n - 1$ zmienne z układu równań).

Krok 3. (*kryterium zatrzymania*)

(a) Wyznaczyć ujemne koszty zredukowane

$$\bar{c}_{ij} = c_{ij} - u_i - v_j$$

dla $(i, j) \notin B$.

(b) Jeśli wszystkie $\bar{c}_{ij}$ są nieujemne, to X jest rozwiązaniem optymalnym i algorytm zatrzymuje się.

Krok 4. (*wymiana bazy*)

(a) Wybrać parę $(i_0, j_0) \notin B$, taką że

$$\bar{c}_{i_0 j_0} = \min\{\bar{c}_{ij} : (i, j) \notin B\}$$

(para (i_0, j_0) zostanie włączona do bazy).

(b) Wyznaczyć pętlę bazową

$$L = \{(i_0, j_0), (i_0, l_1), (k_1, l_1), (k_1, l_2), ..., (k_r, l_r), (k_r, j_0)\}$$

zawartą w zbiorze $\{(i_0, j_0)\} \cup B$ i odpowiadające jej półcykle L^+ i L^-.

(c) Wybrać parę $(i_k, j_k) \in L^-$, dla której

$$x_{i_k j_k} = \Delta = \min\{x_{ij} : (i, j) \in L^-\}$$

(para (i_k, j_k) zostanie usunięta z bazy).

(d) Położyć

$$x_{i_0 j_0} = \Delta.$$

(e) Położyć
$$x_{ij} = x_{ij} - \Delta$$
dla wszystkich $(i, j) \in L^-$.

(f) Położyć
$$x_{ij} = x_{ij} + \Delta$$
dla wszystkich $(i, j) \in L^+$.

(g) Z nowym rozwiązaniem bazowym X przejść do kroku 2.

Twierdzenie 1.3.10. *Jeśli w trakcie realizacji algorytmu transportowego nie wystąpi degeneracja, to rozwiązanie optymalne otrzymuje się w skończenie wielu iteracjach.*

Dowód twierdzenia jest podobny do dowodu skończonej zbieżności algorytmu sympleksowego.

Uwaga 1.3.11. Do tej pory nie pokazano skończonej zbieżności algorytmu transportowego w przypadku wystąpienia degeneracji, jak i również nie znaleziono przykładu zadania transportowego z występującą w trakcie realizacji algorytmu transportowego degeneracją, dla którego algorytm ten nie byłby zbieżny w skończenie wielu krokach.

Uwaga 1.3.12. Ponieważ w trakcie realizacji algorytmu transportowego jedynymi działaniami arytmetycznymi są dodawanie i odejmowanie, więc otrzymane rozwiązanie będzie całkowitoliczbowe, o ile zapasy i zapotrzebowania będą liczbami całkowitymi.

Przykład 1.3.13. Rozpatrzmy zadanie transportowe określone w przykładzie 1.2.4 i rozwiązanie bazowe tego zadania
$$X = \begin{bmatrix} 6 & 4 & & \\ & 1 & 7 & \\ & & 1 & 6 \end{bmatrix}.$$
Za pomocą algorytmu transportowego wyznaczymy rozwiązanie optymalne. Otrzymamy kolejno tablice transportowe, w których naniesiono wartości zmiennych bazowych (tłusty druk), wartości zmiennych dualnych odpowiadających zmiennym bazowym i wartości ujemnych kosztów zredukowanych odpowiadających zmiennym niebazowym. W tablicach tych w lewym górnym rogu każdego okienka podano odpowiednie wartości macierzy kosztów. Każdorazowo pod tablicą podano wartości funkcji celu.

u \ v	7	2	0	-2	
0	7 **6**	2 **4**	4 4	7 9	10
3	9 -1	5 **1**	3 **7**	3 2	8
6	7 -6	7 -1	6 **1**	4 **6**	7
	6	5	8	6	b \ a

.

Wartość funkcji celu $z = 106$.

u \ v	7	2	0	4	
0	7 **5**	2 **5**	4 4	7 3	10
3	9 -1	5 **0**	3 **8**	3 -4	8
0	7 **1**	7 5	6 6	4 **6**	7
	6	5	8	6	a \ b

.

Wartość funkcji celu $z = 100$.

u \ v	7	2	4	4	
0	7 **5**	2 **5**	4 0	7 3	10
-1	9 3	5 4	3 **8**	3 **0**	8
0	7 **1**	7 5	6 2	4 **6**	7
	6	5	8	6	a \ b

.

Wartość funkcji celu $z = 100$.

Ostatnia tablica transportowa przedstawia rozwiązanie optymalne. Widzimy teraz, że druga z kolei tablica przedstawia również rozwiązanie optymalne. Jednak nie zostało to stwierdzone przez algorytm transportowy. Przyczyną jest pojawiająca się pierwotna i dualna degeneracja. Podobnie jak w przypadku algorytmu sympleksowego, możemy – dzięki dualnej degeneracji – wyznaczyć inne rozwiązanie optymalne wybierając w ostatniej tablicy parę $(i, j) = (1, 3)$, która ma być wprowadzona do bazy. Otrzymamy wówczas tablicę

u \ v	7	2	4	4	
0	7 0	2 **5**	4 **5**	7 3	10
-1	9 3	5 4	3 **3**	3 **5**	8
0	7 **6**	7 5	6 2	4 **1**	7
	6	5	8	6	a \ b

.

Tablica ta przedstawia rozwiązanie optymalne dualnie zdegenerowane.

1.4 Zagadnienie transportowe z kryterium czasu

W klasycznym zagadnieniu transportowym przedstawionym w ustępie 1.1, mając dane jednostkowe czasy transportowe wyznaczamy plan transportowy minimalizujący łączne koszty transportowe, czyli plan będący rozwiązaniem zadania

$$\begin{array}{lll} \text{minimalizować} & \sum_{i=1}^{m}\sum_{j=1}^{n} t_{ij}x_{ij} & \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} = a_i, & i = 1, ..., m, \\ & \sum_{i=1}^{m} x_{ij} = b_j, & j = 1, ..., n, \\ & x_{ij} \geq 0, & i = 1, ..., m, j = 1, ..., n, \end{array} \tag{1.8}$$

gdzie t_{ij} oznacza czas dostawy towaru z magazynu A_i do sklepu B_j, $i = 1, 2, ..., m$, $j = 1, 2, ..., n$. Problem ten można oczywiście rozwiązać metodą przedstawioną w poprzednim ustępie. Jednak w praktyce mamy niekiedy do czynienia z sytuacją, w której powinniśmy wyznaczyć plan transportowy minimalizujący maksymalny czas dostaw towaru, mając dane czasy dostaw na poszczególnych trasach. Problem tego typu występuje na przykład w przypadku transportu łatwo psującego się towaru, który zgodnie z zapasami magazynów i zapotrzebowaniami sklepów należy jak najszybciej dostarczyć do wszystkich odbiorców. Oznaczając, podobnie jak poprzednio, a_i – zapasy magazynu A_i, b_j – zapotrzebowanie sklepu B_j oraz t_{ij} - czas dostawy towaru z magazynu A_i do sklepu B_j, $i = 1, 2, ..., m$, $j = 1, 2, ..., n$, i zakładając, że zagadnienie jest zbilansowane, *zagadnienie transportowe z kryterium czasu* ma postać

$$\begin{array}{lll} \text{minimalizować} & \max\{t_{ij} : x_{ij} > 0, i = 1, 2, ..., m, j = 1, 2, ..., n\} & \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} = a_i, & i = 1, ..., m, \\ & \sum_{i=1}^{m} x_{ij} = b_j, & j = 1, ..., n, \\ & x_{ij} \geq 0, & i = 1, ..., m, j = 1, ..., n. \end{array} \tag{1.9}$$

W przeciwieństwie do klasycznego zagadnienia transportowego, zadanie to nie jest zadaniem programowania liniowego, gdyż funkcja celu nie jest liniowa. Jako minimalizacja maksimum funkcji liniowych przy ograniczeniach liniowych jest to szczególny przypadek tzw. zadania minimalizacji wypukłej. Zadanie to można wprawdzie sprowadzić do równoważnego zadania programowania liniowego i rozwiązać je metodą sympleksową, ale istnieje bardziej efektywna metoda rozwiązywania tego zadania. Metoda ta opiera się na algorytmie transportowym przedstawionym w poprzednim ustępie. Na każdym etapie przeprowadza się iterację algorytmu transportowego, przy czym modyfikowana jest macierz kosztów w zależności od rozwiązania otrzymanego w poprzedniej iteracji. Takie postępowanie gwarantuje, że w poszczególnych iteracjach otrzymywane są rozwiązania bazowe. Przedźmy do szczegółów algorytmu.

Algorytm 1.4.1 (transportowy z kryterium czasu)

Wejście: Macierz $T = [t_{ij}]_{m\times n}$, gdzie t_{ij} oznacza czas transportu towaru z i-tego magazynu do j-tego sklepu, $i = 1, 2, ..., m, j = 1, 2, ..., n$, wektor $a = (a_1, a_2, ..., a_m) > 0$ zapasów magazynów, wektor $b = (b_1, b_2, ..., b_n) > 0$ zapotrzebowań sklepów.

Wyjście: Plan transportowy $X = [x_{ij}]_{m\times n}$ będący rozwiązaniem zadania (1.9).

Krok 1. (rozwiązanie początkowe)

(a) Przyjąć $k = 0$.

(b) Wyznaczyć dopuszczalne rozwiązanie bazowe (plan transportowy X^0) zadania (1.8), na przykład przy pomocy metody minimalnego elementu macierzy kosztów albo przy pomocy klasycznego algorytmu transportowego.

Krok 2. (a) Wyznaczyć $t^k := \max\{t_{ij} : x_{ij}^k > 0, i = 1, 2, ..., m, j = 1, 2, ..., n\}$ – najdłuższy czas dla dodatnich transportów towaru (wystarczy wziąć pod uwagę dodatnie elementy rozwiązania bazowego, gdyż dla rozwiązania bazowego współrzędne niebazowe są zerami).

(b) Utworzyć macierz R^k o elementach

$$r_{ij}^k = \begin{cases} 0 & \text{jeśli } t_{ij} < t^k, \\ 1 & \text{jeśli } t_{ij} = t^k, \\ M & \text{jeśli } t_{ij} > t^k, \end{cases}$$

$i = 1, 2, ..., m, j = 1, 2, ..., n$, gdzie $M \geq m + n$ jest dowolną stałą.

(c) Przeprowadzić pojedynczą iterację algorytmu transportowego 1.3.9 przyjmując $c_{ij} = r_{ij}^k$, $i = 1, 2, ..., m, j = 1, 2, ..., n$:

i. Dla danego dopuszczalnego rozwiązania bazowego $X^k = [x_{ij}^k]_{m\times n}$ i dla macierzy kosztów R^k, wyznaczyć odpowiadające mu zmienne dualne $u^k := (u_1^k, u_2^k, ..., u_m^k)$ i $v^k := (v_1^k, v_2^k, ..., v_n^k)$.

ii. Wyznaczyć ujemne koszty zredukowane $\bar{r}_{ij}^k := r_{ij}^k - u_i^k - v_j^k$, $i = 1, 2, ..., m, j = 1, 2, ..., n$. Jeśli $\bar{r}_{ij}^k \geq 0$, $i = 1, 2, ..., m, j = 1, 2, ..., n$, to X^k jest rozwiązaniem optymalnym zadania (1.9) i algorytm kończy swoje działanie.

iii. W przeciwnym przypadku ($\bar{r}_{ij}^k < 0$ dla pewnych $i = 1, 2, ..., m, j = 1, 2, ..., n$) wyznaczyć zmienną wprowadzaną do bazy, pętlę bazową i zmienną usuwaną z bazy.

iv. Zgodnie z regułami algorytmu transportowego wyznaczyć nowe dopuszczalne rozwiązanie bazowe $X^{k+1} = [x_{ij}^{k+1}]_{m\times n}$, położyć $k := k + 1$ i przejść do kroku 2(a).

Dowód skończonej zbieżności Algorytmu 1.4.1 do rozwiązania zadania (1.8) można znaleźć w pracy [Szw66].

Na marginesie zauważmy, że wielkość M w algorytmie 1.4.1 może być dowolną stałą większą niż długość najdłuższej pętli bazowej. Można pokazać, że wówczas ujemne koszty zredukowane $\bar{r}_{ij}^k = r_{ij}^k - u_i^k - v_j^k$ wyznaczane w kroku 2(c) algorytmu będą dodatnie dla dowolnej zmiennej niebazowej x_{ij}, dla której $t_{ij} > t^k$ (wystarczy w tym celu posłużyć się rozumowaniem podobnym do użytego w dowodzie twierdzenia 1.3.7. W konsekwencji taka zmienna nie będzie wprowadzona do bazy. Po tej obserwacji skończoną zbieżność można w skrócie uzasadnić również tym, że w każdej iteracji wyznaczane jest rozwiązanie bazowe, których to rozwiązań jest skończenie wiele. Podobnie jak w przypadku algorytmu sympleksowego, przy założeniu braku degeneracji, w każdej iteracji wartość funkcji celu zmniejsza się (do bazy wprowadzana jest zmienna x_{ij}, dla której $t_{ij} < t^k$. Prowadzi to do tego, że w kolejnych iteracjach najdłuższy czas transportu dla rozwiązania bazowego ulega skróceniu. Zatem po wykonaniu skończonej liczby iteracji otrzymamy rozwiązanie bazowe, dla którego wszystkie ujemne koszty zredukowane będą nieujemne. Takie rozwiązanie w powiązaniu z warunkami wystarczającymi minimalizacji wypukłej jest rozwiązaniem optymalnym zadania (1.9).

Przykład 1.4.2 Dana jest macierz jednostkowych czasów transportu towaru

$$T = \begin{bmatrix} 1 & 9 & 4 & 3 \\ 4 & 1 & 6 & 7 \\ 8 & 3 & 5 & 2 \end{bmatrix},$$

wektor zapasów magazynowych $a = (10, 15, 25)$, wektor zapotrzebowań sklepów $b = (5, 10, 20, 15)$. Wyznaczymy:

(a) plan transportowy minimalizujący łączny czas transportu, czyli plan będący rozwiązaniem zadania (1.8);

(b) plan transportowy X minimalizujący najdłuższy jednostkowy czas transportu, czyli plan będący rozwiązaniem zadania (1.9).

W poniższej tablicy transportowej wyznaczyliśmy bazowe rozwiązanie dopuszczalne metodą najmniejszego elementu macierzy kosztów, zmienne dualne i ujemne koszty zredukowane.

u \ v	1	-1	4	1	
0	1 / **5**	9 / 10	4 / **5**	3 / 2	10
2	4 / 1	1 / **10**	6 / **5**	7 / 4	15
1	8 / 6	3 / 3	5 / **10**	2 / **15**	25
	5	10	20	15	b \ a

Widzimy, że wszystkie ujemne koszty zredukowane są nieujemne, zatem tablica ta przedstawia rozwiązanie optymalne

$$X^0 = \begin{bmatrix} 5 & & 5 & \\ & 10 & 5 & \\ & & 10 & 15 \end{bmatrix}$$

zadania (1.8) z macierzą kosztów T. Na marginesie zauważmy, że najkrótszy łączny czas transportu wynosi $z^* = 145$. Zrealizowaliśmy więc krok 1. algorytmu transportowego z kryterium czasu. Najdłuższy czas transportu dla dodatnich przepływów wyznaczony w kroku 2(a) wynosi $t^0 = 6$. Przyjmijmy $M = 10$. Następnie w kroku 2(b) wyznaczamy macierz R^0:

$$R^0 = \begin{bmatrix} 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 10 \\ 10 & 0 & 0 & 0 \end{bmatrix}.$$

Następnie, w kroku 2(c) przeprowadzamy pojedynczą iterację algorytmu transportowego 1.3.9 dla macierzy kosztów R^0. Poniższa tabela przedstawia wyniki tej iteracji: dla danego dopuszczalnego rozwiązania bazowego wyznaczyliśmy zmienne dualne i ujemne koszty zredukowane:

u \ v	0	-1	0	0	
0	0 **5**	10 11	0 **5**	0 0	10
1	0 -1	0 **10**	1 **5**	10 9	15
0	10 10	0 1	0 **10**	0 **15**	25
	5	10	20	15	b \ a

.

Ponieważ najmniejszy ujemny koszt zredukowany $\bar{r}^0_{21}$ jest ujemny, więc wprowadzamy do bazy zmienną x_{21}, tworzymy pętlę bazową, usuwamy z bazy zmienną x_{23} i wyznaczamy nowe dopuszczalne rozwiązanie bazowe

$$X^1 = \begin{bmatrix} 0 & & 10 & \\ 5 & 10 & & \\ & & 10 & 15 \end{bmatrix}.$$

Przy okazji zauważamy, że jest ono zdegenerowane (jedna ze zmiennych bazowych jest równa 0). Następnie zwiększamy k o 1 (czyli $k = 1$) i przechodzimy do kroku 2(a). Dla dopuszczalnego rozwiązania bazowego X^1 wyznaczamy $t^1 = 5$, tworzymy macierz kosztów

$$R^1 = \begin{bmatrix} 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 10 \\ 10 & 0 & 1 & 0 \end{bmatrix}$$

i ponownie przeprowadzamy pojedynczą iterację algorytmu transportowego 1.3.9 dla macierzy kosztów R^1. Poniższa tabela przedstawia wyniki tej iteracji: dla danego dopuszczalnego rozwiązania bazowego wyznaczyliśmy zmienne dualne i ujemne koszty zredukowane:

u \ v	0	0	0	-1	
0	0 **0**	10 10	0 **10**	0 1	10
0	0 **5**	0 **10**	10 10	10 11	15
1	10 9	0 -1	1 **10**	0 **15**	25
	5	10	20	15	b \ a

.

Ponieważ najmniejszy ujemny koszt zredukowany $\bar{r}^1_{32}$ jest ujemny, więc wprowadzamy do bazy zmienną x_{32}, tworzymy pętlę bazową, usuwamy z bazy zmienną x_{11} i wyznaczamy nowe dopuszczalne rozwiązanie bazowe

$$X^2 = \begin{bmatrix} & & 10 & \\ 5 & 10 & & \\ & 0 & 10 & 15 \end{bmatrix}.$$

Zauważmy przy okazji, że różni się ono od X^1 jedynie tym, że zmienna x_{32} jest teraz zmienną bazową, a zmienna x_{11} jest zmienną niebazową, natomiast przepływy są dla obu planów transportowych identyczne. Jest to spowodowane degeneracją rozwiązania bazowego X^1. Następnie zwiększamy k o 1 (teraz $k = 2$) i przechodzimy do kroku 2(a). Dla dopuszczalnego rozwiązania bazowego X^2 wyznaczamy $t^2 = 5$, tworzymy macierz kosztów

$$R^2 = \begin{bmatrix} 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 10 \\ 10 & 0 & 1 & 0 \end{bmatrix}.$$

Następnie, w kroku 2(c) przeprowadzamy pojedynczą iterację algorytmu transportowego 1.3.9 dla macierzy kosztów R^2. Poniższa tabela przedstawia wyniki tej iteracji: dla danego dopuszczalnego rozwiązania bazowego wyznaczyliśmy zmienne dualne i ujemne koszty zredukowane:

u \ v	-1	-1	0	-1	
0	0 / 1	10 / 11	0 / **10**	0 / 1	10
1	0 / **5**	0 / **10**	10 / 9	10 / 10	15
1	10 / 10	0 / **0**	1 / **10**	0 / **15**	25
	5	10	20	15	a \ b

Ponieważ $\bar{r}^2_{ij} \geq 0$, $i = 1, 2, ..., m, j = 1, 2, ..., n$, to X^2 jest rozwiązaniem optymalnym zadania (1.9) i algorytm kończy swoje działanie. Optymalna wartość funkcji celu (najkrótszy czas transportu) wynosi $t^2 = 5$. Zwróćmy uwagę na to, że dla rozwiązania optymalnego X^0 zadania (1.9), czyli minimalizującego łączny czas transportu, najkrótszy czas transportu wynosi 6, czyli X^0 nie jest rozwiązaniem optymalnym zadania (1.9). Również X^2 nie jest rozwiązaniem optymalnym zadania (1.8), ponieważ dla X^2 łączny czas transportu wynosi $z = 150 > 145 = z^*$. Oznacza to, że oba zadania (1.8) i (1.9) mogą mieć różne rozwiązania optymalne.

1.5 Zagadnienie przydziału

Rzopocznijmy od następującego przykładu

Przykład 1.5.1 W pewnym zakładzie pracy czterem pracownikom P_1, P_2, P_3, P_4 należy przydzielić cztery czynności C_1, C_2, C_3, C_4. Na podstawie testów ustalono koszty wykonywania poszczególnych czynności przez poszczególnych pracowników. Wyniki podane są w następującej tabeli

	C_1	C_2	C_3	C_4
P_1	5	4	5	3
P_2	4	6	3	3
P_3	6	7	4	4
P_4	5	8	5	4

Zadanie polega na przydzieleniu czynności poszczególnym pracownikom tak aby łączny koszt takiego przydziału był jak najmniejszy. Przydziału należy dokonać tak, aby każdemu pracownikowi przydzielono dokładnie jedną czynność i każda czynność była przydzielona dokładnie jednemu pracownikowi.

Możemy teraz przejść do ogólnej definicji *zagadnienia przydziału* (ang. *assignment problem*) Należy obsadzić n stanowisk roboczych przez n osób (pracowników). Znane sa efekty pracy i-tego pracownika na j-tym stanowisku. Efekty te wyrażone są za pomocą macierzy kosztów $C = [c_{ij}]_{n \times n}$, gdzie c_{ij} oznacza koszt przydzielenia i-temu pracownikowi j-tej czynności, $i, j, = 1, 2, ..., n$. W przykładzie 1.5.1 mamy

$$C = \begin{bmatrix} 5 & 4 & 5 & 3 \\ 4 & 6 & 3 & 3 \\ 6 & 7 & 4 & 4 \\ 5 & 8 & 5 & 4 \end{bmatrix},$$

Alternatywnie za c_{ij} zamiast kosztu można przyjąc czas wykonywania j-tej czynności przez i-tego pracownika, $i, j = 1, 2, ..., n$. Przydziału należy dokonać tak, aby łączny koszt przydziału był minimalny, przy czym każdemu pracownikowi należy przydzielić dokładnie jedną czynność i każdą czynność należy przydzielić dokładnie jednemu pracownikowi. Zanim przejdziemy do przedstawienia modelu matematycznego tego zadania wróćmy do przykładu 1.5.1.

Przykład 1.5.2 W przykładzie 1.5.1 istnieje łącznie $4! = 24$ możliwości przydziału. Teoretycznie można więc policzyć łączne koszty dla każdego spośród 24 przydziałów i wybrać najlepszy z nich. Jest to jednak obliczeniowo bardzo pracochłonne. Złożoność obliczeniowa takiego podejścia byłaby wykładnicza.

Oznaczając

$$x_{ij} = \begin{cases} 1 & \text{gdy } i\text{-temu pracownikowi przydzielono } j\text{-tą czynność} \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

$i, j = 1, 2, 3, 4$, z warunków zadania wynika, że $X = [x_{ij}]_{4 \times 4}$ jest rozwiązaniem dopuszczalnym wtedy i tylko wtedy, gdy $x_{ij} \in \{0, 1\}$, $i, j = 1, 2, 3, 4$, oraz

$$
\begin{aligned}
x_{11} + x_{12} + x_{13} + x_{14} &= 1 \\
x_{21} + x_{22} + x_{23} + x_{24} &= 1 \\
x_{31} + x_{32} + x_{33} + x_{34} &= 1 \\
x_{41} + x_{42} + x_{43} + x_{44} &= 1 \\
x_{11} + x_{21} + x_{31} + x_{41} &= 1 \\
x_{12} + x_{22} + x_{32} + x_{42} &= 1 \\
x_{13} + x_{23} + x_{33} + x_{43} &= 1 \\
x_{14} + x_{24} + x_{34} + x_{44} &= 1
\end{aligned}
$$

Pierwsze cztery z tych równości oznaczają, że każdemu pracownikowi przydzielono dokładnie jedną z czterech czynności. Pozostałe cztery równości oznaczają, że każda z 4 czynności została przydzielona dokładnie jednemu pracownikowi. Łaczny koszt przydziału ma postać $\sum_{i=1}^{4} \sum_{j=1}^{4} c_{ij} x_{ij}$, gdzie c_{ij} jest kosztem przydziału j-tej czynności i-temu pracownikowi. Widzimy więc, że opisywany problem jest szczególnym przypadkiem zadania transportowego z macierzą kosztów C i z jednostkowymi zapasami magazynów i jednostkowymi zapotrzebowaniami sklepów. Zadanie to możemy więc rozwiązać za pomocą algorytmu transportowego. Początkowe bazowe rozwiązanie dopuszczalne otrzymane metodą kąta północno-zachodniego ma postać

	C_i	C_2	C_3	C_4
P_1	1	0		
P_2		1	0	
P_3			1	0
P_4				1

i koszt przydziału wynosi 19. Zauważmy, że jest to rozwiązanie zdegenerowane, co ma zwykle miejsce dla zagadnienia przydziału. Natomiast początkowe bazowe rozwiązanie dopuszczalne otrzymane metodą minimalnego elementu macierzy kosztów ma postać

	C_i	C_2	C_3	C_4
P_1				1
P_2			1	
P_3		1	0	0
P_4	1			0

i koszt przydziału wynosi 17. Sprawdzamy, czy to ostatnie jest rozwiązaniem optymalnym. W tym celu zapiszmy to rozwiązanie łącznie z macierzą kosztów w jednej tablicy

	C_i	C_2	C_3	C_4
P_1	5	4	5	3 $\boxed{\mathbf{1}}$
P_2	4	6	3 $\boxed{\mathbf{1}}$	3
P_3	6	7 $\boxed{\mathbf{1}}$	4 $\boxed{\mathbf{0}}$	4 $\boxed{\mathbf{0}}$
P_4	5 $\boxed{\mathbf{1}}$	8	5	4 $\boxed{\mathbf{0}}$

Zadanie dualne ma postać

$$\begin{array}{ll} \text{maksymalizować} & z = \sum_{i=1}^{4} u_i + \sum_{i=1}^{4} v_i \\ \text{przy ograniczeniach} & u_i + v_j \leq c_{ij},\ i, j \in \{1, 2, 3, 4\}, \end{array}$$

gdzie u_i, v_j oznaczają zmienne dualne $i, j = 1, 2, 3, 4$. Zmienne dualne odpowiadające omawianemu początkowemu rozwiązaniu bazowemu wyznacza się z układu równań

$$\begin{aligned} u_1 + v_4 &= 3 \\ u_2 + v_3 &= 3 \\ u_3 + v_2 &= 7 \\ u_3 + v_3 &= 4 \\ u_3 + v_4 &= 4 \\ u_4 + v_1 &= 5 \\ u_4 + v_4 &= 4 \end{aligned}$$

Rowiązaniem tego układu równań jest na przykład $u = (3, 3, 4, 4), v = (1, 3, 0, 0)$. Następnie wyznaczamy ujemne koszty zredukowane $\bar{c}_{ij} = c_{ij} - u_i - v_j$ i wstawiamy je do tablicy

	$v_1 = 1$	$v_2 = 3$	$v_3 = 0$	$v_4 = 0$
$u_1 = 3$	5 1	4 -2	5 2	3 $\boxed{1}$
$u_2 = 3$	4 0	6 0	3 $\boxed{1}$	3 0
$u_3 = 4$	6 1	7 $\boxed{1}$	4 $\boxed{0}$	4 $\boxed{0}$
$u_4 = 4$	5 $\boxed{1}$	8 1	5 1	4 $\boxed{0}$

Widzimy, że rozwiązanie to nie jest optymalne, gdyż $\bar{c}_{12} < 0$. Zatem zmienną x_{12} wprowadzamy do bazy i tworzymy pętlę bazową w celu wyznaczenia zmiennej usuwanej z bazy. Jest nią x_{14} lub x_{32} (obie wartości są najmniejsze w cyklu L_-). Jeśli z bazy usuniemy x_{14}, to nowa tablica transportowa wraz z wpisanymi odpowiadającymi jej zmiennymi dualnymi i ujemnymi kosztami zredukowanymi będzie miała postać

	$v_1 = 5$	$v_2 = 7$	$v_3 = 4$	$v_4 = 4$
$u_1 = -3$	5 3	4 $\boxed{1}$	5 4	3 2
$u_2 = -1$	4 0	6 0	3 $\boxed{1}$	3 0
$u_3 = 0$	6 1	7 $\boxed{0}$	4 $\boxed{0}$	4 $\boxed{1}$
$u_4 = 0$	5 $\boxed{1}$	8 1	5 1	4 $\boxed{0}$

Ponieważ wszystkie ujemne koszty zredukowane odpowiadające zmiennym niebazowym są nieujemne, więc tablica ta przedstawia rozwiązanie optymalne

$$X = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

czyli optymalny przydział wygląda następująco: $P_1 \leftrightarrow C_2, P_2 \leftrightarrow C_3, P_3 \leftrightarrow C_4, P_4 \leftrightarrow C_1$. Minimalny łączny koszt przydziału wynosi $4 + 3 + 4 + 5 = 16$. Ponieważ ostatnia tablica przedstawia rozwiązanie zdegenerowane, istnieją inne przydziały optymalne.

Oznaczmy $I := \{1, 2, ..., m\}$ i $J := \{1, 2, ..., n\}$. W ogólnym przypadku model matematyczny zagadnienia przydziału jest następujący

$$\begin{array}{ll} \text{minimalizować} & \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\ \text{przy ograniczeniach} & \sum_{j=1}^{n} x_{ij} = 1, i \in I, \\ & \sum_{i=1}^{n} x_{ij} = 1, j \in J, \\ & x_{ij} \in \{0, 1\}, i \in I, j \in J. \end{array}$$

Łącznie istnieje $n!$ możliwości przydziału n czynności n pracownikom. Teoretycznie można więc policzyć łączne koszty dla każdego spośród $n!$ przydziałów i wybrać najlepszy z nich. Jest to jednak obliczeniowo bardzo pracochłonne (np $20! \cong 2.4 * 10^{18}$). Złożoność obliczeniowa takiego podejścia jest wykładnicza. Jak zauważyliśmy w przykładzie 1.5.2 zadanie to jest szczególnym przypadkiem zadania transportowego i możemy je rozwiązać za pomocą algorytmu transportowego. Z uwagi jednak na szczególną postać istnieje prostsza metoda rozwiązania zagadnienia przydziału. Zanim przejdziemy do jej opisu poczynimy kilka uwag dotyczących zadania transportowego, w szczególności zagadnienia przydziału.

Uwaga 1.5.3 Po wyznaczeniu bazowego rozwiązania dopuszczalnego $X = [x_{ij}]$, gdzie x_{ij} dla $(i, j) \in B \subseteq I \times J$ są zmiennymi bazowymi (jest ich $2n-1$), a $x_{ij} = 0$ dla $(i, j) \notin B$ i wyznaczeniu zmiennych dualnych $u_i, v_j, i \in I, j \in J$, z układu $2n - 1$ równań z $2n$ niewiadomymi

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

ujemne koszty zredukowane liczone są według wzoru

$$\bar{c}_{ij} = c_{ij} - u_i - v_j, (i, j) \notin B.$$

Utwórzmy nową (pomocniczą) macierz kosztów $C' = [c'_{ij}]_{n \times n}$, gdzie $c'_{ij} = c_{ij} + d_i + e_j$ dla pewnych liczb d_i i e_j, $i \in I, j \in J$. Zauważmy, że dla tego nowego (pomocniczego) zagadnienia X jest nadal rozwiązaniem bazowym. Ponadto wielkości $u'_i := u_i + d_i$ i $v'_j := v_j + e_j$, $i \in I, j \in J$, są odpowiadającymi temu rozwiązaniu zmiennymi dualnymi (spełniają one układ równań $u'_i + v'_j = c'_{ij}, (i, j) \in B$), oraz pozostałe ujemne koszty zredukowane $\bar{c}'_{ij} := c'_{ij} - u'_i - v'_j$, $(i, j) \notin B$, nie zmienią się, ponieważ

$$\bar{c}'_{ij} := c'_{ij} - u'_i - v'_j = c_{ij} + d_i + e_j - (u_i + d_i) - (v_j + e_j) = c_{ij} - u_i - v_j = c_{ij},$$

$i \in I, j \in J$. Zgodnie z twierdzeniem o komplementarności dla zadania transportowego, rozwiązanie dopuszczalne X jest rozwiązaniem optymalnym wtedy i tylko wtedy, gdy $x_{ij}(c'_{ij} -$

$u_i' - v_j') = 0$, $i \in I, j \in J$. Zatem modyfikując w powyższy sposób macierz kosztów (zastępując macierz C przez macierz C') warunki konieczne i wystarczające optymalności nie zmienią się (X jest rozwiązaniem optymalnym dla macierzy kosztów C wtedy i tylko wtedy, gdy X jest rozwiązaniem optymalnym dla zmodyfikowanej macierzy kosztów C'). Okazuje się, że w przypadku zagadnienia przydziału taka modyfikacja może prowadzić do zadania prostszego do rozwiązania. Oznaczmy $d_i = -\min_{j \in J} c_{ij}$ ($-d_i$ jest najmniejszym elementem w i-tym wierszu macierzy kosztów C, $i \in I$) i utwórzmy macierz $C' = [c_{ij}']_{n \times n}$ gdzie $c_{ij}' = c_{ij} + d_i$ (w poszczególnych wierszach macierzy C każdy element pomniejszamy o element najmniejszy w danym wierszu). Dalej oznaczmy $e_j = -\min_{i \in I} c_{ij}'$ ($-e_j$ jest najmniejszym elementem w j-tej kolumnie zmodyfikowanej macierzy kosztów C', $i \in I$) i utwórzmy macierz $C'' = [c_{ij}'']_{n \times n}$ gdzie $c_{ij}'' = c_{ij}' + e_j$ (w poszczególnych kolumnach macierzy C każdy element pomniejszamy o element najmniejszy w danej kolumnie). Zgodnie z poprzednimi rozważaniami zadanie transportowe ze zmodyfikowaną macierzą C'' prowadzi do tych samych rozwiązań optymalnych. Oba zadania są więc sobie równoważne. Zmienia się jedynie wartość optymalna, ale dla wyjściowego zadania dla danego rozwiązania optymalnego X można ją wyznaczyć posługując się wyjściową macierzą kosztów C. W zmodyfikowanej macierzy kosztów C'' pojawi się natomiast co najmniej n zer (co najmniej po jednym w każdej kolumnie).

Przykład 1.5.4 Zastosujemy uwagę 1.5.3 do zadania przydziału opisanego w przykładzie 1.5.1. Macierze C, C' i C'' mają postać

$$C = \begin{bmatrix} 5 & 4 & 5 & 3 \\ 4 & 6 & 3 & 3 \\ 6 & 7 & 4 & 4 \\ 5 & 8 & 5 & 4 \end{bmatrix}, C' = \begin{bmatrix} 2 & 1 & 2 & 0 \\ 1 & 3 & 0 & 0 \\ 2 & 3 & 0 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}, C'' = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 3 & 1 & 0 \end{bmatrix}.$$

Dla każdej z tych macierzy kosztów zadania przydziału są sobie równoważne. Ponadto dowolna macierz permutacji P_4 jest rozwiązaniem dopuszczalnym zagadnienia przydziału. W omawianym przykładzie widzimy, że w rozwiązaniu dopuszczalnym

$$X = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

(będącym macierzą permutacji) wszystkie pozycje jedynek są zgodne z niektórymi pozycjami zer w macierzy C''. Oznacza to, że X jest rozwiązaniem optymalnym (dla macierzy C'' łączny koszt wynosi 0, a więc jest minimalny). Dla wyjściowego zadania X jest więc również rozwiązaniem optymalnym i minimalny łączny koszt przydziału wynosi $4 + 3 + 4 + 5 = 16$. Ten przykład jest jednak szczególny, bowiem w przykładzie tym z ośmiu pozycji dla zer w macierzy C'' można wybrać cztery takie, które dają macierz permutacji. Ogólnie tak nie musi być.

Przykład 1.5.5 Rozważmy zadanie przydziału z macierzą kosztów

$$C = \begin{bmatrix} 82 & 83 & 69 & 92 \\ 77 & 37 & 49 & 92 \\ 11 & 69 & 5 & 86 \\ 8 & 9 & 98 & 23 \end{bmatrix}.$$

Macierze C' i C'' powstałe z macierzy C przez odjęcie kolejno od każdego elementu najmniejszego elementu w danym wierszu, a następnie od każdego elementu najmniejszego elementu w danej kolumnie mają postać

$$C' = \begin{bmatrix} 13 & 14 & 0 & 23 \\ 40 & 0 & 12 & 55 \\ 6 & 64 & 0 & 81 \\ 0 & 1 & 90 & 15 \end{bmatrix}, \; C'' = \begin{bmatrix} 13 & 14 & 0 & 8 \\ 40 & 0 & 12 & 40 \\ 6 & 64 & 0 & 66 \\ 0 & 0 & 90 & 0 \end{bmatrix}$$

W macierzy C'' z sześciu pozycji dla zer nie da się wybrać czterech takich, które dałyby macierz permutacji.

Uwaga 1.5.6 Zauważmy, że macierz kwadratowa $n \times n$ zawierająca co najmniej n zer ma szczególną własność: z pozycji zer w tej macierzy da się wybrać n takich, które dają macierz permutacji wtedy i tylko wtedy, gdy najmniejsza liczba poziomych lub pionowych wierszy zawierających zera wynosi dokładnie n. Tak jest dla macierzy C'' z przykładu 1.5.4, ale nie dla macierzy C'' z przykładu 1.5.5. W takim przypadku można jednak dalej modyfikować macierz C'' (powiększając w niej liczbę zer) tak aby doprowadzić do sytuacji gdy najmniejsza liczba poziomych lub pionowych wierszy zawierających zera wynosi dokładnie n. W poniższym przykładzie opiszemy taką modyfikację.

Przykład 1.5.7 Wróćmy do macierzy C'' z przykładu 1.5.5, dla której najmniejsza liczba poziomych lub pionowych wierszy zawierających zera wynosi 3. Są to wiersze drugi i czwarty oraz kolumna trzecia, zaznaczone w poniższej macierzy na czerwono:

$$\begin{bmatrix} 13 & 14 & \color{red}{0} & 8 \\ \color{red}{40} & \color{red}{0} & \color{red}{12} & \color{red}{40} \\ 6 & 64 & \color{red}{0} & 66 \\ \color{red}{0} & \color{red}{0} & \color{red}{90} & \color{red}{0} \end{bmatrix}$$

Kolumna ta krzyżuje się z obydwoma wierszami w pozycjach $(2, 3)$ i $(4, 3)$. Spośród pozostałych elementów zaznaczonych na czarno wybieramy najmniejszy element. Jest nim 6 na pozycji $(3, 1)$. Odejmujemy ja od wszystkich pozycji zaznaczonych na czarno i dodajemy ją do elementów znajdujących się w miejscach przecięcia czerwonych wierszy i kolumn. Otrzymujemy macierz

$$C''' = \begin{bmatrix} 7 & 8 & 0 & 2 \\ 40 & 0 & 12 & 40 \\ 0 & 58 & 0 & 60 \\ 0 & 0 & 96 & 0 \end{bmatrix}$$

z jednym zerem więcej niż w macierzy C''. W kolejnej uwadze wyjaśnimy, dlaczego zagadnienie przydziału z macierzą kosztów C''' jest równoważne wyjściowemu zagadnieniu. Natomiast już teraz widzimy, że macierz C''' ma już korzystną własność: cztery z siedmiu zer jest w pozycji dającej macierz permutacji.

$$\begin{bmatrix} & & 1 & \\ & 1 & & \\ 1 & & & \\ & & & 1 \end{bmatrix}.$$

Macierz ta przedstawia przydział oprymalny, zaś koszt tego przydziału dla macierzy kosztów C wynosi $69 + 37 + 11 + 23 = 140$.

Uwaga 1.5.8 Rozważmy zagadnienie przydziału z n pracownikami i n czynnościami, z macierzą kosztów $C = [c_{ij}]_{n\times n}$,w której liczba zer wynosi co najmniej n (da się tak zrobić po ewentualnej modyfikacji, patrz uwaga 1.5.3), zaś najmniejsza liczba poziomych lub pionowych wierszy zawierających zera jest mniejsza niż n. Mając dane bazowe rozwiązanie dopuszczalne (jakakolwiek macierz permutacji) możemy wyznaczyć zmienne dualne u_i, v_j, $i, j \in J$, i ujemne koszty zredukowane $\bar{c}_{ij} = c_{ij} - u_i - v_j$, $i, j \in J$, oraz sprawdzić, czy rozwiązanie jest optymalne tak, jak się to robiło dla zadania transportowego. Zmodyfikujmy teraz macierz kosztów i zmienne dualne w następujący sposób. Wyznaczmy najmniejszą liczbę wierszy bądź kolumn macierzy C zawierających zera. Te wiersze i kolumny nazwijmy "czerwone" oraz elementy w tych wierszach bądź kolumnach nazwijmy również "czerwone", natomiast pozostałe elementy nazwijmy "czarne". Spośród wszystkich czarnych elementów wybierzmy element najmniejszy, równy d, i odejmijmy go od wszystkich czarnych elementów oraz dodajmy go do czerwonych elementów należących jednocześnie do czerwonych wierszy i czerwonych kolumn (czyli leżacych w miejscach, gdzie krzyżują się czerwone wiersze i czerwone kolumny). Tak zmodyfikowaną macierz kosztów oznaczmy symbolem C'. Zauważmy, że w w macierzy tej jest co najmniej jedno zero więcej niż w macierzy C (np. w pozycji gdzie znajdował się minimalny czarny element). Zmodyfikujmy teraz zmienne dualne: dla wierszy czarnych $u'_i := u_i - d$ a dla wierszy czerwonych $u'_i = u_i$. Dla kolumn czarnych $v'_j = v_j$ a dla kolumn czerwonych $v'_j = v_j + d$. Nietrudno zauważyć, że dla otrzymanych w ten sposób zmodyfikowanych zmiennych dualnych ujemne kosztu zredukowane nie zmienią się w stosunku do wyjściowych kosztów c_{pr} i wyjściowych zmiennych dualnych u_p, v_r, $p, r \in J$. W tym celu można posłużyć się poniższą zmodyfikowaną tabelą kosztów, w której wprowadzono zmodyfikowane koszty i zmodyfikowane zmienne dualne. Dla uproszczenia wprowadzono tylko jeden wiersz czerwony i tylko jedną kolumnę czerwoną.

	v_1		v_l		$v_j + d$		v_n
$u_1 - d$	$c_{11} - d$	$\cdots$	$c_{il} - d$	$\cdots$	c_{ij}	$\cdots$	$c_n - d$
$\vdots$	$\vdots$		$\vdots$		$\vdots$		$\vdots$
$u_k - d$	$c_{k1} - d$	$\cdots$	$c_{kl} - d$	$\cdots$	c_{kj}	$\cdots$	$c_{kn} - d$
$\vdots$	$\vdots$		$\vdots$		$\vdots$		$\vdots$
u_i	c_{i1}	$\cdots$	c_{il}		$c_{ij} + d$	$\cdots$	c_{in}
$\vdots$	$\vdots$		$\vdots$		$\vdots$		$\vdots$
$u_n - d$	$c_{n1} - d$	$\cdots$	$c_{nl} - d$	$\cdots$	c_{nj}	$\cdots$	$c_{nn} - d$

Oznacza to, że zadanie ze zmodyfikowaną w ten sposób macierzą kosztów C' jest równoważne wyjściowemu zadaniu. Ponieważ po tej modyfikacji liczba zer w macierzy kosztów zwiększy się o co najmniej jedno (np. w pozycji, gdzie znajdował się minimalny czarny element), więc powtarzając tę procedurę modyfikacji co najwyżej $n(n-1)$ razy otrzymamy macierz kosztów z zerami znajdującymi się w pozycjach dających macierz permutacji. Dla tej macierzy permutacji i dla zmodyfikowanej macierzy kosztów łączny koszt wyniesie 0, a więc ta macierz permutacji będzie rozwiązaniem optymalnym. Minimalny łączny koszt dla wyjściowego zadania otrzymamy stosując wyjściową macierz kosztów.

Opisana w uwagach 1.5.3-1.5.8 procedura nosi nazwę *metody węgierskiej* dla rozwiązania zagadnienia przydziału. Nazwa pochodzi od matematyków węgierskich Dénesa Kőniga i Jenő Egerváry'ego, których opracowali tę metodę w I połowie XX wieku. Praca Egerváry'ego w języku węgierskim z 1931 roku została przetłumaczona na angielski i rozwinięta przez Haralda

Kuhna w 1955 roku. Stosunkowo niedawno, bo w roku 2006 odkryto, że problem przydziału rozwiązał już w XIX wieku Carl Gustav Jacobi, który opublikował wyniki w roku 1890.