

*Na prawach rękopisu*

Wojciech Kordecki

## **Przepływy w sieciach transportowych**

materiały dodatkowe  
dla studentów kierunku Informatyka



Collegium Witelona Uczelnia Państwowa w Legnicy  
Legnica 2024

Przedstawione dalej rozwiązanie problemu przepływów w sieciach pochodzi od L. Szamkołowicza, oparte jest na książce [2] i jest nieco odmienne od najczęściej spotykanego w literaturze, na przykład w książkach [1] i [3]. Warto też popatrzeć w internecie na strony [4], [6] oraz [5] i zanalizować podane tam przykładowe implementacje algorytmów.

## 1 Sieci transportowe

Sformułowanie problemu przepływów w sieciach pochodzi od L. Szamkołowicza, oparte jest na książce [2] i jest nieco odmienne od najczęściej spotykanego w literaturze, na przykład w książkach [1] i [3].

Niech  $G = (V, \vec{E})$  będzie grafem skierowanym. Określmy

- $V^-(v) = \{w \in V : (w, v) \in \vec{E}\},$
- $V^+(v) = \{u \in V : (v, u) \in \vec{E}\}.$

*Siecią transportową* nazywamy trójkę  $S = (V, \vec{E}, c)$ , w której  $G = (V, \vec{E})$  jest spójnym grafem skierowanym, acyklicznym, bez wielokrotnych łuków oraz o dokładnie jednym wejściu  $s$  (tzn.  $V^-(s) = \emptyset$ ) i dokładnie jednym wyjściu  $t$  (tzn.  $V^+(t) = \emptyset$ ). Ponadto  $c : \vec{E} \rightarrow \mathbb{Z}^+$ , gdzie  $\mathbb{Z}^+$  oznacza zbiór liczb całkowitych nieujemnych. Funkcja  $c$  jest *przepustowością* sieci  $S$ .

*Strumień* określa się jako dowolną funkcję  $\varphi : \vec{E} \rightarrow \mathbb{Z}^+$  taką, że

$$(A) \quad \varphi(e) \leq c(e) \text{ dla każdego łuku } e \in \vec{E},$$

$$(B) \quad \sum_{u \in V^-(v)} \varphi((u, v)) = \sum_{w \in V^+(v)} \varphi((v, w)), \text{ dla dowolnego wierzchołka } v \in V$$

takiego, że  $v \notin s$  oraz  $v \notin t$ .

### Twierdzenie 1.

Jeśli  $s$  jest wejściem, a  $t$  jest wyjściem sieci  $S$ , to

$$\sum_{v \in V^+(s)} \varphi((s, v)) = \sum_{v \in V^-(t)} \varphi((v, t)) = \Phi_\varphi. \quad (1)$$

Liczba  $\Phi_\varphi$  występująca po prawej stronie wzoru (1) nazywa się *przepustowością* strumienia  $\varphi$ . *Strumień maksymalny* definiuje się jako strumień o największej przepustowości.

*Przekrój*  $C = (U, W)$  określa się jako parę zbiorów taką, że  $U \cap W = \emptyset$ ,  $U \cup W = V$ ,  $s \in U$ ,  $t \in W$ . Przekrój jest *normalny*, gdy podgrafy  $(U, \vec{R})$  i  $(W, \vec{R})$  są spójne. Liczbę

$$P(U, W) = \sum_{\substack{u \in U \\ w \in W}} c((u, w)) = P_C \quad (2)$$

nazywa się przepustowością przekroju  $(U, W)$ . *Przekrój minimalny*, to przekrój o najmniejszej przepustowości. Poniższe tw. 2 z roku 1956 pochodzi od Forda i Fulkersona.

**Twierdzenie 2.**

*W dowolnej sieci transportowej, przepustowość strumienia maksymalnego jest równa przepustowości przekroju minimalnego:*

$$\max_{\varphi} \Phi_{\varphi} = \min_C P_C.$$

## 2 Znajdowania przepływów maksymalnych

Niech  $l = e_1, e_2, \dots, e_{m-1}$  będzie łańcuchem o ciągu wierzchołków  $s = v_1, v_2, \dots, v_m = t$ , gdzie albo  $e_i = (v_i, v_{i+1})$  albo  $e_i = (v_{i+1}, v_i)$ . Jeśli  $e_i = (v_i, v_{i+1})$ , to łuk  $e_i$  jest *zgodnie skierowany* z łańcuchem  $l$ , a jeśli  $e_i = (v_{i+1}, v_i)$ ,  $e_i$  jest *przeciwnie skierowany* do łańcucha  $l$ . Łańcuch  $l$  jest *nienasycony* przez strumień  $\varphi$ , jeśli dla każdego łuku  $e \in l$  zgodnie skierowanego z  $l$  jest  $\varphi(e) < c(e)$ , a dla łuku  $e \in l$  przeciwnie skierowanego do  $l$  jest  $\varphi(e) > 0$ .

**Twierdzenie 3.**

*Jeżeli istnieje łańcuch  $l$  nienasycony przez strumień  $\varphi$ , to  $\varphi$  nie jest maksymalny.*

*Dowód.* Niech

$$\theta = \min_{i,j} (c(e_i) - \varphi(e_i), \varphi(e_j)), \quad (3)$$

gdzie minimum bierzemy łukach  $e_i \in l$  zgodnie skierowanych i łukach  $e_j \in l$  przeciwnie skierowanych skierowanych. Określamy nowy strumień

$$\varphi'(e) = \begin{cases} \varphi(e) & \text{dla } e \notin l, \\ \varphi(e) + \theta & \text{dla } e \text{ zgodnie skierowanych z } l, \\ \varphi(e) - \theta & \text{dla } e \text{ przeciwnie skierowanych do } l. \end{cases} \quad (4)$$

Zauważmy, że  $\varphi'$  spełnia (1), czyli  $\varphi'$  jest strumieniem oraz

$$\Phi_{\varphi'} = \Phi_{\varphi} + \theta > \Phi_{\varphi},$$

więc  $\varphi$  nie jest strumieniem maksymalnym.  $\square$

Wzory (3) i (4) podają metodę zwiększania strumienia, gdy istnieją łańcuchy nienasycone.

**Twierdzenie 4.**

*Jeżeli sieć transportowa nie zawiera żadnego łańcucha nienasyconego przez strumień  $\varphi$ , to  $\varphi$  jest maksymalny.*

Wielokrotnie stosując twierdzenie 3 zwiększamy za każdym razem strumień zgodnie ze wzorem (4). Procedurę zwiększania strumienia kończymy korzystając z twierdzenia 4.

Procedurę znajdowania strumienia maksymalnego rozpoczynamy Algorytmu 1, który wykorzystuje poniższe twierdzenie 5.

**Twierdzenie 5.**

*Warunkiem koniecznym maksymalności strumienia  $\varphi$  jest nasylenie co najmniej jednego łuku  $e$  (czyli  $\varphi(e) = c(e)$ ) dla każdej drogi łączącej  $s$  i  $t$ .*

**Algorytm 1.**

Niech  $D$  będzie zbiorem wszystkich dróg łączących  $s$  z  $t$ .

1. Niech  $d_1 = (e_1^1 \dots, e_{m_1}^1) \in D$ . Określamy  $\varphi_1(e_i^1) = \min_i c(e_i^1)$  dla  $e_i^1 \in d_1$  oraz  $\varphi(e) = 0$  dla  $e \notin d_1$ .
2. Niech  $d_j = (e_1^j \dots, e_{m_1}^j) \in D$  oraz  $d_j$  nie zawiera łuków nasyconych przez  $\varphi_{j-1}$ . Określamy

$$\varphi_j(e_i^j) = \varphi_{j-1}(e_i^j) + \min_i (c(e_i^j) - \varphi_{j-1}(e_i^j))$$

dla  $e_i^j \in d_j$  oraz  $\varphi_j(e) = \varphi_{j-1}(e)$  dla  $e \notin d_j$ . Powtarzamy ten krok dla wszystkich dróg z  $D$ .

Następny algorytm znajduje kolejne łańcuchy nienasycone i parametr  $\theta$ , aby zastosować wzór (3).

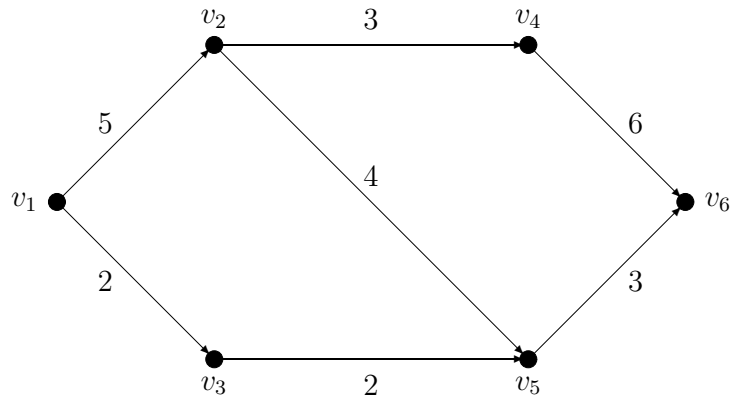
### Algorytm 2.

Wierzchołki  $v_0 = s, v_1, \dots, v_{m_1} = t$  otrzymują indeksy  $N$  i  $\theta$ .

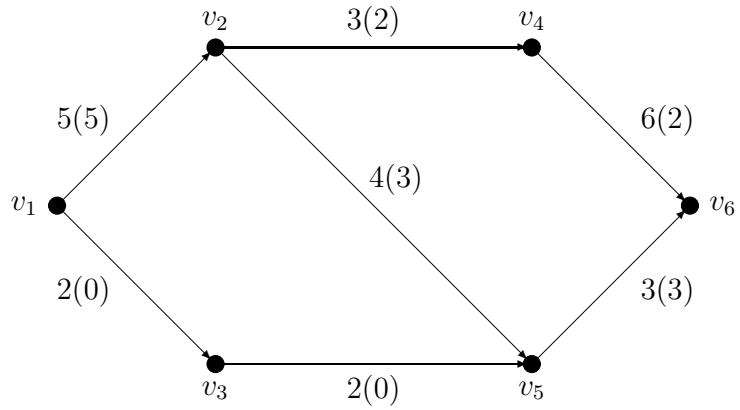
1.  $N(v_0) = 0, \theta(v_0) = \infty$ . Pozostałe wierzchołki nie posiadają indeksów.
2. Niech  $\Omega$  będzie zbiorem wierzchołków posiadających indeksy. Mamy dwie możliwości:
  - (a)  $v_j \in V^+(v_i) \setminus \Omega, c((v_i, v_j)) - \varphi((v_i, v_j)) > 0 \implies N(v_j) = i, \theta(v_j) = \min\{\theta(v_i), c((v_i, v_j)) - \varphi((v_i, v_j))\},$
  - (b)  $v_j \in V^-(v_i) \setminus \Omega, \varphi((v_i, v_j)) > 0 \implies N(v_j) = i, \theta(v_j) = \min\{\theta(v_i), \varphi((v_i, v_j))\},$
3. Jeżeli  $t \in \Omega$ , to istnieje łańcuch nienasycony ( $v_0 = s, v_1, \dots, v_{m-1} = t$ ), w którym  $i_{k-1} = N(v_{i_k})$  oraz  $\theta = \theta(t)$ .

### Przykład.

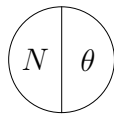
- Na łukach w sieci na rys. 1 są podane przepustowości.
- Wykonujemy algorytm dla kolejnych dróg  $(v_1, v_2, v_4, v_6), (v_1, v_2, v_5, v_6), (v_1, v_3, v_5, v_6)$ , co przedstawia rys. 2. Wszystkie drogi są nasycone i otrzymujemy strumień  $\Phi_\varphi = 5$ .
- Modyfikujemy przepływ: wierzchołki otrzymują indeksy (rys. 3) i na rys. 4 jest przedstawiony łańcuch nienasycony  $(v_0, v_3, v_5, v_2, v_4, v_6)$ .
- Modyfikujemy przepływ (rys. 5) i otrzymujemy  $\Phi_\varphi = 6$ .



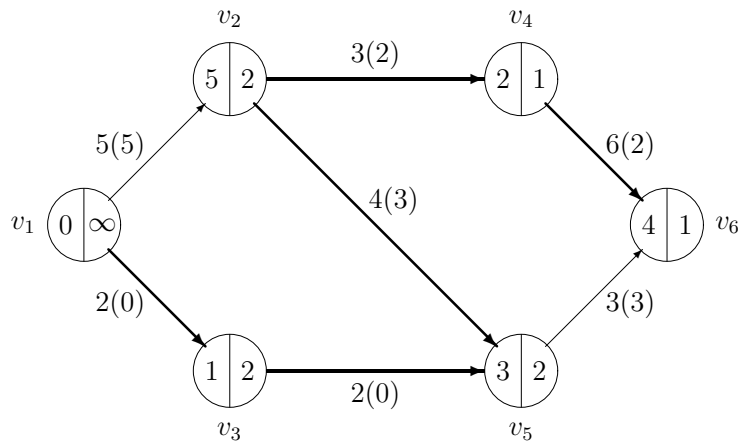
Rysunek 1: Sieć z podanymi przepustowościami



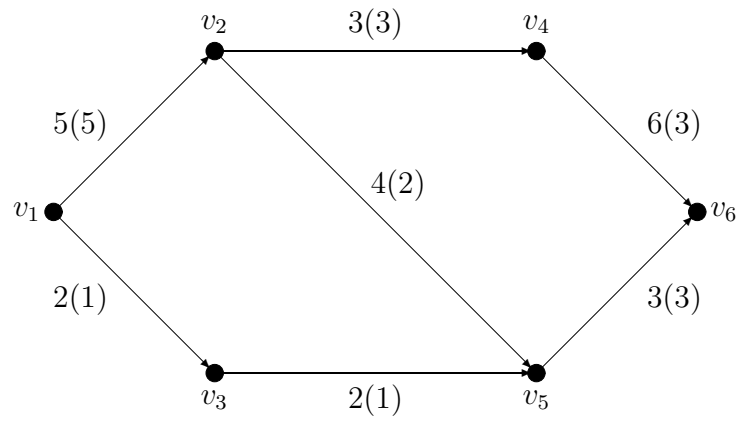
Rysunek 2: Strumień  $\Phi_\varphi = 5$



Rysunek 3: Znajdowanie łańcuchów nienasyconych



Rysunek 4: Łańcuchy nienasycone



Rysunek 5: Strumień maksymalny  $\Phi_\varphi = 6$

## Literatura

- [1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Wprowadzenie do algorytmów*, wydanie 2, WNT, Warszawa 2004.
- [2] W. Kordecki, A. Łyczkowska-Hanćkowiak, *Matematyka dyskretna dla informatyków*, Helion 2018, rozdz. 5.
- [3] W. Lipski, *Kombinatoryka dla programistów*, WNT, Warszawa 2009.
- [4] [https://en.wikipedia.org/wiki/Ford%E2%80%93Fulkerson\\_algorithm](https://en.wikipedia.org/wiki/Ford%E2%80%93Fulkerson_algorithm)
- [5] <https://www.geeksforgeeks.org/ford-fulkerson-algorithm-for-maximum-flow-problem/>
- [6] <https://brilliant.org/wiki/ford-fulkerson-algorithm/>