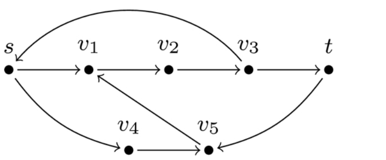
Informatyczny kącik olimpijski (73): Trasowanie



Rys. 1. Optymalne rozwiązanie dla powyższego grafu to $p = (s, v_1, v_2, v_3, t)$ i $q = (t, v_5, v_1, v_2, v_3, s)$. Ścieżki przechodzą łącznie przez 6 wierzchołków.

Tym razem zajmiemy się zadaniem *Routing* z finałów Akademickich Mistrzostw Świata w Programowaniu Zespołowym z 2006 roku. Po przetłumaczeniu historyjki o sieci komputerowej na język teorii grafów brzmi ono następująco. Dany jest skierowany graf $G = (V, E)$, mający n wierzchołków, z których wyróżniamy początkowy s i końcowy t. Chcemy znaleźć w grafie G dwie skierowane ścieżki p i q (pierwszą z s do t, a drugą z t do s) tak, by łączna liczba wierzchołków, przez które przechodzą te ścieżki, była jak najmniejsza (rys. 1).

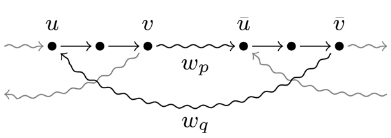
W rozwiązaniu będziemy konstruowali obie ścieżki, poruszając się od wierzchołka s, dla pierwszej z nich zgodnie ze skierowaniem krawędzi, a dla drugiej – przeciwnie. Zdefiniujmy w tym celu ważony graf skierowany $G_2 = (V \times V, E_2)$, w którym z wierzchołka (u, v) wychodzą krawędzie:

Dla zdania logicznego P nawias Iversona $[P]$ jest zdefiniowany następująco: $[P] = 1$, jeżeli zdanie P jest prawdziwe, $[P] = 0$ jeśli jest fałszywe.

(1) do $(\bar{u}, v)$ o wadze $[\bar{u} \neq v]$, jeśli $(u, \bar{u}) \in E$,
(2) do $(u, \bar{v})$ o wadze $[\bar{v} \neq u]$, jeśli $(\bar{v}, v) \in E$,
(3) do (v, u) o wadze $l - 1$, jeśli w G istnieje skierowana ścieżka z u do v i najkrótsza taka ścieżka zawiera l krawędzi.

Krawędzie (1) i (2) w G_2 odpowiadają przesuwaniu się ścieżką p w G zgodnie ze skierowaniem krawędzi i ścieżką q przeciwnie do skierowania krawędzi. Natomiast krawędź (3) w G_2 odpowiada przesunięciu się obiema ścieżkami po tych samych krawędziach w G. Łatwo się więc przekonać, że jeżeli w G_2 istnieje ścieżka z (u, v) do $(\bar{u}, \bar{v})$ o wadze w, to oznacza, że w G istnieją dwie ścieżki, pierwsza z u do $\bar{u}$ i druga z $\bar{v}$ do v, takie że sumaryczna liczba wierzchołków, przez jakie przechodzą (nie licząc u i v), jest równa co najwyżej w. W szczególności, ścieżka w G_2 z (s, s) do (t, t) o wadze $w - 1$ implikuje istnienie rozwiązania zadania o koszcie w.

Pozostaje nam wykazać, że jeśli w optymalnym rozwiązaniu ścieżki p i q przechodzą przez w wierzchołków, to w grafie G_2 istnieje ścieżka z (s, s) do (t, t) o wadze $w - 1$. Niech $A = (s = a_0, a_1, a_2, \ldots, a_r, a_{r+1} = t)$ będzie listą wierzchołków wspólnych dla obu ścieżek w kolejności, w jakiej te wierzchołki występują na ścieżce p. Powiemy, że wierzchołki $a_i, a_{i+1}, \ldots, a_j$ tworzą blok, jeśli są one odwiedzane w tej kolejności również przez ścieżkę q (oznaczymy to przez $\langle a_i, a_j \rangle$). Zauważmy, że z optymalności ścieżek p i q wynika, że na każdej ze ścieżek między wierzchołkami należącymi do tego samego bloku nie występują wierzchołki spoza A. Podobny argument dowodzi tego, że jeśli ścieżka q odwiedza wierzchołek a_j jako najbliższy wierzchołek z A po wierzchołku a_i, to $j \leqslant i + 1$. Jeśli więc A podzielimy na maksymalne bloki w kolejności ich odwiedzania przez ścieżkę p, to ścieżka q będzie je odwiedzać w odwrotnej kolejności. Ponadto pierwszy i ostatni blok podziału będą to odpowiednio $\langle s, s \rangle$ i $\langle t, t \rangle$.



Rys. 2

Pokażemy (rys. 2), że dla kolejnych dwóch bloków $\langle u, v \rangle$ i $\langle \bar{u}, \bar{v} \rangle$ tego podziału istnieje ścieżka w grafie G_2 z (v, u) do $(\bar{v}, \bar{u})$ o wadze $w_p + w_q + l$, gdzie w_p, w_q to odpowiednio liczby wierzchołków spoza A odwiedzanych przez ścieżkę p od v do $\bar{u}$ oraz przez ścieżkę q od $\bar{v}$ do u, natomiast l to rozmiar bloku $\langle \bar{u}, \bar{v} \rangle$. Istotnie, z (v, u) do $(\bar{u}, u)$ przechodzimy $w_p + 1$ krawędziami (1), następnie z $(\bar{u}, u)$ do $(\bar{u}, \bar{v})$ przechodzimy $w_q + 1$ krawędziami (2) o łącznej wadze $w_q + [\bar{u} \neq \bar{v}]$ i ostatecznie (jeśli $\bar{u} \neq \bar{v}$) używamy krawędzi (3) o wadze $l - 2$, aby dostać się z $(\bar{u}, \bar{v})$ do $(\bar{v}, \bar{u})$. W ten sposób konstruujemy ścieżkę w G_2, która odwiedza wszystkie bloki i ma wagę równą $w - 1$.

Tak więc dowiedliśmy, że aby znaleźć rozwiązanie, wystarczy, że wyznaczymy najlżejszą ścieżkę z (s, s) do (t, t) w grafie G_2. Możemy to zrobić algorytmem Dijkstry, na bieżąco konstruując graf G_2. Za każdym razem, gdy z kolejki priorytetowej wyciągamy wierzchołek (u, v), przechodzimy po liście sąsiedztwa u w G i relaksujemy krawędzie (1), następnie przechodzimy po liście sąsiedztwa v w grafie transponowanym G^T i relaksujemy krawędzie (2); w końcu sprawdzamy, czy $u \neq v$ i jeśli tak, relaksujemy krawędź (3).

Graf G_2 ma $O(n^2)$ wierzchołków. Każdy z nich jest początkiem $O(n)$ krawędzi (1) i (2) i co najwyżej jednej krawędzi (3), zatem w G_2 mamy $O(n^3)$ krawędzi. Ponadto uaktualnienie wagi wierzchołka podczas relaksacji będzie wykonywane tylko $O(n^2)$ razy, gdyż krawędzie wchodzące do każdego wierzchołka w G_2 mają co najwyżej trzy różne wagi. Zatem złożoność czasowa rozwiązania wyniesie $O(n^3)$. Tyle też będzie trwało obliczenie długości najkrótszych ścieżek pomiędzy wszystkimi parami wierzchołków w G, jeśli użyjemy do tego algorytmu Floyda–Warshalla.

Tomasz IDZIASZEK