Przedmiot:	Matematyczne narzędzia komputerowe w zastosowaniach telekomunikacyjnych
Prowadzący:	Piotr Chołda piotr.cholda@agh.edu.pl
Kierunek:	Elektronika i Telekomunikacja
Specjalność:	Sieci i usługi
Semestr:	I sem. (letni) studiów magisterskich

Konspekt

Piotr Chołda

3 kwietnia 2018

1 Poszukiwanie najkrótszej ścieżki w grafie

1.1 Poszukiwanie najkrótszej ścieżki w grafie jako problem optymalizacji kombinatorycznej

1. Problem poszukiwania najkrótszej ścieżki w grafie.

2. Algorytm Dijkstry (algorytm etykietowania/cechowania wierzchołków), założenia związane ze stosowalnością tego algorytmu, złożoność algorytmu $\mathcal{O}\left(|V|^2\right)$, wersja algorytmu poszukiwania drzewa najkrótszych ścieżek:

procedure DIJKSTRA($G = (V, A, d)$, $r \in V$)
 ▷ r: korzeń
  ▷ Dla wierzchołków, które nie są sąsiednie, przyjmujemy $d_{ij} = \infty$: $(i, j) \notin A \Rightarrow d_{ij} = \infty$
 ▷ Inicjalizacja:
  $\mathcal{S} \leftarrow \{r\}$
  $predecessor(r) = 0$
  ▷ $\mathcal{S}$: zbiór ocechowanych wierzchołków (dla których znaleziono najkrótszą ścieżkę z wierzchołka r)
  $\mathcal{S}' \leftarrow V \setminus \{r\}$
 ▷ $\mathcal{S}'$: zbiór nieocechowanych wierzchołków
  for all $j \in \mathcal{S}'$ **do**
   $D_{rj} \leftarrow d_{rj}$
   if $D_{rj} < \infty$ **then**
    $predecessor(j) = r$
   end if
  end for
 ▷ Pętla główna:
  while $\mathcal{S}' \neq \emptyset$ **do**
   $k \leftarrow \arg\min_{m \in \mathcal{S}'}\{D_{rm}\}$
   $\mathcal{S} \leftarrow \mathcal{S} \cup \{k\}$
   $\mathcal{S}' \leftarrow \mathcal{S}' \setminus \{k\}$
   ▷ Sprawdzić polepszenie dotychczasowej najkrótszej ścieżki:
   ▷ $\mathcal{N}_k^-$ jest zbiorem następników k (pękiem wyjściowym dla k, *forward star*), $\mathcal{N}_k^- = \{j \in V : (k, j) \in A\}$
   for all $j \in \mathcal{N}_k^- \cap \mathcal{S}'$ **do**
    if $D_{rk} + d_{kj} < D_{rj}$ **then**
     $D_{rj} \leftarrow D_{rk} + d_{kj}$

            predecessor$(j) = k$
        end if
    end for
  end while
end procedure

3. Zmodyfikowany algorytm Dijkstry (dla sieci z ujemnymi wagami łączy, ale bez cykli negatywnych). Pojęcie cyklu negatywnego.

procedure DijkstraNegative(i, $G = (V, A, d)$)
▷ i: korzeń
▷ Nie mogą istnieć w grafie cykle negatywne
▷ Inicjalizacja:
  $\mathcal{S} \leftarrow \{i\}$
    ▷ $\mathcal{S}$: zbiór ocechowanych wierzchołków (dla których znaleziono najkrótszą ścieżkę z wierzchołka i)
  $\mathcal{S}' \leftarrow V \setminus \{i\}$
▷ $\mathcal{S}'$: zbiór nieocechowanych wierzchołków
  for all $j \in \mathcal{S}'$ **do**
    $D_{ij} \leftarrow d_{ij}$
    ▷ Dla wierzchołków, które nie są sąsiednie, przyjmujemy $d_{ij} = \infty$, $(i, j) \notin A \Rightarrow d_{ij} = \infty$
  end for
▷ Pętla główna:
  while $\mathcal{S}' \neq \emptyset$ **do**
    Znaleźć $k = \arg\min_{m \in \mathcal{S}'} \{D_{im}\}$
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{k\}$
    predecessor$(k) = i$
    $\mathcal{S}' \leftarrow \mathcal{S}' \setminus \{k\}$
    ▷ $\mathcal{N}_k^-$ jest zbiorem następników k (pękiem wyjściowym dla k, *forward star*), $\mathcal{N}_k^- = \{j : (k, j) \in A\}$
    for all $j \in \mathcal{N}_k^-$ **do**
      if $D_{ij} > D_{ik} + d_{kj}$ **then**
        $D_{ij} \leftarrow D_{ik} + d_{kj}$
        $\mathcal{S} \leftarrow \mathcal{S} \setminus \{i\}$
        $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{i\}$
      end if
      for all k : *predecessor*$(k) \in \mathcal{S}'$ **do**
        $\mathcal{S} \leftarrow \mathcal{S} \setminus \{k\}$
        $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{k\}$
      end for
    end for
  end while
end procedure

4. Algorytm Bhandari'ego poszukiwania najkrótszej pary ścieżek rozłącznych.

1.2 Zadania

- Proszę podać przykład takiego grafu ważonego z wyróżnionym wierzchołkiem r, w którym: (a) drzewo najkrótszych ścieżek o korzeniu r oraz (b) najkrótsze drzewo rozpinające uzyskane za pomocą algorytmu Prima (przy starcie z wierzchołka r) nie są ze sobą tożsame.

1.3 Lektury

1.3.1 Materiał wykładu

Zagadnienia omówione w ramach tego wykładu są w dużym stopniu opisane w następujących książkach:

- Wayne D. Grover. *Mesh-Based Survivable Networks. Options and Strategies for Optical, MPLS, SONET, and ATM Networks.* Prentice Hall PTR, Upper Saddle River, NJ, 2004: section 4.10.
- Deepankar Medhi and Karthikeyan Ramasamy. *Network Routing. Algorithms, Protocols, and Architectures.* Morgan Kaufmann Publishers—Elsevier, San Francisco, CA, 2007: chapter 2.
- Michał Pióro and Deepankar Medhi. *Routing, Flow and Capacity Design in Communication and Computer Networks.* Morgan Kaufmann Publishers—Elsevier, San Francisco, CA, 2004: appendix C.1-C.2.
- Maciej M. Sysło, Narsingh Deo, and Janusz S. Kowalik. *Algorytmy optymalizacji dyskretnej.* Wydawnictwo Naukowe PWN, Warszawa, 1999: rozdział 3.3.
- Robin J. Wilson. *Wprowadzenie do teorii grafów.* Wydawnictwo Naukowe PWN, Warszawa, 2000: § 8.

1.3.2 Bibliografia uzupełniająca

- Ramesh Bhandari. *Survivable Networks. Algorithms for Diverse Routing.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999: przegląd różnych algorytmów przydatnych w projektowaniu sieci (niezawodnych).
- Maciej M. Sysło, Narsingh Deo, and Janusz S. Kowalik. *Algorytmy optymalizacji dyskretnej.* Wydawnictwo Naukowe PWN, Warszawa, 1999: podstawy teoretyczne do naszego kursu.
- Leonardo Taccari. Integer Programming Formulations for the Elementary Shortest Path Problem. *European Journal of Operational Research*, 252(1):122–130, July 1, 2016: problemy maksymalnego przepływu.
- Robin J. Wilson. *Wprowadzenie do teorii grafów.* Wydawnictwo Naukowe PWN, Warszawa, 2000: zwięzłe wprowadzenie do teorii grafów, trochę algorytmów.