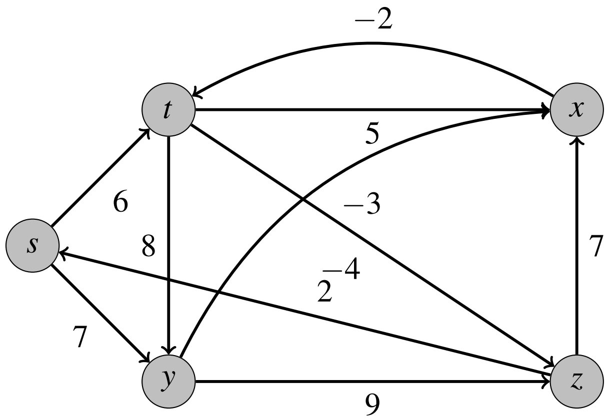
GRAFOVÉ ALGORITMY - CVIČENÍ 8: NEJKRATŠÍ CESTY Z JEDNOHO UZLU DO VŠECH UZLŮ

Převzato z Cormen, Leiserson, Rivest, Stein: Introduction to algorithms. The MIT Press and McGraw-Hill, 2001; kapitola 24.

Bellman-Fordův algoritmus

Příklad 1. Na orientovaný graf na obrázku 1 aplikujte algoritmus Bellman-Ford se zdrojovým uzlem z. Hrany relaxujte v pořadí (t,x), (t,y), (t,z), (x,t), (y,x), (y,z), (z,x), (z,s), (s,t), (s,y). Ukažte průběžné (po každé iteraci) hodnoty d a π. Dále změňte váhu hrany (z,x) na 4 a algoritmus znova aplikujte tentokrát se zdrojovým uzlem s.



Obrázek 1: Příklad 1

Příklad 2. Dokažte následující důsledek důkazu korektnosti Bellman-Fordova algoritmu: Nechť $G=(V,E)$ je ohodnocený orientovaný graf se zdrojovým uzlem s a váhovou funkcí $w\colon E \to \mathbb{R}$. Pak pro každý uzel $v \in V$ existuje cesta z s do v, právě když Bellman-Fordův algoritmus při aplikaci na graf G skončí s nerovností $d[v] < \infty$.

Příklad 3. Mějme zadaný ohodnocený orientovaný graf $G=(V,E)$ bez záporných cyklů (cyklů se záporným ohodnocením). Nechť m je maximum z nejkratších cest (podle počtu hran na cestě) mezi všemi dvojicemi uzlů u, v. Navrhněte jednoduchou změnu Bellman-Fordova algoritmu pro výpočet nejkratších cest (podle váhy cesty) během $m+1$ průchodů. Zdůvodněte korektnost vaší úpravy algoritmu.

Příklad 4. Modifikujte Bellman-Fordův algoritmus, aby nastavoval $d[v]$ na $-\infty$ pro všechny uzly v, pro které existuje záporný cyklus na libovolné cestě ze zdrojového uzlu do v. Zdůvodněte funkčnost algoritmu.

Příklad 5. Mějme ohodnocený orientovaný graf $G = (V, E)$ s váhovou funkcí $w\colon E \to \mathbb{R}$. Vymyslete algoritmus s časovou složitostí $O(VE)$, který pro každý uzel $v \in V$ nalezne hodnotu $\delta^*(v) = \min_{u \in V}\{\delta(u, v)\}$.

Příklad 6. Mějme ohodnocený orientovaný graf $G = (V, E)$ s nějakým záporným cyklem. Navrhněte efektivní algoritmus pro výpis uzlů nějakého cyklu. Dokažte korektnost vytvořeného algoritmu.

Nejkratší cesty v acyklických orientovaných grafech

Příklad 7. Vyzkoušejte si spustit DAG-SHORTEST-PATHS algoritmus nad grafem z přednášek s jiným zdrojovým uzlem (např. uzel r).

Příklad 8. Předpokládejte algoritmus DAG-SHORTEST-PATHS se změněným třetím řádkem následovně:

3: **for** prvních $|V| - 1$ uzlů, brané podle topologického uspořádání, značené u

Ukažte, že i po této změně zůstal algoritmus korektní.

Příklad 9. Uvažujme graf reprezentující úkoly pomocí uzlů a jejich sekvenční časové závislosti prostřednictvím hran. Konkrétněji, hrana (u, v) indikuje, že úkol u musí být hotov před úkolem v. Váhy jsou přiřazeny jednotlivým uzlům (nikoli hranám). Modifikujte algoritmus DAG-SHORTEST-PATHS tak, aby našel nejdelší cestu v orientovaném acyklickém grafu s ohodnocenými uzly v lineárním čase.

Příklad 10. Navrhněte efektivní algoritmus pro zjištění celkového počtu cest v orientovaném acyklickém grafu. Zdůvodněte jeho korektnost a odvoďte jeho časovou složitost.

Dijkstrův algoritmus

Příklad 11. Aplikujte Dijkstrův algoritmus na orientovaný graf z obrázku 2. Nejprve jako zdrojový uzel použijte uzel s a potom uzel z. Podobně jako v přednáškách naznačte hodnoty d, π a obsah množiny S po každé iteraci algoritmu.

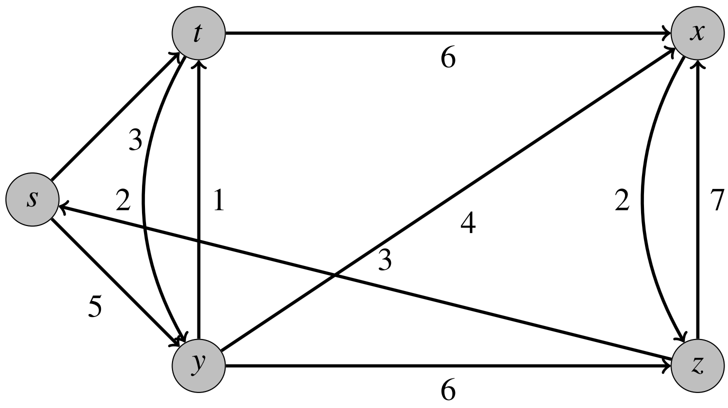
Příklad 12. Vymyslete příklad jednoduchého orientovaného grafu se zápornými hranami, pro který Dijkstrův algoritmus nefunguje. Zdůvodněte, proč neplatí věta o korektnosti algoritmu při povolení záporného ohodnocení hran.

Příklad 13. Předpokládejte algoritmus DIJKSTRA se změněným čtvrtým řádkem následovně:

4: **while** $|Q| > 1$

Tato změna způsobí, že **while**-cyklus provede $|V| - 1$ iterací místo $|V|$ iterací. Zůstal algoritmus i po této změně korektní?

Příklad 14. Mějme dán orientovaný graf $G = (V, E)$, který má každé hraně $(u, v) \in E$ asociovánu hodnotu $r(u, v)$, což je reálné číslo z intervalu $0 \le r(u, v) \le 1$, které reprezentuje



Obrázek 2: Příklad 11

spolehlivost komunikačního kanálu mezi uzly u a v. Hodnota $r(u,v)$ vyjadřuje pravděpodobnost, že kanál z u do v neselže. Předpokládejme nezávislost těchto pravděpodobností. Navrhněte efektivní algoritmus, který nalezne nejspolehlivější cestu mezi dvěma zadanými uzly.

Příklad 15. Nechť $G = (V,E)$ je ohodnocený orientovaný graf s váhovou funkcí $w\colon E \to \{1,2,\ldots,W\}$, kde W je kladné celé číslo. Předpokládejme, že žádné dva uzly nemají stejnou váhu nejkratší cesty ze zdrojového uzlu s. Nyní předpokládejme, že jsme definovali (neohodnocený) orientovaný graf $G' = (V \cup V', E')$ tak, že každá hrana $(u,v) \in E$ s váhou $w(u,v)$ je nahrazena sérií $w(u,v)$ jednotkových hran v E'. Kolik uzlů má G'? Dále předpokládejme, že na G' aplikujeme prohledávání do šířky (BFS). Ukažte, že pořadí obarvení uzlů z V na černo při BFS grafu G' je totožné s pořadím, ve kterém jsou uzly z V vybírány z prioritní fronty na řádku 5 při běhu algoritmu DIJKSTRA na G.

Příklad 16. Nechť $G = (V,E)$ je ohodnocený orientovaný graf s váhovou funkcí $w\colon E \to \{0,1,\ldots,W\}$, kde W je nezáporné celé číslo. Modifikujte algoritmus DIJKSTRA tak, aby počítal nejkratší cesty ze zadaného zdrojového uzlu s v čase $O(WV + E)$.

Příklad 17. Modifikujte algoritmus z předchozího příkladu tak, aby pracoval v čase $O((V+E)\, lgW)$. (Nápověda: Kolik různých odhadů nejkratších cest může být v množině $V - S$ během výpočtu?)

Příklad 18. Předpokládejme, že máme zadán ohodnocený orientovaný graf $G = (V,E)$, ve kterém mohou být hrany vycházející ze zdrojového uzlu s ohodnoceny záporně, ale všechny ostatní hrany jsou ohodnoceny nezáporně a G neobsahuje žádné záporné cykly. Promyslete zachování korektnosti Dijkstrova algoritmu pro nalezení nejkratších cest z uzlu s v takovémto grafu.