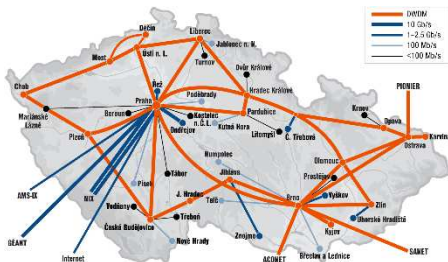


9 Orientované grafy, Toky v sítích

Nyní se budeme zabývat typem **síťových úloh**, ve kterých není podstatná délka hran a spojení, nýbž jejich **propustnost** (jako potrubní nebo počítačové sítě).

Základní úlohou v této oblasti je problém nalezení **maximálního toku** v síti za podmínky respektování daných kapacit hran.

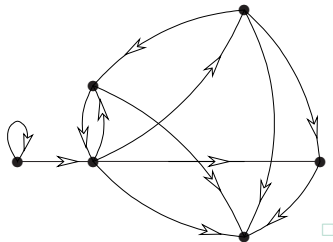


Stručný přehled lekce

- * Definice a některé základní vlastnosti orientovaných grafů, souvislost.
- * Síť s kapacitami hran, hledání maximálního toku a dualita.
- * Důsledky duality toku; vyšší souvislost, bipartitní párování, SRR.

9.1 Základní pojmy orientovaných grafů

Požadavek explicitně vyjádřit **směr hrany** přirozeně vede na následující definici orientovaného grafu, ve kterém hrany jsou **uspořádané** dvojice vrcholů.

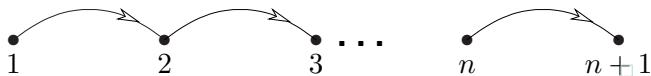


Definice 9.1. Orientovaný graf je uspoř. dvojice $D = (V, E)$, kde $E \subseteq V \times V$. Pojmy podgrafu a isomorfismu se přirozeně přenášejí na orientované grafy. \square

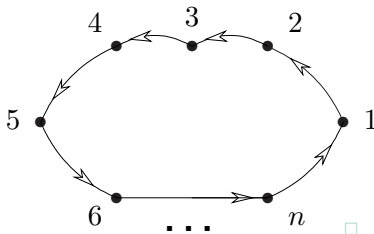
Značení: Hrana (u, v) (zvaná také **šipka**) v orientovaném grafu D **začíná** ve vrcholu u a **končí** ve (míří do) vrcholu v . Opačná hrana (v, u) je různá od (u, v) . Speciálně hrana tvaru (u, u) se nazývá **orientovaná smyčka**. \square

Orientované grafy odpovídají relacím, které nemusí být symetrické.

- *Orientovaná cesta* délky $n \geq 0$ je následujícím grafem na $n + 1$ vrcholech



- a *orientovaná kružnice* (také *cyklus*) délky $n \geq 1$ vypadá takto:



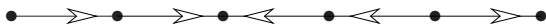
Definice: Počet hran začínajících ve vrcholu u orientovaného grafu D nazveme *výstupním stupněm* $d_D^+(u)$ a počet hran končících v u nazveme *vstupním stupněm* $d_D^-(u)$.

Součet všech výstupních stupňů je přirozeně roven součtu všech vstupních stupňů.

Souvislost na orientovaných grafech

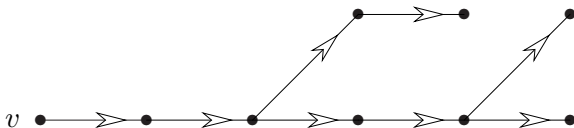
Uvedeme si odstupňovaně tři základní pohledy:

- **Slabá souvislost.** Jedná se o tradiční *souvislost na symetrizaci* grafu D (tj. po „zapomenutí“ směru šipek).

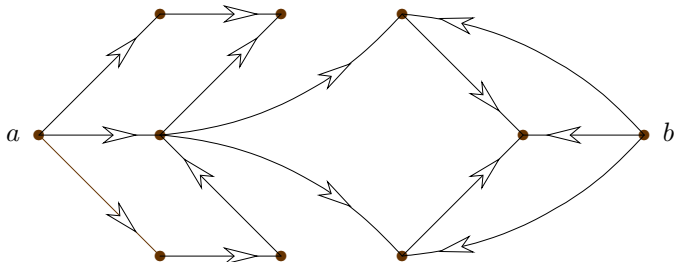


□

- **Dosažitelnost (směrem „ven“).** Orientovaný graf D je *dosažitelný směrem ven*, pokud v něm existuje vrchol $v \in V(D)$ takový, že každý vrchol $x \in V(D)$ je dosažitelný orientovaným sledem z v .

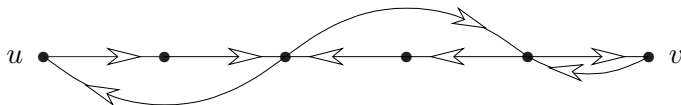


Podrobným zkoumáním následujícího obrázku zjistíme, že tento graf není dosažitelný směrem ven, neboť chybí možnost dosáhnout vrchol b úplně vpravo. Na druhou stranu po vypuštění b je zbylý graf dosažitelný ven z vrcholu a vlevo.



Souvislost na orientovaných grafech, silná

- **Silná souvislost.** Necht' \approx je binární relace na vrcholové množině $V(D)$ orientovaného grafu D taková, že
 - * $u \approx v$ právě když existuje dvojice orientovaných cest – jedna z u do v a druhá z v do u v grafu D . \squarePak \approx je **relace ekvivalence**.

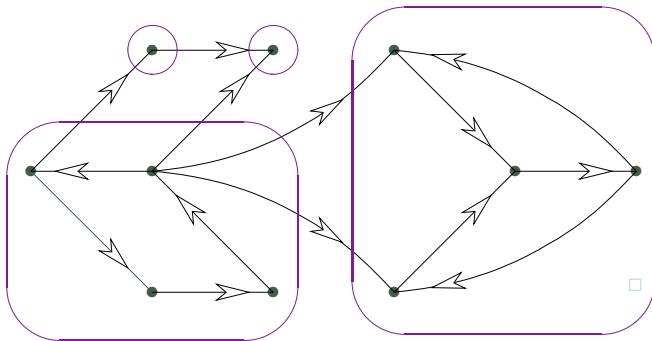


\square

Definice 9.2. Silné komponenty orientovaného grafu D jsou třídy ekvivalence relace \approx uvedené v předchozím. \square

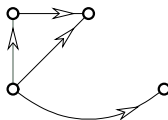
Orientovaný graf D je **silně souvislý** pokud má nejvýše jednu silnou komponentu.

Pro ilustraci si mírně upravíme dříve prezentovaný orientovaný graf tak, že bude dosažitelný z nejlevějšího vrcholu. Je výsledek silně souvislý?



Ne, na obrázku jsou vyznačené jeho 4 silné komponenty.

Zároveň uvádíme pro ilustraci obrázek **kondenzace** silných komponent tohoto grafu, což je acyklický orientovaný graf s vrcholy reprezentujícími zmíněné silné komponenty a směry hran mezi nimi.

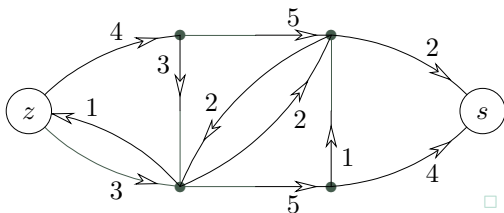


9.2 Definice sítě a toku

Základní strukturou pro reprezentaci sítí je **vážený orientovaný graf** (příčemž implicitní směr hran je v tomto kontextu nezbytný). □

Definice 9.3. **Síť** je čtveřice $S = (G, z, s, w)$, kde

- * G je orientovaný graf,
- * vrcholy $z \in V(G)$, $s \in V(G)$ jsou **zdroj** a **stok**,
- * $w : E(G) \rightarrow \mathbb{R}^+$ je kladné ohodnocení hran, zvané **kapacita hran**.



Poznámka: V praxi může být zdrojů a stoků více, ale v definici stačí pouze jeden zdroj a stok, z něhož / do nějž vedou hrany do ostatních zdrojů / stoků. (Dokonce pak různé zdroje a stoky mohou mít své kapacity.)

Velikost toku v síti

Značení: Pro jednoduchost píšeme ve výrazech značku $e \rightarrow v$ pro hranu e přicházející do vrcholu v a $e \leftarrow v$ pro hranu e vycházející z v . \square

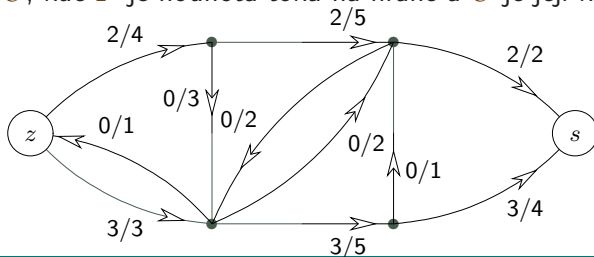
Definice 9.4. Tok v síti $S = (G, z, s, w)$ je funkce $f : E(G) \rightarrow \mathbb{R}_0^+$ splňující

- * $\forall e \in E(G) : 0 \leq f(e) \leq w(e)$, (respektování kapacity)
- * $\forall v \in V(G), v \neq z, s : \sum_{e \rightarrow v} f(e) = \sum_{e \leftarrow v} f(e)$. (zachování substance)

\square

Velikost toku f je dána výrazem $\|f\| = \sum_{e \leftarrow z} f(e) - \sum_{e \rightarrow z} f(e)$. \square

Značení: Tok a kapacitu hran v obrázku síť budeme zjednodušeně zapisovat ve formátu F/C , kde F je hodnota toku na hraně a C je její kapacita.

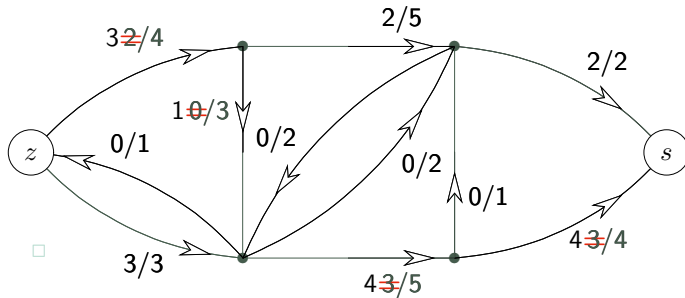


9.3 Nalezení maximálního toku

Naším úkolem je najít co největší tok v dané síti. Pro jeho nalezení existují jednoduché a velmi rychlé algoritmy.

Definice 9.5. Úloha hledání maximálního toku v síti $S = (G, z, s, w)$.
Úkolem je v síti S najít tok f ze zdroje z do stoku s podle Definice 9.4 takový, který maximalizuje velikost $\|f\|$. \square

Tok velikosti 5 uvedený v ukázce v předchozí části nebyl optimální, neboť v této síti najdeme i tok velikosti 6:

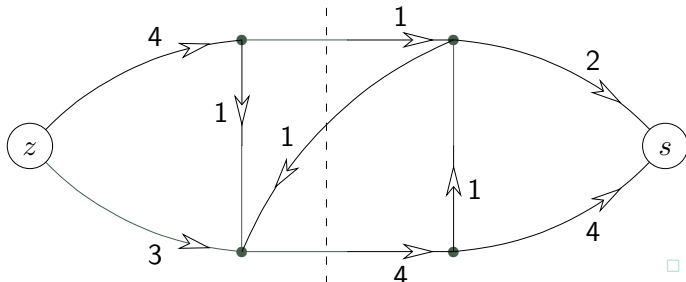


Jak však poznáme, že větší tok již v dané síti neexistuje?

Pojem řezu v síti

Definice 9.6. Řez v síti $S = (G, z, s, w)$ je podmnožina hran $C \subset E(G)$ taková, že v podgrafu $G - C$ (tj. po odebrání hran C z G) nezbude žádná orientovaná cesta ze z do s .

Velikost řezu C rozumíme součet kapacit hran z C , tj. $\|C\| = \sum_{e \in C} w(e)$.



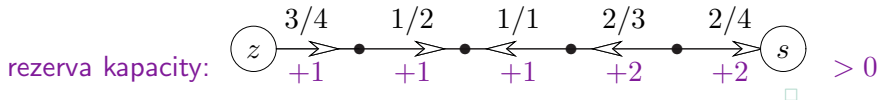
Věta 9.7. Maximální velikost toku v síti je rovna minimální velikosti řezu. □

V uvedeném obrázku nalezneme tok velikosti 5. Proč má vyznačený řez také velikost 5?

Nenasycené cesty v síti

Definice: Mějme síť S a v ní tok f . *Nenasycená cesta* (v S vzhledem k f)

- * je neorientovaná cesta v G z vrcholu u do vrcholu v (obvykle ze z do s), tj. posloupnost navazujících libovolně orientovaných hran e_1, e_2, \dots, e_m , \square
- * kde $f(e_i) < w(e_i)$ pro e_i ve směru z u do v a $f(e_i) > 0$ pro e_i jinak.

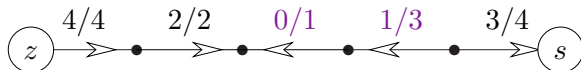
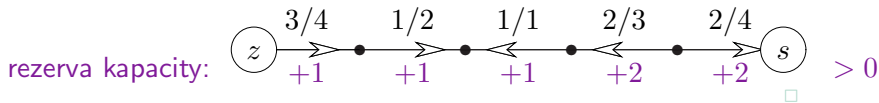


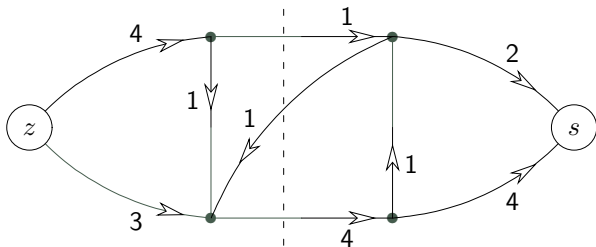
- * Hodnotě $w(e_i) - f(e_i)$ pro hrany e_i ve směru z u do v a hodnotě $f(e_i)$ pro hrany e_i v opačném směru říkáme *rezerva kapacity* hran e_i .

Nenasycená cesta je tudíž cesta s kladnými rezervami kapacit všech hran.

Metoda 9.8. Maximální tok vylepšováním nenasycených cest.

Základní myšlenkou této jednoduché metody hledání maximálního toku v dané síti je prostě opakovaně vylepšovat tok podél nalezených nenasycených cest.





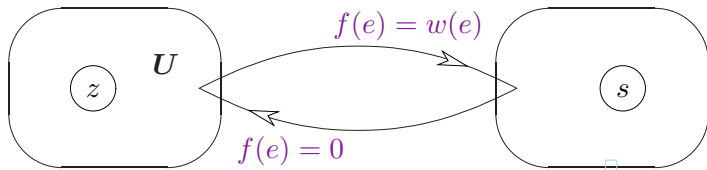
Algoritmus 9.9. Ford–Fulkersonův pro tok v síti.

- **Vstup:** Síť $S = (G, z, s, w)$ podle Definice 9.3.
- Tok $f \leftarrow (0, 0, \dots, 0)$.
- Dále opakujeme následující:
 - * Prohledáváním grafu najdeme množinu U vrcholů G , do kterých se dostaneme ze z po nenasycených cestách.
 - * Pokud $s \in U$, necht' P značí nalezenou nenasycenou cestu v S ze z do s .
 - Zvětšíme tok f o minimální rezervu kapacity hran v P .
- Opakujeme kroky výše, dokud nenastane $s \notin U$.
- **Výstup:** Vypíšeme maximální tok f a také minimální řez jako množinu všech hran vedoucích z U do $V(G) - U$.

Důkaz správnosti Algoritmu 9.9:

Pro každý tok f a každý řez C v síti S platí $\|f\| \leq \|C\|$. Jestliže po zastavení algoritmu s tokem f nalezneme v síti S řez o stejné velikosti $\|C\| = \|f\|$, je jasné, že jsme našli maximální možný tok v síti S . (Pozor, **zastavení** algoritmu jsme zatím nezdůvodnili.) \square

Takže dokažme, že po zastavení algoritmu nastane rovnost $\|f\| = \|C\|$, kde C je vypsáný řez mezi U a zbytkem grafu G . Vezměme tok f v S bez nenasycené cesty ze z do s . Pak množina U z algoritmu neobsahuje s .



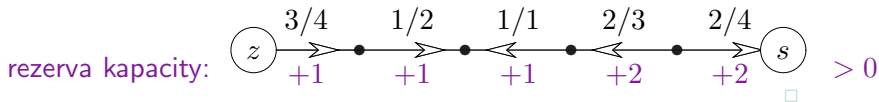
Nyní má každá hrana $e \leftarrow U$ (odch. z U) plný tok $f(e) = w(e)$ a každá hrana $e \rightarrow U$ (přich. do U) tok $f(e) = 0$, takže

$$\|f\| = \sum_{e \leftarrow U} f(e) - \sum_{e \rightarrow U} f(e) = \sum_{e \leftarrow U} f(e) = \sum_{e \in C} w(e) = \|C\| .$$

Důsledky Algoritmu 9.9

Z důkazu Algoritmu 9.9 odvodíme několik zajímavých faktů:

- Pokud Algoritmus 9.9 vždy skončí, dokážeme tím i platnost Věty 9.7. □
- Pro celočíselné kapacity hran sítě S Algoritmus 9.9 vždy skončí.

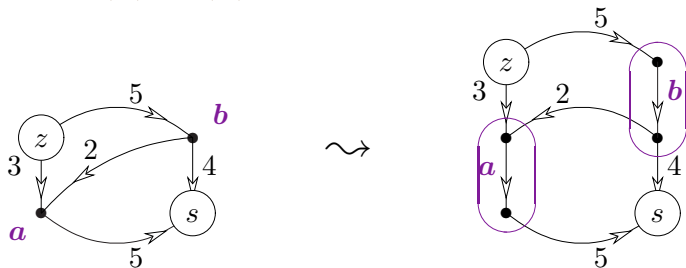


- Pokud jsou kapacity hran sítě S celočíselné, opt. tok také vyjde celočíselně.

9.4 Zobecněné použití sítí

- **Sítě s kapacitami vrcholů:**

U sítě můžeme zadat *kapacity vrcholů*, neboli kapacitní váhová funkce je dána jako $w : E(G) \cup V(G) \rightarrow \mathbb{R}^+$.



□

- **Sítě s dolními kapacitami:**

Pro hrany sítě lze zadat také jejich *minimální kapacity*, tedy dolní meze přípustného toku, jako váhovou funkci $\ell : E(G) \rightarrow \mathbb{R}_0^+$.

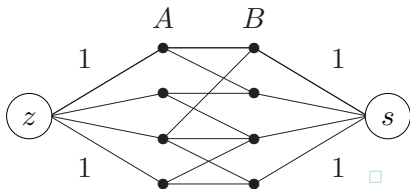
$$\ell(e) \leq f(e) \leq w(e)$$

Bipartitní párování

Definice: *Párování* v (nyní bipartitním) grafu G je podmnožina hran $M \subset E(G)$ taková, že žádné dvě hrany z M nesdílejí koncový vrchol. \square

Metoda 9.11. Nalezení bipartitního párování

Pro daný bipartitní graf G s vrcholy rozdělenými do množin A, B sestrojíme síť S následovně:



- Hrany sítě S orientujeme od zdroje do stoku a přiřadíme jim kapacity 1. \square
- Nyní najdeme (celočíslný) maximální tok v S Algoritmem 9.9. Do párování vložíme ty hrany grafu G , které mají nenulový tok.

Výběr různých reprezentantů

Definice: Necht' M_1, M_2, \dots, M_k jsou neprázdné množiny. *Systemem různých reprezentantů* množin M_1, M_2, \dots, M_k nazýváme takovou posloupnost různých prvků (x_1, x_2, \dots, x_k) , že $x_i \in M_i$ pro $i = 1, 2, \dots, k$. \square

Věta 9.12. (Hall) *Necht' M_1, M_2, \dots, M_k jsou neprázdné množiny. Pro tyto množiny existuje systém různých reprezentantů, právě když platí*

$$\forall J \subset \{1, 2, \dots, k\} : \left| \bigcup_{j \in J} M_j \right| \geq |J|,$$

neboli pokud sjednocení libovolné skupiny z těchto množin má alespoň tolik prvků, kolik množin je sjednoceno.

Důkaz lze podat konstrukcí vhodné sítě podobné té v Algoritmu 9.11: \square

- Speciální vrcholy u a v odpovídají zdroji a stoku,
- další vrcholy reprezentují prvky a množiny naší úlohy, \square
- ostatní hrany přicházející do vrcholu x_j znázorňují všechny z daných množin, které obsahují prvek x_j . \square