Toky v sítích

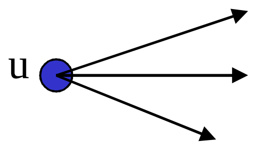
Síť : $\mathbf{S = \langle G, q, s, t \rangle}$ $\mathbf{G} = \langle H, U \rangle$ - orientovaný graf

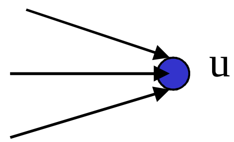
$\mathbf{q} : H \rightarrow R^{+}$ - **kapacita hran,** $q(u,v)$ značíme q_{uv}

s - **zdroj sítě**   **t** - **spotřebič sítě**

Tok v síti S - ohodnocení hran $f : H \rightarrow R^{+}$ splňující

- $\forall (u,v) \in H : 0 \le f(u,v) \le q(u,v)$
- $\sum f(u,v)_{\ (\forall (u,v) \in H)} - \sum f(w,u)_{\ (\forall (w,u) \in H)} = 0$ pro $\forall u \in U,\ u \neq s,t$





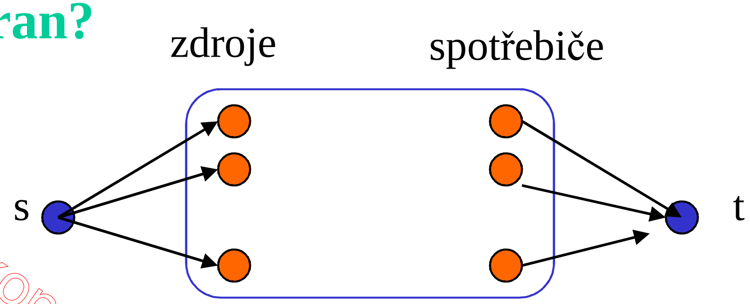
cirkulace ... platí i pro s,t

Velikost toku |f|

$$\mathbf{|f| = \sum f(s,v)_{\ \forall (s,v)} - \sum f(u,s)_{\ \forall (u,s)} = \sum f(u,t)_{\ \forall (u,t)} - \sum f(t,v)_{\ \forall (t,v)}}$$

Základní otázky:

- Jaká je maximální velikost $s \rightarrow t$ toku v síti?
- Jak se určí?
- Jak je rozložen do jednotlivých hran?



Variantní zadání:

- sítě s více zdroji a spotřebiči
- sítě s omezenou kapacitou uzlů

- sítě s omezeným minimálním tokem (řešení ukážeme později)

$$0 \leq r(u,v) \leq f(u,v) \leq q(u,v)$$

- sítě s oceněným tokem

$$c(f) = \sum f(u,v) \,.\, c(u,v) \quad {\scriptstyle (\forall (u,v) \in H)}$$ - cena toku

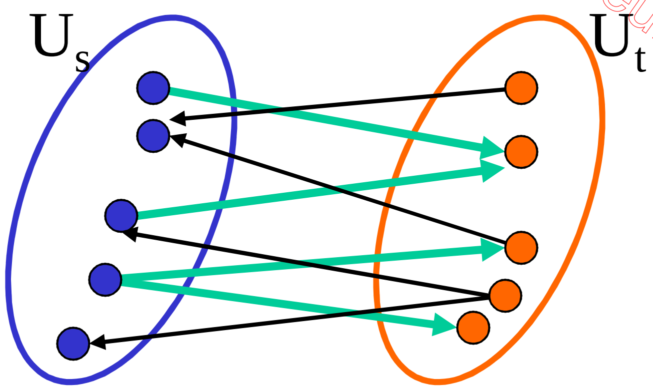
hledá se (přípustná) cirkulace s minimální cenou

Řešení základní úlohy

Řez sítě - hranový řez, který oddělí zdroj a spotřebič

$\{U_s, U_t\}$ - odpovídající rozklad množiny uzlů

$H(U_s \times U_t)$ - hranový řez určený rozkladem uzlů



Kapacita řezu sítě

$$q(U_s \times U_t) = \sum q(u,v)$$

přes hrany (u,v), $u \in U_s$, $v \in U_t$

V: Pro libovolný tok f a řez sítě $H(U_s \times U_t)$ platí

$$|f| \leq q(U_s \times U_t)$$

Velikost (každého) toku tedy nepřekročí kapacitu (žádného) řezu sítě.

Jak poznáme, že daný tok je maximální?

__**V:**__ Tok **f je maximálním** tokem v síti $S=\langle G, q, s, t\rangle$ $\Leftrightarrow$

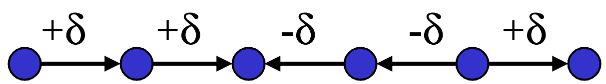
neexistuje (neorientovaná) cesta (tzv. **zlepšující cesta**) $P=\langle u_0,u_1,\ldots,u_n\rangle$, $u_0=s$, $u_n=t$ taková, že platí:

- $f(u_i, u_{i+1}) < q(u_i, u_{i+1})$ pro hrany $(u_i, u_{i+1})\in H$ $(s \rightarrow t)$
- $f(u_{i+1}, u_i) > 0$ pro hrany $(u_{i+1}, u_i)\in H$ $(s \leftarrow t)$

Zvýšení toku podél zlepšující cesty:

$\delta_a = \min(q(u_i, u_{i+1}) - f(u_i, u_{i+1}))$ $\delta_b = \min(f(u_{i+1}, u_i))$

$\delta = \min(\delta_a, \delta_b)$



Algoritmus Ford-Fulkerson

V: (Ford-Fulkerson / max flow - min cut)

Velikost maximálního toku sítě je rovna kapacitě jejího minimálního řezu.

Ford-Fulkerson (S)

```
for každou hranu (u,v)∈H  do f(u,v) := 0
while NajdiCestu(S)
        do ZvyšTok(S)
return f
```

NajdiCestu hledá zlepšující cestu prohledáváním sítě

d[u] průběžně počítané δ, stav[u], p[u] (+ pro $\rightarrow$, - pro $\leftarrow$)

NajdiCestu(S)

```
for každé u∈ U do stav[u]:=FRESH
p[s]:=+s; d[s]:=∞; stav[s]:=OPEN
repeat u:=libovolný otevřený uzel
    stav[u]:=CLOSED
    for každý uzel v∈Γ(U) do
        if (stav[v]=FRESH) and (f(u,v)<q(u,v))
           then stav[v]:=OPEN; p[v]:=+u; d[v]:=min(d[u], q(u,v)-f(u,v))
    for každý uzel v∈Γ⁻¹(U) do
        if (stav[v]=FRESH) and (f(u,v)>0)
          then stav[v]:=OPEN; p[v]:=-u; d[v]:=min(d[u], f(u,v))
until (neexistuje otevřený uzel) or (u = t)
return u = t
```

ZvyšTok(S)

```
x := t;  δ := d[t]
repeat v := x
    if p[v] = +u
        then f(u,v) := f(u,v) + δ
        else f(u,v) := f(u,v) - δ
    x := u
until v=s
```


? Složitost ? NajdiCestu ... **O(|U|+|H|)** ZvyšTok ... **O(|U|)**

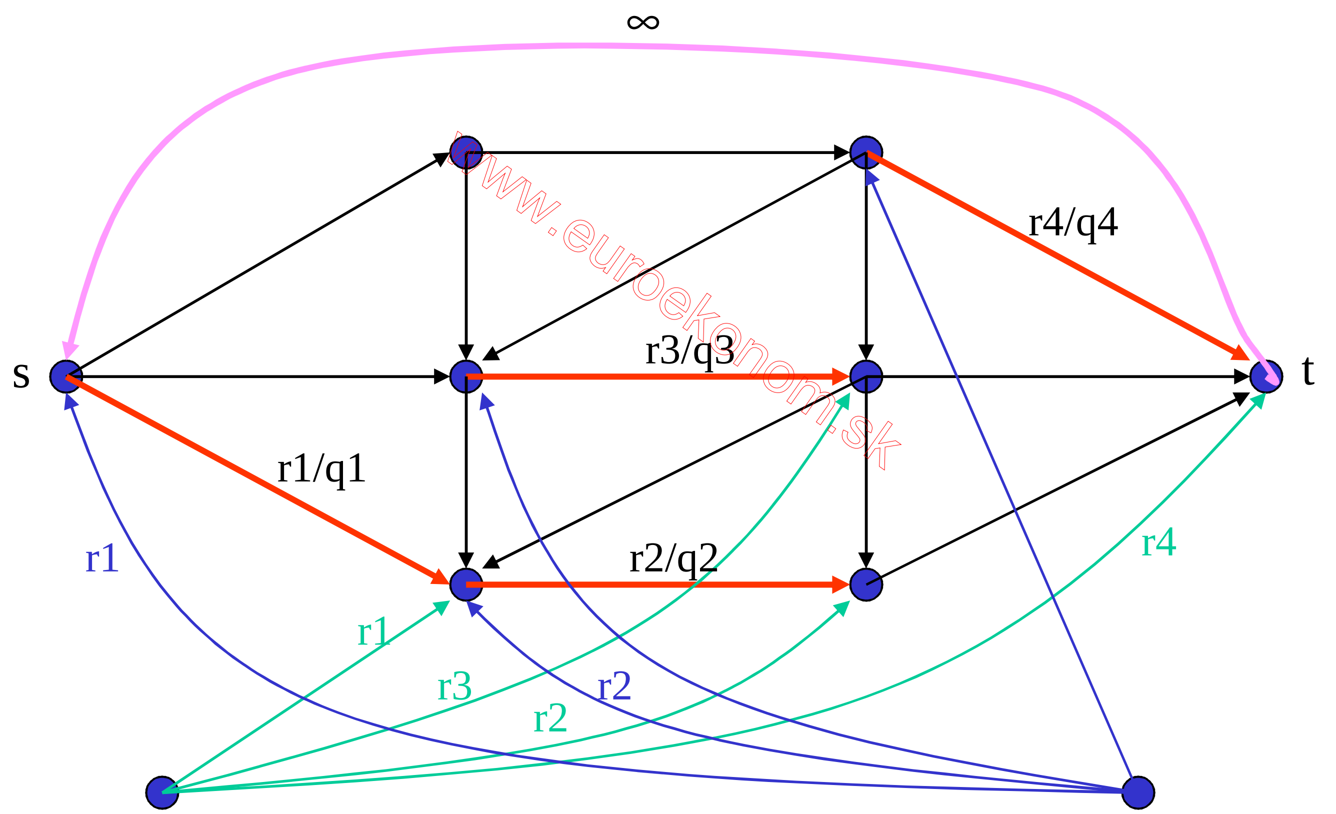
? Celý algoritmus ? O(|U| . |H|2) , O(|U|**2 . |H|), O(|U|**3)**

Další zrychlení pro speciální případy

... až na **O(|U| . lg|U|)** pro planární sítě

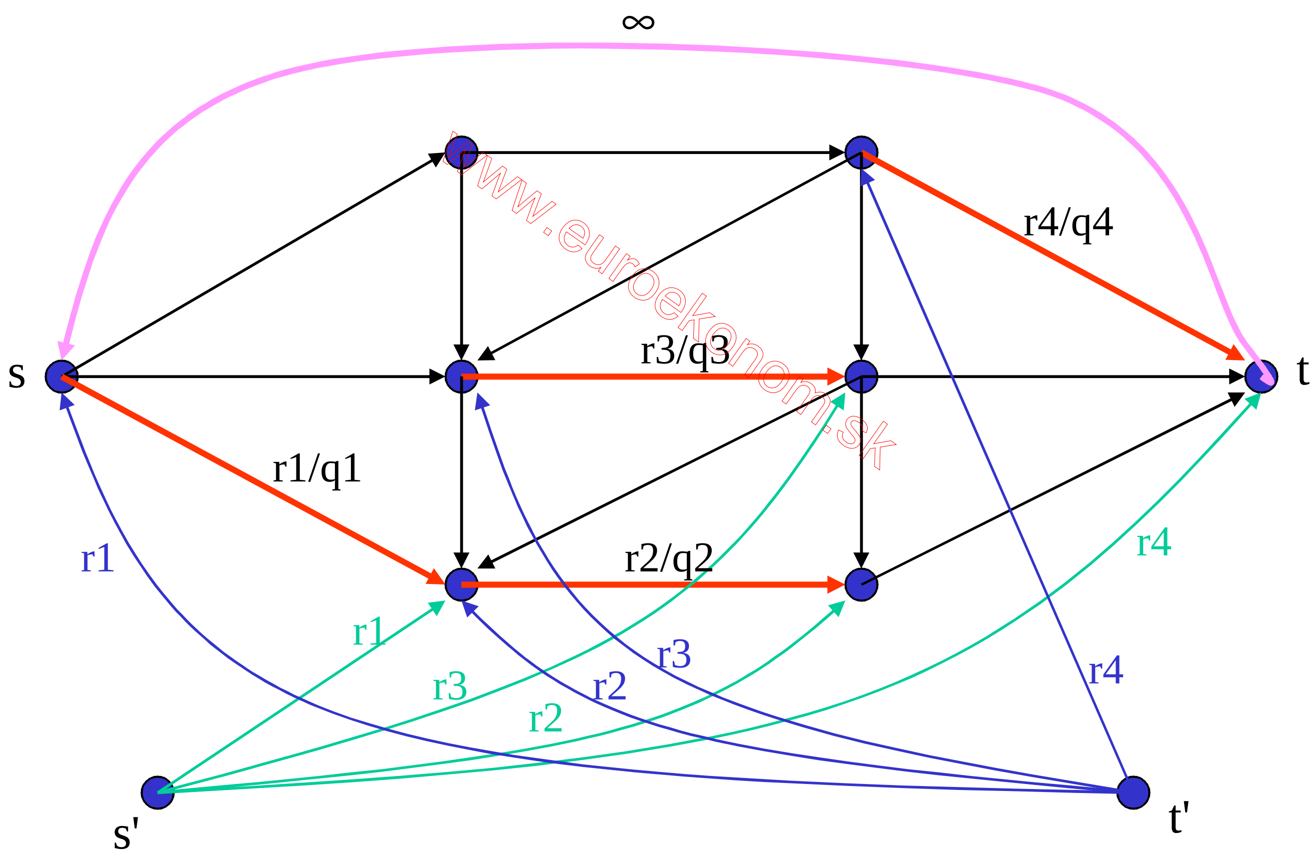
Sítě s omezeným minimálním tokem

Metoda řešení: převod na základní úlohu



Sítě s omezeným minimálním tokem

Metoda řešení: převod na základní úlohu



? Co dál ?

Nalezneme maximální tok s' $\rightarrow$ t'.

?Nasycuje nově přidané hrany?

ANO - máme přípustný tok a zlepšujeme jej standardně podél zlepšujících cest

NE - úloha nemá řešení

Maximální párování

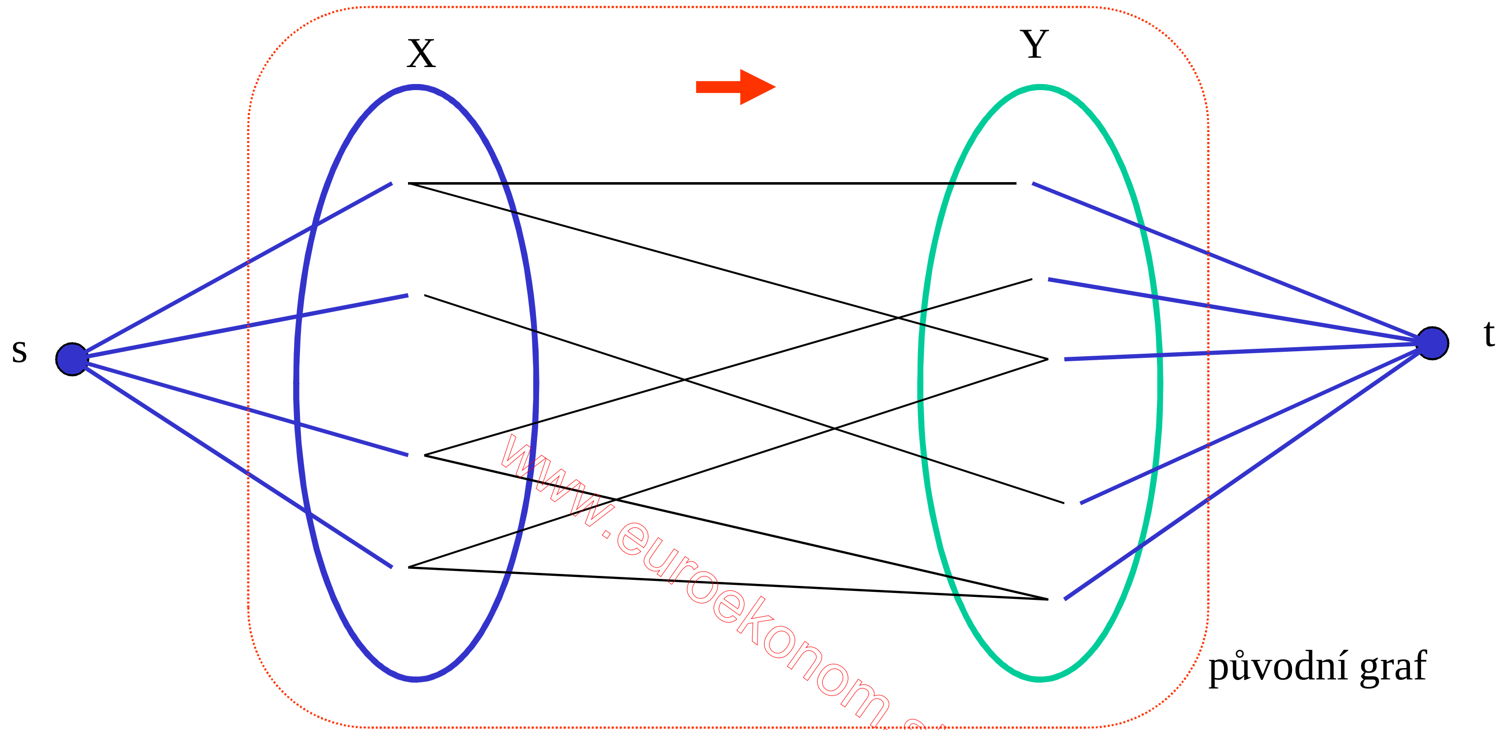
Využití algoritmu max. toků k hledání max. párování

Párování v grafu - "nezávislá" podmnožina hran (žádné dvě nemají společný uzel). **Perfektní párování** pokrývá všechny uzly. Při ohodnocených hranách můžeme hledat nejlevnější nebo nejdražší maximální párování.

Přiřazovací úloha - nejlevnější perfektní párování v úplném bipartitním grafu $K_{n,n}$.

Příklad:

Hledání maximálního párování v neorientovaném bipartitním grafu G s rozkladem uzlů $U = X \cup Y$



Všem hranám přiřadíme kapacitu rovnu 1.

Ve vytvořené síti nalezneme maximální tok pomocí algoritmu Ford-Fulkerson - získáme maximální párování (hrany s nenulovým tokem).