Dynamické grafové algoritmy

- Souvislost, vrcholová a hranová k-souvislost a minimální kostra v neorientovaných grafech.
- Možné způsoby řešení: statické a dynamické algoritmy.
- Hledáme efektivní dynamický algoritmus pro hranovou 2-souvislost.
- Dřívější výsledky:

  $O(\sqrt[3]{n} \cdot \log n)$ pro souvislost (Rauch/King),
  $O(\sqrt{n})$ pro hranovou 2-souvislost (Frederickson),
  $O(PolyLog(n))$ randomizovaně.
- Náš cíl: Polylogaritmické deterministické řešení.
- Potřebujeme:

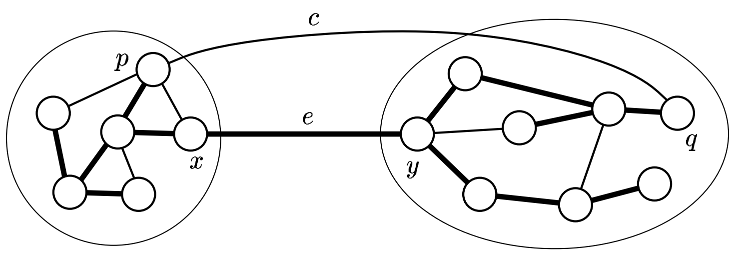
  $\Rightarrow$ zlepšit i algoritmy pro souvislost

  $\Rightarrow$ nalézt vhodné „podkladové" struktury

Algoritmy stromové a semidynamické

- Idea: zvlášť udržujeme kostru a nestromové hrany. Při operacích preferujeme práci s kostrou.
- Dynamické udržování lesů ($O(\log n)$):
  - Sleator-Tarjanovy dynamické stromy (splay stromy, amortizovaná složitost)
  - Fredericksonovy topologické stromy (rekursivní clusterizace, rozklad cest)

  $\Rightarrow$ komponenty lesa, ceny hran a cest
- Kvazidynamické algoritmy:
  - Povoleno mazat nestromové hrany a mosty
  - Souvislost: triviální
  - 2-souvislost: počítáme pokrytí hran

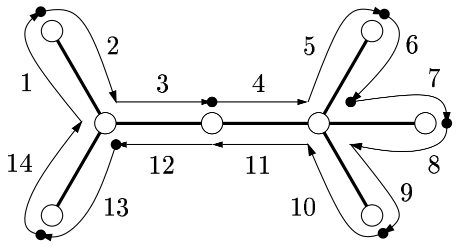


  - Min. kostra: Hledám dražší pokrytou hranu.
  - ST-stromy $\Rightarrow$ čas $O(k \cdot \log n)$ na k operací
- Algoritmus s orákulem:

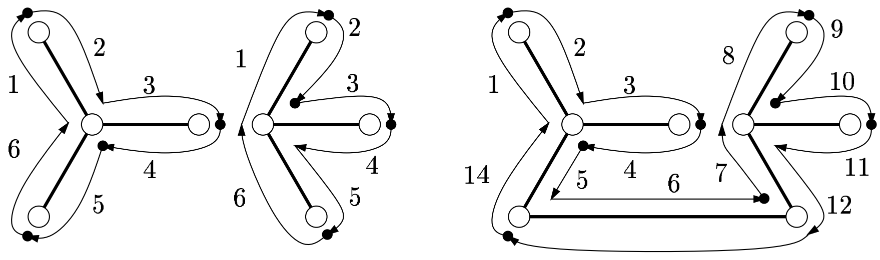
  Souvislost v čase $O(k \cdot \log n)$ na k operací

Eulerovské tahy

- Problém: Mazání stromových hran („rozpad kostry“)
  ⇒ potřebujeme za ni nalézt nestromovou náhradu
  ⇒ hledáme reprezentaci nestromových hran.
- Eulerovské číslování stromových hran a vrcholů:



- Reprezentace kostry pomocí $(a, 2a)$-stromů,
  Link a *Cut* v čase $O(a \cdot \log_a n)$, ostatní $O(\log_a n)$.

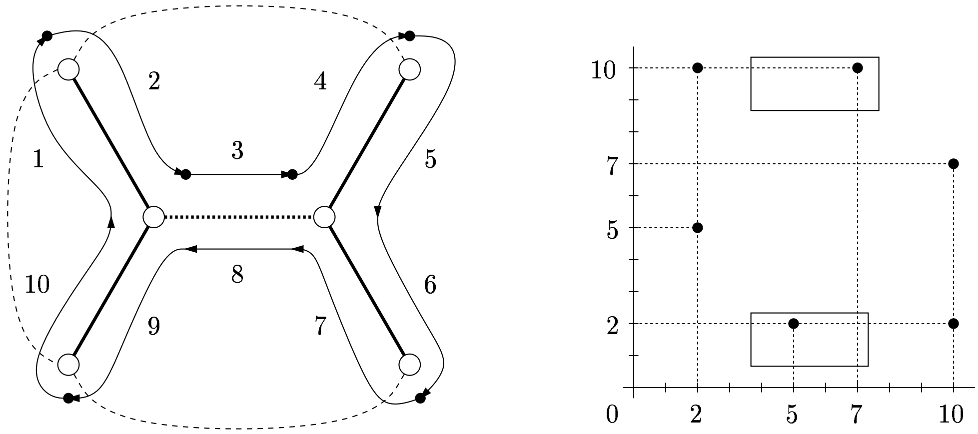


- Nestromové hrany „zavěsíme“ k vrcholům tahu
  ⇒ každá komponenta (vzhledem k rozpadlé kostře) odpovídá souvislé části tahu
  ⇒ odpovídá jednomu $(a, 2a)$-stromu
  ⇒ snadno nalezneme incidentní nestromové hrany

  Problém: potřebujeme hrany *mezi dvěma komp.*

Body v rovině

- Analogie s obdélníkovými dotazy ve 2D-strukturách (kupř. Mehlhornovy range trees)
- Převod nestromových hran na body v rovině:



- Nestromovou hranu hledáme jako bod ležící ve sjednocení $O(1)$ obdélníků.
- *Link* a *Cut* vyžadují přenášení bodů.

$\Rightarrow$ Plane-shift struktura: body, součty, přesuny.

Plně dynamické algoritmy

Nyní můžeme polylogaritmicky redukovat dynamickou (2-)souvislost na práci s body v rovině:

- Souvislost: přímočaře rozšíříme QD algoritmus.
- 2-souvislost:
  - QD algoritmus je bohužel slepou uličkou.
  - Pomůžeme si clusterizací, pro každý cluster evidujeme pokrytí jeho nestromovými hranami.
  - Mnoho informací $\Rightarrow$ sdílené stromy, CoW.
  - Dotaz: rozdělíme cestu podle clusterů, zkoumáme jejich pokrytí.
- Ukázková implementace plane-shift struktury $\Rightarrow$ $PolyLog$-krát pomalejší, ale jednodušší.
- Mezitím bohužel překonáno polylogaritmickými výsledky (Holm, Lichtenberg, Thorup).

Závěrem

Podařilo se:

- odvodit $PolyLog$ redukci na plane-shift strukturu
- ukázat nový pohled na reprezentaci dynam. grafů
- zjednodušit některé známé algoritmy pro $PolyLog$ použití, příp. zjednodušit jejich rozbor
- několik dalších dílčích výsledků (orákula apod.)
- přehledně v rámci jedné teorie popsat existující dynamické algoritmy.

Nepodařilo se:

- zkonstruovat $PolyLog$ plane-shift strukturu.
- být rychlejší než Holm, Lichtenberg a Thorup ☺