Distance oracles for timetable graphs

Vzdialenostné orákula pre grafy reprezentujúce cestovné poriadky

František Hajnovič

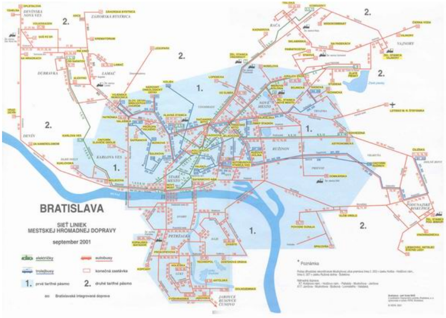
FMFI UK

4. februára 2013

Školiteľ: doc. RNDr. Královič PhD.

Hlavná tématika

- Rýchle hľadanie najkratších spojení v danom cestovnom poriadku



Obr. : Mapa MHD v BA [imh]

Obsah

1. Objasnenie problematiky
   - Využitie
   - Timetable graph
   - Motivácia
   - Distance oracles
2. Teoretická náplň
   - Highway dimension
   - r(n)-separator
   - Scale-free network
   - Výskum
3. Praktická náplň
   - Experimentálne zisťovanie vlastností
   - Úprava a implementácia algoritmov

Objasnenie problematiky

Objasnenie problematiky

Využitie

- Portály typu *cp.sk*, *imhd.sk*...

Využitie

- Portály typu *cp.sk*, *imhd.sk*...
- *Väčší rozsah* - napr. nad mapou Európy

Využitie

- Portály typu *cp.sk, imhd.sk*...
- *Väčší rozsah* - napr. nad mapou Európy
- **Earliest arrival problem (EAP)** - *problém najskoršieho príchodu*

Timetable graf - časovo expandovaný graf

- **Underlying graph** - *podkladový graf* (mapa)

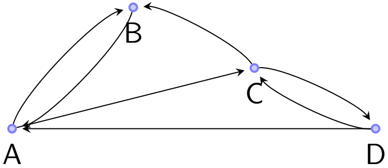
Timetable graf - časovo expandovaný graf

- **Underlying graph** - *podkladový graf* (mapa)
- **Timetable** - *cestovný poriadok* nad daným podkladovým grafom. Množina elementárnych spojení

Timetable graf - časovo expandovaný graf

- **Underlying graph** - *podkladový graf* (mapa)
- **Timetable** - *cestovný poriadok* nad daným podkladovým grafom. Množina elementárnych spojení
- **Time-expanded graph** - *časovo expandovaný graf* [MHSWZ07]

Podkladový graf



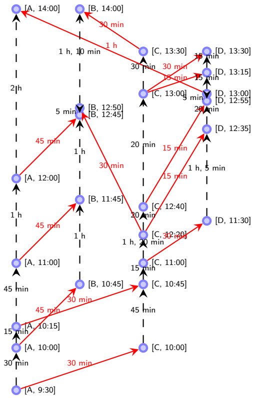
Obr. : Podkladový graf

Cestovný poriadok

Place		Time	
From	To	Departure	Arrival
A	B	10:00	10:45
A	B	11:00	11:45
A	B	12:00	12:45
A	C	9:30	10:00
A	C	10:15	10:45
C	D	11:00	11:30
C	D	13:00	13:30
C	D	12:20	12:35
C	D	12:40	12:55
C	D	13:00	13:15
C	B	12:20	12:50
C	B	13:30	14:00
D	A	13:00	14:00

Tabuľka : Cestovný poriadok nad predošlým grafom

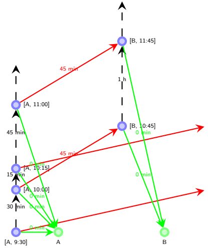
Časovo expandovaný graf



Obr. : Časovo expandovaný graf pre predchádzajúci cestovný poriadok

Vyhľadávanie cez najkratšiu cestu

- Pridáme pre každé mesto podkladového grafu jeden vrchol
- Pridáme orientované hrany s nulovou váhou idúce z každého vrcholu timetable grafu do patričného nového vrchola



Obr. : Riešenie EAP v časovo expandovaných grafoch cez najkratšiu cestu

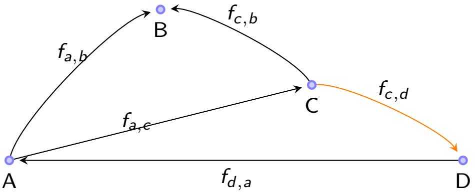
Timetable graf - časovo závislý graf

- **Underlying graph** - *podkladový graf* (mapa)
- **Timetable** - *cestovný poriadok* nad daným podkladovým grafom. Množina elementárnych spojení
- **Time-expanded graph** - *časovo expandovaný graf*
- **Time-dependent graph** - *časovo závislý graf* [MHSWZ07]. Vlastne iba podkladový graf, kde ceny hrán sa určia "on-the-fly" za behu algoritmu

Timetable graf - časovo závislý graf

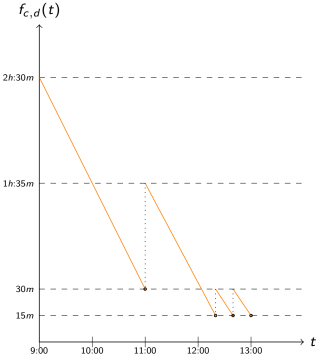
- **Underlying graph** - *podkladový graf* (mapa)
- **Timetable** - *cestovný poriadok* nad daným podkladovým grafom. Množina elementárnych spojení
- **Time-expanded graph** - *časovo expandovaný graf*
- **Time-dependent graph** - *časovo závislý graf* [MHSWZ07]. Vlastne iba podkladový graf, kde ceny hrán sa určia "on-the-fly" za behu algoritmu
  - Iný pohľad: ceny hrán sú *po častiach lineárne* funkcie (piecewise linear functions) udávajúce cenu hrany v každom časovom okamihu [DSSW09]

Časovo závislý graf



Obr. : Časovo závislý graf

Časovo závislý graf



Obr. : Po častiach lineárna funkcia udávajúca čas prechodu z C do D

Place		Time	
From	To	Departure	Arrival
C	D	11:00	11:30
C	D	13:00	13:30
C	D	12:20	12:35
C	D	12:40	12:55
C	D	13:00	13:15

Tabuľka : Spojenia medzi uzlom C a D

Motivácia

- Čo je zlé na Dijkstrovom algoritme?

Motivácia

- Čo je zlé na Dijkstrovom algoritme?
  - "Vytunená" verzia v $\mathcal{O}(m + n \log n)$, Fibonacciho haldy

Motivácia

- Čo je zlé na Dijkstrovom algoritme?
  - "Vytunená" verzia v $\mathcal{O}(m + n \log n)$, Fibonacciho haldy
- Avšak napr. celoeurópska sieť železníc má rádovo 100000 vrcholov $\rightarrow$ cestovný poriadok nad takouto sieťou mnohonásobne viac.

Motivácia

- Čo je zlé na Dijkstrovom algoritme?
  - “Vytunená” verzia v $\mathcal{O}(m + n \log n)$, Fibonacciho haldy
- Avšak napr. celoeurópska sieť železníc má rádovo 100000 vrcholov $\rightarrow$ cestovný poriadok nad takouto sieťou mnohonásobne viac.
- Na queries má server *odpovedať on-line*. Potenciálne veľa queries za časovú jednotku.

Distance oracles (Dištančné orákulá) [TZ05]

- Odpoveď na “pomalé” algoritmy (Dijkstra, A*)

Distance oracles (Dištančné orákulá) [TZ05]

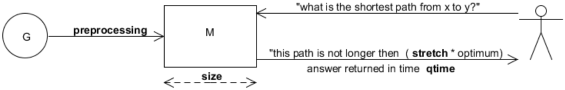
- Odpoveď na “pomalé” algoritmy (Dijkstra, A*)
- Štruktúra, čo odpovedá na otázky “Aká je vzdialenosť z bodu A do bodu B?”

Distance oracles (Dištančné orákulá) [TZ05]

- Odpoveď na “pomalé” algoritmy (Dijkstra, A*)
- Štruktúra, čo odpovedá na otázky “Aká je vzdialenosť z bodu A do bodu B?”
- 4 parametre, čo sa snažíme *tlačiť dole*:
  - **Preprocessing time** - *čas predspracovania*
  - **Size** - výsledná *veľkosť* štruktúry
  - **Query time** - *rýchlosť odpovede*
  - **Stretch** - *presnosť*. Chceme presné algoritmy

Distance oracles (Dištančné orákulá) [TZ05]

- Odpoveď na "pomalé" algoritmy (Dijkstra, A*)
- Štruktúra, čo odpovedá na otázky "Aká je vzdialenosť z bodu A do bodu B?"
- 4 parametre, čo sa snažíme *tlačiť dole*:
  - **Preprocessing time** - *čas predspracovania*
  - **Size** - výsledná *veľkosť* štruktúry
  - **Query time** - *rýchlosť odpovede*
  - **Stretch** - *presnosť*. Chceme presné algoritmy



Obr. : Metóda DO

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*

Teoretické hranice

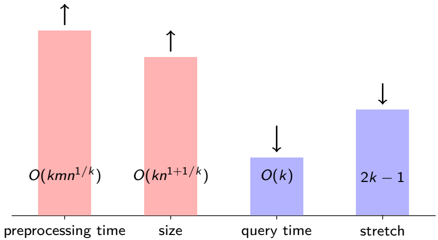
- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna



Obr. : Možné kompromisy DO Thorupa a Zwicka

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna
  - **Gavoille et al.** [GPPR04], Distance labelling (vzdialenostné oštítkovanie): Pre ľubovoľné n existuje graf veľkosti n

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna
  - **Gavoille et al.** [GPPR04], Distance labelling (vzdialenostné oštítkovanie): Pre ľubovoľné n existuje graf veľkosti n
    - Existuje exaktná schéma oštítkovania veľká $n\,3\log n + o(n\log n)$
    - Akákoľvek schéma oštítkovania veľkosti $\leq n^3/2 - \mathcal{O}(n^2\log n)$ má vzdialenostnú funkciu, ktorá je príliš časovo náročná na výpočet z praktického hľadiska

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna
  - **Gavoille et al.** [GPPR04], Distance labelling (vzdialenostné oštítkovanie): Pre ľubovoľné n existuje graf veľkosti n
    - Existuje exaktná schéma oštítkovania veľká $n\,3\log n + o(n\log n)$
    - Akákoľvek schéma oštítkovania veľkosti $\leq n^3/2 - \mathcal{O}(n^2\log n)$ má vzdialenostnú funkciu, ktorá je príliš časovo náročná na výpočet z praktického hľadiska
- Pre viaceré špeciálne triedy grafov existujú *efektívnejšie* metódy

Teoretické hranice

- *Pre všeobecné grafy neexistuje efektívne DO*
  - **Thorup & Zwick** [TZ05], DO pre všeobecné grafy:
    - *Preprocessing* $= \mathcal{O}(kmn^{1/k})$, *Size* $= \mathcal{O}(kn^{1+1/k})$, *Query time* $= \mathcal{O}(k)$, *Stretch* $= 2k - 1$
    - Veľkosť DO je v zásade optimálna
  - **Gavoille et al.** [GPPR04], Distance labelling (vzdialenostné oštítkovanie): Pre ľubovoľné n existuje graf veľkosti n
    - Existuje exaktná schéma oštítkovania veľká $n\,3\log n + o(n\log n)$
    - Akákoľvek schéma oštítkovania veľkosti $\leq n^3/2 - \mathcal{O}(n^2\log n)$ má vzdialenostnú funkciu, ktorá je príliš časovo náročná na výpočet z praktického hľadiska
- Pre viaceré špeciálne triedy grafov existujú *efektívnejšie* metódy
- Existujúce komerčné riešenia často nemajú teoretické garancie pre *exaktnosť* alebo *rýchlosť* odpovede na query [SS05]

Teoretická náplň

Teoretická náplň

Prístup

- Ak má podkladový graf nejakú (želanú) vlastnosť, *má ju aj expandovaný graf*?

Prístup

- Ak má podkladový graf nejakú (želanú) vlastnosť, *má ju aj expandovaný graf*?
- Pre aké typy cestovných poriadkov sa vlastnosť *zachováva* pri expandovaní?

Prístup

- Ak má podkladový graf nejakú (želanú) vlastnosť, *má ju aj expandovaný graf*?
- Pre aké typy cestovných poriadkov sa vlastnosť *zachováva* pri expandovaní?
- Pre grafy so želanou vlastnosťou existuje efektívny algoritmus na hľadanie najkratších ciest → vieme ho *upraviť pre cestovné poriadky nad týmto grafom*?

Prístup

- Ak má podkladový graf nejakú (želanú) vlastnosť, *má ju aj expandovaný graf*?
- Pre aké typy cestovných poriadkov sa vlastnosť *zachováva* pri expandovaní?
- Pre grafy so želanou vlastnosťou existuje efektívny algoritmus na hľadanie najkratších ciest → vieme ho *upraviť pre cestovné poriadky nad týmto grafom*?
- *Vyplýva* jedna vlastnosť z druhej?

Highway dimension

- **Highway dimension** (intuícia) - graf má malú HD, ak pre oblasť každej veľkosti máme malú množinu vrcholov, cez ktoré prechádzajú všetky dostatočne dlhé najkratšie cesty v grafe.

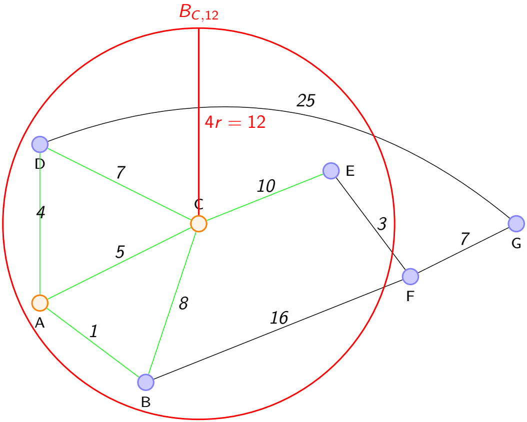
Highway dimension

- **Highway dimension** (intuícia) - graf má malú HD, ak pre oblasť každej veľkosti máme malú množinu vrcholov, cez ktoré prechádzajú všetky dostatočne dlhé najkratšie cesty v grafe.
- **Formálne** - $\forall r \in R^+, \forall u \in V_G, \exists S \subseteq B_{u,4r}, |S| \leq h$, taká, že $\forall v, w \in B_{u,4r}$:

Highway dimension

- **Highway dimension** (intuícia) - graf má malú HD, ak pre oblasť každej veľkosti máme malú množinu vrcholov, cez ktoré prechádzajú všetky dostatočne dlhé najkratšie cesty v grafe.
- **Formálne** - $\forall r \in R^+, \forall u \in V_G, \exists S \subseteq B_{u,4r}, |S| \leq h$, taká, že $\forall v, w \in B_{u,4r}$:
  - ak $|P(v,w)| > r$ a $P(v,w) \subseteq B_{u,4r}$ potom $P(v,w) \cap S \neq \emptyset$

Highway dimension



Obr. : Highway dimension

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD
  - 2005 spoločnosť PTV AG zverejnila európsku cestnú sieť [DSSW09]

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD
  - 2005 spoločnosť PTV AG zverejnila európsku cestnú sieť [DSSW09]
  - Začali sa "dostihy" - kto spraví rýchlejší algoritmus?

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD
  - 2005 spoločnosť PTV AG zverejnila európsku cestnú sieť [DSSW09]
  - Začali sa "dostihy" - kto spraví rýchlejší algoritmus?
  - *Zrýchlenia* (**speed-up)** oproti Dijkstrovému alg. až 3 000 000

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD
  - 2005 spoločnosť PTV AG zverejnila európsku cestnú sieť [DSSW09]
  - Začali sa "dostihy" - kto spraví rýchlejší algoritmus?
  - *Zrýchlenia* (**speed-up)** oproti Dijkstrovému alg. až 3 000 000
  - Algoritmy **Highway hierarchies** (2005), **Transit node routing** (2006), **Contraction hierarchies** (2008)

Malá HD

- Malá HD *garantuje nízku časovú zložitosť* queries viacerých *exaktných* DO metód, ktoré navyše majú rýchle predspracovanie a veľkosť DO iba o čosi väčšiu ako vstup [AFGW10]
- *Cestná sieť* má *predpokladanú* malú HD
    - 2005 spoločnosť PTV AG zverejnila európsku cestnú sieť [DSSW09]
    - Začali sa "dostihy" - kto spraví rýchlejší algoritmus?
    - *Zrýchlenia* (**speed-up)** oproti Dijkstrovému alg. až 3 000 000
    - Algoritmy **Highway hierarchies** (2005), **Transit node routing** (2006), **Contraction hierarchies** (2008)
    - Techniky: **hierarchy** (hierarchia), **shortcuts** (skratky), **landmarks** (význačné body), **contractions** (kontrakcie)

Problémy s úpravou

- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru

Problémy s úpravou

- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru
- V článku [DPW09] označili úpravu spomenutých algoritmov ako *"too challenging"*

Problémy s úpravou

- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru
- V článku [DPW09] označili úpravu spomenutých algoritmov ako *"too challenging"*
- Zatiaľ *"mierne zrýchlenia"*

Problémy s úpravou

- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru
- V článku [DPW09] označili úpravu spomenutých algoritmov ako *"too challenging"*
- Zatiaľ *"mierne zrýchlenia"*
  - [DPW09] zrýchlenie cca. 56

Problémy s úpravou

- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru
- V článku [DPW09] označili úpravu spomenutých algoritmov ako *"too challenging"*
- Zatiaľ *"mierne zrýchlenia"*
  - [DPW09] zrýchlenie cca. 56
  - **Time-dependent SHARC** [Del08] zrýchlenie cca. 26

Problémy s úpravou

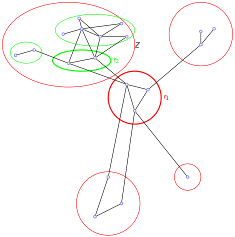
- Grafy cestovných poriadkov nemajú takú jednoduchú (takmer planárnu) štruktúru
- V článku [DPW09] označili úpravu spomenutých algoritmov ako *"too challenging"*
- Zatiaľ *"mierne zrýchlenia"*
  - [DPW09] zrýchlenie cca. 56
  - **Time-dependent SHARC** [Del08] zrýchlenie cca. 26
  - **Time-dependent CH** [BDSV09] zrýchlenie cca. ako SHARC, veľká pamäťová náročnosť

r(n)-separator

- **r(n)-separátor** [GPPR04] - graf má r(n)-separátor ak v ňom existuje množina vrcholov S, $|S| \leq r(n)$, ktorej odstránením sa graf rozpadne na kompenenty veľkostí $\leq 2n/3$, z ktorých každý má r(2n/3)-separátor

r(n)-separator

- **r(n)-separátor** [GPPR04] - graf má r(n)-separátor ak v ňom existuje množina vrcholov S, $|S| \leq r(n)$, ktorej odstránením sa graf rozpadne na kompenenty veľkostí $\leq 2n/3$, z ktorých každý má r(2n/3)-separátor



Obr. : Graf s 3-separátorom (čiže $r(n)$ je konštanta v tomto prípade)

Hľadanie separátora

- Planárne grafy majú $\mathcal{O}(\sqrt{n})$-separátor [Eri]

Hľadanie separátora

- Planárne grafy majú $\mathcal{O}(\sqrt{n})$-separátor [Eri]
- Hľadanie v planárnych grafoch v čase $\mathcal{O}(n)$

Hľadanie separátora

- Planárne grafy majú $\mathcal{O}(\sqrt{n})$-separátor [Eri]
- Hľadanie v planárnych grafoch v čase $\mathcal{O}(n)$
- Všeobecne hľadanie optimálneho separátora je **NP-ťažký** problém

Hľadanie separátora

- Planárne grafy majú $\mathcal{O}(\sqrt{n})$-separátor [Eri]
- Hľadanie v planárnych grafoch v čase $\mathcal{O}(n)$
- Všeobecne hľadanie optimálneho separátora je **NP-ťažký** problém
- Grafy *Internetu, sociálnych sietí* majú malé separátory

Separátor a jeho využitie

- Definujme $\mathbf{R(n)} = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^{\epsilon}$, kde $\epsilon > 0$

Separátor a jeho využitie

- Definujme $\mathbf{R}(\mathbf{n}) = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^{\epsilon}$, kde $\epsilon > 0$
- Pre grafy veľkosti n s r(n)-separátorom existuje schéma oštítkovania: [GPPR04]

Separátor a jeho využitie

- Definujme $\mathbf{R}(\mathbf{n}) = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^{\epsilon}$, kde $\epsilon > 0$
- Pre grafy veľkosti n s r(n)-separátorom existuje schéma oštítkovania: [GPPR04]
  - *Veľkosť schémy* oštítkovania $\leq \mathcal{O}(nR(n) \log W + n \log^2 n)$, kde W je najväčšia váha hrany
  - Vzdialenostná funkcia má *časovú zložitosť* $\mathcal{O}(\log n)$

Separátor a jeho využitie

- Definujme $\mathbf{R(n)} = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^{\epsilon}$, kde $\epsilon > 0$
- Pre grafy veľkosti n s r(n)-separátorom existuje schéma oštítkovania: [GPPR04]
  - *Veľkosť schémy* oštítkovania $\leq \mathcal{O}(nR(n) \log W + n \log^2 n)$, kde W je najväčšia váha hrany
  - Vzdialenostná funkcia má *časovú zložitosť* $\mathcal{O}(\log n)$
- V podstate teda máme DO:

Separátor a jeho využitie

- Definujme $\mathbf{R}(\mathbf{n}) = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^\epsilon$, kde $\epsilon > 0$
- Pre grafy veľkosti n s r(n)-separátorom existuje schéma oštítkovania: [GPPR04]
  - *Veľkosť schémy* oštítkovania $\leq \mathcal{O}(nR(n)\log W + n\log^2 n)$, kde W je najväčšia váha hrany
  - Vzdialenostná funkcia má *časovú zložitosť* $\mathcal{O}(\log n)$
- V podstate teda máme DO:
  - *Preprocessing* = **záleží..**, *Size* = $\mathcal{O}(nR(n)\log W + n\log^2 n)$, *Query time* = $\mathcal{O}(\log n)$, *Stretch* = 1

Separátor a jeho využitie

- Definujme $\mathbf{R(n)} = \sum_{i=0}^{\log_{3/2} n} r(n(2/3)^i)$. Platí $R(n) = \mathcal{O}(r(n))$ ak $r(n) \geq n^{\epsilon}$, kde $\epsilon > 0$
- Pre grafy veľkosti n s r(n)-separátorom existuje schéma oštítkovania: [GPPR04]
  - *Veľkosť schémy* oštítkovania $\leq \mathcal{O}(nR(n)\log W + n\log^2 n)$, kde W je najväčšia váha hrany
  - Vzdialenostná funkcia má *časovú zložitosť* $\mathcal{O}(\log n)$
- V podstate teda máme DO:
  - *Preprocessing* = **záleží..**, *Size* = $\mathcal{O}(nR(n)\log W + n\log^2 n)$, *Query time* = $\mathcal{O}(\log n)$, *Stretch* = 1
  - Napr. pre planárne grafy je čas predspracovania $\mathcal{O}(n^{3/2}\log n)$

Algoritmus (Gavoille)

- *Rekurzívne*:

Algoritmus (Gavoille)

- *Rekurzívne*:
  - Nájdenie separátora, rozdelenie na bloky

Algoritmus (Gavoille)

- *Rekurzívne*:
  - Nájdenie separátora, rozdelenie na bloky
  - Predrátanie vzdialeností do separátora

Algoritmus (Gavoille)

- *Rekurzívne*:
  - Nájdenie separátora, rozdelenie na bloky
  - Predrátanie vzdialeností do separátora
  - V rámci separátora každý s každým

Scale-free network

- **Scale-free network** - *bezškálová sieť* [Som10]

Scale-free network

- **Scale-free network** - *bezškálová sieť* [Som10]
  - Sieť, ktorá má distribúciu stupňov vrcholov podľa *mocninového zákona* (**power law**)

Scale-free network

- **Scale-free network** - *bezškálová sieť* [Som10]
  - Sieť, ktorá má distribúciu stupňov vrcholov podľa *mocninového zákona* (**power law**)
  - $Pr[\deg(v) = x] \propto C \cdot x^{-\tau}$

Scale-free network

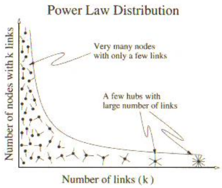
- **Scale-free network** - *bezškálová sieť* [Som10]
  - Sieť, ktorá má distribúciu stupňov vrcholov podľa *mocninového zákona* (**power law**)
  - $Pr[\deg(v) = x] \propto C \cdot x^{-\tau}$
  - C je konštanta, $\tau \in (2, 3)$

Scale-free network

- **Scale-free network** - *bezškálová sieť* [Som10]
  - Sieť, ktorá má distribúciu stupňov vrcholov podľa *mocninového zákona* (**power law**)
  - $Pr[\deg(v) = x] \propto C \cdot x^{-\tau}$
  - C je konštanta, $\tau \in (2, 3)$
- Skoro vždy **small-world** - očakávaný počet skokov pre náhodnú dvojicu vrcholov $\propto \log n$

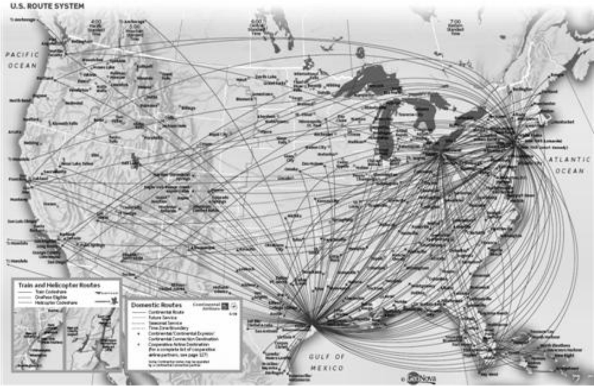
Scale-free network

- **Scale-free network** - *bezškálová sieť* [Som10]
  - Sieť, ktorá má distribúciu stupňov vrcholov podľa *mocninového zákona* (**power law**)
  - $Pr[\deg(v) = x] \propto C \cdot x^{-\tau}$
  - C je konštanta, $\tau \in (2, 3)$
- Skoro vždy **small-world** - očakávaný počet skokov pre náhodnú dvojicu vrcholov $\propto \log n$



Obr. : Distribúcia stupňov vrcholov podľa mocninového zákona

Scale-free network



Obr. : Sieť prepojenia letísk vykazuje charakteristiky bezškálovej siete [Far]

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]
  - Náhodné vybratie význačných bodov (**sampling**)

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]
  - Náhodné vybratie význačných bodov (**sampling**)
  - Pre každý bod určenie vzdialeností do význačných bodov → **sketch** (*črta*) vrchola

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]
  - Náhodné vybratie význačných bodov (**sampling**)
  - Pre každý bod určenie vzdialeností do význačných bodov $\rightarrow$ **sketch** (*črta*) vrchola
  - Pri danom query hľadanie *prieniku čŕt* dvoch vrcholov

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]
  - Náhodné vybratie význačných bodov (**sampling**)
  - Pre každý bod určenie vzdialeností do význačných bodov → **sketch** (*črta*) vrchola
  - Pri danom query hľadanie *prieniku čŕt* dvoch vrcholov
- **Adaptácia DO Thorupa a Zwicka** [Som10]

DO pre bezškálové siete

- Dve metódy, obe *nie úplne exaktné*, obe založené na význačných bodoch
- **Sketch-based DO** [SGNP10]
  - Náhodné vybratie význačných bodov (**sampling**)
  - Pre každý bod určenie vzdialeností do význačných bodov $\rightarrow$ **sketch** (*črta*) vrchola
  - Pri danom query hľadanie *prieniku čŕt* dvoch vrcholov
- **Adaptácia DO Thorupa a Zwicka** [Som10]
  - Za význačné body v ich algoritme sa vezmú vrcholy s *najväčším stupňom*

Vlastnosti rôznych typov sietí

Typ siete (grafu)	Malá HD	Malý separátor	Bezškálovosť
Linky MHD	✔/?	✔/?	✔/?
Medzimestské linky autobusov	✔/?	✔/?	✔/?
Železničná sieť	✔/?	✔/?	✔/?
Sieť leteckých liniek	?	?	✔[WC03]
Cestná sieť	✔/?, slabá [AFGW10]	✔, efektívne [Som10]	✗

Tabuľka : Vlastnosti jednotlivých sietí (nereferencované políčka sú mojou domnienkou)

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:
  - *Nad akým grafom* je daný cestovný poriadok

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:
  - *Nad akým grafom* je daný cestovný poriadok
  - *Akého typu* je sám cestovný poriadok (pravidelná štruktúra, rôzna frekvencia liniek v rôznych oblastiach)

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:
  - *Nad akým grafom* je daný cestovný poriadok
  - *Akého typu* je sám cestovný poriadok (pravidelná štruktúra, rôzna frekvencia liniek v rôznych oblastiach)
- "Divokým" cestovným poriadkom vieme ľahko pokaziť peknú štruktúru podkladového grafu

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:
  - *Nad akým grafom* je daný cestovný poriadok
  - *Akého typu* je sám cestovný poriadok (pravidelná štruktúra, rôzna frekvencia liniek v rôznych oblastiach)
- "Divokým" cestovným poriadkom vieme ľahko pokaziť peknú štruktúru podkladového grafu
- Vo všeobecnosti sa žiadna z vlastností (HD, r(n)-separátor, bezškálovosť, planarita) nezachová

Vlastnosti cestovných poriadkov

- Vlastnosti cestovných poriadkov budú *záležať* od toho:
  - *Nad akým grafom* je daný cestovný poriadok
  - *Akého typu* je sám cestovný poriadok (pravidelná štruktúra, rôzna frekvencia liniek v rôznych oblastiach)
- "Divokým" cestovným poriadkom vieme ľahko pokaziť peknú štruktúru podkladového grafu
- Vo všeobecnosti sa žiadna z vlastností (HD, r(n)-separátor, bezškálovosť, planarita) nezachová
- Inkrementálne povolovať čoraz zložitejšie cestovné poriadky - pokiaľ sa až dá ísť aby sa vlastnosti zachovávali a výpočet bol efektívny

Time-dependent vs. time-expanded prístup

- Časovo závislý vs. časovo expandovaný prístup

Time-dependent vs. time-expanded prístup

- Časovo závislý vs. časovo expandovaný prístup
  - Časovo závislé grafy sú preferované, zaberajú *menej miesta*

Time-dependent vs. time-expanded prístup

- Časovo závislý vs. časovo expandovaný prístup
  - Časovo závislé grafy sú preferované, zaberajú *menej miesta*
  - Na časovo expandované grafy možno vieme *upraviť algoritmy*

Praktická náplň

Praktická náplň

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
    - Náhodne generované cestovné poriadky nad cestnou sieťou

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
  - Náhodne generované cestovné poriadky nad cestnou sieťou
    - Dáta z Open street maps

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
  - Náhodne generované cestovné poriadky nad cestnou sieťou
    - Dáta z Open street maps
    - Dáta PTV AG [DSSW09]

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
  - Náhodne generované cestovné poriadky nad cestnou sieťou
    - Dáta z Open street maps
    - Dáta PTV AG [DSSW09]
    - Husté/riedke/vybalancované/pravidelné/nepravidelné cestovné poriadky...

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
  - Náhodne generované cestovné poriadky nad cestnou sieťou
    - Dáta z Open street maps
    - Dáta PTV AG [DSSW09]
    - Husté/riedke/vybalancované/pravidelné/nepravidelné cestovné poriadky...
  - Ako zisťovať efektívne HD?

Experimentálne zisťovanie vlastností

- *Experimentálne zisťovanie vlastností* grafov pre rôzne typy cestovných poriadkov
  - Náhodne generované cestovné poriadky nad cestnou sieťou
    - Dáta z Open street maps
    - Dáta PTV AG [DSSW09]
    - Husté/riedke/vybalancované/pravidelné/nepravidelné cestovné poriadky...
  - Ako zisťovať efektívne HD?
- C++ - kód
- Python - parsovanie vstupov

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
    - Náhodné generovanie nad mapou Slovenska

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
  - Náhodné generovanie nad mapou Slovenska
  - Medzimestské spoje v okolí Žiliny (cp.sk)

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
  - Náhodné generovanie nad mapou Slovenska
  - Medzimestské spoje v okolí Žiliny (cp.sk)
  - Cestovné poriadky ŽSR

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
  - Náhodné generovanie nad mapou Slovenska
  - Medzimestské spoje v okolí Žiliny (cp.sk)
  - Cestovné poriadky ŽSR
  - Náhodné cestovné poriadky pre London Underground

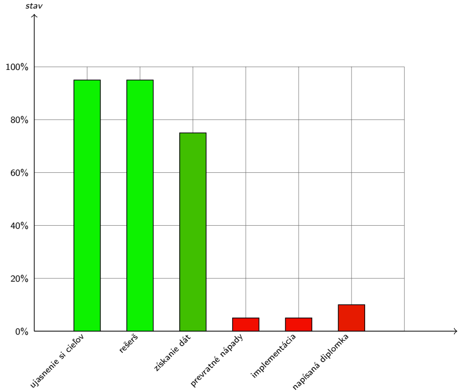
Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
  - Náhodné generovanie nad mapou Slovenska
  - Medzimestské spoje v okolí Žiliny (cp.sk)
  - Cestovné poriadky ŽSR
  - Náhodné cestovné poriadky pre London Underground
  - Cestovné poriadky leteckých liniek - United Airlines, Delta Airlines

Úprava a implementácia algoritmov

- Prostredie na testovanie a štatistiky DO metód
- Úprava existujúcich algoritmov (HH, Transit node routing, CH, Gavoillov alg., Sketch based DO)
- Návrh a implementácia algoritmu pre cestovné poriadky
- Rôzne dáta:
  - Náhodné generovanie nad mapou Slovenska
  - Medzimestské spoje v okolí Žiliny (cp.sk)
  - Cestovné poriadky ŽSR
  - Náhodné cestovné poriadky pre London Underground
  - Cestovné poriadky leteckých liniek - United Airlines, Delta Airlines
- C++ - kód
- Python - parsovanie vstupov

Stav diplomovky



Obr. : Stav diplomovky)

Bibliografia I

[AFGW10] Ittai Abraham, Amos Fiat, Andrew V. Goldberg, and Renato Fonseca F. Werneck. Highway dimension, shortest paths, and provably efficient algorithms. In Moses Charikar, editor, *SODA*, pages 782–793. SIAM, 2010.

[BDSV09] Gernot Veit Batz, Daniel Delling, Peter Sanders, and Christian Vetter. Time-dependent contraction hierarchies. In Irene Finocchi and John Hershberger, editors, *ALENEX*, pages 97–105. SIAM, 2009.

[BFM06] Holger Bast, Stefan Funke, and Domagoj Matijevic. Transit— ultrafast shortest-path queries with linear-time preprocessing, 2006.

[Del08] Daniel Delling. Time-dependent sharc-routing. In Dan Halperin and Kurt Mehlhorn, editors, *ESA*, volume 5193 of *Lecture Notes in Computer Science*, pages 332–343. Springer, 2008. ISBN 978-3-540-87743-1.

[DPW09] Daniel Delling, Thomas Pajor, and Dorothea Wagner. Engineering time-expanded graphs for faster timetable information. In Ravindra Ahuja, Rolf Möhring, and Christos Zaroliagis, editors, *Robust and Online Large-Scale Optimization*, volume 5868 of *Lecture Notes in Computer Science*, pages 182–206. Springer Berlin / Heidelberg, 2009. ISBN 978-3-642-05464-8.

[DSSW09] Daniel Delling, Peter Sanders, Dominik Schultes, and Dorothea Wagner. Engineering route planning algorithms. In *ALGORITHMICS OF LARGE AND COMPLEX NETWORKS. LECTURE NOTES IN COMPUTER SCIENCE*. Springer, 2009.

[Eri] Jeff Erickson. Graph separators.

[Far] Keith Farnish. A matter of scale.

[GPPR04] Cyril Gavoille, David Peleg, Stéphane Pérennes, and Ran Raz. Distance labeling in graphs. *Journal of Algorithms*, 53(1):85 – 112, 2004. ISSN 0196-6774. URL http://www.sciencedirect.com/science/article/pii/S0196677404000884.

Bibliografia II

[GSSD08] Robert Geisberger, Peter Sanders, Dominik Schultes, and Daniel Delling. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In Catherine C. McGeoch, editor, *WEA*, volume 5038 of *Lecture Notes in Computer Science*, pages 319–333. Springer, 2008. ISBN 978-3-540-68548-7.

[imh] imhd.sk. Mapa mhd ba.

[MHSWZ07] Matthias Müller-Hannemann, Frank Schulz, Dorothea Wagner, and Christos Zaroliagis. *Algorithmic Methods for Railway Optimization*, volume 4359 of *Lecture Notes in Computer Science*, chapter Timetable Information: Models and Algorithms, pages 67 – 90. Springer, 2007.

[SGNP10] Atish Das Sarma, Sreenivas Gollapudi, Marc Najork, and Rina Panigrahy. A sketch-based distance oracle for web-scale graphs. In Brian D. Davison, Torsten Suel, Nick Craswell, and Bing Liu, editors, *WSDM*, pages 401–410. ACM, 2010. ISBN 978-1-60558-889-6.

[Som10] Christian Sommer. *Approximate Shortest Path and Distance Queries in Networks*. PhD thesis, Graduate School of Information Science and Technology, The University of Tokyo, 2010.

[SS05] Peter Sanders and Dominik Schultes. Highway hierarchies hasten exact shortest path queries. In Gerth Stølting Brodal and Stefano Leonardi, editors, *ESA*, volume 3669 of *Lecture Notes in Computer Science*, pages 568–579. Springer, 2005. ISBN 3-540-29118-0.

[TZ05] Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, 2005.

[WC03] Xiao F. Wang and Guanrong Chen. Complex networks: small-world, scale-free and beyond. *Circuits and Systems Magazine, IEEE*, 3(1):6–20, 2003. ISSN 1531-636X. URL `http://dx.doi.org/10.1109/MCAS.2003.1228503`.

[Wik] Wikipedia. London underground - wikipedia.

Otázky a nápady

Ďakujem za pozornosť