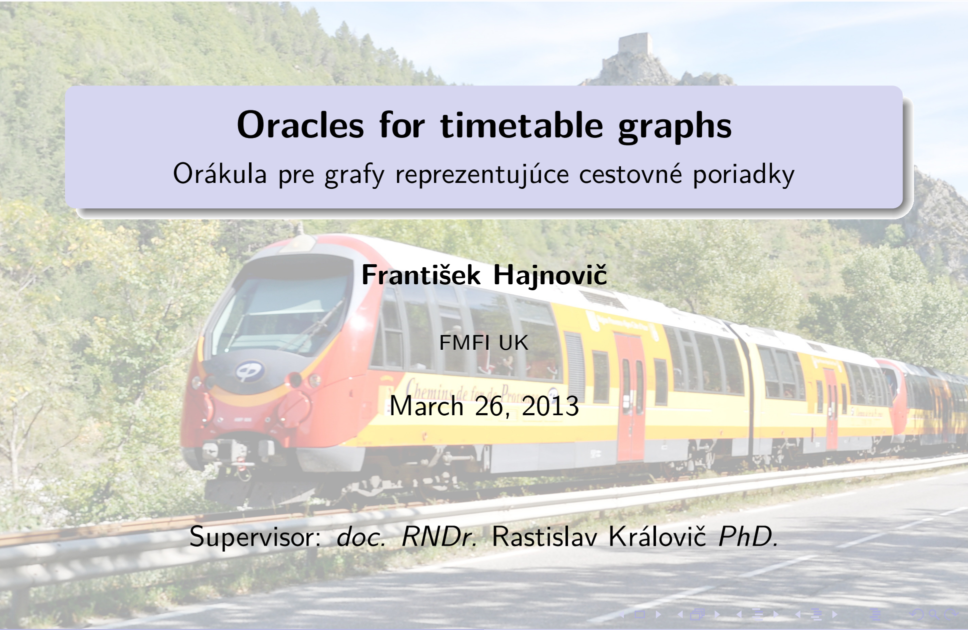


Oracles for timetable graphs

Orákula pre grafy reprezentujúce cestovné poriadky

František Hajnovič

FMFI UK

March 26, 2013

Supervisor: *doc. RNDr.* Rastislav Královič *PhD.*

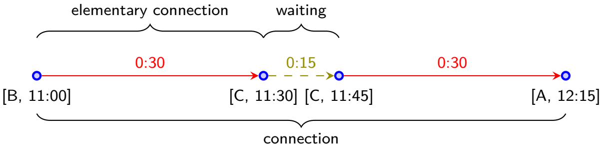
Content

Introduction

Introduction

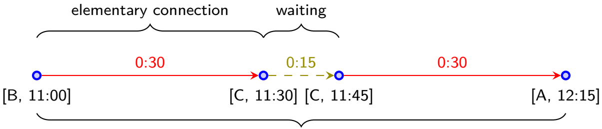
What is it about?

- Given a timetable, we query - (a, t, b) - for
  - **Earliest arrival** (EA) - $t^*_{(a,t,b)}$
  - **Optimal connection** (OC) - $c^*_{(a,t,b)}$

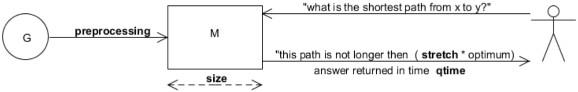


What is it about?

- Given a timetable, we query - (a, t, b) - for
  - **Earliest arrival** (EA) - $t^*_{(a,t,b)}$
  - **Optimal connection** (OC) - $c^*_{(a,t,b)}$



- **Motivation**
  - Large-scale timetable search engines (*cp.sk, imhd.sk*...)
- **Approach**
  - (Distance) oracle-based approach [TZ05] - pre-computation



Timetable and underlying graph

Place		Time	
From	To	Departure	Arrival
A	B	10:00	10:45
B	C	11:00	11:30
B	C	11:30	12:10
B	A	11:20	12:30
C	A	11:45	12:15

Table : **Timetable** - a set of **elementary connections** (between pairs of **cities**)

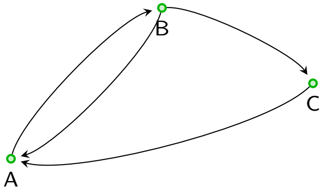


Figure : **Underlying graph**

Timetable and underlying graph

Place		Time	
From	To	Departure	Arrival
A	B	10:00	10:45
B	C	11:00	11:30
B	C	11:30	12:10
B	A	11:20	12:30
C	A	11:45	12:15

Table : **Timetable** - a set of **elementary connections** (between pairs of **cities**)

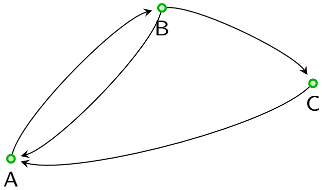


Figure : **Underlying graph**

- **Goals**
  - Devise methods to tackle EA/OC problem
  - Analyse properties of timetables

Contribution

Contribution

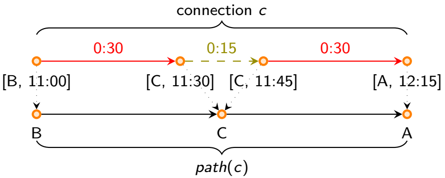
Data

Name	Description	El. conns.	Cities	UG arcs	Time range	Height (h)
air01	domestic flights (US)	601489	287	4668	1 month	24374
cpru	regional bus (SVK)	37148	871	2415	1 day	239
cpza	regional bus (SVK)	60769	1108	2778	1 day	370
montr	public transport (Montreal)	7153	217	349	1 day	363
sncf	country-wide rails (FRA)	90676	2646	7994	1 day	488
zsr	country-wide rails (SVK)	932052	233	588	1 year	60308

Table : Timetables datasets

Idea

- *"Usually we go through the same sequence of cities"*

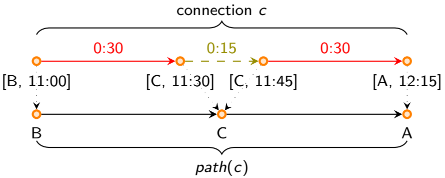


Odchod	Príchod	Dĺžka cesty*	Použité linky	Zóny	Cena*
21:59	22:10	11 min	95	-	0,70 €
22:09	22:20	11 min	95	-	0,70 €
22:19	22:30	11 min	95	-	0,70 €
22:29	22:40	11 min	95	-	0,70 €
22:39	22:50	11 min	95	-	0,70 €

- p is USP $\iff \exists t : path(c^*_{(a,t,b)}) = p$
- we have USP $\rightarrow$ reconstruct $c^*_{(a,t,b)}$

Idea

- *"Usually we go through the same sequence of cities"*



Odchod	Príchod	Dĺžka cesty*	Použité linky	Zóny	Cena*
21:59	22:10	11 min	95	-	0,70 €
22:09	22:20	11 min	95	-	0,70 €
22:19	22:30	11 min	95	-	0,70 €
22:29	22:40	11 min	95	-	0,70 €
22:39	22:50	11 min	95	-	0,70 €

- p is USP $\iff \exists t : path(c^*_{(a,t,b)}) = p$
- we have USP $\rightarrow$ reconstruct $c^*_{(a,t,b)}$
- **Overtaking** [MHSWZ07] causes problems, but can be easily removed

Name	Overtaken edges (%)
air01	1%
cpru	2%
cpza	2%
montr	1%
sncf	2%
zsr	0%

USP-OR

- Pre-compute *all conn.* - space $\mathcal{O}(h\ n^3)$
    - daily height usually $200 < h < 800$

USP-OR

- Pre-compute *all conn.* - space $\mathcal{O}(h\ n^3)$
  - daily height usually $200 < h < 800$
- Pre-compute *all USPs* - space $\mathcal{O}(\tau\ n^3)$
  - $\tau_{A,B}$ - # of USPs between A and B

Name	avg $\tau_{A,B}$	max $\tau_{A,B}$
air01	5.8	30
cpru	7.0	64
cpza	5.1	42
montr	4.3	30
sncf	4.3	24
zsr	2.5	19

Table : Daily, 200 station timetables

Name	avg USP size
air01	3.0
cpru	13.8
cpza	11.1
montr	20.3
sncf	10.5
zsr	13.7

Table : Daily, 200 station timetables

USP-OR

- Pre-compute *all conn.* - space $\mathcal{O}(h\ n^3)$
  - daily height usually $200 < h < 800$
- Pre-compute *all USPs* - space $\mathcal{O}(\tau\ n^3)$
  - $\tau_{A,B}$ - # of USPs between A and B

Prep-time	Size	Q-time	Stretch
$\mathcal{O}(hn^3)$	$\mathcal{O}(\tau\ n^3)$	$\mathcal{O}(\tau\ n)$	1

Table : USP-OR parameters

- τ almost constant, USP size $\approx \sqrt{n}$
- Space $\mathcal{O}(n^{2.5})$ too big anyway

Name	avg $\tau_{A,B}$	max $\tau_{A,B}$
air01	5.8	30
cpru	7.0	64
cpza	5.1	42
montr	4.3	30
sncf	4.3	24
zsr	2.5	19

Table : Daily, 200 station timetables

Name	avg USP size
air01	3.0
cpru	13.8
cpza	11.1
montr	20.3
sncf	10.5
zsr	13.7

Table : Daily, 200 station timetables

USP-OR - τ evolution

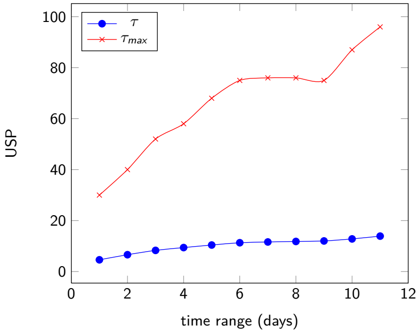


Figure : Changing of τ with increased time range in *air01* dataset. 1 day = about 800 in height

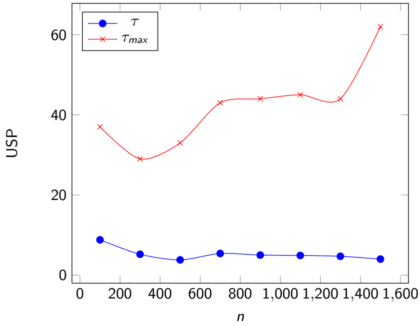
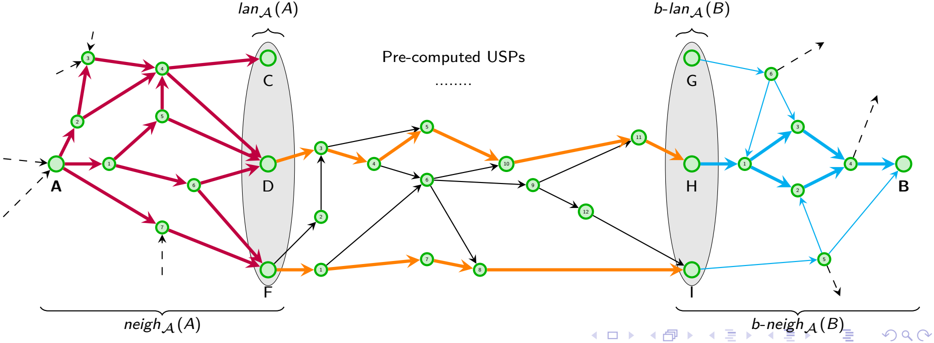


Figure : Changing of τ with increased # of stations in *sncf* dataset

USP-OR-A

- Pre-compute USPs only among *some* cities in UG: set of **access nodes** $\mathcal{A}$
- We would like ($r1$ and $r2$ small constants w.r.t. n):
  - Size $|\mathcal{A}| = \mathcal{O}(\sqrt{n})$
  - Small node neighbourhoods: $\forall v \; |neigh_{\mathcal{A}}(v)| < r_1 \cdot \sqrt{n}$
  - Few local access nodes: $\forall v \; |lan_{\mathcal{A}}(v)| \leq r_2$



Searching for optimal AN set

- NP-complete
  - Reduction to min-set-cover

Figure : text

Searching for optimal AN set

- NP-complete
  - Reduction to min-set-cover

Figure : text

- Heuristics: node v with low $|neigh_{\mathcal{A}}(v) \cap b\text{-}neigh_{\mathcal{A}}(v)|$ but high $\min\{|neigh_{\mathcal{A}}(v)|, |b\text{-}neigh_{\mathcal{A}}(v)|\}$ is good ANs

Existing methods

- **Time-dependent SHARC** [Del08], **Time-dependent CH** [BDSV09]
  - Speed-ups of about 26 / 1500, respectively (EA only)
  - Meant for time-dependent routing in road networks
- **Time-expanded approach** [DPW09]
  - Speed-ups of about 56
  - Remodelling unimportant stations
- **Theory vs. practice** difference
  - Inclusion of transfers, cost...

Neural network approaches

- Multi-layer perceptron, back propagation
- Input layer = events + cities. Output layer:
  1. Arcs of UG → USP
  2. Arcs of UG → routing
  3. Earliest arrival value

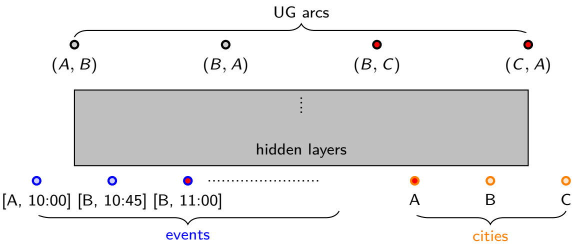


Figure : Approach 1.)

Results

- Tendency to remember USPs
- Long training times

Name	Conn.	Found	Was optimum (%)
air01	931	573	18.7%
cpru	481	281	48%
montr	527	346	86.7%
zsr	672	307	76.2%

Table : Tests of a trained NN on timetables with 30 cities (approach 1.)

Timetable analyzer - TTBlazer

- Works with UG, TE, TD, TT
- Analysis (τ, HD, degrees...), oracles (USP-OR, Dijkstra...), modifications (remove overtaking...), generation (subgraphs, TT $\rightarrow$ TD ...)
- Running & evaluating tests
- Easily extendible

Figure : It's *blazing* fast!

Conclusion

Conclusion

Conclusion

- Trying out novel approaches to solving EAP in timetables
  - *USP-OR*: Exact and quick answers but high space and time preprocessing
  - *NN*: Problem too challenging for NN/try different types of network
- Analysis of **various** real-world timetables
  - Better insight on properties of timetables
- Useful and easily extendible application

Bibliography I

[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.

[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.

[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.

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

Thank you for the attention