Multimodal and multi-criteria shortest paths in transportation networks

Antonin Lentz

under the supervision of Nicolas Hanusse et David Ilcinkas

LaBRI, France
Introduction

Shortest path problem on a road network
Classical methods

Let $G$ be a graph with $n$ vertices and $m$ edges.

Direct methods

- Dijkstra: $O(m \cdot \log n)$
- Bellman Ford: $O(m \cdot n)$
Let $G$ be a graph with $n$ vertices and $m$ edges.

**Direct methods**

- Dijkstra: $O(n \cdot \log n)$
- Bellman Ford: $O(n^2)$
Classical methods

Let $G$ be a graph with $n$ vertices and $m$ edges.

### Direct methods
- Dijkstra: $O(n \cdot \log n)$
- Bellman Ford: $O(n^2)$

### Speed-up
- **landmarks**: compute shortest distances from everyone to a well chosen subset of vertices
- **separators**: efficient for planar graphs
- **contraction hierarchies**: efficient for low tree-width graphs
- and many others.
Classical methods

Comparison of different methods [Bast et al., 2016]

Western Europe: 18.0 millions vertices and 42.5 millions edges
Multimodal multi-criteria shortest paths

We want to take into account:

- other transportation means (foot, bicycle, bus, train, plane, ...)
- individual preferences (fast, cheap, easy, ...)

This emphasizes several issues:

- huge graphs
- data are collected by different providers: trade-off queries/uncertainty.
  → distributed algorithms
Temporal graphs

- Edges available at precise times! Two ways to represent it:

  - One-time edges
  - Dependent-time edges

- Dijkstra-like algorithms [Dibbelt et al., 2013]
- Same speed-up technics, but less convincing improvements.
Temporal graphs

- Edges available at precise times! Two ways to represent it:
  - one-time edges,
  - dependent-time edges.
- Dijkstra-like algorithms [Dibbelt et al., 2013]
- Same speed-up technics, but less convincing improvements.

Merging graphs

Two types of transportation: time dependent and time independent.
Problem: how to modelize a "union graph"? No consensus

Exemple: public transportation graph with small foot complete graphs [Wagner and Zündorf, 2017]
Multi-criteria

- Compare time, costs, transfert number, ... \( \implies d \) dimensions,
- Goal: obtain non dominated solutions (Pareto Set),
- Problem: too many solutions ! If \( \Delta \) is the maximum degree, possibly \( \Delta^n \) solutions.

\[
\begin{align*}
\text{s} & \quad \text{u}_1 & \quad \text{u}_2 & \quad \text{u}_{n-1} & \quad \text{t} \\
(1,0) & \quad (2,0) & \quad (2^k,0) & \quad (2^{n-1},0) \\
(0,1) & \quad (0,2) & \quad (0,2^k) & \quad (0,2^{n-1})
\end{align*}
\]
Multi-criteria

- Compare time, costs, transfer number, ... \( \Rightarrow \) \( d \) dimensions,
- Goal: obtain non-dominated solutions (Pareto Set),
- Problem: too many solutions! If \( \Delta \) is the maximum degree, possibly \( \Delta^n \) solutions.

\[
\begin{array}{cccc}
  s & \rightarrow (1,0) & u_1 & \rightarrow (2,0) \\
  & (0,1) & \rightarrow & (0,2) \\
  u_2 & \rightarrow (2^k,0) & u_{n-1} & \rightarrow (2^{n-1},0) \\
  & (0,2^k) & \rightarrow & (0,2^{n-1}) \\
  t & \\
\end{array}
\]

\( \rightarrow \) Approximation: cover the solution set.
→ Polynomial cover set exists [Papadimitriou and Yannakakis, 2000]

Solution set with two dimensions
→ Polynomial cover set exists [Papadimitriou and Yannakakis, 2000]

Solution set with two dimensions
→ Polynomial cover set exists [Papadimitriou and Yannakakis, 2000]

Solution set with two dimensions
→ Polynomial cover set exists [Papadimitriou and Yannakakis, 2000]

Solution set with two dimensions
→ Polynomial cover set exists [Papadimitriou and Yannakakis, 2000]

Problem: for an \((1 + \epsilon)\) precision, existing algorithms are exponential with respect to \(d\):

\[
\mathcal{O}\left( n^d \cdot m \left( \frac{\log(nK)}{\epsilon} \right)^{d-1} \right) \quad [\text{Tsaggouris and Zaroliagis, 2009}]
\]
Conclusion

Current work

We are trying to find a fully polynomial algorithm. Ideas so far: either correct or exponential.

Future work

- distributed aspects,
- uncertainty
Thanks for your attention!


