Time-Expanded vs Time-Dependent Models for Timetable Information

Evangelia Pirga  
Computer Technology Institute & University of Patras

Frank Schulz  
University of Konstanz

Dorothea Wagner  
University of Konstanz

Christos Zaroliagis  
Computer Technology Institute & University of Patras
Timetable Information

▶ Reduces to computing shortest paths in a specific graph model

▶ Two main models:
  ● Time-Expanded
  ● Time-Dependent

▶ Main approach: Dijkstra-like algorithms
Time-Expanded Model

- Vertices $\leftrightarrow$ Events (e.g., departures, arrivals) at stations
- Edges $\leftrightarrow$ elementary connections between two events
- Weight of edge $(a,b) : \text{Time}(b) - \text{Time}(a)$
Time-Expanded Model (cntd)

- Large size

- Earliest Arrival problem:
  
  size can be reduced by half (all “arrival” vertices, except for those of the target station, can be ignored).
Time-Dependent Model

Vertices $\leftrightarrow$ Stations

Edges $\leftrightarrow$ elementary connections between two stations

Weight of edge $(a, b)$: depends on the arrival time at station $a$
During Dijkstra-like computation:

weight of edge \((a, b)\) is computed on the fly.

Simple method:

binary search on an array associated with \((a, b)\) and sorted w.r.t. departure time from \(a\).
## Size Comparison between Models

- **Data:** DB Timetable of Winter 1996/97

<table>
<thead>
<tr>
<th>Model</th>
<th>#Vertices</th>
<th>#Edges</th>
<th>Average Out-degree</th>
<th>Avg #Elem. Connect./edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-Exp. (full)</td>
<td>931746</td>
<td>1397619</td>
<td>1.5</td>
<td>0.67</td>
</tr>
<tr>
<td>T-Exp. (half)</td>
<td>465888</td>
<td>931776</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>T-Dep.</td>
<td>6961</td>
<td>18664</td>
<td>5.36</td>
<td>23.53</td>
</tr>
</tbody>
</table>
### Time-Expanded Model: Speedup Heuristics

- **[Schulz, Wagner, Weihe, WAE99]**

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Speedup (over Dijkstra)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A: angle restriction</td>
<td>5.7</td>
</tr>
<tr>
<td>S: selection of stations</td>
<td>8.6</td>
</tr>
<tr>
<td>G: goal-directed search</td>
<td>1.4</td>
</tr>
<tr>
<td>A + S + G</td>
<td>30.0</td>
</tr>
</tbody>
</table>

- **[Schulz, Wagner, Zaroliagis, ALENEX2002]**

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Speedup (over Dijkstra)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-level Graphs</td>
<td>11.2</td>
</tr>
</tbody>
</table>
Time-Dependent Model: Simple method (binary search)

↑ # Edges
→ # Elementary Connections

↑ # Edges
→ \log_2(# Elementary Connections)
# Time-Dependent vs Time-Expanded: Simple approaches

<table>
<thead>
<tr>
<th>Model</th>
<th>Algorithm</th>
<th>Avg Query Time [ms]</th>
<th>#Elem. Connect.</th>
<th>#Nodes touched</th>
<th>#Edges touched</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-Exp.(f)</td>
<td>Dijkstra</td>
<td>44.17</td>
<td>18406</td>
<td>33653</td>
<td>47807</td>
</tr>
<tr>
<td>T-Exp.(h)</td>
<td>Dijkstra</td>
<td>29.2</td>
<td>18406</td>
<td>18406</td>
<td>33377</td>
</tr>
<tr>
<td>T-Dep</td>
<td>Bin. Search</td>
<td>5.61</td>
<td>14033</td>
<td>1515</td>
<td>4463</td>
</tr>
</tbody>
</table>
Time-Dependent Model: Speedup Heuristics

- Avoiding Binary Search
- Goal-directed Search (potentials)
Time-Dependent Model: Avoiding Binary Search

- Sort all events of \((v, u_i), 1 \leq i \leq k\).

- \(\forall(v, u_i), \) find first event with \(t \geq t_0\) and put it into the next secondary segment.

- Let \(t_1\) be the arrival time of a primary event \(P^w \in D_w\). Create a pointer from \(P^w\) to the very next primary event \(P^v \in D_v\) with timestamp \(t_2 \geq t_1\).
Time-Dependent Model: Potentials

- For edge $(u, v)$ and $\tau$ the arrival time at $u$:

$$wt'(u, v, \tau) = wt(u, v, \tau) - p[u] + p[v]$$

- $p[ ]$ must be valid, i.e., $wt'(u, v, \tau) \geq 0$.

- For destination node $t \in V$:

$$p[u] = d(u, t)\lambda_t, u \in V, \lambda_t \geq 0$$

where

$$0 \leq \lambda_t = \min_{(u,v)\in E} \frac{\min_{\tau} wt(u, v, \tau)}{d(u, t) - d(v, t)}$$

- $wt(u, v, \tau) - p[u] + p[v] \geq 0 \Rightarrow wt(u, v, \tau) - \lfloor p[u]\rfloor + \lfloor p[v]\rfloor \geq 0$
## Time-Dependent Model: Comparison of Heuristics

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Avg Query Time [ms]</th>
<th>#Nodes Touched</th>
<th>#Edges Consid.</th>
<th>#Edges Unneces.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bin. Search</td>
<td>5.61</td>
<td>1515</td>
<td>2360</td>
<td>452</td>
</tr>
<tr>
<td>Avoid B.S.</td>
<td>6.41</td>
<td>1515</td>
<td>3446</td>
<td>829</td>
</tr>
<tr>
<td>Euclidean Pot.</td>
<td>6.75</td>
<td>984</td>
<td>1584</td>
<td>–</td>
</tr>
<tr>
<td>Eucl. Pot. &amp; int PQ</td>
<td>6.54</td>
<td>988</td>
<td>1590</td>
<td>–</td>
</tr>
<tr>
<td>Manhattan Pot.</td>
<td>6.35</td>
<td>1025</td>
<td>1645</td>
<td>–</td>
</tr>
<tr>
<td>Manh. Pot. &amp; int PQ</td>
<td>5.59</td>
<td>1030</td>
<td>1652</td>
<td>–</td>
</tr>
<tr>
<td>Avoid B.S.++</td>
<td>5.66</td>
<td>1515</td>
<td>2360</td>
<td>–</td>
</tr>
<tr>
<td>Avoid B.S.++ &amp;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manh. Pot. &amp; int PQ</td>
<td>5.38</td>
<td>1030</td>
<td>1652</td>
<td>–</td>
</tr>
</tbody>
</table>

**Considered edges**: distance of tail vertex not yet known

**Unnecessary edges**: those of considered for which the elementary connection departs later than the earliest arrival time at destination
Extensions: Modeling Train Changes

1. Introduce minimum time required for changes (global or local per station)

2. Introduce exceptions when trains depart from the same platform

3. For every train arrival, maintain a list of connecting trains
Modeling Train Changes: Time-Expanded Model

1 & 2

3
Above modeling doesn’t work

Assume

- trains \( T_1 = (u, x) \) and \( T_2 = (v, y) \)
- earliest arrival at \( D \) is through \( u \)
- \( T_1 \) does not reach destination, but \( T_2 \) does

\( \downarrow \)

May miss earlier connection through \( T_2 \) to destination
Conclusions & Further Research

▶ Time-dependent models
   • require less space
   • do not necessarily provide faster query times
   • not trivial to model other optimization criteria

▶ Extensive comparative study among all heuristics in both models

▶ Investigate extension of both models towards other optimization criteria