A low-memory alternative for time-dependent Dijkstra*

Simon Van den Eynde
Ghent University - imec, IDLab
e-mail: simon.vandeneynde@ugent.be

Jeroen Verbrugghe

Pieter Audenaert Ghent University - imec, IDLab
Ben Derudder Ghent University,
Didier Colle Ghent University - imec, IDLab
Mario Pickavet Ghent University - imec, IDLab

30 January 2020

Introduction Congestion is problematic for drivers since it induces delays and frustrations. Congestion is also problematic for routing algorithms because it transforms the edge costs from fixed to time-dependent. From this change, a new branch of routing algorithms has sprouted: the time-dependent routing algorithms. One of the first of these algorithms was presented in 1969 by Dreyfus: Time-Dependent Dijkstra (TD-Dijkstra). More recently, several papers describing time-dependent routing algorithms were published. However, most of these papers focus on query time. While that is admirable, this usually contains a trade-off between ease of implementation and memory requirements. This research presents a novel time-dependent routing heuristic with low memory requirements and a straightforward implementation. The Time-Location Penalty Model (TLPM) combines TD-Dijkstra with a preprocessing step to strongly reduce memory usage.

The algorithm So TLPM is based on TD-Dijkstra. For finding the fastest routes, TD-Dijkstra maintains a table with travel time values for every single edge. To obtain the actual travel time during route calculation, the algorithm accesses the current edge's table and then queries this table with the current time of the route to get the actual travel time. Instead of saving a table of travel times for each edge, TLPM stores two types of values: it associates a location penalty with each edge and a time penalty with each time step.

The penalties are defined in function of $\tau(e,t)$, the time to travel across edge e departing at t and in function of $\tau_{min}(e)$, the minimal travel time across edge e. The location penalty of an edge e is $L(e)$: The average loss on e during the

*This research was partly funded by the Ghent University IOP project "Modelling Uncertainty in Hub Location Planning through Interdisciplinary Research".

day.

$$L(e) = \frac{1}{1440} \sum_t \left(\frac{\tau(e,t) - \tau_{min}(e)}{\tau_{min}(e)} \right)$$

With E the collection of edges, the time penalty at time t is $T(t)$, the average loss on the entire network at time t.

$$T(t) = \frac{1}{|E|} \sum_e \left(\frac{\tau(e,t) - \tau_{min}(e)}{\tau_{min}(e)} \right)$$

In other words, the location penalty is the normalized average delay on an edge during the entire day, while the time penalty is the normalized average delay over the entire network on a single time of the day. After normalization, a penalty of 0 signifies no traffic and 1 signifies that the route takes twice as much time.

Due to the normalization, both penalties sum to the same value. However, their distributions differ strongly. There are some very high location penalties (around 1.8), but most edges have a penalty under 0.2. On the other hand, the edge-penalty distribution has three peaks: at night 0, during the day 0.3, and during the rush hours 0.6. In particular, the maximum is 0.6.

These penalties are used as follows. When routing with TLPM, the basis is a TD-Dijkstra algorithm, but instead of reading the travel times from a table of values, a model that maps the penalties to the travel time is used. The model should express the relation between both penalties. An example model is $a + b \cdot L \cdot T$, where the parameters a, b can be chosen and L, T are, respectively, the location and time penalties.

Evaluation Five models that estimate the travel time from a time and location penalty were proposed and evaluated. These models were trained and tested on an East-Flanders time-dependent dataset[1] consisting of 3825 nodes, 8570 edges and travel times for each minute of the day. TD-Dijkstra is used as a benchmark because when the road network fulfils the first-in-first-out (FIFO) property, which can be enforced, TD-Dijkstra guarantees to find an optimal result. The parameters of the models were calculated with differential evolution. The models were then tested on the graph by summing up the travel times for 10000 random routes. The error was calculated as the difference of the model's results with the total time of TD-Dijkstra. The simplest model performed most reliably: $2.89 \cdot L \cdot T$. On the different tested types of areas - from rural to city-like - this model improves on the relative error of Dijkstra by at least 20%.

Conclusion In summary - for time-dependent routing - Dijkstra requires no additional memory, but it is inaccurate. On the other hand, TD-Dijkstra requires a lot of additional memory. TLPM elucidates this trade-off between memory and accuracy: it is more accurate than Dijkstra while still having low memory consumption. Furthermore, not only is preprocessing linear in function of travel time data, but TLPM is also straightforward to implement. Thus, TLPM provides an alternate view on time-dependent routing and its intricacies.

[1] This 2014 data-set was provided by Be-Mobile for research purposes. At the time of publication, they have already significantly advanced their data-sets.