Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Discrete Mathematics 36 (2010) 487–494
Electronic Notes in DISCRETE MATHEMATICS
www.elsevier.com/locate/endm

Time dependent multiobjective best path for multimodal urban routing

Tristram Gräbener [a,1] Alain Berro [a] Yves Duthen [a]

[a] *IRIT-UT1, UMR 5505, CNRS, Université de Toulouse*
2 rue du doyen Gabriel Marty 31042 Toulouse Cedex 9, France
{`tristram.grabener,alain.berro,yves.duthen`}`@irit.fr`

Abstract

While the fastest path problem has been widely studied with excellent results, little research has been done on the time dependent multiobjective best paths. Applied to multimodal urban routing, this approach offers multiple suggestions adapted to variety of user preferences. We propose a simple model with intersting properties that allows to use traditional algorithms with little modifications. The experimental computation time are acceptable for a real world application.

Keywords: Multiobjective time dependent shortest path, multimodal routing

1 Motivations

With the growth of environmental consciousness and the increasing energy cost, more and more people tend to use public transport or soft means of transport as walking or cycling. However, just one mode of transport can not cover all the needs. Therefore, combining different means of transport can be an interesting possibility in many situations.

[1] PhD student funded by the region *Midi-Pyrénées* and the company *Sodit*

1571-0653/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.endm.2010.05.062

However, every user has different preferences concerning time or money spent, pollution, number of changes, etc. Furthermore, preferences depend on circumstances: if it is raining, the user will not cycle and with heavy luggage, he will avoid changes. *Multiobjective* optimisation gives the user many *equivalent* suggestions where no solution is better on *every* objective.

2 Easy problems

2.1 Fastest route

A recent horse race to get the fastest routing algorithm resulted in impressive results. Computing the shortest path on a road network with constant costs across a continent can be computed in less than a millisecond (see [2] for a survey). Algorithms have been extended to the time dependent problem (time taken on an edge depends on the time) with success [1]. However they focus only on one mode of transport and only consider the time taken as objective, which is an easier problem than having arbitrary costs.

Two approaches are used: the *time-dependent model* and the *time-expanded*. The first keeps the existing topology but uses cost functions instead of constant costs. The second model splits the time in a finite number of time interval and duplicates every node for every time interval. A general trend is to use the time dependent model on road networks and the time expanded in timetable based networks (like public transport). Both approaches have been experimentally confronted on the German railways network in [9]. The authors favor the time-expanded model due easier modelling, while the time dependent model has slightly better performances.

2.2 Multimodal fastest route

Two issues have to be resolved: how to model the multimodal network and how to compute the best path given that network. Considering the modeling of the network, two approaches have been used. The first one builds a layer for every means of transport (*multi-layer* model) and the second has a cost on each edge for every means of transport (*multi-valuated* model). The first model requires a larger graph but can use traditional algorithms, while the second model is a more compact representation but needs adapted algorithms.

An important property to take in account is the Fifo constraint. This condition states that given an edge (u, v), leaving the node u later does not allow to reach the node v earlier. When an user decides to wait for a bus at a

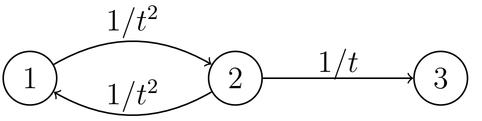

Fig. 1. Infinite mimimum cost path

node instead of walking he will arrive earlier. This problems occurs specially in case of multimodal routing with multivaluated edges.

One of the first work on multimodal routing is [11]. The authors use a multivaluated time-expanded graph with their own algorithm that runs in $\mathcal{O}(q^2n^5)$ to compute the fastest path. This algorithm is able to work even on non-Fifo graph. This allows the authors to merge multiple pedestrian nodes into some significant ones. The number of nodes handled is between 25 and 1000, while the number of time interval range from 50 to 250 for a runtime of the magnitude of some seconds.

In [3] the authors model the acceptable transitions between two modes with an automaton and offer impressive performances. However their approach is restricted to the *fastest* route and defining the automaton in advance will cause the loss of solutions that the user might not have thought about. However, we wonder how that approache based on defining *access points* would behave in a denser environment like a city. Indeed, in the largest instance that covers Europe and north Amercian, the authors only consider 359 airports, while the Parisian area has more than 25000 bus stops. The multimodal part being negligible, the authors use high performance shortest path algorithms on the road network to achieve their excellent results.

3 Hard problems

3.1 Time dependent minimum cost path

When the objective is not the duration, the optimum path might have an inifinite length, even if the costs are strictly positive. The figure 1 shows an example base on [7]; the time taken to walk along an edge is 1. The path of minimum *cost* from 1 to 3 infinitely loops over the nodes 1 and 2 before taking the edge $(2, 3)$ with a total cost of $\pi^2/6$. The authors prove that in order to have a finite optimal path, the cost function must eventually be bounded: $\exists b > 0, T > 0, \forall t > T, C(t) \geq b$.

The *time-expanded* model is able to compute the minimum cost path. In order to avoid the explicit construction the time-space graph, [8] describes the *Chrono-SPT* algorithm that computes the minimum cost path implicitly manipulating the graph and thus saving memory. The complexity of the algorithm is in $\mathcal{O}(m \cdot q)$ where m is the number of edges of the original graph, and q the number of considered time intervals. The authors use the concept of *cost-consistency* to describe acceptable cost functions.

3.2 Multiobjective shortest path

When we want to optimize multiple objective simultaneously, there is rarely a solution that is better on all objectives. Two solutions where the first is better on some objective and the second on the others are said to be *equivalent*. The set of equivalent solutions is called the *Pareto-front*. As the number of elements in the Pareto front grows exponentially with the graph size, so will the runtime of an algorithm computing it. This is why the multiobjective shortest path is a much harder problem than the fastest path. From a decider point of view, providing multiple equivalent solutions is a very interesting approach as there is no need to make any supposition about how he will choose between two equivalent solutions. That is why a Pareto-optimal solution is more interesting than linearizing the objectives into a single one.

The *Martins'* algorithm is an extension of the Dijkstra algorithm to the multiobjective problem. It computes the Pareto-fronts from a single node to every node of the graph. Different authors suggest improvements on that algorithm based on bounds to reduce the search space, in a similar manner to the A* algorithm but with more sophisticated bounds [5]. One work stands out by it computational results ([4]). However, the authors only consider few highly correlated objective (duration, cost and distance). They therefore get only 50 Pareto-optimal solutions on average on a area as big as Europe. A fully polynomial approximation scheme also exists [10], but to with no experimental comparison yet.

To our knowledge, no existing publication considered the problem of time dependent multiobjective paths. Here is an example of edge cost that will result in an infinite number of Pareto elements. Suppose that walking along an edge has a cost of 5 at 8:00 and 2 at 10:00 (an urban toll to reduce the traffic during peak hours). The cost between those two instants is a linear interpolation; in this situation, waiting a tiny moment more, will result in a lower cost and therefore an infinite number of Pareto-optimal solutions. A necessary and sufficient condition to have a finite number of Pareto-optimal solutions is that the cost functions can be split in a finite number of nondecreasing parts.

4 Our model

4.1 The multimodal graph

Every means of transport is represented by one layer in the multimodal graph. Edges connect two layers if it is possible to switch from one layer to another (e.g. from car to foot on car parks) A cost function for each objective is associated to every edge. The cost functions fulfill the conditions presented earlier in order to have a finite number of finite paths.

While this model is trivial, it holds very interesting properties that greatly reduces the complexity of the algorithm running on it: respect the Fifo property; models which modes transitions can be done; no approximation.

4.2 Time dependent Martins' algorithm

The algorithm 2 is a modification of Martins' algorithm. The only is the way to calculate the cost vector of a new label. To simplify the notations, we consider that the first objective is the time. For an edge (u, v), the function $C_{uv} : \mathbb{R} \rightarrow \mathbb{R}^o$ computes the cost vector of the edge at a given time.

A *label* is a tuple of a node, the cost vector and the predecessor label. A label is a representation of a path that can be reconstructed by iteratively following the predecessor label. The o-vector $c(l)$ is the cost of the label l.

The algorithm maintains for every node i two sets of temporary (T_i) and permanent (P_i) labels. The permanent set will hold exclusively Pareto-optimal solutions. For implementations reasons, the predecessor of a label is represented by its node and its rank in the permanent set of the predecessor node.

The algorithm 2 starts from node s at the instant t_0. It iteratively selects the label with the smallest lexical cost, adds it to permanent label set and creates a new label for each successor. The dominated labels are discarded, while the non-dominated labels are added to the temporary set. When all temporary sets are empty, the algorithm stops. As cost functions are such that there is a finite number finite paths in the Pareto front, the algorithm will terminate, but there is no guarantee on the execution time.

4.3 Improvement heuristics

When a path from node s to t is needed, if a label is dominated by any label of P_t, it is discarded. A common heuristic is the *relaxed pareto dominance*;

$P_i \leftarrow \emptyset, T_i \leftarrow \emptyset, \forall i \in \mathcal{N}$; $T_s \leftarrow \{[s, (t_0, \ldots, 0), \_, \_]\}$
while $\bigcup_i T_i \neq \emptyset$ **do**
    $l = [n, c_n, p, r] \leftarrow \arg\min_i T_i$
    $T_n \leftarrow T_n/\{l\}$; $P_n \leftarrow P_n \cup \{l\}$
    for $n', (n, n') \in \mathcal{E}$ **do**
        $l' \leftarrow [n', c_n + C_{nn'}(c_n[1]), n, |T_n|]$; dominated $\leftarrow$ false
        for $l'' \in T_{n'}$ **do**
            If$(c(l') \succ_p c(l''))$: $T_{n'} \leftarrow T_{n'}/\{l''\}$
            ElseIf$(c(l'') \succ_p c(l'))$: dominated $\leftarrow$ true
        If(not dominated): $T_{n'} \leftarrow T_{n'} \cup \{l'\}$

Fig. 2. Time dependent Martins' algorithm

the dominance test is biased in order to eliminate very similar solutions [6,4]. For example a route that lasts two more minutes is not worth considering if the extra cost is 10 euros. Some Pareto-optimal solutions will be lost, but the solutions found are still Pareto optimal. Both approaches reduce the number of considered labels and thus the computation time.

5 Experiments

The goal of this section is to show that our model is fast enough for an actual application. However a comparison with existing is results if difficult. On one hand, there are no similar experiments (existing results are for multiobjective routing with constant costs, consider only few poorly interconnected transport modes or even only compute the fastest route). On the other hand, we favor a simple model that can be used with simple algorithm in over complicated approaches with small computation times, as long as it is acceptable.

5.1 Testing scenario

We used three transport modes: cycling, walking and public transport in San Francisco. The street map is provided by OSM and public transport (Bart and Muni networks) data can be freely downloaded. The user can leave its bike anywhere, but can not use it again later.

We consider up to four different objectives to be minimized: time taken `t`, number of modes switches `ms`, number of lines switches `ls` and positive gain in altitude while cycling `a`. Line switching occurs when changing bus line without moving from the bus stop. The start node is chosen randomly from the cycling layer and the destination node from the pedestrian layer.

	Pareto Optimal		Relaxed dominance		No heuristic
Objectives	Num. of paths	Time	Num. of paths	Time	Time
t + ms	1.2	71.9	1.0	56.8	155
t + ms + ls	1.2	75.5	1.0	61.2	182
t + a	26.9	680	12.7	221	3099
t + a + ms + ls	103	6898	16.6	332	59071

Fig. 3. One-to-one computation time(ms), 36694 nodes, 171443 edges

5.2 Results

The results in figure 3 shows the average number of elements in the Pareto front and the computation time. The results are an average of 10 runs where the source and destination are the same for every test. For every objective the whole Pareto front is computed and compared to the relaxed dominance.

We can see that our model offers sufficient performances for a real world application, excepted maybe with four objectives. We notice that results vary significantly depending on the objectives considered. Indeed the computation time with two objective can be significantly longer than with three objective when the objectives are not correlated (like gain in altitude and duration). When dealing with multiobjective routing, it is very important to think about the nature of the objectives. The relaxed dominance allows significant speed-ups when the objectives generate many equivalent solutions.

Due to lack of space, we don't provide computational results. However, we showed that the simple heuristics can result in an improvement up to a factor 150 in time compared to Martins' algorithm. Also modelling plays a very important role. If we restrict mode changes at some nodes (the bike can only be left at public transport nodes), then the computation time reduced to 100ms on instances that were not tractable in 10 minutes.

6 Conclusion and perspectives

Our performances might seem poor compared to the state of the art fastest path algorithms (time dependent or multimodal). We remind that those are computational easy problems compared to the multiobjective best path. As far as we know, our results are the only one that compute the time dependent multiobjective shortest path. We insist on the fact that our approach doesn't make any approximation and doesn't suppose any preference of the user.

The performances are sufficient to provide a urban route planing tool. However, a strong focus on the performances will be needed to grow in objectives and considered area.

References

[1] Batz, V., D. Delling, P. Sanders and C. Vetter, *Time-dependent contraction hierarchies*, in: *Proceedings of the 11th Workshop on Algorithm Engineering and Experiments (ALENEX09)*, 2009, pp. 97–105.

[2] Bauer, R., D. Delling, P. Sanders, D. Schieferdecker, D. Schultes and D. Wagner, *Combining hierarchical and goal-directed speed-up techniques for Dijkstra's algorithm*, Lecture Notes in Computer Science **5038** (2008), p. 303.

[3] Delling, D., T. Pajor and D. Wagner, *Accelerating Multi-Modal Route Planning by Access-Nodes*, in: *Proceedings of the 17th Annual European Symposium on Algorithms (ESA09), Lecture Notes in Computer Science*, Springer, 2009.

[4] Delling, D. and D. Wagner, *Pareto Paths with SHARC*, , **5526**, Springer, 2009, pp. 125–136.

[5] Mandow, L. and J. de la Cruz, *A New Approach to Multiobjective A ˆ* Search*, , **19**, LAWRENCE ERLBAUM ASSOCIATES LTD, 2005, p. 218.

[6] Muller-Hannemann, M., F. Schulz, D. Wagner and C. Zaroliagis, *Timetable information: Models and algorithms*, Lecture Notes in Computer Science **4359** (2007), p. 67.

[7] Orda, A. and R. Rom, *Minimum weight paths in time-dependent networks*, Networks **21** (1991), pp. 295–319.

[8] Pallottino, S. and M. Scutella, "Shortest path algorithms in transportation models: classical and innovative aspects," Kluwer Academic Publishers, 1998.

[9] Pyrga, E., F. Schulz, D. Wagner and C. Zaroliagis, *Experimental comparison of shortest path approaches for timetable information*, in: *Proceedings of the Sixth Workshop on Algorithm Engineering and Experiments*, 2004, pp. 88–99.

[10] Tsaggouris, G. and C. Zaroliagis, *Improved FPTAS for multiobjective shortest paths with applications*, CTI Techn. Report TR-2005/07/03 (2005).

[11] Ziliaskopoulos, A. and W. Wardell, *An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays*, European Journal of Operational Research **125** (2000), pp. 486–502.