20th EURO Working Group on Transportation Meeting, EWGT 2017, 4-6 September 2017, Budapest, Hungary

Stochastic Route Planning in Public Transport

Kristóf Bérczi [a,b], Alpár Jüttner [a,b], Marco Laumanns [c], Jácint Szabó [c,*]

[a]MTA-ELTE Egerváry Research Group, Pázmány Péter sétány 1/C, Budapest 1117, Hungary
[b] Department of Operations Research, Eötvös Loránd Tudományegyetem, Pázmány Péter sétány 1/C, Budapest 1117, Hungary
[c]IBM Research - Zurich, Säumerstrasse 4, CH–8803 Rüschlikon, Switzerland

Abstract

Journey planning is a key process in public transport, where travelers get informed how to make the best use of a given public transport system for their individual travel needs. A common trait of most available journey planners is that they assume deterministic travel times, but vehicles in public transport often deviate from their schedule. The present paper investigates the problem of finding journey plans in a *stochastic environment*. To fully exploit the flexibility inherent in multi-service public transport systems, we propose to use the concept of a *routing policy* instead of a linear journey plan. A policy is a state-dependent routing advice which specifies a set of services at each location from which the traveler is recommended to take the one that arrives first. We consider *current time dependent policies*, that is, when the routing advice at a given location is based solely on the current time. We propose two heuristic solutions that find routing policies that perform better than deterministic journey plans. A numerical comparison shows the achievable gains when applying the different heuristic policies based on extensive simulations on the public transport network of Budapest. The results show that the probability of arriving on time to a given destination can be significantly improved by following a policy instead of a linear travel plan.

Keywords: public transport; routing policy; stochastic route planning; utility function

 * *E-mail addresses:* berkri@cs.elte.hu (Kristóf Bérczi), alpar@cs.elte.hu (Alpár Jüttner), mlm@zurich.ibm.com (Marco Laumanns), JSZ@zurich.ibm.com (Jácint Szabó)

1. Introduction

Transport infrastructures of large cities are becoming overwhelmed by the growing number of private vehicles. As the use increases, traffic congestion becomes increasingly prevalent, causing significant losses to the economy. Public transport can play an essential role in reducing the traffic load. To facilitate the use of a public transport system, journey planners have been deployed that help passengers to find a route satisfying their needs. When given the start and the destination location of the journey, these applications offer one or multiple alternative routes - i.e. sequences of journey legs in transportation vehicles, potentially connected by interchange activities – that bring the traveler from the start to the destination based on the schedule of all transportation services available.

A common weakness of this approach is that it assumes a deterministic environment. However, disturbances due to congestion as well as unplanned events such as accidents or maintenance work can have strong effects on travel, and when public transportation vehicles are behind or before schedule, the precomputed deterministic journey plan might become infeasible, e.g. because of a missed connection. Motivated by this, we consider a stochastic environment in which travel times follow presumed probabilistic distributions.

In many cases, there are multiple equivalent services which a passenger can choose from, depending on which one comes first; such situations are difficult to handle by a linear journey plan. Public transport providers are trying to take advantage of such options and offer dynamic journey planning capabilities, for example by offering push services to alert travelers of broken connections. However, linear journey plans are not able to capture the full amount of flexibility inherent in multi-service public transport systems as the re-planning always occurs only after some deviation from the original conditions has occurred.

We therefore propose a novel approach that makes use of this flexibility. Instead of linear journey plans, we introduce the notion of a **routing policy**, which is a time-dependent routing advice at each location. A routing advice may consist of more than one service and the passenger is advised to take whichever comes first. Our aim is to determine a routing policy which performs better under uncertainty than deterministic solutions. The performance of a policy is measured in terms of **expected utility**. Travelers may define an arrival time dependent utility value at the destination representing their preferences. The goal is thus to compute a policy that maximizes the expected value of the utility that can be achieved by following the routing advice.

The aim of the present paper is to examine current time dependent policies with multiple service choices for monotone non-increasing utility functions. The monotonicity assumption is plausible as it is always possible to wait at a given stop when using public transportation. The paper is organized as follows. The problem together with basic notation is introduced in Section 3. Besides showing that an optimal policy may not always exist, two heuristic solutions for finding a 'good' current time dependent policy are proposed. Section 4 gives an evaluation of the heuristic solutions by comparing them to deterministic routing plans through extensive simulations, and Section 5 gives some conclusions from the findings and outlook to open questions

2. Previous work

The classical algorithms of Dijkstra (1959), Bellman (1958), Ford (1956) and Floyd (1962) are fundamental methods for route planning. Improved versions, like the A* algorithm of Hart et al. (1968), serve as the basis of many widely used route planning applications. All these methods work with deterministic arc costs and provide a single route as a solution.

One way to add uncertainty to the problem is to model the travel times on the arcs as independent random variables with given probability distributions. In this model both the mean and the variance of the total travel time may serve as performance measures of routes. For time-independent travel time distributions, both problems can be reduced to a standard shortest path problem. Hall (1986) showed that for time-dependent distributions this is not the case. He also introduced a time-adaptive decision rule in which the next node is defined for each step as a function of the arrival time at the node.

Another generalization is given when the effectiveness of a route is defined by a utility function depending on the arrival time. Only a few exact results exist in the literature for this problem due to the computational difficulty of determining the expected utility. Loui (1983) provided a list of utility functions for which an optimal solution can be efficiently found. Frank (1969) defined an optimal path as a path maximizing the probability of arriving before a

given deadline and called this the stochastic on-time arrival (SOTA) problem. Nikolova et al. (2006) characterized a family of realistic models and travel time distributions that are closed under convolution.

Another line of research focuses on finding optimal stochastic journey plans. Polychronopoulos et al. (1996) considered a recourse model for stochastic shortest paths. Fan et al. (2005) applied stochastic dynamic programming to find on-line travel plans, where the next arc is computed depending on the travel times on the path already visited. The SOTA problem was reformulated in terms of stochastic dynamic programming by Fan et al. (2006). Results on the SOTA problem were theoretically and numerically further improved by Samaranayake et al. (2012).

In contrast to the road transport case, multi-modality and interchanging between transport services is a central concept in public transport. Multi-modal public transport journey planning in the presence of uncertainty was investigated by Botea et al. (2013) and Dibbelt et al. (2014). In their model uncertainty appears as times of arrival of transport services. As there are typically multiple services serving the same stop, it is a natural extension to recommend for each node a set of alternative services, instead of just a single service, so that the passenger would take the service from the list that arrives first. This model was introduced by Nonner (2012) and is the starting point of the present paper. Bérczi et al. (2016) further developed the notion of travel alternatives by introducing time dependent journey plans consisting of a set of services, in order to make the resulting traveling advice more robust to uncertainty.

3. Model development

When defining the mathematical model of the public transport system, we use a simplification: we assume throughout that the system consists of individual services between two nodes, meaning that each service has only two stops. Although this simplified model gives only a very rough description of real public transport systems, it allows us to use a simpler notation and description of the algorithms. However, the proposed algorithms can be adapted to the more general case when services have several stops. According to this, the computational results of Section 4 are based on the data of a real-life public transport, namely on the General Transit Feed Specification (GTFS) data of the Budapest public transport network (Centre for Budapest Transport (2017)).

In our stochastic model, the public transport network is represented as follows. The **set of stops** is associated with a set of nodes V, while the **set of services** is denoted by S. Each service s has an **origin** and a **destination node** denoted by o_s and d_s, respectively. The set of services with origin node v is denoted by S_v. The **network** is then represented by a directed graph $D(V,A)$, where A is a **set of directed edges** with vw being in A if and only if there exists a service s with $o_s=v$ and $d_s=w$.

The model considers the probability distributions of arrival, departure and travel times. To formalize this mathematically, let s be a service. The random variable of the **departure time** of s from o_s is denoted by D_s, the random variable of the **arrival time** of s to d_s is denoted by A_s, and the random variable of the **travel time** of s from o_s to d_s is denoted by T_s. Clearly, these random variables are not independent since : $A_s=D_s+T_s$.

For a given random variable X, its **cumulative distribution function** (CDF) is denoted by $F_X(t)$ and its **probability density function** (PDF), if exists, by $f_X(t)$. **The indicator variable** of some logical statement is denoted by χ, for example $\chi_{(x \in X)}$ is 1 if $x \in X$ and 0 otherwise.

Now we are ready to give a definition of a **policy**. Informally, a routing policy is a routing advice that assigns a set of services to each stop and time. As we are considering current time dependent policies, the set of recommended services is continuously updated and only depends on the actual time. Formally, a policy is a function $p: V \times \mathbb{R} \rightarrow 2^S$ (where $\mathbb{R}$ stand for the set of real numbers) so that $p(v,t) \subseteq S_v$. Policies can be interpreted as follows. If a passenger is at stop v in time t, then he is recommended to take any service from $p(v,t)$ if such a service departs at time t (if multiple services are leaving simultaneously, ties are broken uniformly at random). The **set of possible policies** is denoted by P.

The quality of a routing policy will be measured in terms of a utility function, provided by the passenger. A **utility function** is a function $u: \mathbb{R} \rightarrow \mathbb{R}$ that represents the passenger's utility depending on the time of arrival at the destination. It is natural to assume that u is monotone non-increasing as it is always possible to wait at a given node and simply let time pass. We further assume that there is a **deadline** T on the time horizon after which the utility function has value $-\infty$. Given a policy p, the **utility induced by p in stop v at time t** is denoted by $u_p(v,t)$. That is,

$u_p(v,t)$ denotes our expected utility supposing we are in stop v at time t and we continue our journey according to the policy p. The **conditional utility induced by p and service s at stop v and time t** is denoted by $u_p(v,t,s)$ and is defined as our expected utility if we take service s in stop v at time t and continue our journey according to the policy p.

When considering continuous time, the induced utility can be computed as

$$u_p(v,t) = \int_t^\infty \left(\sum_{s \in p(v,t)} u_p(v,t',s) \cdot \prod_{s' \in S_v, s' \neq s} \left(1 - \int_t^{t'} f_{D_{s'}}(t'') \chi_{(s' \in p(v,t''))} dt'' \right) \right) dt' \qquad (1)$$

where the induced utility is given by

$$u_p(v,t,s) = \int_0^\infty u_p(d_s, t+t') f_{T_s}(t') dt'. \qquad (2)$$

3.1. Existence of optimal policies

A natural way to define optimality of a policy is the following. Policy $p \in P$ is called **optimal** if $u_p(v,t) \geq u_{p'}(v,t)$ for every $p' \in P$, $v \in V$ and $t \in \mathbb{R}$. This property is usually fulfilled by optimal policies in Markov decision processes. It would be desirable to find such an optimal policy, but the next example shows that such a policy does not necessarily exist.

Assume that the network consists of two services s_1 and s_2 going from stop v to w. Service s_1 departs in time interval *[0,1]* at uniformly at random while s_2 departs in time interval *[1-ε,2]* uniformly at random. The travel times of s_1 and s_2 are *1* and *1+ε*, respectively. The utility function u in node w is defined as

$$u(t) = \begin{cases} 2, & \text{if } t < 2, \\ 1, & \text{if } 2 \leq t < 3, \\ 0 & \text{otherwise.} \end{cases} \qquad (3)$$

We claim that there exists no optimal policy in the above sense. Indeed, a policy p such that $s_2 \notin p(v,t)$ for some *1-ε $\leq t \leq 1$* cannot be optimal as it may happen that s_1 already left stop v and s_2 departs at time t. In this case the passenger would be advised not to take service s_2, and so the realized utility would be $-\infty$ since no arrival at the destination would be possible at all. That is, if there exists an optimal policy p, then necessarily $s_2 \in p(v,t)$ for any *1-ε $\leq t \leq 1$*. This immediately implies that $u_p(v,0) < 2$. However, the policy $p'(v,t) = \{s_1\}$ for *0 $\leq t \leq 2$* induces $u_{p'}(v,0) = 2$, contradicting the optimality of p.

3.2. Heuristics

The previous example shows that we may not expect to find a policy which is optimal 'globally'. Still, as policies should give a reasonable and natural generalization of deterministic journey plans, it would be interesting to give some heuristic solution which in general provides higher expected utility than deterministic solutions. In the following, we develop two procedures to find such a heuristic solution.

3.2.1. Weak induced policy

The formal definition of a **weak induced policy** is as follows. By going back in time (starting from the deadline T), for each time t, node v and service $s \in S_v$ do the following:

1.   Let *p'* be the policy defined as *p'(w,t')=p(w,t')* for *t'>t* and *w∈V*, and let *p'(v,t)={s}*.
2.   Compute *u(v,t)*, and add *s* to *p(v,t)* if $u_{p'}(v,t) \geq \lim_{t' \to t^+} u_p(v,t')$.

In other words, the concept of weak policies is the following. When determining *p(v,t)* for a given node *v* and time *t*, we compute the expected utility in *v* for time *t'>t* induced by the partial policy determined so far. Note that this value would be our expected utility if our policy simply was to stay in node *v* for time *t*. Then for each service *s∈S_v*, we compare this value to the expected utility that we get if we take service *s* at stop *v* at time t and we continue our travel according the policy determined so far. A service s is added to policy *p(v,t)* if $u_p(v,t) \geq \lim_{t' \to t^+} u_p(v,t')$.

The reason for calling the obtained policy 'weak' is that in some sense this is the naivest approach for defining a policy: a service is recommended for the passenger if taking it induces better expected utility then just staying at the stop and doing nothing.

3.2.2. Strong induced policy

Now we turn to a more sophisticated method and give a definition of **strong induced policies**. By going back in time (starting from the deadline *T*), for each time t and node v do the following:

1.   For each subset *S' ⊆S_v* of services, let *p'* be the policy defined as *p'(w,t')=p(w,t')* for *t'>t* and *w∈V*, and let *p'(v,t)=S'*.
2.   Compute *u_p'(v,t)* and let *X* be the subset of *S_v* for which this value is maximal.
3.   Set *p(v,t):=X*.

The main difference between weak and strong induced policies is that in the latter case policy *p(v,t)* is chosen as the best one among certain subsets of *S_v* while in the weak case services were only compared to a given value one by one. Hence it is natural to expect that strong policies provide better expected induced utility values than the weak induced policy.

4. Computational case study and validation

In this section, we present computational results obtained by the different policies defined in the previous section. The computations are based on the General Transit Feed Specification (GTFS) data of the Budapest public transport Network (see Centre for Budapest Transport (2017)). The algorithms are implemented in C++ with heavy use of the open source LEMON graph optimization library (see Dezső et al. (2011)). The code is platform independent, the actual tests were performed on a Linux-based server with a 3.50 GHz Intel Core i7-5930K CPU and 32 GB RAM.

4.1. Extending the model for the computational experiments

For the computational case study, we considered a discretization of the problem. The time horizon was divided into smaller intervals indexed by 1, 2,…, T (the set of the intervals is denoted by [T]), and every event was assumed to happen in one of these intervals. For the description and analysis of the algorithms above, we have assumed without loss of generality a simplified model where the services were given as simple trips between two nodes without intermediate stops. For the computational case study, we use a more complex and realistic model which, according to the GTFS data format, considers services with several trips (following each other in a prescribed order). A service has several trips consisting of several hops, hence the definition of policy must be modified accordingly. In this case, a policy is a function *p:V ×[T]→2^{S×V}* such that if *(s,w)∈p(v,t)* then
* *s∈S_v*,
* *w≠v* is a stop of service *s* which is after *v*,
* *(s,w')∉p(v,t)* for each stop *w'≠w* of service *s*.

Here, *(s,w)∈ p(v,t)* means that a passenger is advised to take service *s* at stop *v* and time *t*, if it arrives, and to alight from it at stop *w*.

4.2. Data preprocessing and generation of the stochastic model

In order to prepare the necessary input data, the raw GTFS data is preprocessed in several steps. In a first step, irrelevant services are discarded so that only services operating on the date of travel are considered. In a second step the list of services is filtered according to the given passenger preferences, only keeping those services that theoretically may play role in the journey. Simultaneously, a deterministic shortest path is computed between the start and the destination node as a reference solution.

Next, a probabilistic model is set up for the arrival, departure and travel time distributions. The model may be based on the assumption of a simplified distribution (e.g., Gamma distribution for travel times), but can also take into account GTFS real-time data and statistics obtained from real-life measurements.

4.3. Policy generation

As a starting step, an optimal deterministic solution is computed that has the highest utility in the time period defined by the passenger and has the latest departure time from the starting node. This is later used as a reference solution by the method to find a routing policy. Then, a dynamic programming procedure is used to determine the induced utility of being at stop *v* at time *t* for each node and time.

In the main part of the procedure, a queue of 'active' nodes is maintained, which is initialized with the destination node v_d. At the beginning, only v_d is marked as active. In the following iterations, assume that policies and induced utilities are computed for all *v∈V* and *t'>t*. The top node *v* is removed from the queue, and for each service s∈S_v, time *t∈[T]* and stop *w* preceding *v* on the route of *s* we update $u_p(w,t)$ and $u_p(w,t,s)$ according to the policy computation algorithms specified in the previous section, and also change *p(w,t)* if necessary. If an improvement is found for some triple *(s,t,w)*, then stop *w* is pushed to the end of the queue and is marked as 'active'. This Bellman-Ford-like algorithm computes the induced utilities for each stop and each time. The algorithm terminates when no improvement is found or an upper bound on the number of iterations is reached.

4.4. Test instance and results

The above procedure was applied to the GTFS data of the Budapest public transport network. This timetable consists of 5290 stops and 135.959 services. We considered three types of policies:
- deterministic solution given by the Bellman-Ford algorithm,
- weak induced policy,
- strong induced policy.

It is not straightforward how to measure the performance of a policy in a stochastic environment. As the introduction of policies versus simple routes was motivated by the lack of reliability of deterministic journey plans, we measure the performance of a policy by comparing it to the usual deterministic approach. We chose an arbitrary stop *v* as a destination node from the map of Budapest uniformly at random, and determined the above mentioned three policies for each stop different from *v* for a given date and time. We considered the SOTA (Stochastic On Time Arrival) problem in all cases, that is, the utility function at the destination *v* was defined to be 1 until a certain deadline *T* until which the passenger wishes to arrive, and *0* afterwards. The policies thus obtained were tested by running 100 simulations for each node. In each instance of the simulation, we assumed that the departure times follow normal distributions with expected value equal to the deterministic departure time of the corresponding service. The distribution of the travel times was simulated by a simple formula depending on the deterministic travel time and some other parameters such as service-dependent parameters, time of day, number of hops, etc. This method is fast and easy, and also correlation may be built into the model.

For each node *w* different from *v* and each time *t∈[T]*, we checked whether a passenger leaving from node *w* at time *t* and following either the deterministic solution or the weak or the strong induced policy succeeds to get to *v* in time. For each of the three policies and each node *w* we computed the latest time instance when a passenger has to

leave from *w* in order to arrive to *v* before the deadline with probability at least 90%. These values can be compared to each other for the three solutions, hence giving a performance measure for evaluating their effectiveness. In addition, it enables us to introduce the notion of **safety margin,** which tells us how much earlier we should start compared to the latest deterministic departure so that we achieve a 90% probability of arriving in time in the stochastic environment.

Figure 1 provides two illustrative examples of the achievable probability of arriving in time for different departure times and policies from a fixed stop to a destination node. As an optimal current time dependent policy might not exist in general, it may happen that the deterministic journey plan has better expected utility after a certain departure time than a heuristic solution.
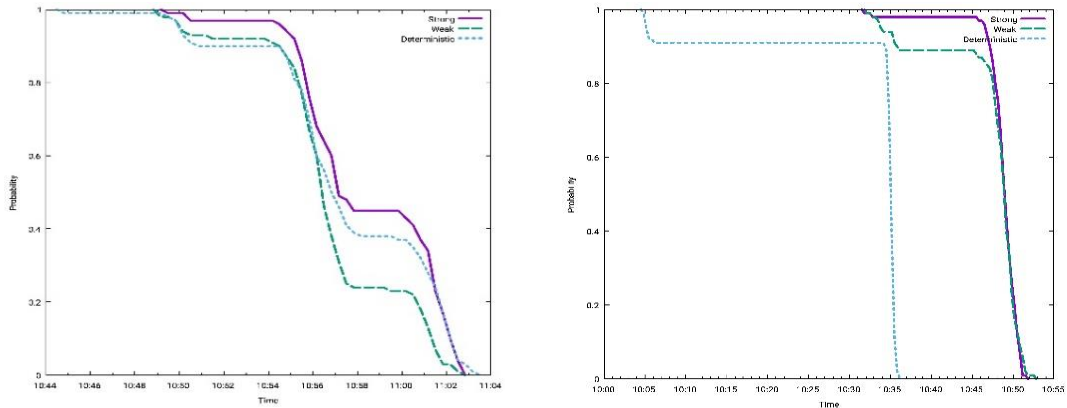

Figure 1. Probability of arriving on time vs. departure time

Figure 2 compares the deterministic solution and the weak induced policy for a given destination node *v*. Each dot on the scatter plot represents a starting node *w* different from *v*. The *x* coordinate of a dot is the time when a passenger should leave from *w* in order to arrive to *v* in time with probability at least 90% when following the route of the weak policy, while its *y* coordinate is defined similarly based on the deterministic journey plan.
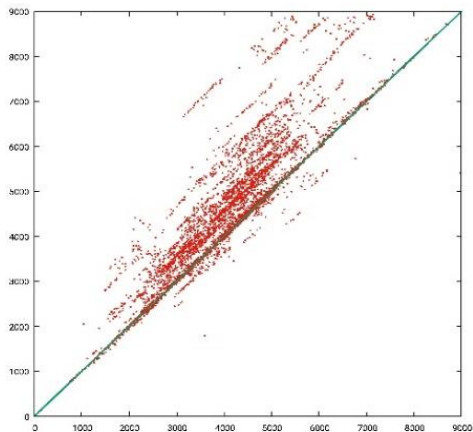

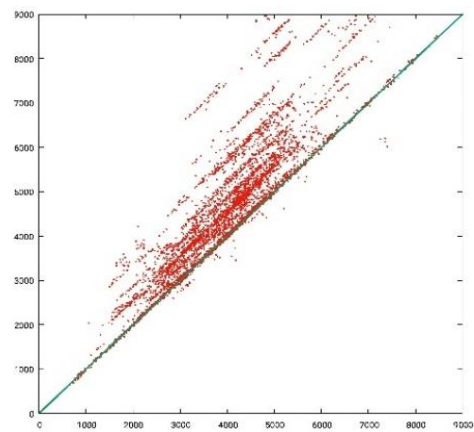Figure 2. Weak policy vs. deterministic route   Figure 3. Strong policy vs. deterministic route

Figure 3. compares the deterministic journey plan with the strong induced policy in the same way. In both cases, dots above the diagonal *y=x* corresponds to stops from where one may leave later when following the induced policy instead of the deterministic route. Note that in general, the stronger solution is only marginally better than the weak one, but it can eliminate basically all cases in which the deterministic route is better than the weak policy.

Figure 4. shows the average safety margin of the methods as a function of the deterministic travel time. The average is taken from computing the policies from all possible starting positions to 4 representative destinations chosen from various parts of the city.
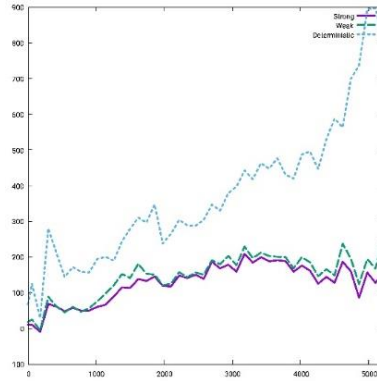

Figure 4. Average safety margin

5. Conclusion

We presented a public transport journey planning approach that can cope with the typically non-deterministic public transport environment. At each location, a policy can prescribe multiple services from which the traveler is advised to take any one depending on which one arrives first, and this selection can even depend on the remaining travel time. Instead of deterministic travel-, arrival- and departure times, our model uses probability distributions, which results in a more robust and on average more effective journey plan. Our approach can handle any passenger-defined monotone non-increasing utility functions, which allows to encode sophisticated passenger preferences beyond just 'arriving in time' or 'as fast as possible'.

References

Bellman, R., 1958. On a routing problem. Quarterly of applied mathematics, 16(1), 87-90.

Bérczi, K., Jüttner, A., Korom, M., Laumanns, M., Nonner, T., Szabó, J., 2016. Stochastic route planning in public transport. U.S. Patent US 9482542 B2.

Botea, A., Nikolova, E., Berlingerio, M., 2013. Multi-modal journey planning in the presence of uncertainty. Proceedings of International Conference on Automated Planning and Scheduling.

Centre for Budapest Transport, 2017. GTFS developer info, http://www.bkk.hu/en/developers/ [Online; accessed 6-March-2017].

Dezső, B., Jüttner, A., Kovács. P., 2011. Lemon – an open source C++ graph template library. Electronic Notes in Theoretical Computer Science 264(5) 23–45. URL http://lemon.cs.elte.hu.

Dibbelt, J., Strasser, B., Wagner, D., 2014. Delay-robust journeys in timetable networks with minimum expected arrival time. In OASIcs-OpenAccess Series in Informatics (Vol. 42).

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numerische Mathematik, 1(1), 269-271.

Fan, Y.Y., Kalaba, R.E., Moore, J.E., 2005. Arriving on time. Journal of Optimization Theory and Applications, 127(3), 497-513.

Fan, Y., Nie, Y., 2006. Optimal routing for maximizing the travel time reliability. Networks and Spatial Economics, 6(3), 333-344.

Floyd, R.W., 1962. Algorithm 97: shortest path. Communications of the ACM, 5(6), 345.

Ford Jr, L.R., 1956. Network flow theory (No. P-923). Rand Corp Santa Monica, CA.

Frank, H., 1969. Shortest paths in probabilistic graphs. Operations Research, 17(4), 583-599.

Hall, R.W., 1986. The fastest path through a network with random time-dependent travel times. Transportation science, 20(3), 182-188.

Hart, P.E., Nilsson, N.J., Raphael, B., 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE transactions on Systems Science and Cybernetics, 4(2), 100-107.

Loui, R.P., 1983. Optimal paths in graphs with stochastic or multidimensional weights. Communications of the ACM, 26(9), 670-676.

Nikolova, E., Brand M., Karger D. R, 2006. Optimal route planning under uncertainty. Proceedings of International Conference on Automated Planning and Scheduling, 31-141.

Nonner, T., 2012. Polynomial-time approximation schemes for shortest path with alternatives. In European Symposium on Algorithms, 755-765.

Polychronopoulos, G.H., Tsitsiklis, J.N., 1996. Stochastic shortest path problems with recourse. Networks, 27(2), 133-143.

Samaranayake, S., Blandin, S., Bayen, A., 2012. A tractable class of algorithms for reliable routing in stochastic networks. Transportation Research Part C: Emerging Technologies, 20(1), 199-217.