

Bicriterion shortest paths in stochastic time-dependent networks

Lars Relund Nielsen*

Department of Genetics and Biotechnology, University of Aarhus, P.O. Box 50, DK-8830 Tjele, Denmark, lars@relund.dk.

Daniele Pretolani

Department of Sciences and Methods of Engineering, University of Modena and Reggio Emilia, Via Amendola 2, I-42100 Reggio Emilia, Italy, daniele.pretolani@unimore.it.

Kim Allan Andersen

Department of Business Studies, University of Aarhus, Fuglesangs Allé 4, DK-8210 Aarhus V, Denmark, kia@asb.dk.

Abstract

In recent years there has been a growing interest in using stochastic time-dependent (STD) networks as a modelling tool for a number of applications within such areas as transportation and telecommunications. It is known that an optimal routing policy does not necessarily correspond to a path, but rather to a time-adaptive strategy. In some applications, however, it makes good sense to require that the routing policy should correspond to a loopless path in the network, that is, the time-adaptive aspect disappears and a priori route choice is considered.

In this paper we consider bicriterion a priori route choice in STD networks, i.e. the problem of finding the set of efficient paths. Both expectation and min-max criteria are considered and a solution method based on the two-phase method is devised. Experimental results reveal that the full set of efficient solutions can be determined on rather large test instances, which is in contrast to the time-adaptive case.

Keywords: Stochastic time-dependent networks, bicriterion shortest path, two-phase method, computational analysis.

1 Introduction

Recently there has been a growing focus on stochastic time-dependent networks¹ (*STD networks*) which often provide a better modelling tool in e.g. transportation applications [3, 9, 10, 12]. Here travel times are modeled as

*Corresponding author

¹Also known as random time-dependent networks, stochastic time-varying networks or stochastic dynamic networks.

random variables with time-dependent distributions. Particular cases, such as non-stochastic time-dependent networks (see [5] for a recent overview) and time-independent stochastic networks (see [15]) have been widely studied, but are not considered here.

STD networks were first addressed by Hall [4] who pointed out several ways to formulate the route selection problem in STD networks. If the driver is allowed to react to revealed (actual) arrival times at intermediate nodes, the best route is not necessarily a path, but rather a time-adaptive *strategy* that assigns optimal successor arcs to a node as a function of leaving time. This is referred to as *time-adaptive route choice*. If a *loopless path* must be specified before travel begins, and no deviations from the route are permitted, the path is selected *a priori* on the basis of only the probability distributions of the arc travel-times. This is referred to as a *priori route choice* and may be useful for routing highly sensitive substances for which the path travelled must be preapproved. The problem of finding a minimum expected travel time path under a priori route choice is NP-hard [14].

It is quite obvious that multicriteria a priori route choice in STD networks is a relevant and difficult problem. However, only two papers exist on the subject. Miller-Hooks and Mahmassani [6] consider bicriterion a priori route choice in discrete STD networks, the objectives being minimizing expected travel time and cost. A label-correcting procedure is described, which guarantees that all the efficient paths can be obtained. Computational results are presented on a single road network. Chang, Nozick, and Turnquist [1] consider multicriterion a priori route choice in a continuous time STD network, where travel times are normally distributed. They devise a heuristic method based on the first two moments of the distributions, where an approximate stochastic dominance criterion is adopted to compare paths. Computational results are presented on an example network and a single road network.

In this paper we consider bicriterion route choice problems in STD networks under a priori route choice. More specifically we consider the problem of finding the set of efficient paths between an origin and a destination node, when leaving the origin at time zero. We assume that departure times are integer and that travel times are discrete random variables. The paper differs from previous work in the following aspects:

1. We propose a new algorithm using the two-phase method to determine the set of efficient paths as opposed to the labelling approach proposed by Miller-Hooks and Mahmassani [6].
2. We perform a reasonably wide computational experience on grid networks, where we address the case of two cost criteria, besides the (somehow easier) time/cost case; this allows us to evaluate the effect of uncorrelated and correlated cost criteria; furthermore, we consider both expectation and min-max criteria.
3. In our computational setting we do not consider a “steady state” with deterministic travel times at the end of a single peak period in contrast to [6]. During the travel, times are stochastic and time dependent, and several peak periods are encountered.
4. Since our algorithms solve the bicriterion problem exactly on the set of

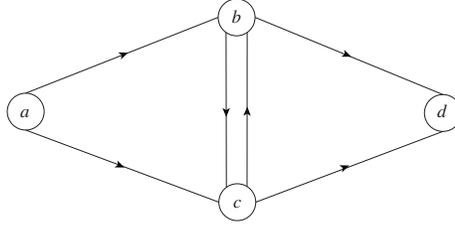


Figure 1: The topological network G .

instances addressed here, we are able to compare the nondominated points found under a priori and time-adaptive route choice.

The paper is organized as follows. In Section 2 we briefly introduce STD networks. In Section 3 we give a short description of the two-phase method, and describe the procedure we use for its implementation. Computational results are reported in Section 4.

2 Stochastic time-dependent networks

Let $G = (N, A)$ be a directed graph, referred to as *the topological network* and let $o, d \in N$ denote two different nodes which represent the *origin* and the *destination* node in G , respectively. Throughout this paper we consider routing from o towards d when leaving node o at time zero.

Departure and arrival times belong to a finite *time horizon*, i.e. a set $H = \{0, 1, \dots, t_{\max}\}$. Let $X(u, v, t)$ denote the arrival time at node v when leaving node u at time t along arc (u, v) which is a discrete random variable with density

$$\Pr(X(u, v, t) = t_i) = \theta_{uvt}(t_i), \quad t_i \in I(u, v, t)$$

where $I(u, v, t) = \{t_1, \dots, t_{\kappa(u, v, t)}\}$ denotes the set of $\kappa(u, v, t)$ possible arrival times at node v . We assume that travel times are positive, and no waiting is allowed in the nodes. For the situation where waiting is allowed see [11].

Costs are considered in the model by letting $c(u, v, t)$ denote the *travel cost* of leaving node u at time t along arc (u, v) and $g(t)$ the *penalty cost* of arriving at node d at time t .

A *strategy* S is a function which provides routing choices for travelling from o at time zero towards d . That is, if $S(u, t) = (u, v)$ a traveller leaving node u at time t travels along arc (u, v) . Note that a strategy must provide routing choice for all possible arrival times at an intermediate node. Under time-adaptive route choice finding the best route with respect to some criterion corresponds to finding the best strategy. Under a priori route choice we must travel along a loopless path in G , that is, we must adopt a *path-strategy*, where the successor arcs of a node for different leaving times are time-independent. From now on, we shall identify paths with path-strategies. Let us denote by \mathcal{S} the set of all strategies and with \mathcal{S}_P the set of all path-strategies. Clearly, $\mathcal{S}_P \subset \mathcal{S}$.

Example 1 Consider the topological network in Figure 1, where a is the origin node and d is the destination node. For each arc in G , the possible departure

$(u, v), t$	$(a, b), 0$	$(a, c), 0$	$(b, c), 1$	$(b, c), 3$	$(b, d), 1$
$I(u, v, t)$	$\{1, 3\}$	$\{2\}$	$\{2\}$	$\{4, 5\}$	$\{3\}$
$c(u, v, t)$	$(1, 1)$	$(3, 0)$	$(2, 2)$	$(0, 4)$	$(3, 8)$
$(u, v), t$	$(b, d), 3$	$(c, b), 2$	$(c, d), 2$	$(c, d), 4$	$(c, d), 5$
$I(u, v, t)$	$\{6, 7\}$	$\{3\}$	$\{3, 4\}$	$\{5, 6\}$	$\{6\}$
$c(u, v, t)$	$(3, 6)$	$(2, 1)$	$(4, 2)$	$(3, 3)$	$(1, 5)$

Table 1: Input parameters.

and arrival times are listed in Table 1. Here a pair $((u, v), t)$ corresponds to a possible leaving time t from node u along arc (u, v) . For the sake of simplicity, we assume that $X(u, v, t)$ has a uniform density, i.e., for each $t' \in I(u, v, t)$, we have $\theta_{uvt}(t') = 1/|I(i, j, t)|$. For example, if we leave node b at time 3 along arc (b, c) , we arrive at node c at time 4 or 5 with the same probability $1/2$. Two possible strategies are

$$\begin{aligned}
S_1 : S_1(a, 0) &= (a, b), S_1(b, 1) = (b, d), S_1(b, 3) = (b, d); \\
S_2 : S_2(a, 0) &= (a, b), S_2(b, 1) = (b, d), S_2(b, 3) = (b, c), \\
&S_2(c, 4) = (c, d), S_2(c, 5) = (c, d).
\end{aligned}$$

Strategy S_1 is a path-strategy and corresponds to the path $a - b - d$ while for strategy S_2 we travel different routes depending on the leaving time from node b . \blacksquare

In this paper we assume that two values are associated with a strategy, namely travel time and cost, where cost is considered in general terms, e.g., a risk measure or the economic travel cost. Furthermore, different criteria are considered, namely, *expectation criteria* (minimize expected travel time or cost) and *min-max criteria* (minimize maximum possible travel time or cost). The value associated with a strategy according to these criteria can be formally defined by means of sets of recursive equations, see Pretolani [14]. Given a strategy $S \in \mathcal{S}_P$ let $W(S) = (W_1(S), W_2(S))$ denote the 2-dimensional vector, where $W_i(S)$, $i = 1, 2$ is the value associated with S with respect to one of the above four criteria. In this paper we face the following problem:

$$\begin{aligned}
\min \quad & W(S) = (W_1(S), W_2(S)) \\
\text{s.t.} \quad & S \in \mathcal{S}_P.
\end{aligned} \tag{1}$$

That is, we want to find the set of *efficient* (Pareto optimal) path-strategies $\mathcal{S}_E \subseteq \mathcal{S}_P$ or equivalently the set of *nondominated* points $\mathcal{W}_E = \{W(S) \in \mathbb{R}^2 \mid S \in \mathcal{S}_E\}$ in the criterion space \mathcal{W} . We will follow the predominant thought within bicriterion optimization which is to identify all nondominated points, providing one corresponding efficient solution for each nondominated point.

Nondominated points can be partitioned into two sets, namely supported and unsupported. The supported ones can be further subdivided into extreme and nonextreme as illustrated in the following example.

Example 1. (continued) Assume that two costs $c_i(u, v, t)$, $i = 1, 2$, are given for each leaving time t from node u along arc (u, v) , see Table 1.

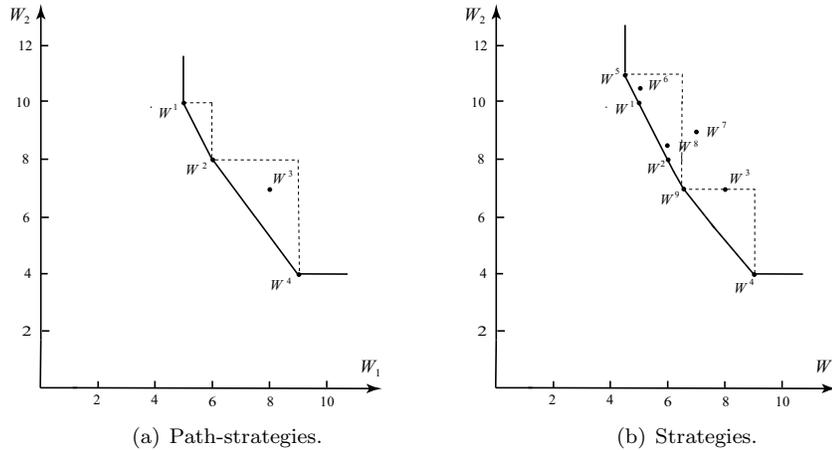


Figure 2: Criterion spaces under a priori and time-adaptive route choice.

Consider (1) when both criteria are to minimize expected cost. The criterion points corresponding to the four possible loopless paths in G are illustrated in Figure 2(a). In this example all four points are nondominated. W^1, W^2 and W^4 are supported points all of which are extreme. The extreme points define two triangles, shown with dashed lines, in which it may be possible to find unsupported nondominated points such as W^3 .

In general the total number of path-strategies is significantly lower than the number of strategies. Under time-adaptive route choice, i.e. we consider (1) with the constraint replaced with $S \in \mathcal{S}$ instead, we have nine possible strategies, i.e. five points do not correspond to a path. All nine points are illustrated in Figure 2(b). Under time-adaptive route choice the five points W^1, W^2, W^4, W^5 and W^9 are supported nondominated points of which W^1 and W^2 are non-extreme. Points which do not lie inside the triangles such as W^3 and W^7 are dominated. Moreover, the two points W^6 and W^8 are dominated by W^1 and W^2 , respectively. Further details and illustrations of the (path-)strategies can be found in [11].

Please note that the set of nondominated points under time-adaptive route choice will always dominate the set of nondominated points under a priori route choice. However, a nondominated point under a priori route choice (such as W^3) may be dominated, if time-adaptive route choice is considered. ■

3 Solution method

In this section we devise a solution method finding all efficient paths under a priori route choice based on the two-phase method. The two-phase approach is a general method for solving bicriterion discrete optimization problems such as (1). As the name suggests, the two-phase method divides the search for nondominated points into two phases. In phase one, extreme supported non-

dominated points are found. These extreme points define a number of triangles in which unsupported nondominated points may be found in phase two. For a description of a generic two-phase method see [13].

Both phases make use of a parametric function $\gamma : (\mathcal{W}, \mathbb{R}_+) \rightarrow \mathbb{R}_+$ which denotes the *parametric cost* of a path-strategy $S \in \mathcal{S}_P$.

$$\gamma(W(S), \lambda) = W_1(S)\lambda + W_2(S). \quad (2)$$

It is well-known that given $\lambda > 0$ the path-strategy S with minimum parametric cost $\gamma(W(S), \lambda)$ corresponds to a supported nondominated point and hence is efficient. As a result all supported extreme nondominated points can be found in phase one by solving (2) for different values of λ , see [2].

Phase two searches each triangle using an algorithm for ranking path-strategies with respect to the parametric weight (2), where λ is a function of the slope of the line joining the two points defining the triangle. The search stops when the parametric weight reaches an upper bound, which in turn is dynamically updated (decreased) when new nondominated points are found.

It must be kept in mind that in both phases we have to solve a sequence of difficult problems, since a priori routing even for the single criterion case is NP-hard. In order to solve these problems we adopted the newly developed algorithm for ranking paths in STD networks (*procedure K-BPS*), see Nielsen et al. [10]. However, the effectiveness of this approach is quite different for expectation and min-max criteria.

As long as two expectation criteria are used, the parametric cost $\gamma(W(S), \lambda)$ of a strategy S is equal to the cost of S with respect to the cost vector $c_\lambda = c_1 \cdot \lambda + c_2$. This result has been proved in [9, Th. 2] for strategies, and clearly holds for path-strategies too. As a consequence, we can rank paths with respect to the parametric cost by applying procedure *K-BPS* with the costs c_λ . This procedure is also used in phase one, stopping as soon as the best parametric path is found.

Unfortunately, the above result does not hold, if min-max criteria are considered. In this case, c_λ only provides us with a *lower bound* $W_\lambda(S) \leq \gamma(W(S), \lambda)$ (see [9, Th. 3, 4]). By applying procedure *K-BPS* with costs c_λ we generate path-strategies in non-decreasing order of $W_\lambda(S)$. In phase one, for each value of λ we let procedure *K-BPS* run until $W_\lambda(S)$ reaches the parametric cost of the best path-strategy generated so far. In phase two a triangle is searched until the lower bound $W_\lambda(S)$ reaches the upper bound of the triangle. In phase one it may happen, due to ranking according to $W_\lambda(S)$, that procedure *K-BPS* generates many paths that actually fall inside the triangle defined by a certain λ . In order to take advantage of this fact, in our computational tests, we adopt a *hybrid* algorithm, where the two phases are combined. More precisely, when a new triangle is identified in the first phase we search inside the triangle by letting procedure *K-BPS* continue until the lower bound $W_\lambda(S)$ reaches the upper bound.

4 Computational experience

We implemented the algorithm in C++ and compiled the source code with the GNU C++ compiler with optimize option -O on a 1 GHz PIII computer with

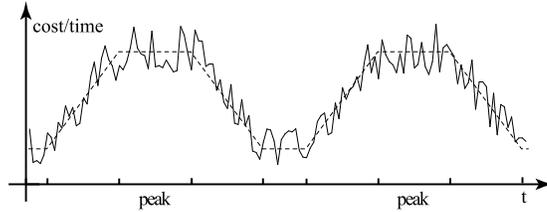


Figure 3: Peak effect and random perturbation for an arc.

Size	H	Peaks	I_T	I_C	κ
5×8	144	2	[3,20]	[1,2200]	6
10×10	288	4	[3,20]	[1,2200]	6

Table 2: Test parameters.

1GB RAM using a Linux Red Hat operating system. The main goals and results of our computational experience have been anticipated in the introduction, where we pointed out the original contributions of our work.

4.1 Test instances

The STD network test instances are generated using the newly developed generator TEGP (*Time-Expanded Generator with Peaks*) which includes several features inspired by typical aspects of road networks. For more details see [8].

Two grid graphs, with sizes 5×8 and 10×10 , are considered; the length of the time horizon is 144 and 288, respectively. Each grid arc is randomly assigned an *off peak* mean travel time, so that the mean travel time changes as a function of leaving time, increasing up to 100% during *peak periods*. For the 5×8 grid the mean travel time follows the two-peaks pattern shown by the dotted line in Figure 3, while for the 10×10 grid the pattern in Figure 3 is repeated, obtaining four peaks. The travel time distribution is randomly generated around the travel time mean, as a discrete approximation of a normal distribution. Arc costs are generated independently from travel times and increase due to the peak effect with a further *random perturbation* (10% at most) thus following a pattern similar to the solid line in Figure 3.

The network parameters are summarized in Table 2. Here I_T and I_C denote the range of possible travel times and costs, and κ is the average size of the travel time distributions. For further details about the test instances see [11].

We report results on three different combinations, namely: T/C , corresponding to travel time and cost; $C/C_{neg_{cor}}$, where the two costs are negatively correlated; and $C/C_{no_{cor}}$, where no correlation between the two costs is assumed. Recall that $C/C_{neg_{cor}}$ is usually considered to be harder. In all combinations the penalty costs are zero. For each setting of costs five test instances were generated.

		$ \mathcal{W}_{SE} $	CPU_{SE}	$ \mathcal{W}_\Delta $			CPU_Δ	
				Δ	ave	max	ave	max
5×8	T/C	5	0.54	4	1	4	0.16	0.33
5×8	C/C <i>no_{cor}</i>	4	0.57	3	1	4	0.20	0.40
5×8	C/C <i>neg_{cor}</i>	8	1.45	7	3	11	0.73	2.22
10×10	T/C	6	8.90	5	2	8	3.79	22.91
10×10	C/C <i>no_{cor}</i>	8	21.45	7	3	17	5.76	47.57
10×10	C/C <i>neg_{cor}</i>	11	25.04	10	6	26	12.08	43.05

Table 3: Results expectation criteria.

		$ \mathcal{W}_{SE} $	Δ	$ \mathcal{W}_\Delta $		CPU_Δ	
				ave	max	ave	max
5×8	C/C <i>no_{cor}</i>	4	3	1	4	0.32	0.68
5×8	C/C <i>neg_{cor}</i>	7	6	4	14	2.01	5.65
10×10	C/C <i>no_{cor}</i>	8	7	2	12	17.91	80.12
10×10	C/C <i>neg_{cor}</i>	10	9	7	59	100.42	439.72

Table 4: Results min-max criteria.

4.2 Results

We first consider the case of two expectation criteria. The results are reported in Table 3. Here $|\mathcal{W}_{SE}|$ is the number of supported extreme nondominated points and CPU_{SE} is the average CPU time (in seconds) used to find the supported extreme nondominated points. $|\Delta|$ is the number of triangles, while $|\mathcal{W}_\Delta|$ is the number of nondominated points in a triangle (points defining the triangle not included). Finally, CPU_Δ is the CPU time used to find the nondominated points in a triangle (average and maximum).

In phase one all extreme supported nondominated points can be determined in a reasonable amount of time. The same holds true for phase two (which is the most time-consuming phase). That is, we can find the nondominated set for all the test instances considered.

Comparing the different combinations of criteria, we see that the time-cost case is in general easier than the cost-cost cases, and that negatively correlated costs are harder than uncorrelated costs. The plots in Figure 4 give an intuition of the difference between *C/C neg_{cor}* and *C/C no_{cor}*. Indeed, negatively correlated costs produce more nondominated points that are spread in a larger area of the criterion space. As a result, we have more triangles to search, and it takes longer time to search each one of them; see [11] for further details. This fact was also observed under time-adaptive route choice [9] and is a general feature for discrete bicriterion optimization problems, see e.g. [13].

The results for two min-max criteria are presented in Table 4; we only consider the (more difficult) cost/cost combinations here. Compared to expectation criteria the total number of nondominated points is about the same in average. However, the CPU time spent is considerably higher, as we may expect, since the parametric problem is harder to solve and the lower bound used for ranking

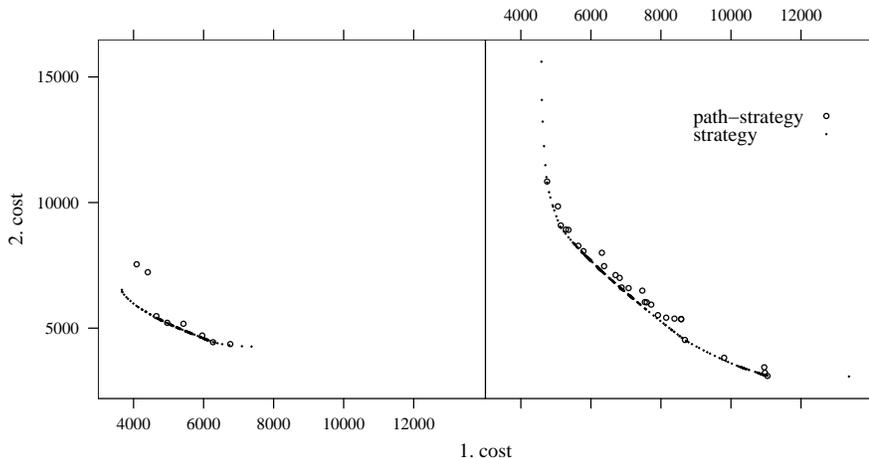


Figure 4: \mathcal{W}_E for an uncorrelated (left) and negatively correlated (right) test instance (both criteria are minimizing expected cost).

is not very tight. Also in this case, the problems with negatively correlated costs are much more difficult.

4.3 Comparison to the time-adaptive case

Comparing the results for the a-priori case to previous results for the time-adaptive case [7, 9] allows us to point out interesting differences. Here we restrict ourselves to expectation criteria, since for min-max criteria the approximation found in [9] is usually rather weak.

First of all, recall that the set of nondominated points has been found for all the instances considered. This is in deep contrast to time-adaptive route choice, where not even an ε -approximation with $\varepsilon = 1\%$ could be found for the same set of instances [9]. This result may be viewed as surprising, since finding the best strategy in the single criterion case is easy (can be done in linear time) while finding the best path-strategy is NP-hard. A reasonable explanation of this apparent paradox is that the solution space is much more dense in the time-adaptive case, that is, the total number of path-strategies is much lower than the total number of strategies. Therefore the ranking procedure used in the second phase does not have to rank as many solutions.

In order to get a deeper insight in this issue, we made plots comparing the nondominated set for the a priori case with an approximation of the nondominated set for the time adaptive case, obtained using the algorithms from [9]. Figure 4 shows two instances on a 5×8 grid with uncorrelated costs (left) and negatively correlated costs (right).

First, as noted above, negatively correlated costs produce more nondominated points, spread in a wider area; this situation arises for both a priori and adaptive routing.

Second, in some cases the a priori nondominated set may contain points close to the time-adaptive nondominated set. Hence solutions found when priori

routing must be adopted, due e.g. to outside regulations, may still be as good as those found without this regulation. However, in other cases you may have to pay higher costs, if paths must be adopted (e.g. if the first cost is below 4500 in the left plot in Figure 4).

Finally, in general for our instances there are large variations in the values of ε for which the a priori nondominated set turns out to ε -dominate (see [16]) the time-adaptive nondominated set. On average $\varepsilon = 0.1$, the minimum ε value found was 0.03 and the maximum 0.25.

References

- [1] T.S. Chang, L.K. Nozick, and M.A. Turnquist. Multiobjective path finding in stochastic dynamic networks, with application to routing hazardous materials shipments. *Transportation Science*, 39(3):383–399, 2005. doi:10.1287/trsc.1040.0094.
- [2] J. Cohen. *Multiobjective Programming and Planning*. Academic Press, New York, 1978.
- [3] S. Gao and I. Chabini. Optimal routing policy problems in stochastic time-dependent networks. *Transportation Research Part B*, 40:93–122, 2006. doi:10.1016/j.trb.2005.02.001.
- [4] R.W. Hall. The fastest path through a network with random time-dependent travel times. *Transportation Science*, 20(3):182–188, 1986.
- [5] H.W. Hamacher, S. Ruzika, and S.A. Tjandra. Algorithms for time-dependent bicriteria shortest path problems. *Discrete Optimization*, 3: 238–254, 2006. doi:10.1016/j.disopt.2006.05.006.
- [6] E. Miller-Hooks and H.S. Mahmassani. Optimal routing of hazardous materials in stochastic, time-varying transportation networks. *Transportation Research Record*, 1645:143–151, 1998. doi:10.3141/1645-18.
- [7] L.R. Nielsen. *Route Choice in Stochastic Time-Dependent Networks*. PhD thesis, Department of Operations Research, University of Aarhus, 2004. URL <http://www.imf.au.dk/publs?id=499>.
- [8] L.R. Nielsen. Texp - time-expanded generator with peaks, February 2006. URL <http://www.research.relund.dk>. Reference manual.
- [9] L.R. Nielsen, K.A. Andersen, and D. Pretolani. Bicriterion shortest hyperpaths in random time-dependent networks. *IMA Journal of Management Mathematics*, 14(3):271–303, 2003. doi:10.1093/imaman/14.3.271.
- [10] L.R. Nielsen, D. Pretolani, and K.A. Andersen. K shortest paths in stochastic time-dependent networks. Working Paper WP-L-2004-05, Department of Accounting, Finance and Logistics, Aarhus School of Business, 2004.
- [11] L.R. Nielsen, D. Pretolani, and K.A. Andersen. Bicriterion a priori route choice in stochastic time-dependent networks. Working Paper WP-L-2006-10, Department of Business Studies, Aarhus School of Business, 2006.

- [12] S. Opananon and E. Miller-Hooks. Multicriteria adaptive paths in stochastic time-varying networks. *European Journal of Operational Research*, 173: 72–91, 2006. doi:10.1016/j.ejor.2004.12.003.
- [13] C.R. Pedersen, L.R. Nielsen, and K.A. Andersen. The bicriterion multi modal assignment problem: Introduction, analysis, and experimental results. *Informs Journal on Computing*, 20(3):400411, 2008. doi:10.1287/ijoc.1070.0253.
- [14] D. Pretolani. A directed hypergraph model for random time-dependent shortest paths. *European Journal of Operational Research*, 123(2):315–324, 2000. doi:10.1016/S0377-2217(99)00259-3.
- [15] J.S. Provan. A polynomial-time algorithm to find shortest paths with recourse. *Networks*, 41(2):115–125, 2003. doi:10.1002/net.10063.
- [16] A. Warburton. Approximation of pareto optima in multiple-objective, shortest-path problems. *Operations Research*, 35(1):70–79, 1987.