Time-adaptive and history-adaptive multicriterion routing in stochastic, time-dependent networks

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.

Lars Relund Nielsen

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

Kim Allan Andersen

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

Matthias Ehrgott

Department of Engineering Science, The University of Auckland, Private Bag 92019, Auckland 1142, New Zealand, m.ehrgott@auckland.ac.nz.

Abstract: We compare two different models for multicriterion routing in stochastic time-dependent networks: the classic "time-adaptive" model and the more flexible "history-adaptive" one. We point out several properties of the sets of efficient solutions found under the two models. We also devise a method for finding supported history-adaptive solutions.

Keywords: Multiobjective programming, shortest paths, stochastic time-dependent networks, time-adaptive strategies, history-adaptive strategies.

1 Introduction

In *stochastic time-dependent* (*STD*) networks (also known as random and time-varying) travel times are modelled as random variables with time-dependent distributions. STD networks were first addressed by Hall [3], who showed that the best route between two nodes is not necessarily a path, but rather a *time-adaptive strategy* that assigns optimal successor arcs to each node as a function of leaving times. This is referred to as *time-*

*Corresponding Author. Tel.: +39 0522 522229; fax: +39 0522 522609

adaptive route choice, and represents the standard model for routing in STD networks. A survey on the subject and a literature review can be found in the paper by Gao and Chabini [2], who also discuss a more general framework, where online information and stochastic dependency are taken into account.

When the arcs in an STD network carry multiple attributes, we are faced with *multicriterion* routing problems, where the solution is no longer a single optimal strategy but rather a set of *efficient* (Pareto optimal) strategies. Finding the efficient set is well-known to be NP-hard also in deterministic networks. Nielsen [4] and Nielsen, Andersen, and Pretolani [5] address the bicriterion routing problem under time-adaptive route choice; they propose solution methods for the *weighted sum scalarization* of the problem (see e.g. [1]) and apply them in a *two-phase method* for finding (or approximating) the set of efficient strategies. Opasanon and Miller-Hooks [6] consider an arbitrary number of criteria and a generalization of time-adaptive route choice. More precisely, they propose a model where routing decisions at a node are a function of time as well as of the traveller's *history*, i.e., arrival times at previous nodes. We refer to this model as *history-adaptive route choice*. For this model, Opasanon and Miller-Hooks [6] point out some properties, and propose a label correcting method (*Algorithm APS*) for finding the efficient set. Moreover, they devise two algorithms for solving a weighted sum scalarization (referred to as "disutility").

In this paper we investigate the relationships between time-adaptive and history-adaptive route choice in a multicriterion setting. First we describe the structure of the solutions and propose a classification of the two models; then we point out some relevant theoretical properties; finally we address computational issues, proposing possible improvements to scalarization algorithms. Throughout the paper we adopt a standard terminology of multiobjective programming while keeping notation and formal definitions to a minimum. Further technical details can be found in the extended version [8]. Most of the results will be illustrated by means of examples. For this purpose we adopt, as a graphical tool, the representation of an STD network as a *time-expanded hypergraph*; the reader is referred to [7, 4] for a theoretical treatment of the subject. We remark that the results provided in the paper hold for an arbitrary number of criteria, even if examples are limited to the bicriterion case.

The structure of the paper is as follows. In the next section we introduce STD networks with a running example, which allows us to describe the structure of the solutions. Properties of the two models are discussed formally in Section 3. Computational issues are addressed in Section 4. A summary of the results is given in Section 5.

2 STD networks, strategies, and labels

Let $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ be a directed graph, referred to as the *topological network*. We consider *discrete* STD networks, where arrival and departure times to/from nodes are integers in the interval $H = [0, I]$. For each arc $(i, j) \in \mathcal{A}$, $t \in [0, I]$ the set T_{ij}^t contains the possible travel times when leaving node i at time t along arc (i, j). The set $A_{ij}^t = \{t + t' : t' \in T_{ij}^t\}$ contains the corresponding possible arrival times at j. Each travel time $t' \in T_{ij}^t$ occurs

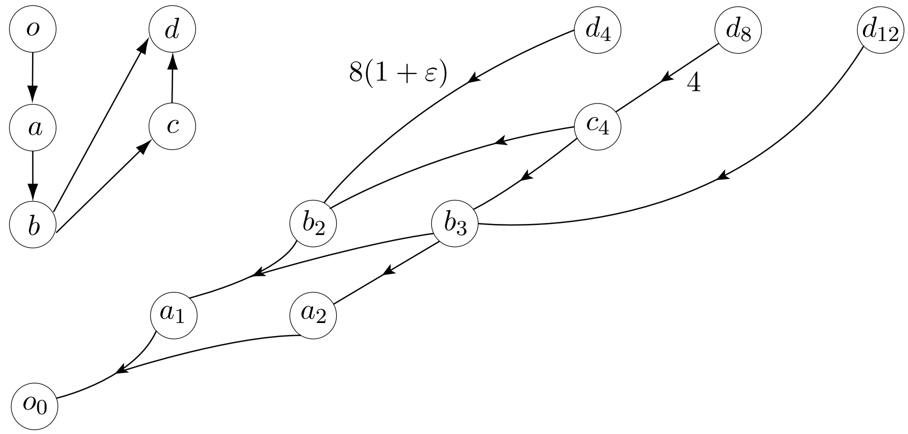


Figure 1: Topological network and time-expanded hypergraph

$(i,j),t$	$(o,a),0$	$(a,b),1$	$(a,b),2$	$(b,d),2$	$(b,d),3$	$(b,c),2$	$(b,c),3$	$(c,d),4$
T_{ij}^t	$\{1,2\}$	$\{1,2\}$	$\{1\}$	$\{2\}$	$\{9\}$	$\{2\}$	$\{1\}$	$\{4\}$
A_{ij}^t	$\{1,2\}$	$\{2,3\}$	$\{3\}$	$\{4\}$	$\{12\}$	$\{4\}$	$\{4\}$	$\{8\}$

Table 1: Travel times and arrival times

with probability $p_{ij}^t(t')$. Waiting at nodes is not permitted.

We consider a set of $r \geq 2$ criteria, where the first criterion is identified with travel time. For each arc $(i,j) \in \mathcal{A}$, $t \in [0, I]$ and $1 < k \leq r$ we denote by $c_{ij}^k(t)$ the cost according to criterion k of travelling along arc (i,j) leaving i at time t. Note that this definition extends the one given in [7] for a single cost criterion. Opasanon and Miller-Hooks [6] adopt a more detailed description of the STD network that can be shown to be equivalent (see [8], Appendix A) to the definition adopted here.

Example 1 Consider the topological network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ shown in the top left corner of Figure 1. We assume that a traveller leaves the origin node o at time zero towards the destination node d. Since waiting is not allowed, we only consider departure times corresponding to possible arrival times at intermediate nodes. For each arc (i,j) and relevant time t, the set T_{ij}^t of travel times and the set A_{ij}^t of arrival times are given in Table 1.

We only have two non-deterministic travel times, namely arc (o,a) at departure time 0 and arc (a,b) at departure time 1; in both cases we assume that travel times have the same probability 1/2, that is $p_{oa}^0(1) = p_{oa}^0(2) = p_{ab}^1(1) = p_{ab}^1(2) = 1/2$. Note that routing decisions are needed (actually, possible) only at node b.

We represent the STD network by means of a *time-expanded hypergraph*, as shown in Figure 1. For each node $i \in \mathcal{N}$ and relevant time t we introduce a hypergraph node i_t; for each arc (i, j) and departure time t we introduce a *hyperarc* $e_{ij}(t)$ that connects node i_t to the set $\{j_\theta : \theta \in A_{ij}^t\}$ of hypergraph nodes corresponding to possible arrival times at j. For example, hyperarc $e_{ab}(1)$ connects node a_1 to the node set $\{b_2, b_3\}$, since $A_{ab}^1 = \{2, 3\}$.

We assume $r = 2$, and we refer to criterion 2 as *cost*; the cost is zero for each arc and departure time, except for the two cases shown in Figure 1, namely: arc (c, d) at departure time 4, with cost $c_{cd}^2(4) = 4$, and arc (b, d) at departure time 2, with cost $c_{bd}^2(2) = 8(1+\varepsilon)$, where $0 < \varepsilon < 1$. □

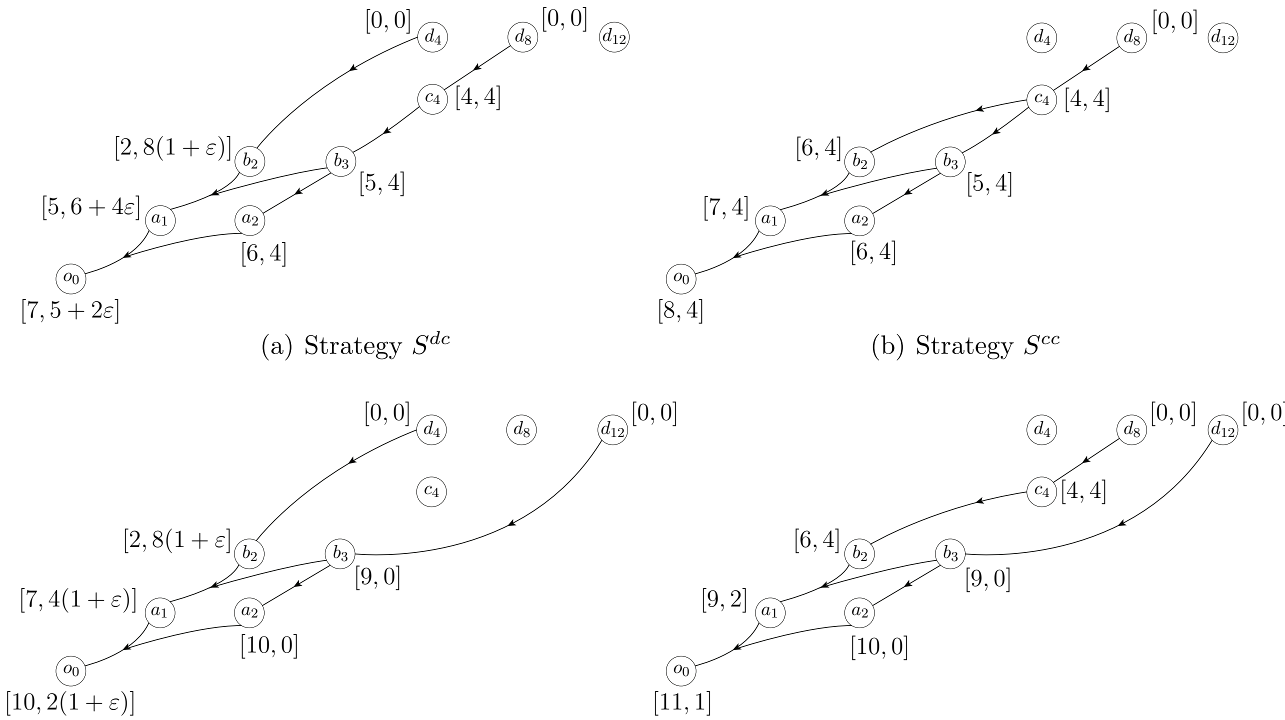
According to time-adaptive route choice, a time-adaptive strategy (TAS) in a discrete STD network is defined by choosing a single successor arc for each node $i \neq d$ and time t. Each strategy determines, for each node i, time t and $k = 1 \ldots r$, the expected value of criterion k for travelling from i to the destination, leaving i at time t. Given a strategy, the corresponding expected values can be formally defined by means of a set of recursive equations, see e.g. Pretolani [7]. In practice, the computation of these values consists of a labelling process that we illustrate with our running example.

Example 1 (continued) In order to define a TAS, we must choose a successor for node b at time 2 and at time 3; for the other nodes, only one successor is available. Since two choices are possible at node b, namely going to the destination d or to the intermediate node c, we can define four possible strategies. We denote these strategies by S^{dd}, S^{cd}, S^{dc} and S^{cc}, where u and v in S^{uv} denote the successor of b at time 2 and 3, respectively. The four strategies are shown in Figure 2. Each one is represented by the corresponding *hyperpath* that contains the hyperarcs representing the chosen successor arcs. Namely, if (i, j) is the successor arc of node i at departure time t, then the hyperpath contains the hyperarc $e_{ij}(t)$.

Each strategy assigns to each hyperpath node i_t a label $\lambda_i(t) = [\lambda_i^1(t), \lambda_i^2(t)]$, where $\lambda_i^1(t)$ is the expected travel time and $\lambda_i^2(t)$ is the expected cost for traveling from node i at departure time t to the destination. For each destination node d_t the label is $[0, 0]$. If (i, j) is the successor arc of node i at departure time t, then the label of node i_t is obtained as a weighted sum of the labels at nodes $\{j_\theta : \theta \in A_{ij}^t\}$, using probabilities p_{ij}^t as weights.

Figure 2 reports the labels assigned to hyperpath nodes by each strategy. For instance, consider strategy S^{dc}; here label $[4, 4]$ for c_4 is obtained from $[0, 0]$ at d_8, since both travel time and cost are 4 for arc (c, d) at time 4. The label $[5, (6 + 4\varepsilon)]$ for a_1 is obtained from labels $[2, 8(1 + \varepsilon)]$ and $[5, 4]$ (nodes b_2 and b_3); the expected travel time is $(1 + 2)/2 + (2 + 5)/2 = 5$, while the expected cost is $8(1+\varepsilon)/2+4/2 = (6+4\varepsilon)$. Note that the probabilities $p_{ab}^1(1) = p_{ab}^1(2) = 1/2$ are used here. □

Opposite to the time-adaptive route choice, under history-adaptive route choice we have that in a history-adaptive strategy (HAS) the successor of a node i at time t is not necessarily unique; a traveller can choose different successors, and thus different substrategies, depending on the travel time experienced in previous arcs. As a consequence, different labels assigned to the same hypergraph node can be combined in the labelling process. Again, we illustrate the resulting labelling process using our running example.



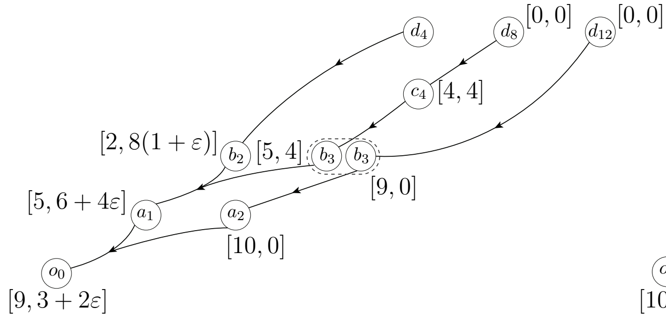
(a) Strategy S^{dc} (b) Strategy S^{cc}

(c) Strategy S^{dd} (d) Strategy S^{cd}

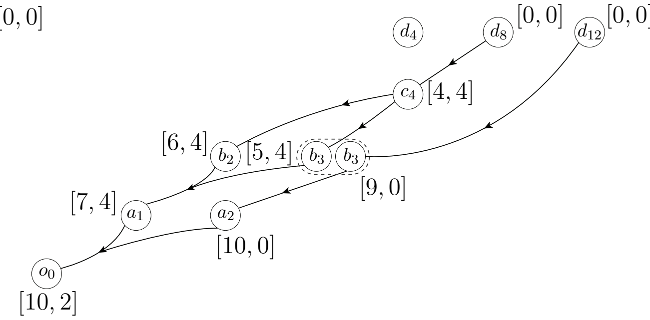
Figure 2: Time-adaptive strategies and corresponding time/cost labels.

Example 1 (continued) Observe that a traveller can reach node b at time 3 along two different "histories", namely, leaving node a at time 1 or 2. Moreover node b has two possible successors. Thus there are four possible history-adaptive choices for the successor of node b at time 3. In fact, this is the only case where history-adaptive route choice can occur; indeed, nodes o, a and c have a unique possible successor, while node b at time 2 has a unique "history", that is, leaving a at time 1. Since there are two possible choices at node b and time 2 we have eight HAS overall. Four of them, where the successor of node b at time 3 is independent of the leaving time from a, correspond to the time-adaptive strategies shown in Figure 2. The other four are shown in Figure 3, where we "split" node b_3 to point out the history-adaptive behavior. Extending the previous notation, u and v in $S^{w,uv}$ denote the successor of b at time 3 when leaving a at time 1 and 2, respectively, while w is the successor of b at time 2. We assume that $S^{w,uv}$ denotes the TAS S^{wv} if $v = u$. Note that in each strategy of Figure 3 two different labels are assigned to hypergraph node b_3. One of these is used to obtain the label for a_1, while the other is used to obtain the label for a_2. □

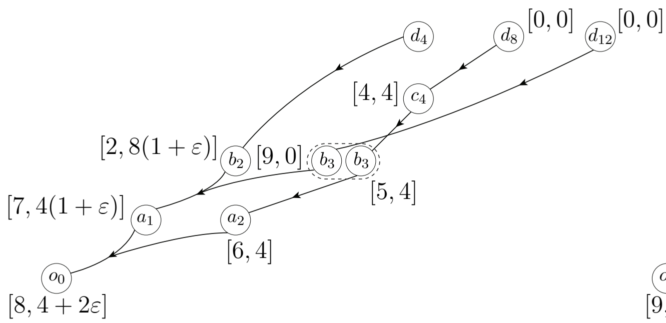
Terminological note Pretolani [7] proved that TAS define hyperpaths in the time-expanded hypergraph. However, this relation does not hold for HAS that are not TAS.



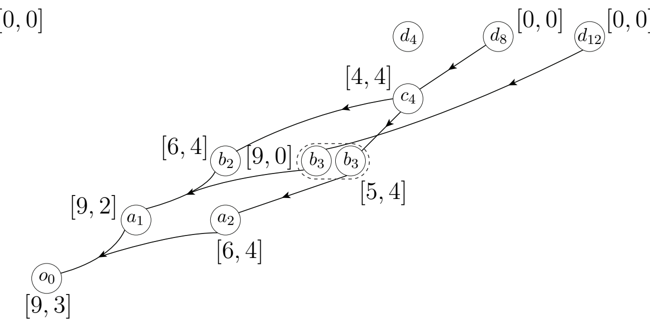
(a) Strategy $S^{d,cd}$



(b) Strategy $S^{c,cd}$



(c) Strategy $S^{d,dc}$



(d) Strategy $S^{c,dc}$

Figure 3: History-adaptive strategies and corresponding time/cost labels.

Opasanon and Miller-Hooks [6] refer to HAS as “hyperpaths”, but this term should be intended informally as a collection of paths, rather than a formal definition of the solution structure.

3 Properties of adaptive routing models

As a first step we classify routing models according to the taxonomy proposed by Gao and Chabini [2]. The classification is based on the amount of current information which is available to the traveller. The information depends on two factors, namely *network stochastic dependency* and *information access*. The former defines the link- and time-wise stochastic dependency between travel time random variables. One extreme is that all link travel time random variables are completely independent, and the other extreme is that they are completely dependent. The latter defines which link time realizations are available to the traveller at any given time and given node. It is characterized according to whether *perfect online information*, *partial online information* or *no online information* is available to the traveller. Models with no online information belong to a single class, referred to as *NOI*. Models with perfect or partial online information are further subdivided into groups, also depending on the stochastic dependency between random variables. If these variables

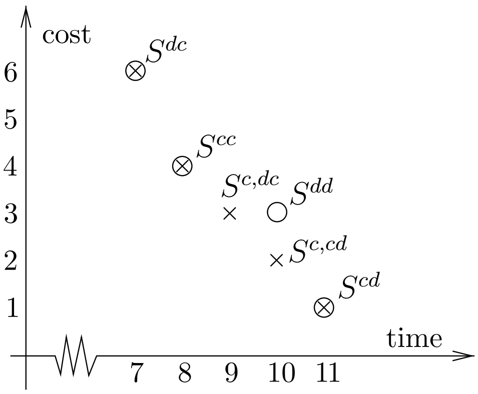


Figure 4: Strategies and efficient labels ($\varepsilon = 1/2$).

are completely independent, models belong to *Group 1*. Otherwise, they belong to *Group 2* if perfect online information is available and to *Group 3* if only partial online information is available.

Time-adaptive route choice corresponds to the *NOI* class, since the traveller is assumed to have no information other than current node and time. History-adaptive routing falls in Group 1, and precisely in the case with partial on-line information available. Indeed, a history provides no information on future link travel times, that is, stochastic independence of random variables is assumed. Moreover, the information provided by a history is limited to those links previously used by the traveller, and does not extend to the whole network. Gao and Chabini remark that the class NOI and Group 1 are different in principle, although computationally equivalent in a single criterion setting. Their claim is supported by the fact that these two models are no longer equivalent in a multicriterion setting.

We can point out several properties of TAS and HAS observing the results of our running example. In Figure 4 we plot (assuming $\varepsilon = 0.5$) the labels $\lambda_o(0)$ for the four time-adaptive strategies (circles) and for the five efficient history-adaptive strategies (crosses).

Let us consider time-adaptive route choice first. As can be seen in Figure 4, for $\varepsilon = 0.5$ (actually, for $0 \leq \varepsilon < 1$) the four labels turn out to be nondominated. However, if we consider the labels associated to node a_1 in Figure 2, we note that the label $[7, 4]$ assigned by S^{cc} dominates the label $[7, 4(1 + \varepsilon)]$ assigned by S^{dd}. Therefore, a traveller following strategy S^{dd} has a nonzero probability (actually, probability 1/2) of arriving at a at time one and, thereafter, of following a dominated (i.e., non efficient) *substrategy*. Note also that no other strategy yields the same label $\lambda_o(0)$ as S^{dd}. Let us say that a TAS is *strongly efficient* if all its substrategies are efficient. Thus S^{dd} is efficient but not strongly efficient. We can state the following theorem.

Theorem 1 *There may exist an efficient TAS which is not strongly efficient, and yields a label that cannot be obtained from a strongly efficient TAS.*

Note that similar results (for the bicriterion case) can be found in [4], where expected as well as *maximum possible* values are considered. Theorem 1 shows that a well-known property of deterministic bicriterion shortest paths, where subpaths of efficient paths are efficient, does not extend to time-adaptive route choice. On the contrary, the property holds for history-adaptive route choice, i.e., an efficient HAS is strongly efficient, see Lemma 1 in Opasanon and Miller-Hooks [6]. Since label correcting methods (such as Algorithm APS in [6]) only generate strongly efficient strategies, we have the following relevant consequence.

Corollary 1 *A label correcting algorithm may not find all the nondominated labels corresponding to efficient TAS, in particular, it will miss efficient TAS that are not strongly efficient.*

In our example, each TAS except S^{dd} is strongly efficient and is *extreme*, i.e., its label defines an extreme point of the time-adaptive set

$$\mathcal{Y}_T^{\geqq} = \text{conv}(\mathcal{Y}_T) \oplus \mathbb{R}_+^r = \{\lambda + y : \lambda \in \text{conv}(\mathcal{Y}_T), \ y \in \mathbb{R}_+^r\},$$

where $\mathcal{Y}_T$ is the set of TAS labels and "conv" denotes the convex hull; in this case, $r = 2$ and $\mathcal{Y}_T = \{[7, 5 + 2\varepsilon], [8, 4], [10, 2(1 + \varepsilon], [11, 1]\}$. Thus every extreme TAS is strongly efficient in our example: as we shall see later, this is the case in general.

Let us now consider history-adaptive route choice which, as expected, provides a more dense solution set. Five out of the eight HAS turn out to be efficient; three of them correspond to extreme TAS, the other two, namely $S^{c,cd}$ and $S^{c,dc}$, dominate the TAS S^{dd}. Note that $S^{c,cd}$ and $S^{c,dc}$ are *supported* solutions, that is, they belong to the boundary of the history-adaptive set

$$\mathcal{Y}_H^{\geqq} = \text{conv}(\mathcal{Y}_H) \oplus \mathbb{R}_+^r,$$

where $\mathcal{Y}_H$ is the set of HAS labels; however, they are not extreme points in $\mathcal{Y}_H^{\geqq}$. Moreover, extreme points in $\mathcal{Y}_H^{\geqq}$ correspond to TAS, in other words we have $\mathcal{Y}_H^{\geqq} = \mathcal{Y}_T^{\geqq}$ in our example. Theorem 3 will show that this is no coincidence.

Let us define a *weighted sum scalarization* (*WSS*) of the problem under history-adaptive route choice. We are given a vector of weights $w \in W^+ = \{w \in \mathbb{R}^k : w_k > 0, 1 \leq k \leq r\}$, where w_k is the weight of criterion k. We must find a HAS that minimizes the weighted sum $w^T \lambda_o(0)$ of the expected values of the criteria. We say that one such HAS is *WSS-optimal* for the weights w. Since both the scalarization and the labels are defined by means of linear equations, the following quite intuitive result follows.

Lemma 1 *A WSS-optimal HAS defines WSS-optimal substrategies, i.e., minimum values $w^T \lambda_i(t)$, for each node i and time t.*

This result corresponds to Lemma 3 in [6], where an optimal HAS for a WSS is referred to as a "*LED hyperpath*". The following lemma establishes another key property of WSS.

Lemma 2 *For each weight vector $w \in W^+$ there exists a WSS-optimal TAS.*

Proof Let S be a WSS-optimal HAS, and assume that S assigns two or more different successors to node i at time t, depending on different histories. As follows from Lemma 1, the labels obtained by these successors must be both WSS-optimal, that is, minimize the product $w^T \lambda_i(t)$. But then, we can choose one of the optimal successors, and use it for all histories, still obtaining a WSS-optimal strategy at node i and time t. By iterating this process we end up with a TAS that fulfills the requirements, and the claim follows. □

We can now prove the general properties mentioned above.

Theorem 2 *Extreme TAS are strongly efficient.*

Proof Assume that the TAS S yields the extreme point $\bar{\lambda}$ and is WSS-optimal for weights $w \in W^+$. Suppose that S defines a dominated substrategy $S_i(t)$ for node i and time t. Since $w_i > 0$ for each $1 \leq k \leq r$, we have $w^T \lambda_i(t) > w^T \lambda_i'(t)$, where $\lambda_i(t)$ and $\lambda_i'(t)$ denote labels assigned by S and by another TAS S', respectively. Thus the substrategy $S_i(t)$ is not WSS-optimal for w. However, it follows from Lemma 1 and Lemma 2 that a WSS-optimal TAS must define optimal substrategies, which implies a contradiction. □

Theorem 3 *Each extreme point in $\mathcal{Y}_H^{\geqq}$ is an extreme point in $\mathcal{Y}_T^{\geqq}$, i.e., $\mathcal{Y}_H^{\geqq} = \mathcal{Y}_T^{\geqq}$.*

Proof Let $\bar{\lambda}$ be an extreme point in $\mathcal{Y}_H^{\geqq}$. It is well-known that some $w \in W^+$ exist such that $\bar{\lambda}$ is the *unique* solution of

$$\min_{\lambda \in \mathcal{Y}_H^{\geqq}} w^T \lambda.$$

Therefore, any optimal solution to the WSS with weights w yields the label $\bar{\lambda}$. By Lemma 2, at least one such optimal TAS exists, thus $\bar{\lambda} \in \mathcal{Y}_T^{\geqq}$ and the claim follows. □

Theorem 3 implies that every HAS yields a label that belongs to the time-adaptive set $\mathcal{Y}_T^{\geqq}$. This fact can be related to the taxonomy of Gao and Chabini [2] observing that, in both models, no information on future link travel times is available. Theorem 3 suggests that both online information *and* stochastic dependency (i.e., models in groups 2 and 3) are necessary in order to find solutions outside $\mathcal{Y}_T^{\geqq}$. In our context, this means that a history should provide information on travel time distributions at future times.

4 Computational issues

As pointed out in [4, 5] the multicriterion problem for time-adaptive route choice is computationally intractable, also for $r = 2$ and for instances of reasonable size. Similar conclusions are drawn by Opasanon and Miller-Hooks [6] for history-adaptive route choice. In fact, the latter case is likely to be even more difficult, for at least two reasons (see [8], Appendix B):

- the number of HAS can be exponentially larger than the number of TAS;

- while the size of a TAS is linear in the size of the STD network, a single HAS can require exponential space.

Therefore, for practical purposes, efficient methods for solving WSS become crucial. Existing methods can solve a WSS quite efficiently, actually in polynomial time in the input size [5, 6]. However, these methods only return a single WSS-optimal TAS. If a decision maker is primarily interested in extreme solutions a single TAS may be sufficient. However, non-extreme supported solutions may provide a much better representation of the entire solution set, and this can also be interesting to a decision maker. Thus we should be able to find *all* the optimal labels for a WSS, including labels corresponding to WSS-optimal HAS that are not TAS. In general, this is a difficult task, since the number of optimal labels for a single WSS can be exponential in the input size, see Pretolani et al. [8]. To the best of our knowledge, no methods have been proposed for this task, except of course finding all the efficient HAS. Here we propose a solution method that exploits the theoretical properties given in the previous section. We illustrate our approach on the running example.

Example 1 (continued) The two supported HAS $S^{c,cd}$ and $S^{c,dc}$, as well as the extreme TAS S^{cd} and S^{cc}, are optimal solutions to the WSS with weight $w = [1, 1]$. Only S^{cd} and S^{cc} can be found by existing methods, even though $S^{c,cd}$ and $S^{c,dc}$ may be more attractive, since they offer a better time/cost trade-off.

Note that the two successors of node b at time 3 are both optimal for the WSS with weight $w = [1, 1]$, since $w^T[9, 0] = w^T[5, 4]$. This is not the case for node b at time 2, where (b, c) is the only optimal successor, since $w^T[6, 4] < w^T[2, 8(1 + \varepsilon)]$. If we forbid the non-optimal successor (b, d) at time 2 (i.e., we remove hyperarc $e_{bd}(2)$ from the time-expanded hypergraph) the remaining efficient solutions are exactly $S^{c,cd}$, $S^{c,dc}$, S^{cd} and S^{cc}. □

Given a weight vector w our method performs the following two steps.

1. find a WSS-optimal TAS, keeping track of all the optimal successor arcs for each intermediate node and time;

2. apply a labelling algorithm where, for each intermediate node and time, only the optimal successor arcs tracked in the previous step are used.

Both steps require minor changes in existing algorithms. It is rather easy to show (we omit details here) that the above method finds all the nondominated labels in $\mathcal{Y}_H$ corresponding to WSS-optimal HAS for w.

Note that for the bicriterion case the above approach suggests a "hybrid" between labelling and two-phase methods. For $r = 2$ each face of $\mathcal{Y}_T^{\geqq}$ is a segment, defined by a unique weight w that can be found in polynomial time, see [5]. By applying the method above to each w defining a face we can find all the supported solutions under history-adaptive route choice. Clearly, this process is intractable in general, however, the overall computational effort may be affordable as long as, for each w, the second step works on a small fraction of the whole STD network.

5 Final remarks

In this paper we investigated relations and differences between two known models for multicriterion routing in STD networks. Our results can be summarized as follows.

- we described the structure of the solutions for the two models;
- we classified the two models according to the taxonomy given by Gao and Chabini [2];
- we showed that, in contrast to HAS, an efficient TAS is not necessarily strongly efficient; however, extreme efficient TAS are strongly efficient;
- we showed that a WSS always admits an optimal TAS, which implies that the two models define the same extreme nondominated points;
- we investigated the computational aspects related to the number and size of (supported) efficient solutions under the two models.

Finally, by exploiting the above theoretical results, we proposed a method finding supported HAS that are not TAS, which may be an interesting subject for further research.

References

[1] M. Ehrgott. *Multicriteria optimization*. Springer-Verlag, Berlin, 2nd edition, 2005.

[2] 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.

[3] R.W. Hall. The fastest path through a network with random time-dependent travel times. *Transportation Science*, 20(3):182–188, 1986.

[4] 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`.

[5] 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.

[6] S. Opasanon 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.

[7] 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.

[8] D. Pretolani, L.R. Nielsen, K.A. Andersen, and M. Ehrgott. Time-adaptive versus history-adaptive strategies for multicriterion routing in stochastic time-dependent networks. Technical Report WP-L-2008-05, Department of Accounting, Finance and Logistics, Aarhus School of Business, 2008. URL `http://asb.dk/article.aspx?pid=597`.