WORKING PAPER L-2008-05

WP

Daniele Pretolani, Lars Relund Nielsen, Kim Allan Andersen & Matthias Ehrgott

Time-adaptive versus history-adaptive strategies for multicriterion routing in stochastic time-dependent networks

CORAL - Centre for Operations Research Applications in Logistics

Time-adaptive versus history-adaptive strategies for 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.

August 26, 2008

Abstract: We compare two different models for multicriterion routing in stochastic time-dependent networks: the classic "time-adaptive" route choice and the more flexible "history-adaptive" route choice. We point out some interesting properties of the sets of efficient solutions ("strategies") found under the two models. We also suggest possible directions for improving computational techniques.

Keywords: Multiple objective 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

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

successor arcs to each node as a function of leaving times. This is referred to as *time-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 a STD network carry multiple attributes, we are faced with *multi-criterion* 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 [7] 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 [7] 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. Most of the results will be ilustrated by means of examples. For this purpose we adopt, as a graphical tool, the representation of a STD network as a *time-expanded hypergraph*; the reader is referred to [4, 8] 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.

Some of the content of this paper already appeared, in a different form, in a previous note [6]. However, that note was based on the assumption that Opasanon and Miller-Hooks [7] considered routing under time-adaptive, rather than history-adaptive route choice. Consequently, the wrong conclusion is drawn that most of the results in Opasanon and Miller-Hooks [7] are not correct. Later we realized that Opasanon and Miller-Hooks [7] actually consider history-adaptive route choice. Apart from this misinterpretation, some observations in [6] are correct, and are summarized in Theorem 1 and Corollary 1 in the present paper. Moreover, [6] contains some algorithmic improvements that are not reported here.

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, including scalarization algorithms, are addressed in Section 4. A summary of the results, and some suggestions for further research, are given in Section 5. Appendix A shows that the STD representation adopted by Opasanon and Miller-Hooks [7] can be simplified, and Appendix B considers some complexity issues related to the number and size of time-adaptive and history-adaptive strategies.

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 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 [8] for a single cost criterion. Opasanon and Miller-Hooks [7] adopt a more detailed description of the STD network that can be shown to be equivalent to the definition adopted here, see Appendix A for details.

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

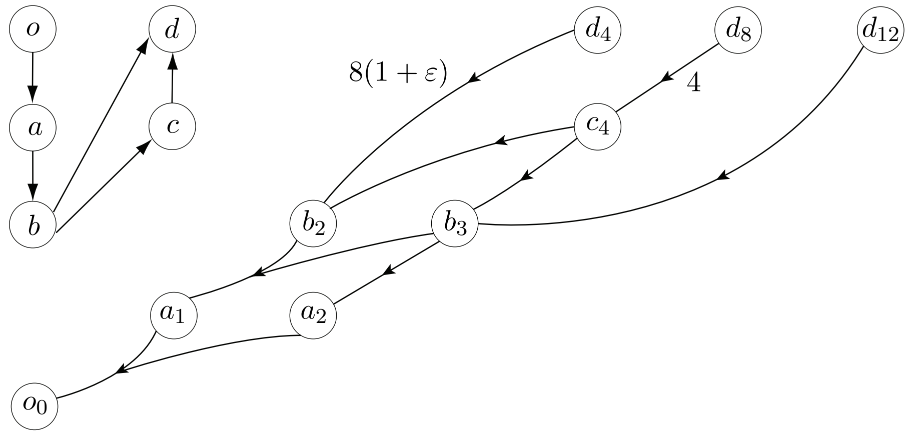


Figure 1: Topological network and time-expanded hypergraph

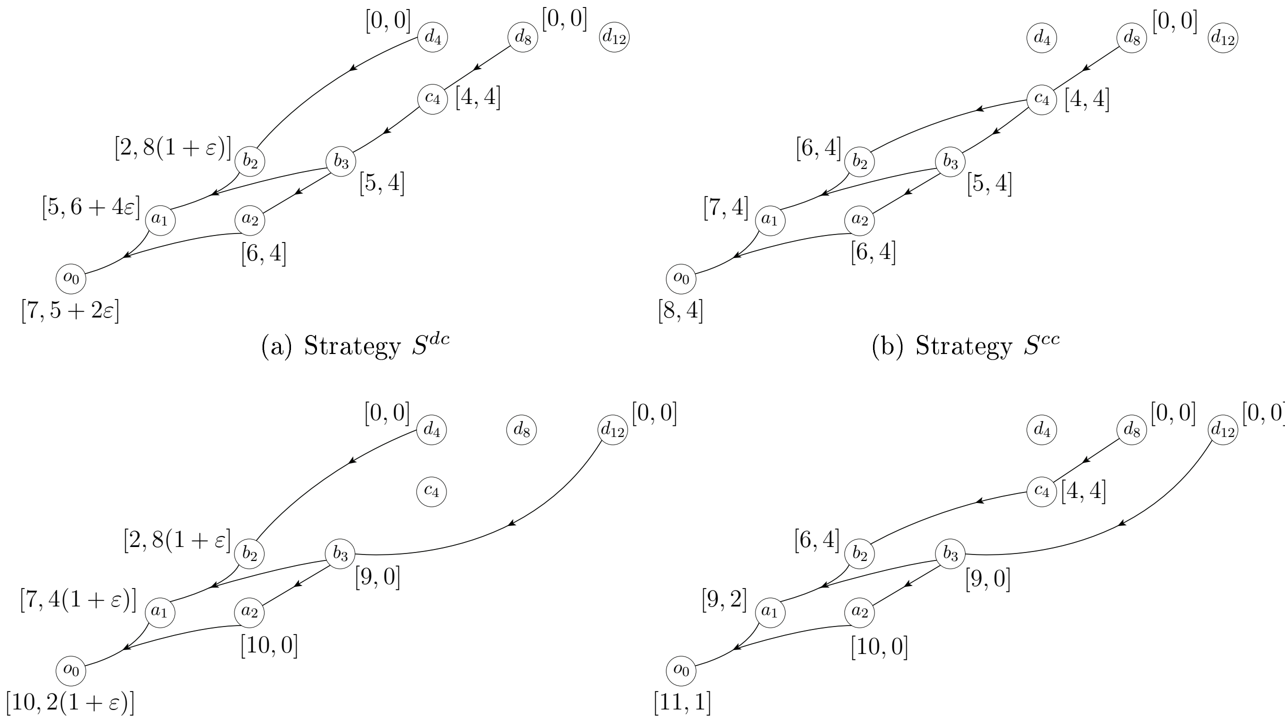
$(i,j),t$	$(o,a),0$	$(a,b),1$	$(a,b),2$	$(b,d),2$	$(b,d),3$	$(b,c),2$	$(b,c),3$	$(c,d),4$
T_{ij}^t	$\{1,2\}$	$\{1,2\}$	$\{1\}$	$\{2\}$	$\{9\}$	$\{2\}$	$\{1\}$	$\{4\}$
A_{ij}^t	$\{1,2\}$	$\{2,3\}$	$\{3\}$	$\{4\}$	$\{12\}$	$\{4\}$	$\{4\}$	$\{8\}$

Table 1: Travel times and arrival times

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 [8]. 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.



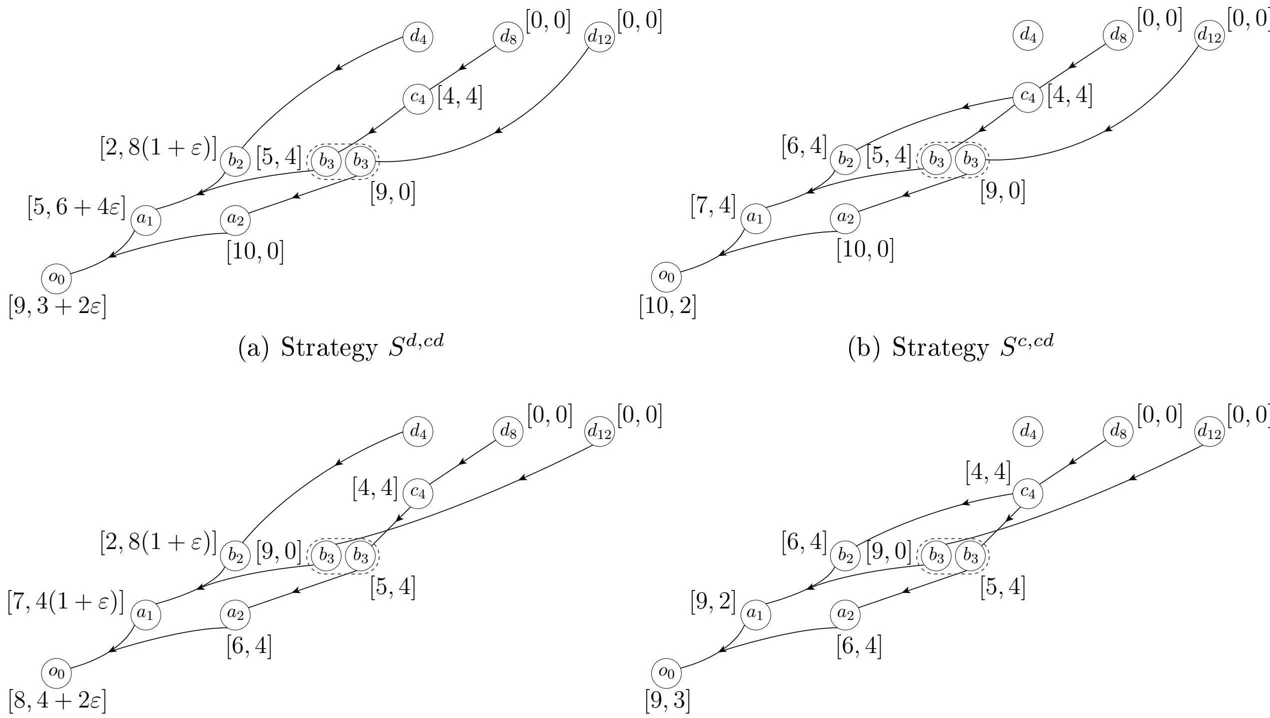
(a) Strategy S^{dc} (b) Strategy S^{cc} (c) Strategy S^{dd} (d) Strategy S^{cd}

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

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 $p^1_{ab}(1) = p^1_{ab}(2) = 1/2$ 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.

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



(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.

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 [8] proved that time-adaptive strategies define hyperpaths in the time-expanded hypergraph. This property holds because, under time-adaptive route choice, each hypergraph node i_t is assigned a *unique* “predecessor” hyperarc $e_{ij}(t)$, which is not always the case for history-adaptive route choice. Opasanon and Miller-Hooks [7] refer to history-adaptive strategies 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 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

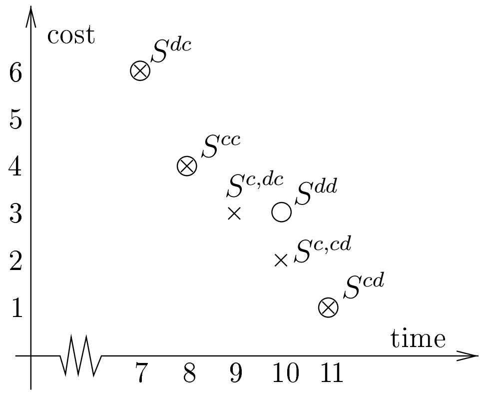


Figure 4: Strategies and efficient labels ($\varepsilon = 1/2$).

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 [7]. Since label correcting methods (such as Algorithm APS in [7]) 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

$$\mathcal{Y}_T^{\geqq} \;=\; \mathrm{conv}(\mathcal{Y}_T) \oplus \mathbb{R}^r_+ \;=\; \{\lambda + y :\; \lambda \in \mathrm{conv}(\mathcal{Y}_T),\; y \in \mathbb{R}^r_+\},$$

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

$$\mathcal{Y}_H^{\geqq} = \mathrm{conv}(\mathcal{Y}_H) \oplus \mathbb{R}^r_+,$$

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 shows 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 [7], 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 $w \in W^+$ a time-adaptive strategy exists that is WSS-optimal and defines WSS-optimal substrategies.*

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. ■

In Appendix B it is shown that the number of HAS may be exponential in the number of TAS. Despite this fact, Theorem 3 shows that the extreme points in $\mathcal{Y}_H^{\geqq}$ and $\mathcal{Y}_T^{\geqq}$ are the same. This means that if a decision maker is primarily interested in one of the criteria, it is sufficient to consider TAS. On the other hand, non-extreme HAS give a much better representation of the entire solution space inside the set $\mathcal{Y}_T^{\geqq}$. This might also be interesting to a decision-maker.

Theorem 3 states that history-adaptive route choice does not allow to "jump out" of the time-adaptive set $\mathcal{Y}_T^{\geqq}$. This may be related to the taxonomy of Gao and Chabini [2], observing that both models assume stochastic independence, even if they assume different online information. An extensive interpretation of Theorem 3 would suggest that stochastic dependency (groups 2 and 3 in [2]) should be taken into account in order to find solutions outside $\mathcal{Y}_T^{\geqq}$. In our context, stochastic dependency means that a history may provide information on travel time distributions at future times.

4 Computational issues

We address some computational and algorithmic issues in this section; claims on the number and size of adaptive strategies are proved in Appendix B. As pointed out in [4, 5] the multicriterion problems for time-adaptive route choice are computationally intractable, also for $r = 2$ and for instances of reasonable size. Indeed, the solution space is extremely dense, and thus very hard to explore with current state-of-the-art techniques. Similar conclusions are drawn by Opasanon and Miller-Hooks [7] for history-adaptive route choice. In fact, the latter case is likely to be even more difficult, for at least two reasons (see Appendix B):

- the number of HAS may be much larger (in some cases, 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.

For the above reasons, solution methods for weighted sum scalarizations become crucial. Existing methods can solve a WSS efficiently, actually in polynomial time in the input size [5, 7]; however, these methods only return one single WSS-optimal TAS. According to Theorem 3, a single TAS suffices as long as extreme solutions are searched. However, non-extreme supported solutions may be relevant as well, as shown by our 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 a WSS with weight $w = [1, 1]$. Only the extreme TAS 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, 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. 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}. □

In practice, we may be interested in finding all the non-dominated labels corresponding to optimal solutions to 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 Appendix B. Up to our knowledge, no methods have been proposed for this task, except of course finding all the efficient HAS. The above observations on our example suggest a possible approach. Given a weight vector w we proceed as follows.

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.

It follows from Lemma 2 (we omit details here) that the above method finds all the non-dominated labels in $\mathcal{Y}_H$ corresponding to WSS-optimal HAS for w. Both steps require minor changes in existing algorithms.

Note that the above approach defines a sort of "hybrid" between labelling and two-phase methods, which seems to be quite suitable for the bicriterion case. 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 reasonably 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, opposed 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 showed that the number and size of the solutions grow exponentially when moving from time-adaptive to history-adaptive route choice; this remains true even if only supported solutions are considered;
- we proposed a hybrid two-phase/labeling method finding supported HAS that are not TAS.

Due to the inherent intractability of multicriterion routing problems for both models, further research should concentrate on heuristic methods, e.g., scalarization techniques or ε-approximations. To this aim, the theoretical results provided in this paper may provide a useful guidance. In particular, the hybrid approach proposed here 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] L.R. Nielsen, D. Pretolani, and K.A. Andersen. A note on "multicriteria adaptive paths in stochastic, time-varying networks". Technical Report WP-L-2006-11, Department of Accounting, Finance and Logistics, Aarhus School of Business, 2006.

[7] 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.

[8] 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.

Appendix A: representation of STD networks

The STD network model adopted by Opasanon and Miller-Hooks [7] specifies a distribution of possible values for each criterion, arc (i, j) and departure time t. In particular, $C^k = \{c_{ij}^{kz_k}(t) : \ z_k = 1, \ldots, D\}$ denotes the set of possible values for criterion k when travelling along arc (i, j) at departure time t. Note that $C^1 = \{c_{ij}^{1z_1}(t) : \ z_1 = 1, \ldots, D\}$

is the set of possible travel times. Each value $c_{ij}^{kz_k}(t)$ occurs with probability $\rho_{ij}^{kz_k}(t)$, thus $\sum_{z_k=1,\dots,D}\rho_{ij}^{kz_k}(t)=1$. This model is unnecessarily detailed, since for each criterion $k>1$, it suffices to know the *expectation*

$$c_{ij}^k(t)=\sum_{z_k=1}^{D} c_{ij}^{kz_k}(t)\cdot\rho_{ij}^{kz_k}(t).$$

In order to prove that the simplified model is correct, it suffices to show that the formula used to compute new labels for criteria other than travel time (see Step 3 of Algorithm APS, Section 4 in [7]) can be simplified as follows:

$$\begin{aligned}
\eta_i^k(t) &= \textstyle\sum_{(z_1,x)\in Q}\sum_{z_k=1}^{D}\ \left[\, c_{ij}^{kz_k}(t)+\lambda_{jx}^k\big(t+c_{ij}^{1z_1}(t)\big)\,\right]\cdot\rho_{ij}^{1z_1}(t)\cdot\rho_{ij}^{kz_k}(t)\\
&= \textstyle\sum_{(z_1,x)\in Q}\ \rho_{ij}^{1z_1}(t)\cdot\left[\sum_{z_k=1}^{D} c_{ij}^{kz_k}(t)\cdot\rho_{ij}^{kz_k}(t)\ +\ \sum_{z_k=1}^{D}\lambda_{jx}^k\big(t+c_{ij}^{1z_1}(t)\big)\cdot\rho_{ij}^{kz_k}(t)\right]\\
&= \textstyle\sum_{(z_1,x)\in Q}\ \rho_{ij}^{1z_1}(t)\cdot\left[c_{ij}^k(t)\ +\ \lambda_{jx}^k\big(t+c_{ij}^{1z_1}(t)\big)\right]\\
&= c_{ij}^k(t)\ +\ \textstyle\sum_{(z_1,x)\in Q}\ \rho_{ij}^{1z_1}(t)\lambda_{jx}^k\big(t+c_{ij}^{1z_1}(t)\big)
\end{aligned}$$

Note that this simplification is purely algebraic, and does not depend on the choice of the labels to combine, i.e. on the value of x. Since the new label is obtained as a function of the expectation $c_{ij}^k(t)$ (and of the travel time distribution) the model assigning a single expected value can be adopted without loss of generality.

Clearly, in both models the distribution is necessary for travel times; indeed, for $k=1$ the two models are equivalent: given $t'=t+c_{ij}^{1z_1}$ we have $t'\in A_{ij}^t$ and $p_{ij}^t(t')=\rho_{ij}^{1z_1}$.

Appendix B: number and size of strategies

We define a (somehow pathological) STD network that may help to figure out the inherent difficulty of history-adaptive route choice.

Given $K\geq 1$ consider the topological network $\mathcal{G}=(\mathcal{N},\mathcal{A})$ satisfying:

$$\mathcal{N}=\{u^i:\ 0\leq i\leq K+1\}\quad\cup\quad\{v^i:\ 0\leq i\leq K\};$$

$$\mathcal{A}=\{(u^i,v^i),\,(v^i,u^{i+1}):\ 0\leq i\leq K\}\quad\cup\quad\{(u^K,u^{K+1})\}.$$

We identify the origin $o=u^0$ and the destination $d=u^{K+1}$. Note that $\mathcal{G}$ is an o-d path plus the single arc (u^K,d). We define the STD network as follows.

- Each arc (u^i,v^i) with $i<K$ is stochastic but time independent: possible travel times are 1 and 2, with equal probability 1/2.
- Each arc (v^i,u^{i+1}) with $i<K$ is time-dependent but deterministic: when leaving v^i at time t the travel time is 2 if t modulo $3=1$ and 1 otherwise.
- Arcs (u^K,v^K), (u^K,d) and (v^K,d) have static travel time 1.

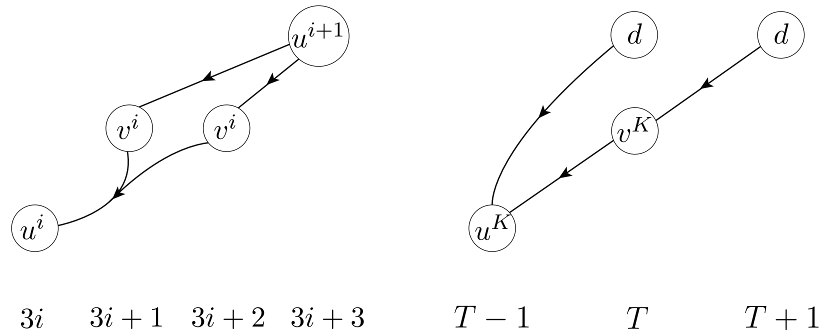


Figure 5: Fragments of the time-expanded hypergraph

- Each arc has a static and deterministic zero cost, except arc (u^K, d) that has cost one.

Figure 5 may help to understand the structure of the time-expanded hypergraph. The left part shows the fragment corresponding to arcs (u^i, v^i) and (v^i, u^{i+1}), with leaving time from node u^i at time $3i$; the right part involves arcs (u^K, v^K), (u^K, u^{K+1}) and (v^K, u^{K+1}). Note that the time subscript index of each node in the hypergraph is not shown. The STD network satisfies:

1. If leaving node o at time 0, then the only possible arrival/departure time at each node u^i with $i \leq K$ is $3i$.

2. Since the leaving time from node u^K is $3K = T - 1$ the possible arrival times at d are $T = 3K + 1$ and $T + 1 = 3K + 2$.

3. Routing decisions are possible only at node u^K and time $T - 1$. Hence there are only two time-adaptive strategies, corresponding to successor arcs (u^K, d) and (u^K, v^K), denoted by S^d and S^v, respectively.

4. Setting the time horizon $H = [0, T + 1]$, a complete description of the STD network or its input size is $O(K^2)$.

Theorem 4 *The number of history-adaptive strategies can be exponential in the number of time-adaptive strategies.*

Proof There are 2^K possible histories leading to node u^K. For each of these histories either node d or v^K may be the successor node of u^K leading to 2^{2^K} history-adaptive strategies compared to only two time-adaptive strategies. ■

Theorem 5 *The number of non-dominated labels corresponding to supported HAS that are optimal for a single WSS can be exponential in the input size and also in the number of supported TAS.*

Proof Time-adaptive strategies S^d and S^v yield labels $[T, 1]$ and $[T+1, 0]$, and are supported extreme solutions. Under history-adaptive route choice we have a set of $2^K + 1$ nondominated labels

$$\mathcal{Y}_H = \{\lambda^{(j)} = [T + \delta j,\, 1 - \delta j] : \; 0 \leq j \leq 2^K\}$$

where $\delta = 2^{-K}$, and each $\lambda^{(j)}$ is obtained by choosing arc (u^K, v^K) for j out of the 2^K possible histories at node u^K. Note that $\lambda^{(0)}$ and $\lambda^{(2^K)}$ correspond to S^d and S^v, respectively. Labels in $\mathcal{Y}_H$ identify points in the segment joining $[T, 1]$ and $[T+1, 0]$, thus they correspond to supported solutions, in particular, optimal solutions to a WSS with weights $w = [1, 1]$. ■

Note all HAS are supported HAS.

Corollary 2 *The number of (supported) efficient HAS can be exponential in the number of (supported) efficient TAS.*

Moreover, since the number of HAS is 2^{2^K} corresponding to $2^K + 1$ different labels we have.

Corollary 3 *The number of supported HAS corresponding to the same non-dominated label can be exponential in the input size.*

Let us now consider efficient labels at intermediate nodes, as they are generated by the labelling algorithm APS proposed by Opasanon and Miller-Hooks [7]. For the sake of simplicity, we associate these labels to nodes $u^0, \ldots, u^K$ in $\mathcal{G}$, since for each u^i there is a unique arrival/departure time $3i$. For node u^K we have two efficient labels $[1, 1]$ and $[2, 0]$. Now suppose that we have two labels $[a, b]$ and $[a', b']$ at node u^{i+1}. It is easy to verify that at node u^i we can obtain three efficient labels $[3+a, b]$, $[3+a', b']$ and $[3+(a+a')/2, (b+b')/2]$; note that the last label is the mid-point of the segment joining the first two ones. In this way we obtain three efficient labels at node u^{K-1}, five efficient labels at node u^{K-2}, …, $2^{K-i} + 1$ efficient labels at node u^i. Clearly, this gives $|\mathcal{Y}_H| = 2^K + 1$ efficient labels at node u^0.

Theorem 6 *The size of a single supported history-adaptive strategy, by means of the data structure devised by Opasanon and Miller-Hooks [7], can be exponential in the input size, i.e. exponential in the size of a TAS, which is linear in the input size.*

Proof Assume that K is even. We know we have at least $2^{K/2}$ efficient labels at node $u^{K/2}$; we can use these labels to obtain $2^{K/2-1}$ labels at node $u^{K/2-1}$, $2^{K/2-2}$ labels at node $u^{K/2-2}$,…, $2^0 = 1$ label at node u^0. In this way we define a single history-adaptive strategy that requires at least 2^K different labels to be represented, i.e., $O(2^K)$ space. By contrast, a TAS requires at most linear space in the input size. ■

Working Papers from CORAL – Centre for Operations Research Applications in Logistics

L-2008-05 Daniele Pretolani, Lars Relund Nielsen, Kim Allan Andersen & Matthias Ehrgott: Time-adaptive versus history-adaptive strategies for multicriterion routing in stochastic time-dependent networks.

L-2008-04 Christian Larsen: Comments to Özkaya et al. (2006).

L-2008-03 Christian Larsen: Derivation of confidence intervals of service measures in a base-stock inventory control system with low-frequent demand.

L-2008-02 Jens Lysgaard: The Pyramidal Capacitated Vehicle Routing Problem.

L-2008-01 Jens Lysgaard & Janni Løber: Scheduling participants of Assessment Centres.

L-2007-03 Christian Larsen: Note: Comments on the paper by Rosling (2002).

L-2007-02 Christian Larsen, Claus Hoe Seiding, Christian Teller & Anders Thorstenson: An inventory control project in a major Danish company using compound renewal demand models.

L-2007-01 Christian Larsen: The *Q(s,S)* control policy for the joint replenishment problem extended to the case of correlation among item-demands.

L-2006-11 Daniele Pretolani, Lars Relund Nielsen & Kim Allan Andersen: A note on "Multicriteria adaptive paths in stochastic, time-varying networks".

L-2006-10 Lars Relund Nielsen, Kim Allan Andersen & Daniele Pretolani: Bicriterion a priori route choice in stochastic time-dependent networks.

L-2006-09 Christian Larsen & Gudrun P. Kiesmüller: Developing a closed-form cost expression for an *(R,s,nQ)* policy where the demand process is compound generalized Erlang.

L-2006-08 Eduardo Uchoa, Ricardo Fukasawa, Jens Lysgaard, Artur Pessoa, Marcus Poggi de Aragão, Diogo Andrade: Robust Branch-Cut-and-Price for the Capacitated Minimum Spanning Tree Problem over a Large Extended Formulation.

L-2006-07 Geir Brønmo, Bjørn Nygreen & Jens Lysgaard: Column generation approaches to ship scheduling with flexible cargo sizes.

L-2006-06 Adam N. Letchford, Jens Lysgaard & Richard W. Eglese: A Branch-and-Cut Algorithm for the Capacitated Open Vehicle Routing Problem.

L-2006-05 Ole Mortensen & Olga W. Lemoine: Business integration between manufacturing and transport-logistics firms.

L-2006-04 Christian H. Christiansen & Jens Lysgaard: A column generation approach to the capacitated vehicle routing problem with stochastic demands.

L-2006-03 Christian Larsen: Computation of order and volume fill rates for a base stock inventory control system with heterogeneous demand to investigate which customer class gets the best service.

L-2006-02 Søren Glud Johansen & Anders Thorstenson: Note: Optimal base-stock policy for the inventory system with periodic review, backorders and sequential lead times.

L-2006-01 Christian Larsen & Anders Thorstenson: A comparison between the order and the volume fill rates for a base-stock inventory control system under a compound renewal demand process.

L-2005-02 Michael M. Sørensen: Polyhedral computations for the simple graph partitioning problem.

L-2005-01 Ole Mortensen: Transportkoncepter og IT-støtte: et undersøgelsesoplæg og nogle foreløbige resultater.

L-2004-05 Lars Relund Nielsen, Daniele Pretolani & Kim Allan Andersen: K shortest paths in stochastic time-dependent networks.

L-2004-04 Lars Relund Nielsen, Daniele Pretolani & Kim Allan Andersen: Finding the K shortest hyperpaths using reoptimization.

L-2004-03 Søren Glud Johansen & Anders Thorstenson: The (r,q) policy for the lost-sales inventory system when more than one order may be outstanding.

L-2004-02 Erland Hejn Nielsen: Streams of events and performance of queuing systems: The basic anatomy of arrival/departure processes, when focus is set on autocorrelation.

L-2004-01 Jens Lysgaard: Reachability cuts for the vehicle routing problem with time windows.

Handelshøjskolen i Århus

Aarhus School of Business

ISBN 9788778823076

Department of Business Studies

Aarhus School of Business
University of Aarhus
Fuglesangs Allé 4
DK-8210 Aarhus V - Denmark

Tel. +45 89 48 66 88
Fax +45 86 15 01 88

www.asb.dk