

Optimal routing policy problems in stochastic time-dependent networks

Song Gao ^{a,*}, Ismail Chabini ^b

^a *Caliper Corporation, 1172 Beacon Street, Newton, MA 02461, USA*

^b *Massachusetts Institute of Technology, Room 1-263, 77 Massachusetts Avenue, Cambridge, MA 02139, USA*

Received 3 December 2001; received in revised form 4 February 2005; accepted 7 February 2005

Abstract

We study optimal routing policy problems in stochastic time-dependent networks, where link travel times are modeled as random variables with time-dependent distributions. These are fundamental network optimization problems for a wide variety of applications, such as transportation and telecommunication systems. The routing problems studied can be viewed as counterparts of shortest path problems in deterministic networks. A routing policy is defined as a decision rule that specifies what node to take next at each decision node based on realized link travel times and the current time. We establish a framework for optimal routing policy problems in stochastic time-dependent networks, which we believe is the first in the literature. We give a comprehensive taxonomy and an in-depth discussion of variants of the problem. We then study in detail one variant that is particularly pertinent in traffic networks, where both link-wise and time-wise stochastic dependencies of link travel times are considered and online information is represented. We give an exact algorithm to this variant, analyze its complexity and point out the importance of finding good approximations to the exact solution. We then overview several approximations, and present a summary of theoretical and computational analyses of their effectiveness against the exact algorithm.

© 2005 Elsevier Ltd. All rights reserved.

Keywords: Routing policy; Adaptive routing; Stochastic; Dynamic; Network optimization

* Corresponding author. Tel.: +1 617 527 4700; fax: +1 617 527 5113.
E-mail address: song@caliper.com (S. Gao).

1. Introduction

Road traffic congestion is a significant problem of modern society. It is commonly recognized that building more infrastructure, which is usually politically, financially, and environmentally constrained, is not the only remedy to congestion. Nowadays, traffic measures to relieve congestion are generally based on the concept of making best use of the current infrastructure with advances in information technology, which is the underlying idea of intelligent transportation systems (ITS). Among various sub-systems of ITS, an advanced traveler information system (ATIS) aims to provide travelers with updated and useful information about network conditions, in hope that a better informed traveler can make a better decision, and collectively better decisions by many travelers would result in a relief from congestion. The value of ATIS is most evident when traffic conditions are stochastic. For example, when an incident happens, a timely notice by ATIS to travelers who plan to take the route with incident would be quite beneficial. Otherwise, in a network where traffic quantities are almost certain, travelers are already quite well informed and ATIS has little to provide.

An outstanding feature of congested traffic networks is the stochasticity in traffic quantities, such as travel time, link volume, queue length, and so on, on a day-to-day base. The travel time from home to work on a Monday morning could be different from that on a Tuesday morning, or another Monday morning of the next week. The randomness can come from multiple sources. One of the most significant sources is the disturbances to traffic networks, such as incidents, vehicle breakdown, bad weather, work zones, special events, and so on. Some of these disturbances are completely unpredictable, such as incidents and vehicle breakdown. Others are predictable to some extent, such as bad weather, work zones and special events, but usually there are prediction errors. A weather forecast is usually in a probabilistic format, e.g. a precipitation probability of 90%. Work zones and special events are scheduled, so they are less unpredictable in some sense, but the schedules might not be available to the travelers in a timely manner, and thus are actually unpredictable to travelers. These disturbances cause non-recurrent congestion which is a significant part of the total congestion, as described in the 2003 Urban Mobility Report by the Texas Institute of Transportation (Schrank and Lomax, 2003, p. V): “Crashes, vehicle breakdown, weather, special events, construction and maintenance activities greatly affect the reliability of transportation systems; these delays account for about 50% of all delay on the roads”. Traffic conditions with recurrent congestion, on the other hand, are also usually different from day to day, largely because of fluctuations in origin–destination (OD) trips. The fluctuations can be in both the total number of OD trips and the spread of OD trips over departure times. Travelers with non-commuting trip purposes might decide not to take a trip at a particular day, due to other personal business, and the no-travel decisions collectively result in a random number of OD trips. Travelers might also respond to congestion by shifting departure times from day to day, and thus there exists a random pattern in OD trips’ spread. Fluctuations in OD trips combined with random disturbances in supply can bring a lot of stochasticity to a traffic network.

Travelers make decisions on destination, mode, departure time, and route based on their information about the traffic network. The information can be obtained through a wide range of means: travelers’ own experience, word of mouth, radio broadcast, variable message signs (VMS), in-vehicle communication system, and so on. The information can be classified as a priori

or online. A priori information is about the general picture of the day-to-day fluctuations of traffic quantities, e.g. the travel time on a bridge is 30 s on average, but roughly once in a month, the travel time is unusually high, due to various reasons. Online information is about the traffic condition on a specific day, e.g. an incident just occurred on this bridge, and it will probably last for 20–30 min. This classification is meaningful only when there is stochasticity in the network, such that online information is different from a priori information. Destination, departure time and mode decisions are usually made at origins only and can be hardly changed en route, while routing decisions can be changed en route more easily and thus benefit more from online information. ATIS can provide both a priori and online information. Travelers only have personal experience on their selected routes. In order to obtain a priori information about the whole network, they need to go beyond their personal experience, and one of the good sources is ATIS. ATIS can provide travelers with reports of traffic conditions in the past and possibly predictions about the future, for the temporal and spatial ranges and in formats specified by travelers. Combining all sources of a priori information, travelers can form their own general pictures about the network. Nevertheless, the benefit of ATIS is primarily embodied through the provision of online information, especially in a network disturbed randomly by incidents, vehicle breakdowns, bad weather, work zones, special events, and so on.

There are various mechanisms for providing online information, i.e. various implementations of ATIS, and they differ in the spatial and temporal availability, the quality, and the format of information provided. A VMS is usually fixed in location and thus only travelers passing it can obtain the information. It is also limited in the amount of information it can provide, due to the limitation of the display panel. Usually it simply tells traveler that an incident happened somewhere, and sometimes with estimated delay on an affected major route. Radio-based or telephone-based systems can provide information to travelers anywhere in the radio coverage or when a telephone is available (with the popularity of cell phones, this actually can be anywhere). More detailed information is available compared to VMS, such as information on a number of alternative routes, or on some specific locations of interest to the traveler. There is still limitation in the amount of information provided, as travelers who listen to the radio or call an information provider usually process information in their mind with limited processing ability, given that driving already poses a large workload. More advanced in-vehicle systems usually contain a database of road map, travel times under normal conditions, records of past incidents, and so on, and can communicate with information centers to obtain very detailed and updated information. They usually have significant processing power, and can process a large amount of information, and display customized processing results as requested by travelers.

Travelers' routing decisions in a stochastic network with online information is conceivably different from those in a deterministic network. It is generally believed that adaptive routing will save travel time and enhance travel time reliability. For example, in a network with random incidents, if one does not adapt to an incident scenario, he/she could be stuck in the incident link for a very long time. However, if adequate online information is available about the incident and the traveler adapts to it by taking an alternative route, he/she can save travel time compared to the non-adaptive case. The adaptiveness also ensures that the travel time is not prohibitively high in incident scenarios, and thus provides a more reliable travel time.

It is therefore a very interesting research question how an individual traveler makes adaptive routing decisions in a stochastic and time-dependent network. The network is time-dependent,

since in an operational context, within-day dynamics should be considered. Preferably the research on adaptive routing should address the following two issues:

1. Traffic variables are not only random; they are also usually correlated link-wise and time-wise. We use link travel times as an example. If the randomness comes from weather, then link travel times of the whole network over a certain time period are correlated. If the randomness comes from incidents, then link travel times around the incident location and around the incident duration are correlated. It is desirable to capture these correlations.
2. As discussed before, there are a large variety of information provision mechanism and thus a large variety of information accessibility situations. It is desirable to formulate the problem so that simplified assumptions on information accessibility are avoided. Common simplified assumptions include no online information, and on the other extreme, full information (knowing the future for sure).

Specifically, we define a *stochastic time-dependent* (STD) network as a network whose link travel times are random variables with time-dependent distributions. We use the term “routing” in this paper to denote the task to move entities from one location to another in a given network. The movements need to be optimized according to one or multiple criteria, such as in minimum cost or maximum reliability. In this paper, optimal routing policy problems in stochastic time-dependent networks, abbreviated as “ORP problems in STD networks”, are studied. A *routing policy* is a decision rule that specifies which node to take next at each decision node based on the current time and realized network link travel times. An *optimal routing policy* (ORP) is a routing policy that moves a traveler on a network from the origin to the destination in least expected travel time. We will define these terms in a mathematical way later in this paper.

The focus of the discussion of this section has so far been on the routing policy. In fact, routing in an STD network can take one of the two forms: a path and a routing policy. A *path* is a pre-specified set of concatenated links. Travelers who follow a path make decisions a priori and take a fixed set of links, regardless of the network conditions revealed during their trips. In contrast, travelers who follow a routing policy make decisions en route and therefore can take different set of links, depending on actual network conditions. Two illustrative examples in Figs. 1 and 2 aim to show the difference between path-based and policy-based routings.

In Fig. 1, we seek to do routing from node 1 to node 4 and two paths: $a-b$ and $c-d$ are available. a and c are links, b and d are abstract representation of paths. The network is stochastic but not time-dependent. The data beneath the network shows the probabilistic distribution of travel times

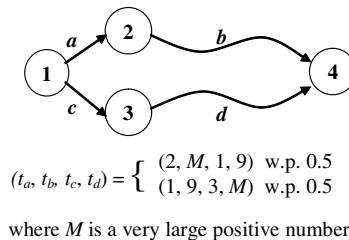


Fig. 1. Paths vs. routing policies in a time-invariant and stochastically dependent network.

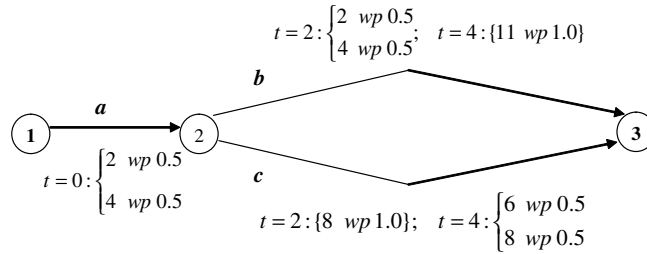


Fig. 2. Paths vs. routing policies in a time-dependent and stochastically independent network.

of different parts of the network. One of the paths will be blocked at any given time. Assume that one can learn actual realizations of travel times of link *a* and link *c* when one arrives at node 1.

If the routing outputs least expected travel time paths, we will choose path *a–b*, and the expected O–D travel time is $(2 + M) \times 0.5 + (1 + 9) \times 0.5 = 6 + M/2$. If the routing outputs least expected travel time routing policies, we will do the following: when the travel time on link *a* is 2, we choose path *c–d*; otherwise we choose path *a–b*. The expected O–D travel time of the policy is $(1 + 9) \times 0.5 + (1 + 9) \times 0.5 = 10$. An optimal routing policy defers the decision until some useful information is collected. In this example, the decision is delayed until one knows which path is blocked.

The example in Fig. 1 shows the value of information in a stochastic routing problem. The value of information in the example is due to the stochastic dependency of link travel times. When we learn the realizations on some of the links, we can make inference about other links so that we can make better decisions. There are cases when the link travel times are not stochastically dependent, but their time-dependency makes information valuable. An example in Fig. 2 shows the case.

Data beside each link shows the probability mass function (PMF) of that link travel time at each departure time in interest. Note that these link travel time random variables are stochastically independent. We also assume that only the arrival times are available to the traveler, so that a traveler does not know the actual realizations of travel times on links *b* or *c* at node 2. This assumption is made in order to show that the knowledge of arrival times can also benefit the routing decision-making in an STD network. We seek to do routing from node 1 to node 3 for departure time 0. The least expected travel time path is path *a–b*, with an expected travel time of $(2 + 2) \times 0.25 + (2 + 4) \times 0.25 + (4 + 11) \times 0.5 = 10$, while the expected travel time of path *a–c* is $(2 + 8) \times 0.5 + (4 + 6) \times 0.25 + (4 + 8) \times 0.25 = 10.5$. The optimal routing policy is as follows: when the arrival time at node 2 is 2, take link *b* next; if the arrival time at node 2 is 4, take link *c* next. The expected travel time of the routing policy is $(2 + 2) \times 0.25 + (2 + 4) \times 0.25 + (4 + 6) \times 0.25 + (4 + 8) \times 0.25 = 8$. The decision in a routing policy is delayed until the arrival time at a decision node is known. Two paths are involved in the routing policy, and the better one is chosen at any situation. The flexibility of choice can be viewed as the reason for the superiority of routing policies over simple paths.

The least expected travel time path ignores the possible information collected during the trip, and thus is generally less effective than an optimal routing policy. Indeed, a path is a special routing policy, where the next arc for a given node is the same for any departure time from that node and for all possible information collected until then. Thus we can view the least expected travel

time path problem as an optimal routing policy problem with additional side constraints. From the theory of mathematical programming, we know that the less constrained problem can yield solutions at least as good as the more constrained problem.

We see the value of information, either due to the stochastic dependency or the time dependency of link travel times, in the above two examples. We also see that different assumptions about the stochastic dependency of link travel times or assumptions about available link travel time realizations can be made. It follows that there are numerous variants of ORP problems in STD networks, some of which are studied in the literature, and some of which are to be explored.

The paper is organized as follows. In Section 2, we give a literature review and summarize the contributions of this paper. In Section 3, we give a framework for ORP problems in STD networks, and present a comprehensive taxonomy based on the network stochastic dependency and information access. An in-depth discussion of the variants is given. In Section 4, we study a problem variant which is particularly pertinent in traffic networks. An algorithm (Algorithm DOT-SPI) which provides an exact optimal solution to the variant is presented and implemented. The complexity analysis shows that the solution of the problem variant requires an exponential running time. Hence there is a need to develop effective and efficient approximations. In Section 5, computational results are given. They show the role of Algorithm DOT-SPI as a benchmark to study the effectiveness of approximations. An example is given in [Appendix A](#) to illustrate the working of Algorithm DOT-SPI. Finally in Section 6, we make some concluding remarks and give future research directions.

2. Literature review

Different variants of the optimal routing policy problem have been studied in the literature. Most of them assume that the underlying network is static (not dependent on time). Some other variants studied in the literature work with special cases of dynamic stochastic networks. They do not represent time explicitly. A limited number of papers have studied ORP problem in STD networks where time is explicitly represented, with various assumptions on network stochasticity and information availability. A comprehensive study of the problem is not available in the literature.

First let us look at the problem variants in static networks. The study of ORP problems in static networks is useful to the study of their dynamic versions. [Andreatta and Romeo \(1988\)](#) study the problem in a static network where the topology is stochastic. A stochastic topology is defined by a deterministic set of nodes N and a random set of links $A \in N \times N$. Each possible topology has a positive probability. A random link can be either active or not. When it is active, it is included in the network; when it is not active, it is removed from the network. The decision maker (DM) can learn whether a link is active or not once he/she reaches the node from which the link emanates. The DM can reroute once he/she finds out the next link is inactive. The notion “stochastic shortest path” is used, yet actually a routing policy problem is studied. A stochastic dynamic programming formulation of the problem is provided, with the definition of “information state” which gives the active/inactive links of the network revealed to the decision maker so far and based on which the recourse decision is made. It is pointed out that the complexity of the algorithm can grow exponentially with the number of links. Therefore a restricted version of the problem is studied and it is shown that polynomial algorithms exist for this particular case.

Polychronopoulos and Tsitsiklis (1996) extend the work of Andreatta and Romeo (1988). They study the problem both in networks with link travel times that are correlated and in networks with independent link travel times. For the dependent case, a joint distribution of link travel times is used to represent the stochastic network. We can see that the stochastic topology in Andreatta and Romeo (1988) is actually one special form of the joint distribution of link travel times. It is assumed that the travel time realizations of outgoing links of a given node are known and remembered by the traveler once he/she arrives at this node, and the realizations remain unchanged afterwards. As the traveler moves on the network from the origin to the destination, more link travel time realizations are learned, and the network becomes closer to a deterministic one. The concept of information set is introduced to represent the traveler's knowledge about the network. An information set is composed of support points that are consistent with the link travel times revealed so far. A "support point" is defined as a distinct value that a discrete random variable can take or a distinct vector of values that a discrete random vector can take, depending on the context. Thus a probability mass function (PMF) of a random variable (vector) is a combination of support points and the associated probabilities. When the information set becomes a singleton, the network becomes deterministic. A dynamic programming approach is presented where the stage of dynamic programming is labeled by the cardinality of the information set, starting from the smallest. Some of the main concepts in this paper originate from Polychronopoulos and Tsitsiklis (1996). A similar approach is designed for the independent case, with changes in the manner that the information set is defined. The algorithms, however, have exponential running times: the algorithm for the dependent case has a running time exponential in the number of support points, and the algorithm for the independent case exponential in the number of links. Some heuristics are given and the relationships between results from heuristics and exact algorithms are studied.

Cheung (1998) studies the problem with the same independent network assumptions as those in Polychronopoulos and Tsitsiklis (1996), except the assumption that two visits to the same node result in different realizations of outgoing link travel times. This assumption (which is termed as "reset" later by Provan (2003)) actually makes ambiguous the statement that the network is static, as the same link can take different travel times at different times, although the different travel times are generated from the same distribution. On the other hand, the reset assumption makes possible a simple recursive equation for the minimum expected travel times. An approach that mimics the classical label-correcting algorithm is presented. Computational tests are carried out to compare different implementations of the label-correcting approach. Waller and Ziliaskopoulos (2002) study the problem in a static network with limited spatial and time arc cost dependencies. In the one-step spatial dependence variant, it is assumed that given the cost of predecessor arcs, no further information is obtained through spatial dependence. This limited dependence is described by distributions of the cost of an arc, conditional the cost of each one of its upstream arcs. In fact, what is implied by this one-step spatial dependence is that a traveler has online information only about arcs traversed by himself/herself, otherwise, more probabilistic descriptions of the network is needed. The limited temporal dependence variant is the same as that studied by Cheung (1998). Algorithms are designed for the two variants and a third variant with a combination of the above two dependences. Provan (2003) studies the same problem as defined by Cheung (1998) with the extension that link travel times can be dependent. However, this relaxation from independent to dependent networks does not make the problem harder. In fact, the reset assumption makes the

term “dependent” limited in a sense, as one can never make inferences about travel times on links other than those going out of the current node. The same recursive equation as in [Cheung \(1998\)](#) is presented, but a different polynomial-time algorithm is designed. The author also gives a good classification of ORP problems in time-invariant networks, and shows the relationship between them. [Fu \(2001\)](#) studied a variant where travel time realizations of outgoing links of the current node are available to the vehicle before it enters the link. Independent link travel time random variables are assumed implicitly, according to the formulation of the problem. The time-dependency of link travel times is not explicitly considered. An efficient probabilistic approximation is proposed to solve the formulated problem and the advantage of adaptive routing systems is shown.

Some researchers studied ORP problem variants with stationary Markovian link costs and these variants can be viewed as an infinite horizon version of the dynamic ORP problem. [Polychronopoulos \(1992\)](#) assumes global information access and defines a combination of travel times of all links as a state. It is further assumed that the transition probability matrix is available, and that the occurrence of transition in unit time is dependent on network conditions. A dynamic programming formulation of the problem is proposed and it is shown that any standard Markov decision algorithm can solve the problem. [Psaraftis and Tsitsiklis \(1993\)](#) assume travel times of outgoing links of a given node are functions of condition at this node, which evolves as a Markovian chain. Markovian chains at different nodes are assumed to be independent and the network is acyclic. Under these assumptions, polynomial algorithms are developed.

[Hall \(1986\)](#) studies for the first time the time-dependent version of the ORP problem. It is shown that in an STD network, routing policies are more effective than paths. A dynamic programming approach is provided, where the stages of the dynamic program are the number of links from the destination node. It is implicitly assumed by [Hall \(1986\)](#) as well as by [Chabini \(2000, 2002\)](#) that routing policies are based only on arrival times at decision nodes. [Chabini \(2000\)](#) gives a dynamic programming algorithm based on the concept of decreasing order of time (DOT). The algorithm is optimal in the sense that no algorithms with better theoretical complexity exist. [Miller-Hooks and Mahmassani \(2000\)](#) study the ORP problem assuming time-wise and link-wise stochastically independent link travel time random variables. This assumption leads to routing policies based on arrival times at decisions nodes. A label-correcting algorithm is developed to solve the problem. Insight into the difference between an optimal routing policy problem and a least expected time path problem is provided. [Miller-Hooks \(2001\)](#) compares the label-correcting algorithm ([Miller-Hooks and Mahmassani, 2000](#)) and the dynamic programming algorithm working in decreasing order of time ([Chabini, 2000](#)) in both sparse transportation networks and dense telecommunication data networks. Recently, [Yang and Miller-Hooks \(2004\)](#) also extend the study of the time-adaptive routing policies to a signalized network.

[Bander and White \(2002\)](#) study the same time-adaptive routing policy problem, i.e. the link travel time random variables are time-dependent, but stochastically independent from each other, and thus the decision is dependent on the (time, node) pair. The major contribution of this paper is the design of a heuristic approach with a promising feature: it will terminate with an optimal solution if one exists, given that the heuristic function underestimates the true cost-to-go. The proposed heuristic has a significant computational advantage compared to dynamic programming, shown through computational tests.

We also make a brief literature review on the least expected time path problem in an STD network, as it is closely related to the ORP problem. [Fu and Rilett \(1998\)](#) model link travel times as a

continuous-time stochastic process. It is assumed that travel times on individual links at a particular point in time are stochastically independent, and the dependency between link travel times are modeled through the time-dependency of link travel time distributions. Relationships between the mean and variance of the travel time of a given path and the mean and variance of link travel times on that path are identified. A heuristic is designed in recognition of the computational intractability of the problem. Miller-Hooks and Mahmassani (2000) study the least expected time path problem under the same assumptions for the ORP problem. They establish a dominance rule for paths in STD networks and design a label-correcting-like algorithm. The worst-case complexity of the algorithm is exponential as a function of the network size, but computational tests on sparse transportation networks show the actual performance is practically linear with respect to the network size.

In summary, only some of the ORP problem variants have been studied. If we look at the ORP problem in a time-dependent context, even less work has been done. The only studied variant is the one with routing policies depending on arrival times at decision nodes. Traffic network variables, such as link travel times, are generally stochastically dependent, especially in congested networks, yet no study of the ORP problem in these networks exists in the literature. Furthermore, advanced traffic information systems (ATIS) and advanced traffic management systems (ATMS) require the modeling of information on traffic variables such as link travel times, yet the role of information in ORP problems in STD networks has not been studied in the literature.

This paper contributes to the knowledge base of ORP problems in STD networks in the following aspects:

1. Establish the first framework for ORP problems in STD networks;
2. Provide a comprehensive taxonomy of the studied problem, based on information access and network stochastic dependency;
3. Design an algorithm (Algorithm DOT-SPI) to one of the variants particularly pertinent in traffic networks, where the network dependency and the value of information are taken into account;
4. Point out the importance of designing good approximations for the ORP problems. Show the role of the exact algorithm as both a basis for designing approximation methods and a test bed to test the effectiveness of approximation methods;

The theoretical framework of ORP problems and the associated Algorithm DOT-SPI for the perfect-online-information variant are the main contributions of this paper. One should note that Algorithm DOT-SPI are not intended to be deployed in practice, as in practice it is difficult to obtain the a priori joint realization and the run time of the algorithm is high due to the fact that the problem variant it solves is an intrinsically difficult problem. Purposes served by the content of this paper include: (1) provision of a conceptual tool to understand the basic components of optimal routing policy problems, (2) understanding the computational complexity of ORP problems, (3) establishing a basis for designing efficient and effective approximation methods, and (4) development of exact algorithms that would serve as benchmarks in assessing the effectiveness of approximation methods. Due to space limit, not all details are provided in this paper and interested readers are advised to refer to Gao (2004) for an elaboration of topics discussed here.

3. Framework

3.1. The framework for optimal routing policy problems in stochastic time-dependent networks

From the literature review, we find that there is no formal definition of the routing problem in a stochastic time-dependent network. Various assumptions are made to define a stochastic network and to define how the realizations of the stochastic network are revealed to the travelers (decision makers). For example, in [Andreatta and Romeo \(1988\)](#), the topology of the network is stochastic; in [Polychronopoulos and Tsitsiklis \(1996\)](#), the whole static network is described by a joint distribution of link travel costs in the dependent case, and by marginal distributions of link travel times in the independent case; in [Polychronopoulos \(1992\)](#) and [Psaraftis and Tsitsiklis \(1993\)](#) the link costs evolve as Markov chains; in [Hall \(1986\)](#), [Chabini \(2000\)](#), [Miller-Hooks and Mahmassani \(2000\)](#), [Miller-Hooks \(2001\)](#), [Yang and Miller-Hooks \(2004\)](#), and [Bander and White \(2002\)](#), time-dependent networks are described by marginal distributions of link travel times. As for the revealing of the stochastic network, some assume that one learns the realization of a link travel cost once he/she arrives at the node from which the link emanates ([Andreatta and Romeo, 1988](#); [Polychronopoulos and Tsitsiklis, 1996](#); [Cheung, 1998](#); [Fu, 2001](#); [Waller and Ziliaskopoulos, 2002](#); [Provan, 2003](#)), while others do not state explicitly how travelers learn about the network conditions other than the arrival times at decision nodes ([Psaraftis and Tsitsiklis, 1993](#); [Chabini, 2000](#); [Miller-Hooks and Mahmassani, 2000](#); [Miller-Hooks, 2001](#); [Yang and Miller-Hooks, 2004](#); [Bander and White, 2002](#); [Pretolani, 2000](#)). Therefore the routing policies are time-adaptive, i.e. dependent on the (time, node) pair. However, we can generalize these various descriptions and assumptions. We also realize that the routing process in a stochastic network is a mapping from some knowledge of the network to a decision, and what knowledge is available and/or useful depends on specific assumptions about the network and the information access. A general set of optimality conditions is then possible with the formal definitions of the problem.

We establish the framework to provide a unified view of the optimal routing policy problem in a stochastic time-dependent network. In the rest of the paper, we will use “ORP problems” to denote the problems we are to study. Otherwise indicated, an ORP problem is a problem in a stochastic time-dependent network. We will be able to see the connections among various variants in the literature with the aid of the framework, and generate new variants that are required by specific applications. The generic optimality conditions can provide a general way of designing solution algorithms for variants of the problem.

3.1.1. The network

Let $G = (N, A, T, P)$ be a stochastic time-dependent network. N is the set of nodes and A is the set of arcs. The number of nodes and arcs are denoted respectively as $|N| = n$ and $|A| = m$. T is the set of time periods $\{0, 1, \dots, K - 1\}$.

Throughout this paper, a symbol with a \sim over it is a random variable, while the same symbol without the \sim is one specific support point of the random variable. The travel time on each link (j, k) at each time period t is a random variable $\tilde{C}_{jk,t}$ with finite number of discrete, positive and integral support points. Beyond time period $K - 1$, travel times are static and deterministic, i.e. travel times on link (j, k) at any time $t \geq K - 1$ is equal to $C_{jk, K-1}$. P is the probabilistic description of link travel times. Different descriptions exist because of different assumptions about

network probabilistic distributions. The most general one is in the form of a joint probability distribution of all link travel time random variables: $P = \{v_1, v_2, \dots, v_R\}$, where v_i is a vector with a dimension $K \times m$, $i = 1, 2, \dots, R$, and R is the number of support points. The r th support point has a probability p_r , and $\sum_{r=1}^R p_r = 1$. $C_{jk,t}^r$ is the travel time of arc (j, k) at time t in the r th realization. Note that we assume the underlying topology of the network is deterministic, as a network with stochastic topology can be transformed into a network with deterministic topology where one realization of the link travel times for the stochastic links can be infinity (or computationally, a very large positive number).

3.1.2. The decision

Assume the traveler can make decisions only at nodes. The decision is what node k to take next at each node, based on the current state $x = \{j, t, I\}$, where j is the *current-node*, t is the *current-time*, and I is the *current-information*. Current-information I is defined as a set of realized link travel times at current-node j and current-time t that are useful in making inferences about future link travel times. In the example of Fig. 1, the current-information would be the combination of travel time realizations of links a and c . The decision at node 1 can then be described as: when the state is $(1, t, \{2, 1\})$, take node 3 next; when the state is $(1, t, \{1, 3\})$, take node 2 next, for all t . Note that current-information is one component of a state and refers to link travel time realizations based on which the current decision is made, while a reference to “information” alone is in the general sense. A *routing policy* $\mu(x)$ is defined as a mapping from states to decisions (next nodes).

We look ahead of time at the current state after making the decision. The next state $y = \{k, \tilde{t}, \tilde{I}\}$ the traveler will occupy is uncertain, i.e. \tilde{t} and \tilde{I} are random variables. The link travel time of link (j, k) at time t conditional on I could be uncertain, resulting in an uncertain arrival time \tilde{t} at node k . The next current-information \tilde{I} is also uncertain, as \tilde{t} itself is uncertain. Even if \tilde{t} is certain, link travel time realizations between t and \tilde{t} could take multiple values, as the network is still stochastic to the traveler at the current state. However, for a given current state and a given decision, probabilities of all possible next states can be evaluated from the network probabilistic description P . Define a *state chain* $\{x_0, x_1, \dots, x_S\}$ as the series of states a traveler experiences during the trip, where x_S is a state with the destination node as its current-node. Current-nodes of a state chain form a path, and S is the number of links involved in the path. With a given initial state x_0 and a routing policy μ , one could experience multiple state chains. For example, the routing policies in Figs. 1 and 2 involve more than one path. Denote this set of possible state chains as $M(x_0, \mu)$.

3.1.3. The minimization problem

Assume we want to reach the destination in an optimal travel time. Since link travel times are random variables, there exist multiple criteria on what optimal travel times are. The expected travel time is used in this paper, as generally it is the primary criterion in routing choices. Other criteria, such as travel time variance and expected travel time schedule delay, and a combination of some of the criteria, will be explored in future research. Define t_x as the current-time of state x . A statement of the *optimal routing policy problem in a stochastic time-dependent network* is to find

$$\mu^* = \arg \min_{\mu} \{E_{\{x_0, x_1, \dots, x_S\} \in M(x_0, \mu)} (t_{x_S} - t_{x_0})\}$$

for a given initial state x_0 . The expectation is taken over all possible state chains, and the minimum is taken over all routing policies.

3.1.4. The optimality conditions

Let $e_\mu(x)$ denote the *expected travel time* to the destination node d when the initial state is x and the routing policy μ is applied. Define $A(j)$ as the set of adjacent nodes of node j , $\tilde{C}_{jk,t} \mid I$ as a travel time random variable of link (j,k) at time t conditional on current-information I , and $\tilde{I}' \mid I$ as a current-information random variable at the next node k and at time $t + \tilde{C}_{jk,t} \mid I$. Denote $Z(j, t)$ as the set of all possible current-information at node j and time t . For $\forall j \in N \setminus \{d\}$, $\forall t \in T$, $\forall I \in Z(j, t)$, the minimum expected travel time from initial state x , $e_{\mu^*}(x)$, and the corresponding optimal routing policy μ^* are solutions of the following system of equations:

$$e_{\mu^*}(j, t, I) = \min_{k \in A(j)} \left\{ E_{\tilde{C}_{jk,t}} \left[\tilde{C}_{jk,t} + E_{\tilde{I}'} \left[e_{\mu^*}(k, t + \tilde{C}_{jk,t}, \tilde{I}') \mid \tilde{C}_{jk,t} \right] \mid I \right] \right\}$$

$$\mu^*(j, t, I) = \arg \min_{k \in A(j)} \left\{ E_{\tilde{C}_{jk,t}} \left[\tilde{C}_{jk,t} + E_{\tilde{I}'} \left[e_{\mu^*}(k, t + \tilde{C}_{jk,t}, \tilde{I}') \mid \tilde{C}_{jk,t} \right] \mid I \right] \right\}$$

with the boundary conditions: $e_{\mu^*}(d, t, I) = 0$, $\mu^*(d, t, I) = d$, $\forall t \in T$, $\forall I \in Z(j, t)$. Since $\tilde{I}' \mid I$ is dependent on $\tilde{C}_{jk,t} \mid I$, we first take the expectation over $\tilde{I}' \mid I$ with given $\tilde{C}_{jk,t} \mid I$, and then take the expectation over $\tilde{C}_{jk,t} \mid I$. Note that we assume the realization of the decision is deterministic, i.e. the traveler will end up at node k if he/she chooses node k as his/her next node. Croucher (1978) studies the problem where the realization of the decision itself is stochastic. We do not discuss this case, as our original initiative in studying the ORP problem is for traffic applications where this case does not arise.

3.2. Taxonomy

The framework is abstract, as no concrete form of current-information I is specified. The current-information depends on two factors: *network stochastic dependency* defining the stochastic dependency of link travel time random variables, and *information access* defining which link travel time realizations are available to the travelers at any given time and given node. The taxonomy of the ORP problem is therefore along these two dimensions.

3.2.1. The taxonomy

Network stochastic dependency is characterized by link-wise and time-wise stochastic dependencies of link travel times. At one extreme, all the link travel time random variables are independent, both link-wise and time-wise. At the other extreme, all the link travel time random variables are completely dependent. There are numerous cases in between these two extremes, and we denote them as partial stochastic dependency.

Information access has the following four categories: (1) full information; (2) perfect online information; (3) partial online information; (4) no online information. Travelers with *full information* have knowledge of the realizations of all link travel times before the trip. Travelers with

Table 1
Taxonomy of the ORP problem

Network stochastic dependency	Information access			
	Full information	Perfect online information	Partial online information	No online information
No link-wise and no time-wise dependency	WS (wait-and-see)	Group 1	Group 1	NOI
Complete dependency	WS (wait-and-see)	Group 2	Group 3	NOI
Partial dependency	WS (wait-and-see)	Group 2	Group 3	NOI

perfect online information have knowledge of the realizations of all link travel times up to current time period. Travelers with *partial online information* only have knowledge of part of the link travel time realizations and the restrictions in online information can be either temporal or spatial or both. Travelers with *no information* have no knowledge of any of the realizations and the only knowledge they have about the current state is the current-node and current-time. Table 1 gives a possible taxonomy along the two dimensions.

3.2.2. Discussion of variants

In the discussion of the variants listed in Table 1, we focus on the specification of current-information for each variant and the resulting implications for algorithm design. A general rule in determining current-information is as follows: information access determines which link travel times have the potential to be included in current-information, while network stochastic dependency determines whether all the available link travel time realizations are necessary. The unnecessary link travel times can be eliminated so that the dimension of current-information is minimized.

For example, assume we are equipped with the most advanced traffic information system so that we know the realizations of all link travel times up to the current time (i.e. perfect online information). Presumably we hope we can make use of all the available information. However, assume all the link travel time random variables are stochastically independent, implying that knowledge about one link at one time cannot help travelers make inferences about any other link at any time, then a large part of the information is not useful and the current-information could contain much fewer link travel times than those available. This rather extreme case shows how the two factors act together to characterize the current-information, and we will see more in the following discussion. Throughout the discussion, we assume that a user always knows his/her current-node and current-time (not necessarily remembering the arrival times at past nodes), and any knowledge about link travel time realizations can only be represented by the current-information. This implies that if we say the current-information is empty, then the user cannot obtain the travel time on a link by taking the difference between the arrival times at the two end nodes of that link.

3.2.2.1. The WS variant. The full information variant is a special case. We borrow a term from stochastic programming to denote the variant as WS (wait-and-see). In WS, the current-information I includes all the link travel times, so travelers can know the network deterministically a priori. This variant is not realistic, but it is discussed here as it results in lower bounds for all other

variants of the ORP problem. Thus it can be used as a benchmark in the robustness analysis of solutions to other variants. Under full information, the ORP problem reduces to multiple deterministic dynamic shortest path problems, each of which works on a deterministic network defined by one of the R support points $\{v_1, v_2, \dots, v_R\}$. Since we are working only on deterministic networks, network stochastic dependency does not make any difference in algorithm design. Algorithm DOT (Chabini, 1997) with a running time of θ ($SSP + nK + mK$) for all-to-one shortest path problem, where SSP is the running time of a classical static shortest path algorithm, can be used to solve the individual deterministic dynamic shortest path problems.

3.2.2.2. The no-online-information variant. The other extreme is the case when no online information (NOI) is available. The lack of online information prevents travelers from making any useful inferences about future network conditions. The current-information is an empty set at any point in space and time, and decisions depend only on current-node and current-time. This is true for any kind of stochastic dependency, which is another example besides the WS variant showing the role of information in defining an ORP problem. The optimal conditions in Section 3.1.4 thus reduce to

$$e_{\mu^*}(j, t) = \min_{k \in A(j)} \left\{ \min_{\tilde{C}_{jk,t}} E \left[\tilde{C}_{jk,t} + e_{\mu^*}(k, t + \tilde{C}_{jk,t}) \right] \right\}$$

$$\mu^*(j, t) = \arg \min_{k \in A(j)} \left\{ \min_{\tilde{C}_{jk,t}} E \left[\tilde{C}_{jk,t} + e_{\mu^*}(k, t + \tilde{C}_{jk,t}) \right] \right\}$$

Several algorithms have appeared in the literature to solve this variant (Hall, 1986; Chabini, 2000; Miller-Hooks and Mahmassani, 2000; Miller-Hooks, 2001). It should be noted that the assumption that link travel times are independent is neither sufficient nor necessary to validate the formulation. It is not necessary, because variants with dependent link travel times and no online information can also have this formulation. It is not sufficient, because if the realizations of outgoing link travel times at current-time are available in an independent network, the current-information is no longer an empty set and thus the above formulation of the problem is no longer valid. Algorithm DOT-S (Chabini, 2000) to this variant has an optimal running time of θ ($SSP + nKQ + mKQ$) for all-to-one-all-departure-time problems, where Q is the maximum number of realizations of a single link travel time.

3.2.2.3. The independent variants. In the rest of the section, we discuss variants with some online information access. The above discussion shows that sometimes information access alone can determine the current-information, as in the case of variants WS and NOI. On the other hand, network stochastic dependency sometimes can play a very important role in determining the current-information. This can be shown by the variants in Group 1. Variants in Group 1 have stochastically independent link travel times, thus no inferences about the future can be made, even if we could have perfect online information. The only useful knowledge is about the adjacent links of the current-node at the current-time. As long as this amount of information is available, variants in Group 1 share the same problem formulation, where the current-information I is the com-

bination of adjacent link travel times at current-time. When the knowledge about the adjacent links of the current-node at the current-time is not available, the current-information becomes an empty set and the problem has the same formulation as variant NOI. Note that the name “NOI” represents a variant where current-information I is an empty set. No online information is only a sufficient condition to validate the formulation. We choose “NOI” as the name, as it is intuitive to obtain the idea of an empty current-information from the no-online-information assumption. However, we should remember that there are other conditions that can validate the “NOI” formulation. Besides the one we discussed above, one of them is: no time-wise dependency and no knowledge about the adjacent links of the current-node at the current-time.

3.2.2.4. Variants with complicated information access and stochastic dependency. Variants in Group 2 and Group 3 generally have complex current-information. All available link travel time realizations are potentially useful and could be included in the current-information I . Network stochastic dependency can be utilized to eliminate unnecessary link travel times from the current-information, but the judgment sometimes requires very smart work and the resulted dimension reduction of current-information may not compensate for the extra effort needed to distinguish them. In summary, the determination of current-information for variants in these two groups depends largely on the actual assumptions on both information access and network stochastic dependency.

Most transportation networks belong to Group 2 and Group 3. For example, a typical urban traffic network can be divided into several zones and we can assume that traffic within one zone is highly dependent, while has weak relationship with traffic in the others. On the other hand, we can assume that only traffic conditions within the last 1 h are helpful in predicting the future. It is also very likely that there are several local traffic information centers that provide information to vehicles within their respective functional ranges. All these assumptions about network stochastic dependency and information access complicated the problem, and careful modeling is required.

We distinguish between Group 2 and Group 3, because complexity of algorithms for Group 2 and Group 3 could differ greatly. The complexity of an algorithm for the ORP problem depends on the maximum number of possible current-information. For the sake of convenience of presentation, assume the partial online information is only partial in the spatial dimension, not in the temporal dimension. With perfect online information, the current-information is composed of all link travel times up to current-time t , and the maximum number of current-information is just the maximum number of joint distinctive values that these $(t+1)m$ random variables can take, which is at most R . With partial spatial online information, however, the current-information is composed of travel times on links around the path (what specific links are included depends on specific assumptions about “partial” spatial dependency) from the origin to the current-node. Therefore the current-information can be $2^{(t+1)m}$ different sets of link travel times. As each set of link travel times has at most R distinctive values, the maximum number of possible current-information is $2^{(t+1)m}R$. The maximum numbers of current-information in these two groups differ in a ratio of $2^{(t+1)m}$, which is significant. It is shown by Polychronopoulos and Tsitsiklis (1996) that in a static network, the maximum number of current-information with partial online information is $2^R - 1$. However, this is a quite loose upper bound. A tighter upper bound obtained by applying the above logic would be $2^m R$.

The dynamic shortest path problem in acyclic networks with Markovian arc costs studied by Psaraftis and Tsitsiklis (1993) can be viewed as an infinite horizon version in Group 3. The

assumption of acyclic networks implies that no link will be visited twice, so it is not helpful to keep information of any already traversed links. Therefore the dimension problem of current-information with partial spatial online information as discussed above does not exist in this case. This assumption along with the infinite horizon assumption makes a polynomial running time algorithm possible.

4. Variants with perfect online information (POI)

We study in detail the variant with perfect online information. We assume the current-information includes all available link travel times. As discussed in the previous section, a variant with perfect online information is easier to study than variants with partial online information. The algorithm to variant POI also provides many basic elements that can be used in developing algorithms for other variants with online information. In this section, we present an operational algorithm DOT-SPI for the perfect online information variant. The joint distribution representation of a network is adopted. We then proceed with the complexity analysis and point out the importance of finding good approximations for ORP problems.

4.1. Algorithm DOT-SPI

We have a network as described in Section 3.1.1. We seek to find the least expected travel times from all nodes at all departure times with all possible current-information to a certain destination node d . We assume that travelers have perfect online information about link travel times. Mathematically speaking, at any time t , any traveler has knowledge of the realizations of $\tilde{C}_{jk,t'}$, $\forall (j,k) \in A$, $\forall t' \leq t$.

We use a different way to represent the concept of current-information. The current-information defined in the framework of the ORP problem is composed of link travel times. This definition is not convenient for the implementation of the algorithm. At each current-time t , each joint realization of the $(t+1)m$ random variables, $\tilde{C}_{jk,t'}$, $\forall (j,k) \in A$, $\forall t' \leq t$, corresponds to a unique subset of $\{v_1, v_2, \dots, v_R\}$, therefore we define a new term as the counterpart of current-information in algorithm design. Let $\pi_{jk,t}$ be the realization of $\tilde{C}_{jk,t}$ we have already learned until the current-time. Define the *event collection* $EV := \{v_r \mid C_{jk,t'}^r = \pi_{jk,t'}, \forall (j,k) \in A, \forall t' \leq t, \text{ for a certain } t\}$. This is the set of support point candidates after we collect information at time t . As we collect more information (i.e. t increases), the size of EV remains the same or decreases. When EV becomes a singleton, we obtain a deterministic network and can apply any deterministic dynamic shortest path algorithm. Let $EV(t)$ be the set of all possible event collections at time t and the element of $EV(t)$ is an event collection $EV = \{v_r \mid C_{jk,t'}^r = \pi_{jk,t'}, \forall (j,k) \in A, \forall t' \leq t\}$. Specifically, $EV(K-1) = \{\{v_1\}, \{v_2\}, \dots, \{v_R\}\}$. All the possible event collections can be generated in preprocessing. Here are some important facts about event collections:

1. There is no overlapping among elements of $EV(t)$ for a given t , so there are at most R event collections at any certain time t ($|EV(t)| \leq R$), and at most RK event collections in total.
2. Any element of $EV(t)$ is a subset of an element of $EV(t-1)$.
3. $|EV(t)| \geq |EV(t-1)|$.

A possible scheme of event collections is in Fig. 3. The rows represents time points in increasing order, i.e. the first row represents the first time point. Each cell in the last row represents a single support point v_r , implying that the network becomes deterministic beyond time period 5. At each time t , cells within the bold boundary form an event collection. For example, at time 0, $\{v_1, \dots, v_{10}\}$ is one event collection, and $\{v_{11}, \dots, v_{15}\}$ is the other. At time 1, when more link travel time realizations are available, $\{v_1, \dots, v_{10}\}$ is split into three event collections $\{v_1\}$, $\{v_2, \dots, v_5\}$, and $\{v_6, \dots, v_{10}\}$. Other event collections are obtained similarly.

Let $e_{\mu^*}(j, t, EV)$ be the minimum expected travel time to the destination node d if the departure from node j happens at time t with the event collection EV . Let $\mu^*(j, t, EV)$ be the next arc to take out of node j to realize $e_{\mu^*}(j, t, EV)$. Assume we select arc (j, k) out of node j . At the end of the journey along arc (j, k) , we have a new event collection EV' which is one of the possible event collections at time $t + \pi_{jk,t}$. The probability of a certain EV' can be evaluated as following:

$$\Pr(EV' | EV) = \frac{\sum_{r|r \in EV' \cap EV} p_r}{\sum_{r|r \in EV} p_r}, \quad \forall EV' \in EV(t + \pi_{jk,t}), \quad \forall EV \in EV(t)$$

Note that $EV' \cap EV = \emptyset$ or EV' .

The optimality conditions for the problem are:

$$\begin{aligned} e_{\mu^*}(j, t, EV) &= \min_{k \in A(j)} \{ \pi_{jk,t} + E_{EV'}[e_{\mu^*}(k, t + \pi_{jk,t}, EV')] \} \\ &= \min_{k \in A(j)} \left\{ \pi_{jk,t} + \sum_{EV' \in EV(t + \pi_{jk,t})} e_{\mu^*}(k, t + \pi_{jk,t}, EV') \times \Pr(EV' | EV) \right\}, \quad \text{for } j \neq d, \\ e_{\mu^*}(d, t, EV) &= 0, \quad e_{\mu^*}(j, t \geq K - 1, EV) = e(j, K - 1, EV) \quad \forall t \in T, \quad \forall EV \in EV(t) \end{aligned}$$

The solution of these equations can be carried out in a decreasing order of time, since the evaluation of $e_{\mu^*}(j, t, EV)$ only depends on $e_{\mu^*}(j, t', EV')$, where $t' > t$. At time $K - 1$ or beyond, the network becomes deterministic and static, and we can use any deterministic static shortest path algorithm to compute $e_{\mu^*}(j, t, EV)$, $\forall j \in N$, $\forall t \geq K - 1$, $\forall EV \in EV(K - 1)$. Denote the algorithm as DOT-SPI, a counterpart of Algorithm DOT (Chabini, 1997, 1998) to solve a stochastic problem with perfect information. The statement is as follows. An illustrative example to show how Algorithm DOT-SPI works can be found in Appendix A.

$t=0$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
$t=1$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
$t=2$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
$t=3$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
$t=4$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}
$t=5$	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}

Fig. 3. A possible scheme of event collections.

Algorithm DOT-SPI*Initialization*

- Step 1: Construct $EV(t)$, $t = 0, \dots, K - 1$
 Call Generate_Event_Collection (see the statement below)
 Step 2: Compute $e_{\mu^*}(j, K - 1, EV)$, $\forall j \in N$, $\forall EV \in EV(K - 1)$
 Step 3: $e_{\mu^*}(j, t, EV) \leftarrow +\infty$, $\forall j \in N \setminus \{d\}$,
 $e_{\mu^*}(d, t, EV) \leftarrow 0$,
 $\forall t < K - 1$, $\forall EV \in EV(t)$

Main Loop

- Step 4: For $t = K - 1$ down to 0
 For each $EV \in EV(t)$
 For each arc $(j, k) \in A$
 $temp_e = \pi_{jk,t} + \sum_{EV' \in EV(t + \pi_{jk,t})} e_{\mu^*}(k, t + \pi_{jk,t}, EV') \times \Pr(EV' | EV)$
 If $temp_e < e_{\mu^*}(j, t, EV)$ then
 $e_{\mu^*}(j, t, EV) = temp_e$
 $\mu^*(j, t, EV) = k$

Generate_Event_Collection

- $D = \{\{v_1, \dots, v_R\}\}$
 For $t = 0$ to $K - 1$
 For each arc $(j, k) \in A$
 For each disjoint set $S \in D$
 $w =$ number of distinct values among $C_{jk,t}^r$, $\forall r | v_r \in S$;
 Divide S into disjoint sets S'_1, S'_2, \dots, S'_w ,
 such that $C_{jk,t}^r$ is constant $\forall r | v_r \in S'_i$, $i = 1, \dots, w$ and $\bigcup_i S'_i = S$;
 $D' \leftarrow D \setminus \{S\} \cup \{S'_1, S'_2, \dots, S'_w\}$;
 Next S
 $D \leftarrow D'$
 Next (j, k)
 $EV(t) \leftarrow D$;
 Next t

4.2. Complexity analysis

The basic step in Generate_Event_Collection is the division of S into disjoint sets. This can be done by sorting the elements in S in time $\theta(s \ln s)$, where s is the cardinality of S . For a given time and a given link, all S are mutually exclusive and collectively exhaustive over all support points. Assume there are u such disjoint sets for a given time and a given link, S_1, S_2, \dots, S_u , and $1 \leq u \leq R$. Therefore the sorting of all the u sets takes time $\theta(\sum_{i=1}^u s_i \ln s_i) = \theta(\ln \prod_{i=1}^u s_i^{s_i}) = O(\ln(s_1 + s_2 + \dots + s_u)^{s_1 + s_2 + \dots + s_u}) = O(R \ln R)$. On the other hand, the sorting has to retrieve all

the R joint supports at least once, so the running time is also $\Omega(R)$. Altogether constructing event collections takes time $O(mKR \ln R)$ and $\Omega(mKR)$. Step 2 is solving R static shortest path problems, so the running time is $\theta(R \times \text{SSP})$, where SSP is the running time to solve a static shortest path problem. Step 3 takes time $\theta(KRn)$. At a given time t and for a given link (j, k) , the evaluation of all $\Pr(EV'|EV)$ takes time $\theta(R)$. There are altogether K time periods and m links, so the main loop has a running time of $\theta(mKR)$.

To sum up, Algorithm DOT-SPI has a complexity of $O(mKR \ln R + R \times \text{SSP})$ and $\Omega(mKR + R \times \text{SSP})$. This algorithm is strongly polynomial in R , however R could be an exponential function of m . If the link travel times are highly dependent, we expect that R is much less than Q^m , where Q is the maximum number of support points for a single link travel time. Other variants with less online information could also have running time exponential in number of link travel time random variables involved in current-information.

5. Computational tests

In this section, we study Algorithm DOT-SPI and some approximation algorithms computationally. The objectives of the computational tests are two-folded: (1) to show that Algorithm DOT-SPI can be implemented; and (2) to demonstrate the role of Algorithm DOT-SPI as a laboratory test bed to study the effectiveness of approximation methods. We introduce the four approximations to be compared and results are presented and discussed.

5.1. Four approximations

We study four approximations: (1) *certainty equivalent (CE)* approximation; (2) *no-online-information (NOI)* approximation; (3) *open-loop feedback certainty equivalent (OLFCE)* approximation; and (4) *open-loop feedback no-online-information (OLFNOI)* approximation. The CE approximation replaces every link travel time random variable by its expected value. Thus it transforms a stochastic network into a deterministic network. It then applies any dynamic shortest path problem algorithm to obtain an “optimal” path. The running time of CE is the same as that of a deterministic dynamic shortest path algorithm. The NOI variant can be solved in polynomial time as stated in the discussion of variants in Section 3.2.2, and could serve as an approximation to POI. OLFCE is an improved certainty equivalent approximation. At each decision node, travelers employ a CE that replaces every link travel time random variable in later times by its expected value conditional on the network conditions realized so far. Travelers follow the resulted “optimal” path until a new decision node is reached. At that time, a CE is applied again, conditional on the updated network conditions. The running time of OLFCE is $\min(K, R)$ times the time to solve one CE. The transition from CE to OLFCE suggests a new approximation OLFNOI developed from the NOI approximation. Similar to OLFCE, at each decision node, travelers employ an NOI approximation that works on the marginal distributions of link travel times conditional on the network conditions realized so far. Travelers follow the resulted “optimal” routing policy until a new decision node is reached. At that time, an NOI approximation is applied again, conditional on the updated network conditions. Theoretical study shows that (1) Algorithm DOT-SPI always gives solutions with less expected travel times than any of the four approximations

do; and (2) solutions from approximations could be arbitrarily worse in absolute value than those obtained by running the exact algorithm. For a more detailed study of the four approximations, please refer to [Gao and Chabini \(2002\)](#).

5.2. Computational tests

In this section, computational tests are designed to use Algorithm DOT-SPI as a laboratory test bed to study the effectiveness of the four approximations presented in the previous subsection. Details on the test design can be found in [Gao and Chabini \(2002\)](#).

The results of the tests are shown in [Figs. 4–7](#). The upper graph in each figure shows the results for all the four approximations. The lower graph in each figure shows the results for only the two open loop feedback approximations, as they are different in scale from the other two. The Y-axis

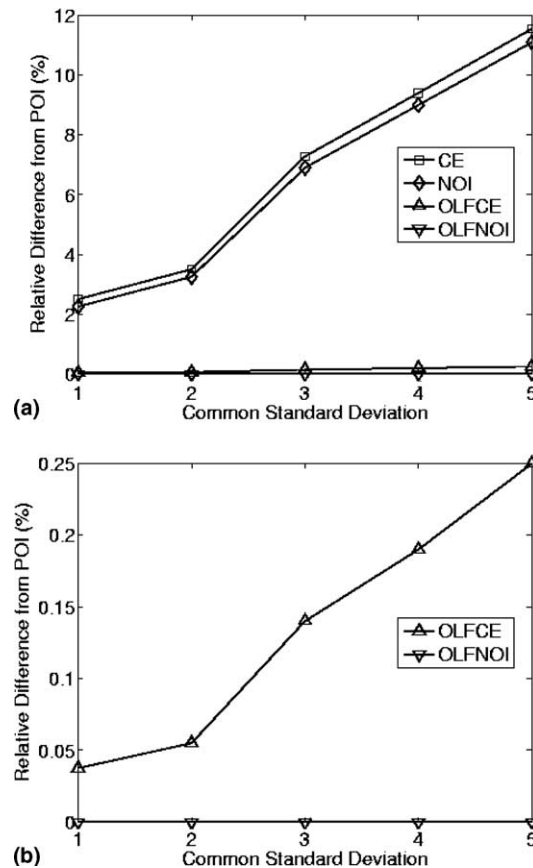


Fig. 4. Percent differences of approximations relative to Algorithm DOT-SPI as functions of the common standard deviation of link travel times (with 10 nodes, 30 links, 20 time periods, 100 support points, 10 as the common mean link travel time, and 0.5 as the common correlation coefficient of link travel times).

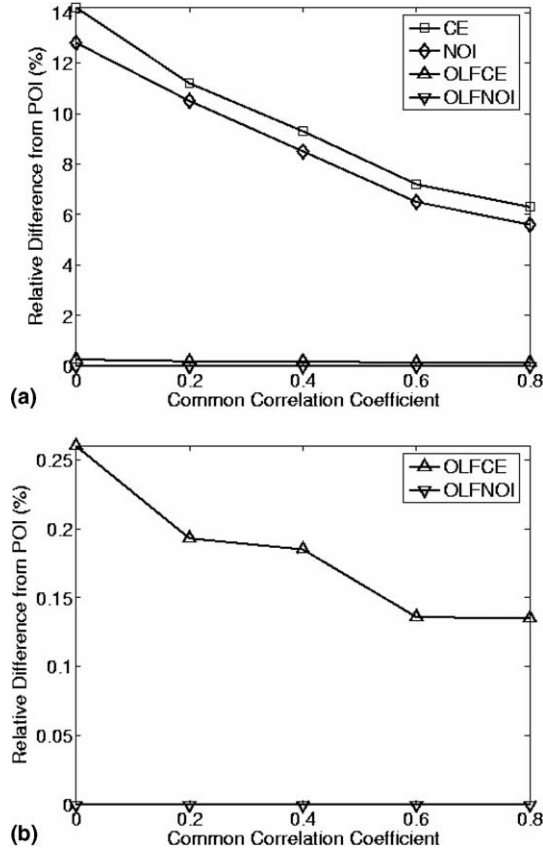


Fig. 5. Percent differences of approximations relative to Algorithm DOT-SPI as functions of the common correlation coefficient of link travel times (with 10 nodes, 30 links, 10 time periods, 100 support points, 5 as the common mean link travel time, and 1 as the common standard deviation of link travel times).

shows the percent relative difference between results from Algorithm DOT-SPI and results from the four approximations. The definition of the percent relative difference is as follows:

$$\Delta_{\text{approximation}} = \frac{\sqrt{\sum_{t=0}^{K-1} \sum_{j=1}^N (\text{POI}(j, t) - \text{approximation}(j, t))^2}}{\sqrt{\sum_{t=0}^{K-1} \sum_{j=1}^N (\text{POI}(j, t))^2}}$$

where approximation can take the value CE, NOI, OLFCE, or OLFNOI, $\text{POI}(j, t) = E_{I \in Z(j, t)} [\text{POI}(j, t, I)]$, $\text{POI}(j, t, I)$ is the expected travel time from state (j, t, I) to destination d when the routing policy generated by Algorithm DOT-SPI is followed, and $\text{approximation}(j, t) = E_{I \in Z(j, t)} [\text{approximation}(j, t, I)]$, $\text{approximation}(j, t, I)$ is the expected travel time from state (j, t, I) to destination d when the routing policy generated by the approximation is followed. In the computation of percent relative differences, the weight of each (j, t) pair is the same. This implies that we assume the demand to the destination node is distributed evenly across both space and time.

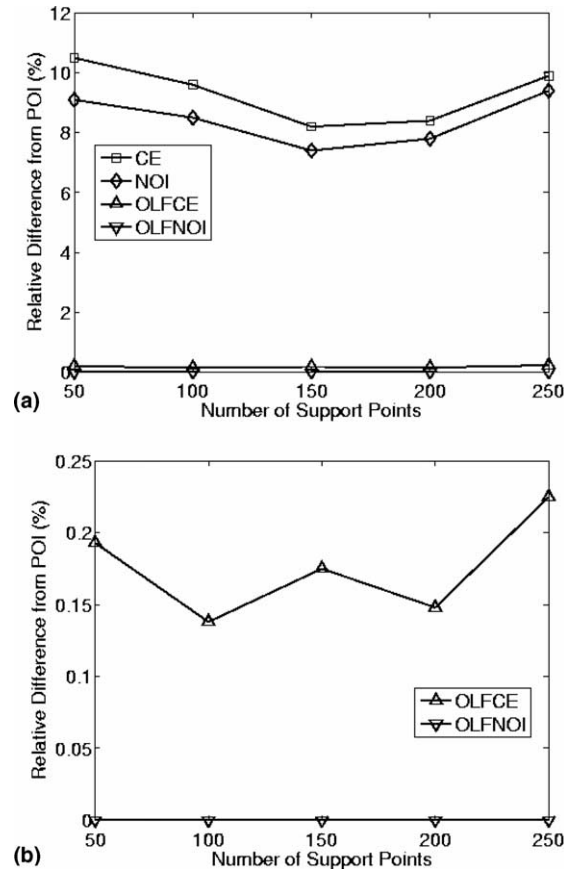


Fig. 6. Percent differences of approximations relative to Algorithm DOT-SPI as functions of the number of support points (with 10 nodes, 30 links, 10 time periods, 5 as the common mean link travel time, 2 as the common standard deviation of link travel times, and 0.5 as the common correlation coefficient of link travel times).

There are some general observations across all figures. The magnitude of the percent relative difference for CE and NOI is around 10, and that for OLFCE and OLFNOI is very close to zero. The performance of the two open loop feedback approximations are very close to that of Algorithm DOT-SPI, partly due to the small number of support points. When the number of support points is small, online information is quite valuable and the network becomes deterministic very soon. Note that OLFCE and OLFNOI do make use of this valuable online information. Since in a deterministic network, CE, NOI and Algorithm DOT-SPI give the same expected travel times, it is not surprising that OLFCE and OLFNOI have very close results as Algorithm DOT-SPI in this situation. Another interesting observation is that generally NOI performs better than CE and OLFNOI better than OLFCE, although the theoretical study shows that CE could perform better than NOI and OLFCE could perform better than OLFNOI. This is reasonable, as NOI outputs routing decisions that are dependent on actual arrival times at intermediate nodes, while CE outputs paths that totally ignore such information. It is conjectured that when the average number of

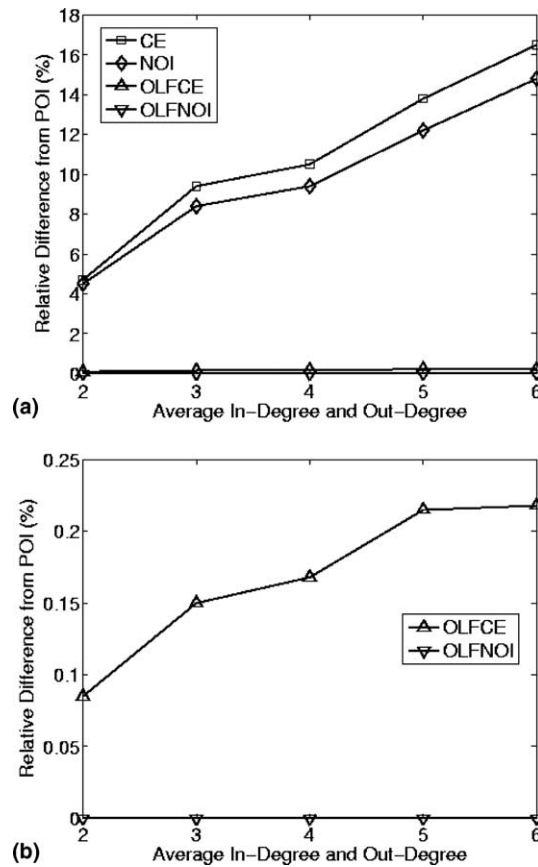


Fig. 7. Percent differences of approximations relative to Algorithm DOT-SPI as functions of the average in-degree and out-degree (with 15 nodes, 10 time periods, 100 support points, 5 as the common mean link travel time, 2 as the common standard deviation of link travel times, 0.5 as the common correlation coefficient of link travel times, and 2 times the average in- and out-degree as the maximum in- and out-degree).

possible next nodes is small, the performance of CE and NOI is close, since routing policies would nearly reduce to paths in this situation.

Fig. 4 shows the percent relative difference as an increasing function of the common standard deviation of link travel times. Note the mean link travel time is also common across time and space. When the standard deviation is large, the link travel times are more dispersed, and thus the expected travel times of different paths (routing policies) are more likely to be apart from each other. This magnifies the difference between optimal and sub-optimal solutions. Fig. 5 shows the percent relative difference as a decreasing function of the common correlation coefficient of link travel times. A positive correlation coefficient of random variables X and Y provides a measure of extent to which the signs of $x - E(X)$ and $y - E(Y)$ “tend” to be the same. As we have a common mean for all link travel times, the positive correlation coefficient actually indicates how close x and y are. When link travel times are close, the difference between optimal and sub-optimal solutions is reduced. Fig. 6 shows the percent relative difference as a function of the number of support

points. The number of support points represents, among others, the extent of discretization. There is no definite relationship shown in the figure. Further computational tests are needed to study the effect of discretization in a larger range. Fig. 7 shows the percent relative difference as an increasing function of average in-degree and out-degree. The two degrees are set to be equal in the tests. As the average degree increases, the travelers have more choices of the next node. Therefore more paths are involved in an optimal routing policy, and the optimal routing solutions have more chance to achieve lower travel times than the sub-optimal solutions.

6. Conclusions and future directions

We study optimal routing policy problems in stochastic time-dependent networks. There are many variants of ORP problems in STD networks, however they can be integrated in a framework. We establish such a framework, including a general description of the STD network, the decision process, the problem statement, and the optimality conditions. We provide a taxonomy of the ORP problem, based on network stochastic dependency and information access. These two factors determine the current-information based on which the routing decisions are made. Numerous variants exist according to the taxonomy, and we provide insights into most of them, focusing on the specification of the current-information. We then study in detail a variant (POI) that is particularly pertinent to the modeling and routing applications in traffic networks. Variant POI takes into account the stochastic dependency among link travel times and the role of information in routing decision making, which is a realistic depiction of traffic systems equipped with ATIS and/or ATMS. An exact algorithm (Algorithm DOT-SPI) is designed and implemented for this variant.

The complexity analysis of Algorithm DOT-SPI reveals the need to design good approximations for the ORP problem. Four approximations are presented. Their properties are studied both theoretically and computationally. There is a trade-off between effectiveness and efficiency for all approximations, i.e. they could have satisfactory running times, but their results could be arbitrarily worse in absolute value than those obtained by running the exact algorithm. The computational tests assess the performance of approximations as a function of certain parameters of STD networks.

Future research on ORP problems in STD networks can be in the following directions:

1. Identify variants with realistic assumptions on network stochastic dependency and information access that are suitable for traffic applications. Intuitively local information access and partial stochastic dependency is the right choice, but further research work is required to obtain specific forms that trade off realism and model tractability.
2. The mechanism to deploy ORP algorithms and approximations in actual traffic applications. For example, how to obtain the support points of link travel times needed for Algorithm DOT-SPI? What if the observed link travel time realizations do not comply with the a priori distributions? For the two open-loop approximations, how to obtain a new estimate of link travel time marginal distributions at each decision point? In the computational tests we derive the marginal distributions from the joint distribution, but this is not practical in real world applications.

3. Design approximations that are both computationally feasible and realistic in the context of transportation traffic networks, and test the corresponding solution algorithms using real-world data. One possible approximation is the *aggregate states* approximation. It is also referred as feature extraction in dynamic programming. So far our network conditions have been in a very disaggregate level, i.e. each possible link travel time realization can possibly change the state. Sometimes, however, aggregate states could be used to reduce the dimension of the state space while still give a satisfactory representation of the network conditions. One possible aggregate state in traffic applications is the level of service, A, B, C, D, E, or F.
4. Travel time reliability is another important factor in travelers' routing decision making in stochastic networks. Two possible measures of travel time reliability is variance and expected schedule delay. A routing policy with less travel time variance or less expected schedule delay is viewed as more reliable. Schedule delay is defined as the difference between the actual arrival time and the desired arrival time. For commuters, the desired arrival time in the morning might be some time around the work starting time. For a traveler catching a plane, the desired arrival time might be roughly 1 h before the plane departure. Optimal routing policy problems with maximum reliability as well as those with multiple optimality criteria such as mean travel time and reliability are future research topics.

Acknowledgments

This research benefited from the support of the USA National Science Foundation (NSF). This support was under the CAREER Award number CMS-9733948, and the ETI Award numbered CMS-0085830.

Appendix A. An illustrative example for algorithm DOT-SPI

We use an example to illustrate how Algorithm DOT-SPI works. The small network in Fig. A-1 has three nodes, three links and the number of time periods is 2. The values of the travel time realizations are in Table A-1. Each of the three support points has a probability of 1/3. The network is designed to be very small to make the understanding of the algorithm easier. Note that travelers starting from node *b* or node *c* have no choice but to take node *c* as the next node. It is suggested that the reader pay attention to how routing decision at node *a* is affected by time and online information.

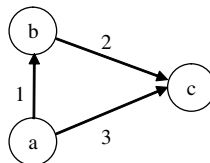


Fig. A-1. Algorithm DOT-SPI: A small network.

Table A-1

Support points for the small network

Time	Link	v_1	v_2	v_3
0	1	1	1	1
	2	2	2	1
	3	3	3	2
1	1	1	1	2
	2	1	2	1
	3	3	2	2

Step 1: Construct $EV(t)$, $t = 0, 1$

Call Generate_Event_Collection

$$D = \{\{v_1, \dots, v_3\}\}$$

$$t = 0$$

$$(j, k) = 1$$

$$S = \{v_1, \dots, v_3\}$$

$$w = 1$$

$$S'_1 = S$$

$$D' = \{\{v_1, \dots, v_3\}\}$$

$$D = \{\{v_1, \dots, v_8\}\}$$

$$(j, k) = 2$$

$$S = \{v_1, \dots, v_3\}$$

$$w = 2$$

$$S'_1 = \{v_1, v_2\}, S'_2 = \{v_3\}$$

$$D' = \{\{v_1, v_2\}, \{v_3\}\}$$

$$D = \{\{v_1, v_2\}, \{v_3\}\}$$

$$(j, k) = 3$$

$$S = \{v_1, v_2\}$$

$$w = 1$$

$$S'_1 = S$$

$$D' = \{\{v_1, v_2\}, \{v_3\}\}$$

$$S = \{v_3\}$$

$$w = 1$$

$$S'_1 = S$$

$$D' = \{\{v_1, v_2\}, \{v_3\}\}$$

$$D = \{\{v_1, v_2\}, \{v_3\}\}$$

$$EV(0) = \{\{v_1, v_2\}, \{v_3\}\}$$

$$t = 1$$

$$(j, k) = 1$$

$$\begin{aligned}
& S = \{v_1, v_2\} \\
& \quad w = 1 \\
& \quad S'_1 = S \\
& \quad D' = \{\{v_1, v_2\}, \{v_3\}\} \\
& S = \{v_3\} \\
& \quad w = 1 \\
& \quad S'_1 = S \\
& \quad D' = \{\{v_1, v_2\}, \{v_3\}\} \\
& D = \{\{v_1, v_2\}, \{v_3\}\} \\
& (j, k) = 2 \\
& \quad S = \{v_1, v_2\} \\
& \quad \quad w = 2 \\
& \quad \quad S'_1 = \{v_1\}, S'_2 = \{v_2\} \\
& \quad \quad D' = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& \quad S = \{v_3\} \\
& \quad \quad w = 1 \\
& \quad \quad S'_1 = S \\
& \quad \quad D' = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& \quad D = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& (j, k) = 3 \\
& \quad S = \{v_1\} \\
& \quad \quad w = 1 \\
& \quad \quad S'_1 = S \\
& \quad \quad D' = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& \quad S = \{v_2\} \\
& \quad \quad w = 1 \\
& \quad \quad S'_1 = S \\
& \quad \quad D' = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& \quad S = \{v_3\} \\
& \quad \quad w = 1 \\
& \quad \quad S'_1 = S \\
& \quad \quad D' = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& \quad D = \{\{v_1\}, \{v_2\}, \{v_3\}\} \\
& EV(1) = \{\{v_1\}, \{v_2\}, \{v_3\}\}
\end{aligned}$$

A summary of the results of constructing event collections is as follows.

$$\begin{aligned}
EV(0) &= \{\{v_1, v_2\}, \{v_3\}\} \\
EV(1) &= \{\{v_1\}, \{v_2\}, \{v_3\}\}
\end{aligned}$$

Step 2: Compute $e_{\mu^r}(j, 1, EV)$, $\forall j \in N$, $\forall EV \in EV(1)$ This step involves solving deterministic static shortest path problems with each single support point v_r , $r = 1, 2, 3$. Any classical shortest path algorithm can be used. In our small network, this can be done by observation. The

Table A-2

Results in the static deterministic period (time $t \geq 1$, $EV(1) = \{\{v_1\}, \{v_2\}, \{v_3\}\}$)

Node	EV		
	$\{v_1\}$	$\{v_2\}$	$\{v_3\}$
c	0 (node c)	0 (node c)	0 (node c)
b	1 (node c)	2 (node c)	1 (node c)
a	2 (node b)	2 (node c)	2 (node c)

results are listed in Table A-2. In each result cell, the minimum expected travel time is given and the corresponding next node is in the parentheses.

Step 3:

$$e_{\mu^*}(j, t, EV) \leftarrow +\infty, \forall j \in N \setminus \{d\},$$

$$e_{\mu^*}(d, t, EV) \leftarrow 0, \forall t < K - 1, \forall EV \in EV(t)$$

Step 4: (main loop)

$$t = 0$$

$$EV = \{v_1, v_2\}$$

$$EV'_1 = \{v_1\}, \Pr(EV'_1|EV) = 0.5$$

$$EV'_2 = \{v_2\}, \Pr(EV'_2|EV) = 0.5$$

$$(j, k) = 1$$

$$\begin{aligned} temp_e &= 1 + e_{\mu^*}(b, 0 + 1, EV'_1) \Pr(EV'_1|EV) + e_{\mu^*}(b, 0 + 1, EV'_2) \Pr(EV'_2|EV) \\ &= 1 + 1 \times 0.5 + 2 \times 0.5 = 2.5 < +\infty \end{aligned}$$

$$e_{\mu^*}(a, 0, \{v_1, v_2\}) = 2.5, \mu^*(a, 0, \{v_1, v_2\}) = \text{node } b$$

$$(j, k) = 2$$

$$\begin{aligned} temp_e &= 2 + e_{\mu^*}(c, 0 + 2, EV'_1) \Pr(EV'_1|EV) + e_{\mu^*}(c, 0 + 2, EV'_2) \Pr(EV'_2|EV) \\ &= 2 + 0 \times 0.5 + 0 \times 0.5 = 2 < +\infty \end{aligned}$$

$$e_{\mu^*}(b, 0, \{v_1, v_2\}) = 2, \mu^*(b, 0, \{v_1, v_2\}) = \text{node } c$$

$$(j, k) = 3$$

$$\begin{aligned} temp_e &= 3 + e_{\mu^*}(c, 0 + 3, EV'_1) \Pr(EV'_1|EV) + e_{\mu^*}(c, 0 + 3, EV'_2) \Pr(EV'_2|EV) \\ &= 3 + 0 \times 0.5 + 0 \times 0.5 = 3 > 2.5 = e_{\mu^*}(a, 0, \{v_1, v_2\}) \end{aligned}$$

$$EV = \{v_3\}$$

$$EV'_1 = \{v_3\}, \Pr(EV'_1|EV) = 1$$

$$(j, k) = 1$$

$$\begin{aligned} temp_e &= 1 + e_{\mu^*}(b, 0 + 1, EV'_1) \Pr(EV'_1|EV) \\ &= 1 + 1 \times 1 = 2 < +\infty \end{aligned}$$

$$e_{\mu^*}(a, 0, \{v_3\}) = 2, \mu^*(a, 0, \{v_3\}) = \text{node } b$$

$$(j, k) = 2$$

$$\begin{aligned} temp_e &= 1 + e_{\mu^*}(c, 0 + 1, EV'_1) \Pr(EV'_1|EV) \\ &= 1 + 0 \times 1 = 1 < +\infty \end{aligned}$$

$$e_{\mu^*}(b, 0, \{v_3\}) = 1, \mu^*(b, 0, \{v_3\}) = \text{node } c$$

$$(j, k) = 3$$

$$\begin{aligned} temp_e &= 2 + e_{\mu^*}(c, 0 + 2, EV'_1) \Pr(EV'_1|EV) \\ &= 2 + 0 \times 1 = 2 = e_{\mu^*}(a, 0, \{v_3\}) \end{aligned}$$

Table A-3

Results at time 0 (time $t = 0$, $EV(0) = \{\{v_1, v_2, v_3\}\}$)

Node	EV	
	$\{v_1, v_2\}$	$\{v_3\}$
c	0 (node c)	0 (node c)
b	2 (node c)	1 (node c)
a	2.5 (node b)	2 (node b)

The summary of results at time 0 is in [Table A-3](#).

References

- Andreatta, G., Romeo, L., 1988. Stochastic shortest paths with recourse. *Networks* 18, 193–204.
- Bander, J., White III, C.C., 2002. A heuristic search approach for a nonstationary stochastic shortest path problem with terminal cost. *Transportation Science* 36 (2), 218–230.
- Chabini, I., 1997. A new algorithm for shortest path problems in discrete dynamic networks. In: *Proceedings of the 8th IFAC Symposium on Transport Systems*, Chania, Greece, pp. 551–556.
- Chabini, I., 1998. Discrete dynamic shortest path problems in transportation applications: Complexity and algorithms with optimal run time. *Transportation Research Record* 1645, 170–175.
- Chabini, I., 2000. Minimum expected travel times in stochastic time-dependent networks revisited, working paper.
- Chabini, I., 2002. Algorithms for k -shortest path problems and other routing problems in time-dependent networks. forthcoming in *Transportation Research Part B*.
- Cheung, R.K., 1998. Iterative methods for dynamic stochastic shortest path problems. *Naval Research Logistics* 45, 769–789.
- Croucher, J., 1978. A note on the stochastic shortest-route problem. *Naval Research Logistics Quarterly* 25, 729–732.
- Fu, L., 2001. An adaptive routing algorithm for in-vehicle route guidance systems with real-time information. *Transportation Research Part B* 35, 749–765.
- Fu, L., Rilett, L.R., 1998. Expected shortest paths in dynamic and stochastic traffic networks. *Transportation Research Part B* 32 (7), 499–516.
- Gao, S., 2004. Optimal adaptive routing and traffic assignment in stochastic time-dependent networks, Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA.
- Gao, S., Chabini, I., 2002. The best routing policy problem in stochastic time-dependent networks. *Transportation Research Record* 1783, 188–196.
- Hall, R.W., 1986. The fastest path through a network with random time-dependent travel times. *Transportation Science* 20 (3), 182–188.
- Miller-Hooks, E.D., 2001. Adaptive least-expected time paths in stochastic, time-varying transportation and data networks. *Networks* 37 (1), 35–52.
- Miller-Hooks, E.D., Mahmassani, H.S., 2000. Least expected time paths in stochastic, time-varying transportation networks. *Transportation Science* 34 (2), 198–215.
- Polychronopoulos, G.H., 1992. Stochastic dynamic shortest distance problems, Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA.
- Polychronopoulos, G.H., Tsitsiklis, J.N., 1996. Stochastic shortest path problems with recourse. *Networks* 27, 133–143.
- Pretolani, D., 2000. A directed hyperpath model for random time dependent shortest paths. *European Journal of Operational Research* 123, 315–324.
- Provan, J.S., 2003. A polynomial-time algorithm to find shortest paths with recourse. *Networks* 41 (2), 115–125.

- Psaraftis, H.N., Tsitsiklis, J.N., 1993. Dynamic shortest paths in acyclic networks with Markovian arc costs. *Operations Research* 41 (1), 91–101.
- Schrank, D., Lomax, T., 2003. The 2003. annual urban mobility report. Technical report, Texas Transportation Institute. Available from: <<http://mobility.tamu.edu/>>.
- Waller, S.T., Ziliaskopoulos, A.K., 2002. On the online shortest path problem with limited arc cost dependencies. *Networks* 40 (4), 216–227.
- Yang, B., Miller-Hooks, E., 2004. Adaptive routing considering delays due to signal operations. *Transportation Research Part B* 38, 385–413.