2-2012

Real-Time Information and Correlations for Optimal Routing in Stochastic Networks

He Huang
University of Massachusetts Amherst, jie31best@gmail.com

REAL-TIME INFORMATION AND CORRELATIONS FOR OPTIMAL ROUTING IN STOCHASTIC NETWORKS

A Dissertation Presented

by

HE HUANG

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

February 2012

Civil and Environmental Engineering

REAL-TIME INFORMATION AND CORRELATIONS FOR OPTIMAL ROUTING IN STOCHASTIC NETWORKS

A Dissertation Presented

by

HE HUANG

Approved as to style and content by:

____
Song Gao, Chair

____
John Collura, Member

____
Hari Balasubramanian, Member

____
Marco (Yu) Nie, Member

____
Richard Palmer, Department Head
Department of Civil and Environmental Engineering

DEDICATION

For BOSA.

ACKNOWLEDGMENTS

The research is funded by the University of Massachusetts (UMass) Amherst, the Department of Civil and Environmental Engineering and the School of Engineering at UMass Amherst.

The author is also grateful to the California Department of Transportation and University of California Berkeley for providing access to the PeMS data.

The author wishes to express his gratitude to his supervisor, Dr. Gao, for providing him the chance to work in the research group BOSA and her supervision, guidance and support over the four years. This dissertation and the research would not have been possible without the support of her. Deepest gratitude is also due to the members of the committee, Prof. Collura, Prof. Balasubramanian and Prof. Nie, for their significant contributions to the dissertation.

The author also wishes to express his love and gratitude to his beloved ones, who have always been there, for their understanding and endless support.

ABSTRACT

REAL-TIME INFORMATION AND CORRELATIONS FOR OPTIMAL ROUTING IN STOCHASTIC NETWORKS

FEBRUARY 2012

HE HUANG, B.E., TSINGHUA UNIVERSITY

M.S., UNIVERSITY OF MASSACHUSETTS AMHERST

PH.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Song Gao

Congestion is a world-wide problem in transportation. One major reason is random interruptions. The traffic network is inherently stochastic, and strong dependencies exist among traffic quantities, e.g., travel time, traffic speed, link volume. Information in stochastic networks can help with adaptive routing in terms of minimizing expected travel time or disutility. Routing in such networks is different from that in deterministic networks or when stochastic dependencies are not taken into account.

This dissertation addresses the optimal routing problems, including the optimal *a priori* path problem and the optimal adaptive routing problem with different information scenarios, in stochastic and time-dependent networks with explicit consideration of the correlations between link travel time random variables. There are a number of studies in the literature addressing the optimal routing problems, but most of them ignore the correlations between link travel times. The consideration of the correlations makes the problem studied in this dissertation difficult, both conceptually and computationally.

The optimal path finding problem in such networks is different from that in stochastic and time-dependent networks with no consideration of the correlations. This

dissertation firstly provides an empirical study of the correlations between random link travel times and also verifies the importance of the consideration of the spatial and temporal correlations in estimating trip travel time and its reliability. It then shows that Bellman's principle of optimality or non-dominance is not valid due to the time-dependency and the correlations. A new property termed purity is introduced and an exact label-correcting algorithm is designed to solve the problem.

With the fast advance of telecommunication technologies, real-time traffic information will soon become an integral part of travelers' route choice decision making. The study of optimal adaptive routing problems is thus timely and of great value. This dissertation studies the problems with a wide variety of information scenarios, including delayed global information, real-time local information, pre-trip global information, no online information, and trajectory information. It is shown that, for the first four partial information scenarios, Bellman's principle of optimality does not hold. A heuristic algorithm is developed and employed based on a set of necessary conditions for optimality. The same algorithm is showed to be exact for the perfect online information scenario.

For optimal adaptive routing problem with trajectory information, this dissertation proves that, if the routing policy is defined in a similar way to other four information scenarios, i.e., the trajectory information is included in the state variable, Bellman's principle of optimality is valid. However, this definition results in a prohibitively large number of the states and the computation can hardly be carried out. The dissertation provides a recursive definition for the trajectory-adaptive routing policy, for which the information is not included in the state variable. In this way, the number of states is small, but Bellman's principle of optimality or non-dominance is invalid for a similar reason as in the optimal path problem. Again purity is introduced to the trajectory-adaptive routing policy and an exact algorithm is designed based on the concept of decreasing order of time.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

1.1 Stochastic Networks

Congestion, as described in Schrank and Lomax (2009), is a problem in the United States' 439 urban areas and has gotten worse in regions of all sizes. One major reason for congestion is random disruptions, e.g., crashes, vehicle breakdown, bad weather, special events, construction and maintenance activities. They greatly affect the reliability of transportation systems, and the resulting delays account for about 50 percent of all delay on the roads (Schrank and Lomax, 2003). Some of the disturbances are completely unpredictable, such as incidents and vehicle breakdown, while others are predictable to some extent, such as bad weather, work zones and special events, but usually with prediction errors or limitations. A weather forecast is usually in a probabilistic format, e.g. a precipitation of rain with chance of precipitation 60%. Work zones and special events are usually scheduled, but the schedules might not be available to the travelers in a timely manner, and thus are still unpredictable to travelers.

Congested traffic networks are inherently uncertain with those random disruptions, and there exists randomness in traffic quantities, such as travel time, link volume, queue length, and so on, on a day-to-day base. For example, 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 a different week. The randomness can come from multiple sources. One of the most significant sources is the random disturbances, as described in the previous paragraph. Another major reason is 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. For example, travelers with non-commuting trip purposes might decide not to take a trip at a particular day, and this kind of 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.

1.2 Information in Stochastic Networks

In developed countries where building more infrastructures is usually politically, financially and environmentally constrained, a lot of efforts have been devoted to making best use of current infrastructure system with the help of Intelligent Transportation Systems (ITS). For example, advanced traveler information systems (ATIS) aim 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 a large number of 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 on a highway, a timely notice by ATIS to travelers who plan to take the highway 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.

In stochastic networks, travelers make decisions (destination, mode, departure time, and route) based on the information they have about the traffic network. The information can be obtained through a wide range of means, e.g., travelers' own experience, word of mouth, and ATIS. 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 one minutes 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 to 30 minutes. This classification is meaningful only when there is randomness 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 hardly changed en route, while route 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 good source is ATIS. ATIS can provide travelers with reports of traffic conditions in the past and possibly predictions about the near 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 stochastic networks, where there are random disruptions, e.g., crashes, vehicle breakdown, bad weather, special events, construction and maintenance activities.

There are various implementations of ATIS, and they differ in the spatial and temporal availability, the quality, the format, and limitation of information provided. For example, a variable message sign (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 systems can provide information to travelers anywhere in the radio coverage. Relatively more detailed information is available compared to VMS, yet still the coverage is usually limited to major highways and arterials. Besides the limitation on the spatial side, there is also limitation on the temporal side. Usually radio broadcast provides traffic condition information every 15 minutes for example, and so for travelers there is a time lag with the information. Internet can also be an access to ATIS, providing travelers with information such as camera images, travel time estimations, work zone and event schedule, and travel advisories. However, once travelers are en route, they can hardly have access to internet, and so internet-based ATIS implementation is usually viewed as a pre-trip planner. More advanced in-vehicle systems are also emerging, possibly with a database of road map, travel times under normal conditions, records of past incidents, etc., and can communicate with information centers to obtain very detailed and updated information.

1.3 Correlations in Stochastic Networks

Traffic quantities (e.g., link travel times, travel speed, etc.) in stochastic networks are not only random, but there also usually exist strong time-wise and link-wise dependencies among them, largely due to traffic flow propagations over time and space, or an event that affects road capacities in a wide area. Take link travel times for example. If the randomness comes from incidents, then link travel times around the incident location and around the incident duration are correlated. If the randomness comes from weather, then link travel times of the whole network over a certain time period are correlated. Specifically, when an incident occurs, congestion will build up upstream of the incident location, and thus the high travel time on the incident link at 8:00 AM will likely suggest a high travel time on an upstream link at 8:10 AM. When a heavy thunderstorm hits a region, all links affected by the weather will experience long delays, and thus high travel times on highways suggest high travel times on arterials.

Network stochastic dependencies are generally required to capture the benefits of online information for network routing, since only through the dependencies over time and space can the knowledge of traffic conditions at the current time and specific location result in a better prediction of traffic conditions in the future and elsewhere. It is generally believed that the smaller the temporal and/or spatial distance is between two time-location pairs, the more correlated their traffic conditions are. For example, travelers are provided with the information on the traffic conditions of a section of highway at 9 AM. With the information, travelers can make a respectively accurate prediction on the traffic conditions of the same section of highway or elsewhere nearby in the near future, e.g., the traffic conditions of the same section of highway or the nearest on-ramp or parallel arterial at 9:10 AM. However, the information is of no help for travelers to get a clue what the traffic conditions will be like on the same section of highway or elsewhere nearby at 9 PM or somewhere else 50 miles away.

1.4 Routing in Stochastic Networks

There exist two possible types of routing problems in stochastic networks: non-adaptive and adaptive. Non-adaptive routing does not take into account the fact that information on arrival times at intermediate nodes and/or link travel time realizations will be available during a trip, and thus a fixed *a priori* path is determined at the origin node and followed regardless of the actual realizations of the stochastic network. On the other hand, adaptive routing considers intermediate decision nodes, and a next link (or sub-path) is chosen based on information collected thus far.

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 travelers does not adapt to an incident, they could be stuck in the incident link for a very long time. However, if adequate online information is available about the incident and travelers adapt to it by taking an alternative route, they can save travel time compared to the non-adaptive case. The adaption also ensures that the travel time is not prohibitively high in incident scenarios, and thus provides a more reliable travel time.

Although adaptive routing is more effective than *a priori* path, *a priori* paths are still useful in many circumstances. In practice, travelers usually begin a trip bearing in mind a pre-planned path, and en-route rerouting occurs only when the travel time on the pre-planned route exceeds a certain threshold. Furthermore, when travelers do rerouting, e.g., when they need to exit a congested freeway, the new route they plan for the rest of the trip is usually still a path from the intermediate decision node to the destination. Last but not least, an optimal adaptive routing policy may suggest cycling to avoid large travel time in some cases, a counterintuitive guidance that travelers are unlikely to follow. On the contrary, an optimal *a priori* path may not contain cycles.

In stochastic networks, the definition of optimal routing, including adaptive routing and *a priori* path, can be ambiguous. In the literature, a variety of optimality definitions have been made. One of the most commonly used definitions is the minimum expected travel time. Take *a priori* path for example. Unlike deterministic networks, in which travelers can determine a single optimal path with shortest travel time, when travelers are making decisions in stochastic networks, they might find that several paths have positive probabilities of attaining the minimum travel time for some realization of the network. A set of non-dominated (sometimes referred to as Pareto-optimal) paths can be identified based on first-order stochastic dominance, and the one with the minimum expected travel time is defined as optimal path.

However, the minimum expected travel time definition for optimal routing does not take into account the effect of travel time reliability on route choice. For example, consider the case where one path bears a deterministic travel time of 15 minutes, while another one have random travel time of either 10 or 20 minutes, both with probability of 0.5. The expected travel times on the two paths are the same, but only risk-seeking travelers will choose the latter one. In reality, most travelers are risk-averse when making routing decisions in stochastic networks, and so reliability of travel time is important. Various forms of disutility functions of travel time can be defined to take into account travel time reliability, and the routing with the minimum expected disutility is defined as the optimal, following the classical von Neumann and Morgenstern paradigm in decision under risk (von Neumann and Morgenstern, 1944). The disutility function can be either linear or non-linear, and is usually an increasing function of travel time. Travel time itself can be viewed as a special case of the disutility function. More general convex non-linear disutility functions can capture travelers' risk-averse behavior and take into account travel time reliability. The disutility function can also be a linear combination of the mean and the variance (or standard deviation) of travel time, and the objective is to minimize the disutility.

1.5 Thesis Objectives

In this thesis, the following questions are to be answered:

- As stated in Section 1.3 , it is generally believed that the closer two time-location pairs are in time and/or space, the more correlated their traffic conditions, e.g., link travel time, traffic speed, link volume, are, but how do correlations exist among traffic quantities over time and space?

In order to answer this question, real-life traffic data from an urban freeway segment are to be obtained from PeMS database. Spatial and temporal Pearson's correlation coefficients among traffic variables over a number of links and time periods will be calculated. A regression model will be created based on the calculated correlation coefficients, and the model will be able to tell how correlations change over temporal and spatial distances and other properties of correlations.

- When the previous question is answered, empirical evidences of stochastic dependencies among traffic variables in a traffic network will be provided. However, most researches on optimal routing (including adaptive routing and non-adaptive routing) do not take correlations into account, and those studies that do consider stochastic dependencies just assume a certain level of correlations on random link travel time variables over time and/or space A natural question is how far off a routing strategy will be in terms of minimizing expected travel time or other criteria, if stochastic dependencies are ignored, compared with a more realistic case where they are taken into account, e.g., where the regression model on correlation coefficients obtained from the answer to the previous question is applied?

In order to answer this question, an efficient routing algorithm with realistic assumptions on network stochastic dependencies is to be designed. The theoretical complexity of the developed algorithms is to be studied, and it is to be determined whether the consideration of stochastic dependencies significantly complicates the routing algorithm design. If yes, a reasonable compromise between modeling stochastic dependencies realistically and computing routing strategies efficiently needs to be found. Besides, computational tests of the developed algorithms will be conducted in hypothetical and real-life networks to investigate whether the consideration of stochastic dependencies significantly increase the algorithm average running time and also to answer the question.

- As stated in Section 1.2 , a pre-assumption of ATIS is that better informed travelers can make better decisions. However, is that true? Is more information always better for optimal adaptive routing? Note that it is assumed that the information is without any error, and the optimality of the routing is with respect to individual travelers rather than the system. In other words, we do not consider the interaction between demand and supply. In Gao and Chabini (2006), perfect information scheme is assumed, and in that case, Bellman's principle of optimality is valid. However, does it still hold for imperfect information schemes? If not, how will this affect the algorithm designing? If an exact algorithm is difficult to develop, will a heuristic algorithm be available? If yes, how does the heuristic algorithm perform?

In order to answer this question, a generic description of online information is to be provided, based on which different types of imperfect online information schemes can be derived. It is to be determined through theoretical analysis whether Bellman's principle of optimality is valid in imperfect information case. An efficient algorithm is to be designed to solve optimal adaptive routing problems in different imperfect online information schemes. Theoretical and computational analyses are to be carried out to study the performance of the algorithm and to show whether more error-free information is always better for optimal adaptive routing in flow-independent networks.

- For a stochastic network where the complete dependencies between link travel times are considered, how the optimal *a priori* path finding problem is to be solved? Earlier studies show that Bellman's principle of optimality does not hold for such problem in a stochastic network where no dependencies between link travel times are considered. Does it apply to the complete dependency case? If not, is there any property of the quantity of the path that can satisfy Bellman's principle? Will that help solve the problem?

In order to answer these questions, a theoretical analysis is needed to investigate Bellman's principle for *a priori* paths in such a network. An efficient algorithm is to be designed according to the analysis result to solve the optimal path problem. Theoretical and computational analyses are to be carried out to study the performance of the algorithm and to show how the optimal solution is affected by the parameters of the problem.

- The least amount of information a traveler can obtain *en route* even without any external information source is trajectory information. When a traveler makes routing decisions adaptive to trajectory information, he/she is making a trajectory-adaptive routing. Does Bellman's principle hold for trajectory-adaptive routing? If not, how will this affect the algorithm designing?

In order to answer these questions, a theoretical analysis is needed to investigate Bellman's principle for trajectory-adaptive routing. An efficient algorithm is to be designed according to the analysis result to solve the optimal trajectory-adaptive routing problem. Theoretical and computational analyses are to be carried out to study the performance of the algorithm and to show how the optimal solution is affected by the parameters of the problem.

1.6 Thesis Organization

The thesis is organized as follows. A literature review on correlations, information, and routing (including adaptive routing and non-adaptive routing) in stochastic networks is given in CHAPTER 2. In CHAPTER 3, correlations in stochastic networks are studied. The existence of correlations among link travel times is shown by actual data from a real-life network, and linear regression is conducted to show how correlations change with temporal and spatial distances. Theoretical analysis and simulation show how correlations affect travelers' routing in stochastic networks. CHAPTER 4 deals with information and adaptive routing in stochastic networks. It is shown that more error-free information is always better (or at least not worse) for optimal adaptive routing in flow-independent networks. A heuristic algorithm is designed for the optimal adaptive routing problem with the three partial and no online information schemes, based on a set of necessary conditions for optimality. The effectiveness of the heuristic is shown to be satisfactory over the tested random networks. CHAPTER 5 and CHAPTER 6 study the problem of finding the optimal *a priori* paths and the optimal trajectory-adaptive routing policies in a stochastic network. Exact algorithms are designed to solve such problems. It is shown that the benefit of being adaptive to trajectory information in terms of minimizing the expected disutility of travel time increases with travelers' risk aversion, the correlation between link travel times and the network size. CHAPTER 7 gives a summary of the thesis work and findings and discusses future directions of research.

CHAPTER 2

LITERATURE REVIEW

2.1 Correlations in Stochastic Networks

A number of studies in the literature on optimal routing problem take network stochastic dependencies into account. Several of them are on optimal *a priori* path problem. Sivakumar and Batta (1994) discuss the variance-constrained shortest path problem and uses covariance matrices to model the correlation across links. Sen et al. (2001) use similar approach, and they assume that removing a cycle results in a route whose total variance is strictly less than that associated with the route containing the cycle. They observe that this assumption does not rule out negatively correlated link travel times. In Nie and Wu (2009), travel time correlations are restricted only to adjacent links, and non-dominated paths over the states on the next link are generated to find the one with maximum arrival time reliability.

Some researches on adaptive routing have considered network stochastic dependencies. Psaraftis and Tsitsiklis (1993) assume link travel times are known functions of certain environment variables at network nodes and each of these variables evolves according to an independent Markov process. Travelers learn the current state of the Markovian chain at any time. Waller and Ziliaskopoulos (2002) are concerned with the adaptive routing problem with limited forms of spatial and temporal link cost dependencies. They assume one-step arc dependence, that is, given the cost of predecessor links, no further information is obtained through spatial dependence. The limited temporal dependency assumes that the cost of a link is known once the entrance node is reached. Fan et al. (2005) address the adaptive routing problem in static and stochastic networks with correlated link service levels. A limited correlation structure which is similar to that in Waller and Ziliaskopoulos (2002) is employed whereas link states are restricted to be either congested or not. The correlations between the states of adjacent nodes are taken into account by introducing conditional probabilities of downstream node state given upstream node state. In Boyles (2006), conditional probabilities of adjacent link travel costs are utilized and travelers are assumed to remember only the travel time on the last link they traverse. In Gao and Chabini (2002), Gao (2005), Gao and Chabini (2006), complete dependencies are assumed, where all travel times on all links at all time periods are correlated, and a joint distribution of travel time random variables is applied.

All the algorithm designs in the above studies just assume a certain level of correlations (dependencies) on random link travel time variables over time and/or space, not to mention most other researches on route choice do not take correlations into account. However, a lack of data support is noted. Conceivably with higher level of dependencies assumed, the algorithm complexity is higher, but it is to be found what a good compromise between tractability and realism is. It is important to gain an understanding of stochastic dependencies of link travel times from real life data. Intuitively such dependencies exist in reality; however it is valuable to provide empirical evidences of stochastic dependencies among link travel times in a traffic network through actual data and to provide guidelines on the scope of spatial and temporal dependencies which will help validate assumptions used in routing algorithm design.

Another major application of link travel time correlations is travel time prediction. Prediction of short-term future traffic condition on real-time basis is important as it can allow travelers to avoid traffic congestion and react to the traffic incidents immediately after they occur. A number of travel time forecasting models have been developed in the past two decades. Some of them take into account the correlations of travel times over time and space.

Gajewski and Rilett (2004) focus on link travel time correlation estimation using Bayesian statistical inference. They use natural cubic splines, which is a nonparametric regression technique, to model the mean link travel time and develop a Bayesian-based methodology for estimating the distribution of the correlation of travel times between links along a corridor. It is shown that an estimate of the correlation coefficient of travel times can be calculated along with associated intervals.

Goel et al. (2005) propose Bayesian and non-Bayesian strategies to improve Average Annual Daily Traffic (AADT) estimation by exploiting the inherent underlying correlations between link flows. These correlations arise partially, because the inflows and outflows to a node are always constrained. In addition, when the network has a large number of OD zones, and a relatively smaller number of links, the correlation between the link flows can be large.

Eom et al. (2006) propose a spatial regression model that considers spatial correlation effect. They show that, if spatial correlation between AADT at one location and those at its neighbors exists, the overall predictive capability of the spatial regression model is much better than that of ordinary regression model. It is also shown that, since the spatial correlation depends on the distance among the stations, the closer stations are located to each other, the higher spatial dependency is.

Tam and Lam (2009) use the historical travel time estimates together with their updated temporal variance-covariance relationships to predict the travel times in the next five-minute interval, and they show that use of the updated temporal variance-covariance relationships of travel times can greatly improve the accuracy of the short-term travel time prediction.

The above researches study the properties of correlations of random link travel time variables. However, they do not show how the correlations affect the reliability of trip travel time and travelers' route choice decisions in a traffic network.

2.2 Information in Stochastic Networks

There are a large number of studies on traveler information since two decades ago. One critical problem is how to represent various types of information situations in a network. Under a traffic equilibrium framework, some (e.g., Hall, 1996; Yang, 1998; Levinson, 2003) assume full information for travelers with access to ATIS, which is sometimes too ideal. In Mahmassani and Jayakrishnan (1991), Hall (1996) and Engelson (2003), travelers are assumed to switch routes based on instantaneous path travel times, rather than those that they will actually experience. This assumption circumvents the need to retrieve future link travel times. In Yin and Yang (2003) and Lo and Szeto (2004), the imperfection of various ATIS is represented through random errors added to the true path travel times, and different degrees of errors suggest different information systems. Under a dynamic process framework, information could be included in travelers' learning process to represent traffic conditions from the previous day or time period (e.g., Ben-Akiva et al., 1991; Friesz et al., 1994; Emmerink et al., 1995; Jha et al., 1998; Mahmassani and Liu, 1999). A common shortcoming of these studies is that the information representation cannot be directly related to real life situations, e.g., the spatially or temporally limited information systems discussed in Section 1.2 .

There is another school of information theoretic studies on simplified networks. Arnott et al. (1999) study effects of online information in a two-link network with random capacities under equilibrium in both departure time and route, using the bottleneck model to calculate congested travel times. Rigorous studies of zero information, full information, and imperfect information are carried out. Other studies in this school include Arnott et al. (1991, 1996), Emmerink et al. (1998), de Palma and Picard (2006) and Chorus et al. (2006). Denant-Boemont and Petiot (2003) evaluate travel information value using human subjects' willingness to pay in an experimental setting with limited mode and route choices.

2.3 Optimal *a Priori* Path Problem in Stochastic Networks

A large number of studies have been done addressing the optimal path problem, ever since the early researches of Bellman (1958), Dijkstra (1959), and Dantzig (1960). Different assumptions and constraints have been made in terms of time-dependency of link travel times, randomness of link travel times, and network stochastic dependencies among link travel times over time and/or space. In this literature review, the focus is on stochastic networks.

In deterministic networks, Dijkstral-type algorithms can be applied in either static case or time-dependent case (Dreyfus, 1969). However, such Dijkstral-type algorithms are generally not available for the optimal path problem in stochastic networks, due to the invalidity of Bellman's principle of optimality (Miller-Hooks and Mahmassani, 2000). Moreover, unlike deterministic networks, in which one can determine a single optimal path, when a traveler is facing a stochastic network, he/she might find that several paths have positive probabilities of attaining the minimum disutility for some realization of the network, and a set of non-dominated (sometimes referred to as Pareto-optimal) paths can be identified.

Several papers have worked on defining minimum path travel time distribution in static and stochastic networks. Frank (1969) and Mirchandani (1976) have addressed the problem of determining the probability distribution of the minimum path travel time. Frank (1969) assumes continuous probability distributions for link travel times and computes the probability that the minimum path travel time is less than some given threshold. Mirchandani (1976) assumes independent discrete probability distributions for link travel times and develops an algorithm to compute the probability mass function of the minimum path travel time. Sigal et al. (1980) compute the probability that a given path is shorter than all the others, and suggests considering the path with the maximum probability of being the shortest path as the optimal path.

A common optimality criterion is minimum expected travel time (METT) or minimum expected disutility (MED). Several works (Loui, 1983; Eiger et al., 1985; Mirchandani and Soroush, 1985; Murthy and Sarkar, 1996; Murthy and Sarkar, 1998) present procedures for finding optimal paths when various forms of disutility functions are defined. It is shown that Bellman’s principle of optimality holds when affine or exponential functions are used. More general non-linear disutility functions that capture risk-averse behavior may be approximated by piecewise-linear and convex functions, and Murthy and Sarkar (1998) develop exact algorithms to solve large problem instances.

The METT criterion does not consider the effect of travel time reliability on route choice, while MED with a convex (concave) disutility function models risk aversion (seeking). There are other approaches to considering travel reliability in optimal path finding, for example, a bicriteria shortest path problems that trade off the mean and variance of path travel times. The bi-criteria problems can be formulated using generalized dynamic programming (Carraway et al., 1990) based on the non-dominance relationship. The mean-variance tradeoff can also be treated in other ways. For example, in Sen et al. (2001), the objective function of stochastic routing becomes a parametric linear combination of mean and variance. Nie and Wu (2009) study the problem of finding shortest paths to guarantee a given probability of arriving on-time and develop a label-correcting algorithm.

The optimal path problem in dynamic and stochastic networks is more difficult. For example, to find an METT path in a static and stochastic network (with or without stochastic dependency), one can simply set each link travel time random variable to its expected value and solve an equivalent shortest path problem in the converted static and deterministic network. This method will not work in a time-dependent network, as a path travel time is a composition of link travel times at the time of arrival of each intermediate node, and the travel time at an "expected arrival time" is generally not the expected travel time over random arrival times. Hall (1986) proposes a branch-and-bound procedure for finding the METT path on this type of network. Miller-Hooks (1997) and Miller-Hooks and Mahmassani (2000) explore the definition of optimality based on first-order stochastic dominance and definite stochastic dominance. Label-correcting algorithms are proposed to find non-dominated paths under the stochastic dominance rules. Recognizing that the exact algorithm does not have a polynomial bound, heuristics are considered to limit the size of the retained non-dominated paths by a predetermined number. However, these heuristics may not identify any non-dominated paths, as noted in Miller-Hooks (1997).

2.4 Optimal Adaptive Routing Problem in Stochastic Networks

Various assumptions have been made to define stochastic networks and how the realizations of the stochastic networks are revealed to the travelers.

Studies in both static and time-dependent (and stochastic) networks are reviewed. In Andreatta and Romeo (1988), the topology of the static 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), Psaraftis and Tsitsiklis (1993) and Kim et al. (2005), the link costs evolve as Markov processes; in Hall (1988), Chabini (2000), Miller-Hooks and Mahmassani (2000), Pretolani (2000), Miller-Hooks (2001), Yang and Miller-Hooks (2004), Bander and White (2002), Fan et al. (2005b) and Opasanon and Miller-Hooks (2006), time-dependent networks are described by marginal distributions of link travel times; in Gao and Chabini (2006), time-dependent networks are described by joint distribution of travel times of all links at all times; and in Waller and Ziliaskopoulos (2002), Fan et al. (2005a) and Boyles (2006), conditional probabilities of adjacent link travel costs are utilized.

As for the revealing of network conditions, in Andreatta and Romeo (1988), Polychronopoulos and Tsitsiklis (1996), Cheung (1998), Fu (2001), Waller and Ziliaskopoulos (2002) and Provan (2003) it is assumed that one learns the realization of a link travel cost once he/she arrives at the node from which the link emanates; in Chabini (2000), Miller-Hooks and Mahmassani (2000), Miller-Hooks (2001), Yang and Miller-Hooks (2004), Bander and White (2002), Pretolani (2000), Fan et al. (2005b), Opasanon and Miller-Hooks (2006) it is not stated explicitly how travelers learn about the network conditions other than the arrival times at decision nodes, hence the term “time-adaptive”; in Waller and Ziliaskopoulos (2002), Fan et al. (2005a) and Boyles (2006) it is assumed that travelers remember only the travel time on the last link they traverse; in Gao and Chabini (2006) it is assumed that travelers have knowledge about all link travel time realizations up to the current time; and in Psaraftis and Tsitsiklis (1993) and Kim et al. (2005) it is assumed that Markovian travel times and thus travelers learn the current state of the Markovian chain at any time.

The optimal adaptive routing problem studies in stochastic time-dependent (STD) networks are summarized in Table 0.1 with a taxonomy developed by Gao and Chabini (2006). A more detailed review follows.

Table 0.1 Taxonomy of the optimal routing policy problem

Information / Network	Perfect online information	Partial online information	No online information (time-adaptive)
No link-wise and time-wise dependency		Opasanon and Miller-Hooks (2006)	See the note below*
Complete dependency	Gao and Chabini (2002, 2006)	**This dissertation**	
Partial dependency		Psaraftis and Tsitsiklis (1993), Kim et al. (2005), Boyles (2006)	

* Hall (1987), Miller-Hooks and Mahmassani (2000), Chabini (2000), Pretolani (2000), Miller-Hooks (2001), Bander and White (2002), Nielson et al. (2003), Yang and Miller-Hooks (2004), Fan et al. (2005b), Fan and Nie (2006), Pretolani et al. (2009).

In the studies of no time-wise or link-wise dependencies and no online information, marginal distributions of link travel times are used and the routing is only adaptive to arrival times at decision nodes (hence the name time-adaptive). 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. 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) develop a label-correcting algorithm. Insight into the difference between an optimal routing policy problem and a least expected time path problem is provided. Later Miller-Hooks (2001) compares the said label-correcting algorithm and the dynamic programming algorithm working in decreasing order of time (Chabini, 2000) in both sparse transportation networks and dense telecommunication data networks. Yang and Miller-Hooks (2004) also extend the study of the time-adaptive routing policies to a signalized network. Nielson et al. (2003) study the bicriterion time-adaptive problem.

Pretolani (2000) uses a hyper-path representation of the adaptive routing problem based on arrival times. Bander and White (2002) design 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. Fan et al. (2005b) maximize the probability of arriving on time with continuous probability density functions on link travel times. Later in Fan and Nie (2006), algorithmic issues are explored for the same problem.

In the case of partial online information, Opasanon and Miller-Hooks (2006) study the multicriterion adaptive routing problem with information on traversed link travel times in a statistically independent network. Later on Pretolani et al. (2009) distinguish between time-adaptive and history-adaptive routing in a multicriteron optimization context.

Psaraftis and Tsitsiklis (1993) study the problem in acyclic networks, implying that no link would be visited twice, so it is not helpful to keep information of any already traversed links. This assumption along with the infinite horizon assumption makes a polynomial running time algorithm possible. Kim et al. (2005) study a similar problem in a general network with a wider information range. Boyles (2006) studies the problem with minimum expected disutility, which is a general piece-wise polynomial function of arrival time at the destination. Gao and Chabini (2002, 2006) study the problem in a general STD network with both time-wise and link-wise dependencies with perfect online information.

2.5 Thesis Contributions

The contributions of the thesis are summarized as follows.

The literature review shows that there exists the gap of lack of empirical quantification of spatial-temporal patterns, as many of the aforementioned research areas rely on correlations. The thesis fills the gap with an empirical study on the properties of the correlations on random link travel times and we also verify the importance of spatial and temporal correlations in estimating trip travel time and its reliability.

There are few studies addressing optimal routing problem (*a priori* path or adaptive routing) in stochastic networks with the consideration of complete dependencies. This paper fills the gap and presents algorithms for such problems.

The thesis expands upon past research by examining the optimal adaptive routing policy problem in such networks with partial or no online information. A heuristic, rather than exact, algorithm is designed and employed based on a set of necessary conditions for optimality.

Theoretical and computational analyses show that stochastic dependencies affect optimal path finding in a stochastic network, and the effect depends on the level of link travel time correlations and travelers' risk aversion. The thesis shows that Bellman's principle is invalid if the optimality or non-dominance of a path and its sub-paths is defined with respect to (w.r.t.) the universal set of departure times and travel time probabilistic outcomes. A new property termed purity is introduced for which the Bellman's principle is valid, and it is proved that there must exist an optimal path with this property. An exact label-correcting algorithm is designed to find the optimal paths based on this property.

For optimal trajectory-adaptive routing problem, the thesis proves that, if the routing policy is defined in a similar way to other four information scenarios, i.e., the trajectory information is included in the state variable, Bellman's principle of optimality is valid. However, this definition results in a prohibitively large number of the states and the computation can hardly be carried out. The dissertation provides a recursive definition for the trajectory-adaptive routing policy, for which the trajectory information is not included in the state variable. In this way, the number of states is small, but

Bellman's principle of optimality or non-dominance is invalid for a similar reason as in the optimal path problem. Again purity is introduced to the trajectory-adaptive routing policy and an exact algorithm is designed based on the concept of decreasing order of time (DOT), which can find the optimal trajectory-adaptive routing policies. It is shown that stochastic dependencies affect optimal routing policy finding as well as the benefits of being adaptive and of traveler information in a stochastic network, and the impact is related to the level of correlation and risk attitudes.

CHAPTER 3

CORRELATIONS IN STOCHASTIC NETWORKS

3.1 Introduction

In this chapter, we use real-life data and study the properties of the correlations on random link travel times. We also verify the importance of spatial and temporal correlations in estimating trip travel time reliability, test if route choice prediction will be biased if correlation is not taken into account, and investigate how sensitive route shares are to the level of correlation and risk attitudes.

Specifically, we investigate a simple network where there are only two paths between an OD pair, one freeway path and the other local path. Freeway path consists of a series of freeway links whose travel times are correlated random variables, while local path travel time is deterministic. We first carry out theoretical analysis where we assume identical correlation coefficient between any pair of freeway link travel time random variables and evaluate the role of correlation in route choice. Then we process data from an urban freeway segment and use a linear regression model to estimate the correlation between different links at different time periods on the path. Simulation is conducted based on the data and sensitivity analysis is carried out to further evaluate the role of correlation as well as travelers' risk attitude in route choice.

This chapter is organized as follows. In Section 3.2 , the problem is defined and theoretical analysis is given in Section 3.3 . Data from a real-life network is processed in Section 3.4 and simulation is run in Section 3.5 . In Section 3.6 , conclusions are made and future directions given.

3.2 Problem Statement and Methodology

Suppose in a transportation network, between origin node *O* and destination node *D*, there are two paths: one is freeway path, which consists of a series of freeway links; the other local path, which contains only local links. It is assumed that freeway links/path bear stochastic travel times, while local link/path travel times are static and deterministic. Figure 0.1 shows both paths between node *O* and *D*.

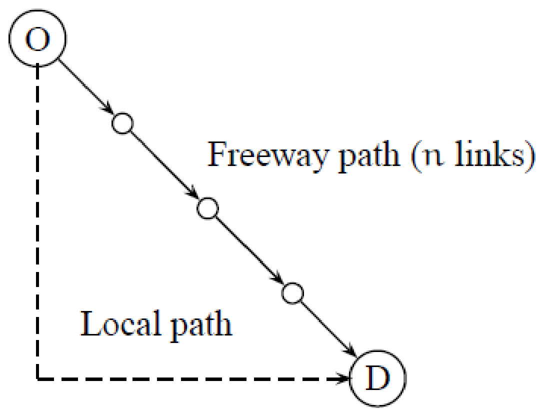


Figure 0.1 Freeway Path and Local Path

Suppose the freeway path consists of *n* freeway links, whose travel time random variables are X_1, X_2, …, X_n with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ. Let *Y* denote freeway path travel time: $Y = \sum_{i=1}^{n} X_i$. Then the expected path travel time is $E[Y] = sum(\mu)$, and the standard deviation is $std[Y] = \sqrt{sum(\Sigma)}$, where *sum* means the summation of all elements in the vector/matrix. It is also assumed that the local path travel time is fixed *Z*.

The problem is to decide which path is optimal, given the distribution of freeway link travel time random variables. The optimality criterion is more than minimum expected travel time. Note that, in real-life transportation networks, freeway link/path travel time is generally shorter but with higher risk than local link/path travel time. In general, travelers are risk-averse when making route choice in a stochastic network. For example, suppose freeway path has travel time of 10 or 20 minutes, each with probability 1/2, and local path travel time is fixed 15 minutes. Under such circumstance, travelers tend to choose local path, as it takes no risk, though both paths have the same expected travel time. We adopt two approaches to modeling travelers' risk-averse attitude. The first one follows the expected utility theory from economics and minimizes an expected disutility function of travel times (Mirchandani and Soroush, 1985). The other one minimizes a disutility function that is a linear combination of mean travel time and standard deviation (std), which is a common method used in empirical studies of travel time reliability (Lam and Small, 2001).

In order to take into account individual error and other factors, stochastic choice model is applied instead of deterministic choice model. In deterministic choice model, given the expected utility/disutility of both paths, travelers either choose freeway path or local path with probability of 1; in stochastic choice model, the probability is smaller than 1, that is, part of travelers will not choose optimal path solution. Logit model is assumed, so the portion of travelers choosing freeway path is given as follows:

$$\begin{aligned} P(freeway) &= \frac{\exp\left(-V(freeway)\right)}{\exp\left(-V(freeway)\right)+\exp\left(-V(local)\right)} \\ &= \frac{1}{1+\exp\left(-V(local)+V(freeway)\right)} \end{aligned} \tag{0.1}$$

where V(freeway) is the systematic disutility of the freeway path, and V(local) the systematic disutility of the local path.

There are two specifications for the systematic utility, one is expected utility (e.g., exponential disutility function), and the other is mean-standard deviation (e.g., a linear combination of mean travel time and standard deviation).

- Exponential disutility

Generally, for path travel time Y, the disutility is $\exp(aY)$, where a is risk aversion factor, a positive parameter which represents traveler's risk-averse attitude. Specifically, if path travel time has normal distribution $Y \sim \mathrm{N}(\mu, \sigma^2)$, the disutility has log-normal distribution $\exp(aY) \sim \text{Log-N}(a\mu, a^2\sigma^2)$, and the expected disutility is $V[Y] = \mathrm{E}[\exp(aY)] = \exp(a\mu + a^2\sigma^2/2)$.

Generally, when risk aversion parameter a is larger, the traveler is more risk-averse, and so the freeway is less attractive. When a is close to 0, the traveler is close to risk-neutral. Traveler's risk-averse attitude grows fast with a. For example, suppose freeway path has stochastic travel time of 10 or 20 minutes, each with probability 1/2, and local path travel time is fixed x minutes. Table 0.1 shows different a value and the corresponding x value such that a traveler is indifferent in choosing either path. Note that the traveler becomes extremely risk-averse when $a \geq 1.0$, and this is not usual in real life.

Table 0.1 Traveler's Risk-Averse Attitude

a	0.01	0.1	0.2	0.5	1.0	1.5	2.0	3.0
x	15.1	16.2	17.2	18.6	19.3	19.5	19.7	19.8

For exponential disutility function, the highway share (the portion of travelers choosing freeway path) calculated by the Logit model is:

$$P(freeway) = \frac{1}{1 + \exp\left(-e^{aZ} + e^{a\mu + a^2\sigma^2/2}\right)} \tag{0.2}$$

- Mean-standard deviation disutility

Generally, for path travel time Y with mean μ and standard deviation σ, the mean-standard deviation disutility is $V(Y) = c_1\mu + c_2\sigma$, where c_1 and c_2 are systematic parameters. Note that c_1 and c_2 are generally positive to represent traveler's risk-averse attitude, the degree of which is shown by the ratio c_2/ c_1.

For mean-standard deviation disutility, the highway share is:

$$P(freeway) = \frac{1}{1 + \exp\left(-c_1(Z - \mu) + c_2\sigma\right)} \tag{0.3}$$

3.3 Theoretical Analysis

Suppose freeway link travel time random variables $X_1, X_2, \ldots, X_n$ are multivariate normally distributed with mean vector $\boldsymbol{\mu}$ and covariance matrix $\sum$: $X_1, X_2, \ldots, X_n \sim \mathrm{MVN}(\boldsymbol{\mu}, \sum)$. Assume all freeway link travel times are with the same normal distribution (i.e., the same mean μ and variance σ^2): $X_i \sim \mathrm{N}(\mu, \sigma^2)$, and the correlation coefficient between any pair of freeway link travel times is the same ρ. Note that, in order to ensure such $\sum$ can be a covariance matrix, it has to be semi-positive definite, so it is required that $\rho \geq -1/(n-1)$. Thus, the expected path travel time is $E[Y] = n\mu$, and the standard deviation is $std[Y] = \sqrt{n(1+(n-1)\rho)}\sigma$. It is also assumed that the local path travel time is fixed $Z = kn\mu$.

For exponential disutility function, the expected disutility of freeway path travel time is $V[Y] = \mathrm{E}[\exp(aY)] = \exp(an\mu + a^2n(1+(n-1)\rho)\sigma^2/2)$, while the disutility of local path travel time is $V[Z] = \mathrm{E}[\exp(aZ)] = \exp(kan\mu)$. The highway share is:

$$P(freeway) = \frac{1}{1 + \exp\left(-e^{kan\varpi} + e^{an\mu + a^2\sigma^2 n(1+(n-1)\rho)/2}\right)} \qquad (0.4)$$

When risk aversion $a < \frac{2(k-1)\mu}{(1+(n-1)\rho)\sigma^2}$, travelers tend to choose freeway path.

For mean-standard deviation disutility, the disutility of freeway path travel time is $V(Y) = c_1 n\mu + c_2\sqrt{n(1+(n-1)\rho)}\sigma$, while the disutility of local path travel time is $V(Z)$ = $c_1 kn\mu$. the highway share is:

$$P(freeway) = \frac{1}{1 + \exp\left(-c_1(k-1)n\mu + c_2\sqrt{n(1+(n-1)\rho)}\sigma\right)} \qquad (0.5)$$

When risk aversion $c_2 / c_1 < \frac{(k-1)\sqrt{n}\mu}{\sqrt{1+(n-1)\rho}\sigma^2}$, travelers tend to choose freeway path.

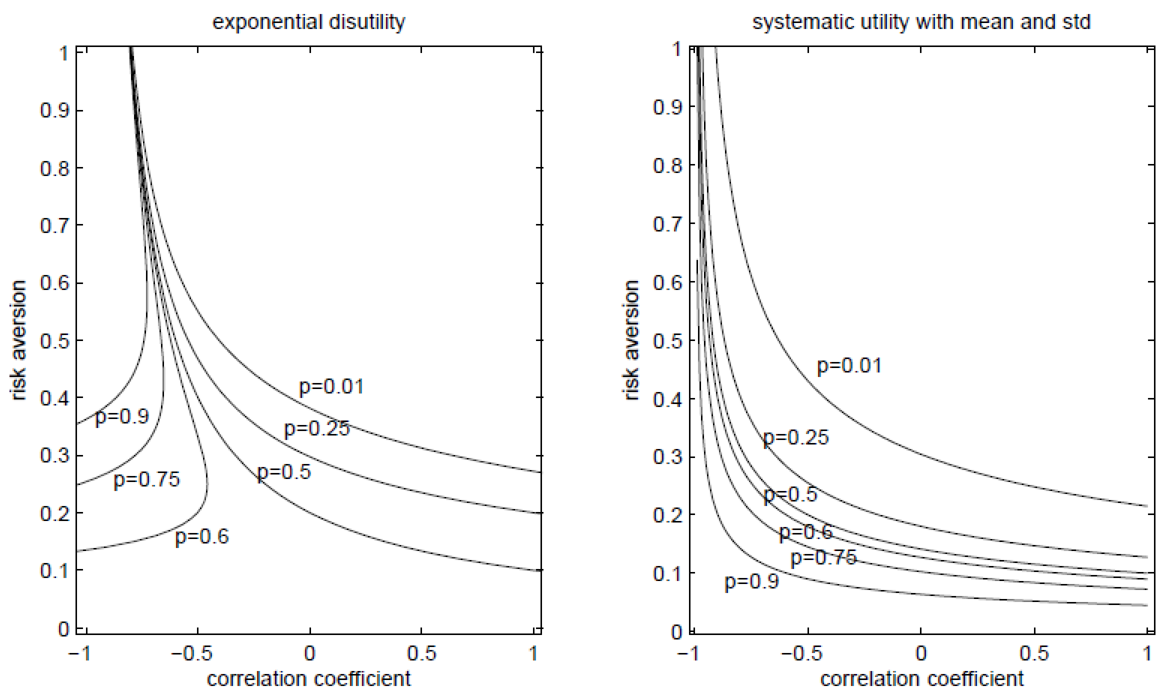


Figure 0.2 Highway Share and the Corresponding Risk Aversion and Correlation Coefficient

Figure 0.2 shows the relationship between highway share contour and risk aversion and correlation coefficient. In order to make the correlation coefficient span the range from -1 to 1, we set the number of freeway links as $n = 2$. Other preset parameter values are: $k = 1.1$, $c_1 = 0.2$, $\mu = 4$, $\sigma = 2$. Both plots show that, given a positive correlation coefficient, when travelers are less risk averse, highway share is higher; and given risk aversion, when the correlation is lower, highway share is higher.

Note that there are anomalies in the contour plot for exponential disutility when the correlation coefficient is negative. For example, when $\sigma = -0.5$, the highway share first increase and then decrease with risk aversion parameter. This counter-intuitive result can be explained by examining further the Logit model based on the expected disutility. The expected disutility for freeway path is $V[Y] = \exp(2a\mu + a^2(1+\rho)\sigma^2)$, which is always increasing with a when $a > 0$. The disutility for the local path is $V[Z] = \exp(2ka\mu)$, which is also increasing with a. The highway share, however, depends on the difference of the expected disutilities, which is not necessarily monotonic with a. When a is relatively small, the disutility of the local path might increase more than proportionally of the freeway path expected disutility increase.

The disutility function in general describes how people value outcomes, and a convex one says that people have increasing sensitivity to the travel time – a 10 minutes increase from 100 to 110 minutes is more onerous than the same 10 minutes increase from 10 to 20 minutes. However, it is more reasonable to assume a diminishing sensitivity – the increase doubles the total travel time in the latter case but only worsen the trip marginally in the former case. Given the counter-intuitive result from the exponential disutility model and also our concern over the validity of "increasing sensitivity” to travel time, we decide to adopt only the mean-standard deviation formulation for the simulation analysis next.

3.4 Data Processing

In order to investigate the characteristics of correlations among random link travel times in a real-life traffic network, we process traffic data on a road section, which is a 4.79 mile (7.71 km) segment of Interstate 10 E in Los Angeles, California, as shown in Figure 0.3.

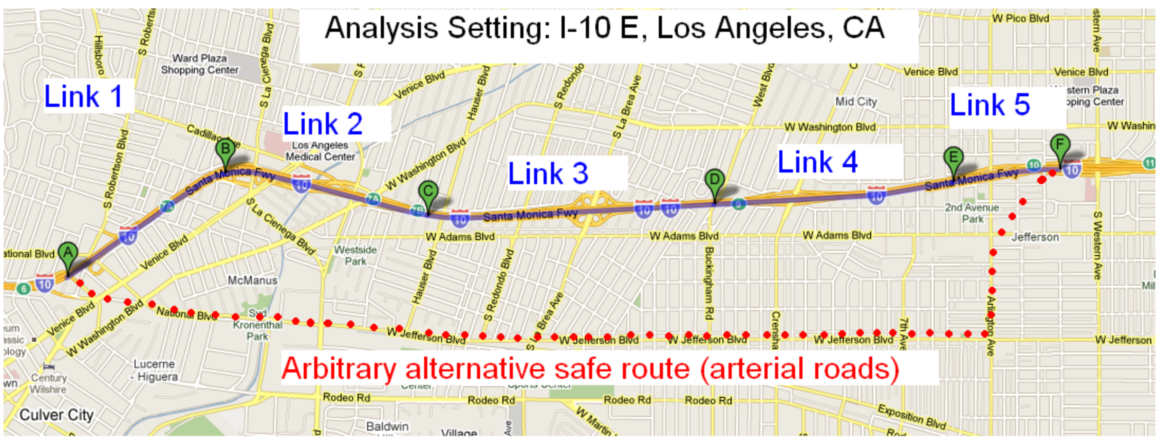


Figure 0.3 Analysis Setting

It stretches between 5.64 mile (9.07 km) marker (or exit 6) and 10.43 mile (16.78 km) marker (or exit 10). The primary reasons for this choice are high levels of congestion and large traffic volumes. The freeway is monitored by California Department of Transportation Performance Measurement System (Caltrans PeMS), which provides traffic information in an online database. It has been divided into 5 consecutive links, each approximately 1 mile (1.61 km) in length. The main criteria for link limits were detector locations directly downstream or upstream of exit ramps; downstream locations are preferred to minimize impact of ramps queues on lane detector readings.

5-minute speed data aggregates have been gathered from PeMS for a total of 87 non-holiday weekdays between March 1st, 2010 and June 30th, 2010 from all 17 detectors along the studied freeway segment. The period between 7:00-10:59:59 AM has been chosen for two reasons: it includes the morning peak hour (estimated to be approximately 7:45-9:30 AM) as well as time right before and after the peak. This allowed us to observe correlations for the peak and off-peak periods. The length of each link has been divided by a harmonic mean of speed detector readings on that particular link to obtain the approximate travel times from the speed data.

For the entire segment, the mean travel time is 7.20 min, with minimum 4.12 min, and maximum 18.48 min. There are 240 random variables, each with 87 observations. Note that we have time-dependent travel time random variables to study both spatial and temporal correlations.

Pearson’s correlation coefficients between each pair of the 240 time-dependent random link travel time variables are calculated in MATLAB for the observed travel time data. Figure 0.4 depicts spatial and temporal correlations for travel time with regard to link 1 at different times.

Intuitively, correlations should drop over temporal and spatial distance – this presumption is correct and the steady drop is clearly shown in the figure. It also shows the dropping rate along the time dimension depends on the distance of the two links. For example, consider the figure for Link 1 at 9:00. The correlation is the highest (1) with Link 1 at 9:00 (itself), and it decreases within the same link with time periods either earlier or later than 9:00. The peaks can also be observed on other links at around 9:00, however the curves get flatter when the other link is farther way. Off-peak period (e.g., 7:00 AM), however, is characterized by significantly lower correlations. This also follows the intuition, as off-peak periods usually have considerably lower traffic densities than peak hours, and thus probably see less interactions and dependencies (e.g. those through queue spillbacks) among link variables.

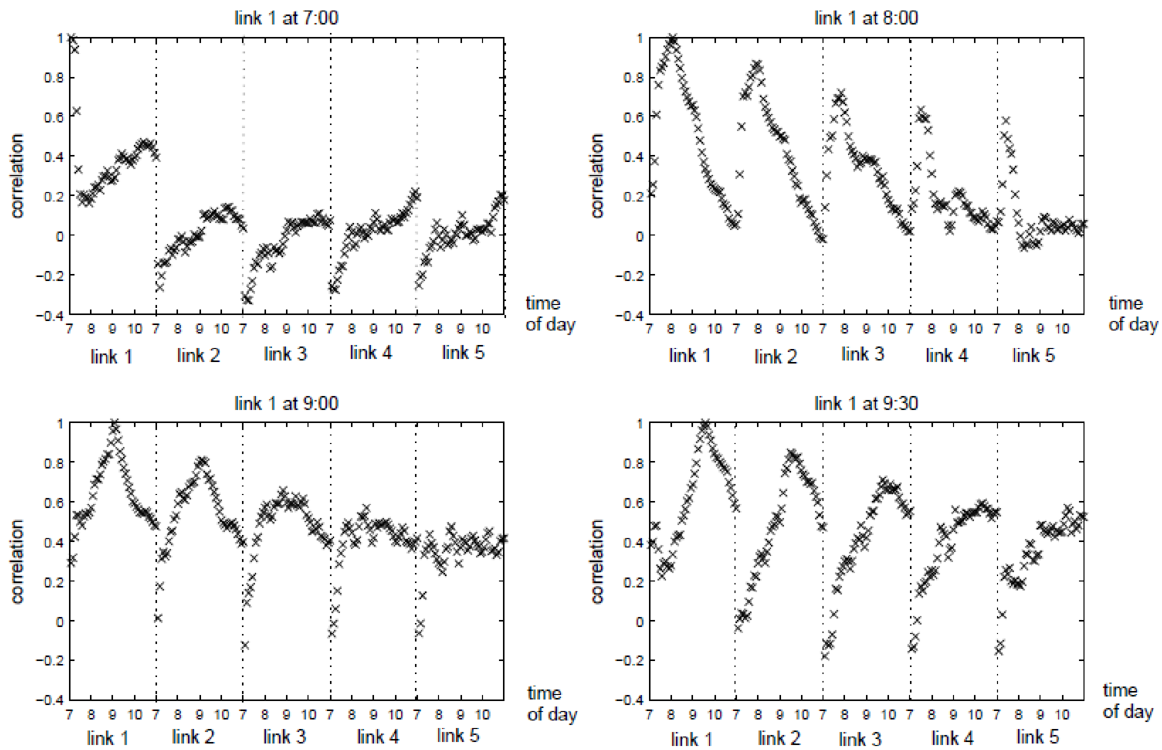


Figure 0.4 Link 1 Correlation Patterns

Another distinctive characteristic of the period before the peak hour is the presence of negative correlations, inexistent or insignificant during the peak hour. One of the possible explanations is that commuting drivers are usually well aware of daily traffic fluctuations and try to escape the congestion by speeding up and getting off the highway as soon as possible before hitting the peak hour when the downstream links are starting to slow down. This explanation is to be verified in future research by collection and analysis of larger samples and possibly data on origin-destination trip rates. As mentioned, the dependencies for peak hour are much stronger, and Figure 0.5 extends that statement over the entire road segment under the study.

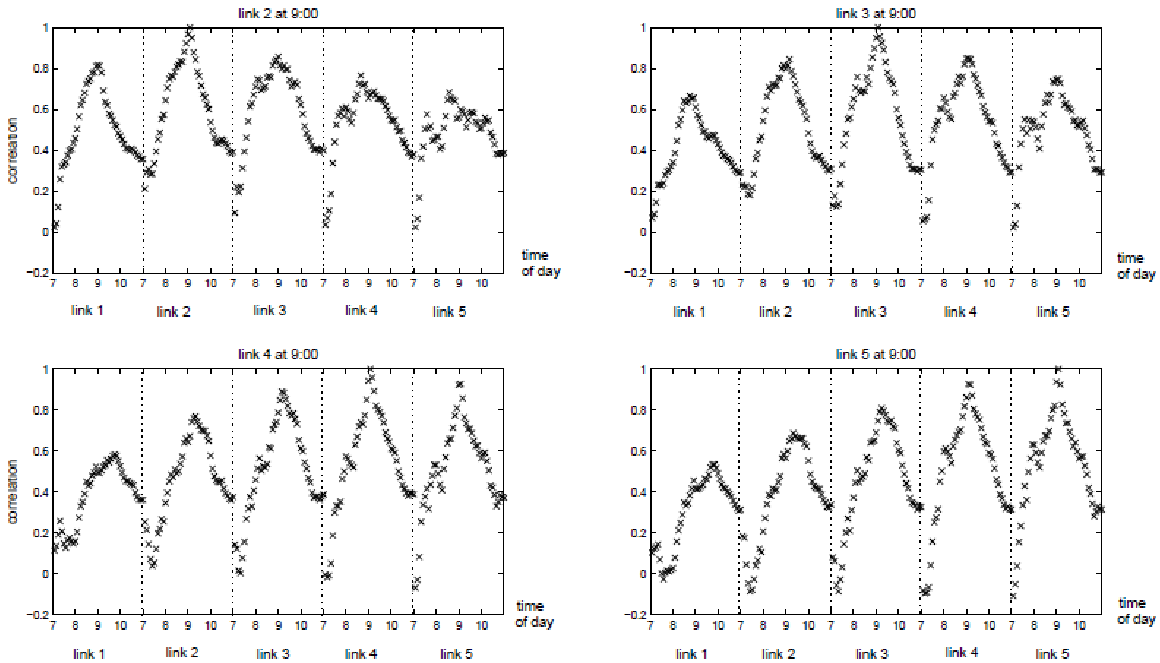


Figure 0.5 Peak Hour Correlations over the Freeway Segment

In order to quantify the correlation drop over time and distance, a multiple linear regression model is fitted to the data using two predictor variables – time difference and distance, and the responses vector – correlations. As presented in Table 0.2, the model consists of three components, each aiming to describe a different case in traffic condition. Note that the first constant in the model has been fixed to the value of 1 for all cases to force correlations with self to the correct prediction, thus making model more reflecting the reality. The variable "distance" denotes the difference in the number of link, e.g., the distance between link 1 and 4 is 3. Since there are in total 5 links, the range of "distance" is from 0 to 4. The variable “time_diff” denotes the difference in time with unit of minute. The study period is from 7:00 AM to 10:59:59 AM, with intervals of 5 minutes, so the range of “time_diff” is from 0 to 235. “OP_dummy” and “OO_dummy” are dummy variables. “OP_dummy” is 1 if one random link travel time variable is in peak period and the other in off-peak and 0 other wise; similarly, “OO_dummy” is 1 if both random link travel times are in the off-peak period and 0 otherwise.

Table 0.2 Regression Results

	Regression Model			N=87	R^2=0.6826
Variables	Peak-Peak	Offpeak-Peak	Offpeak-Offpeak	standard error	t-test
constant	1(fixed)				
distance	-0.1591			0.001324	-120.191
time_diff	-0.0059			2.8E-05	-210.57
interaction (distance*time_diff)	0.0011			2.03E-05	56.24307
distance*OP_dummy		-0.0909		0.0028	-32.4505
time_diff*OP_dummy		-0.0011		0.001834	17.70257
interaction*OP_dummy		0.0012		4.81E-05	-22.7568
distance*OO_dummy			0.0325	3.59E-05	14.2987
time_diff*OO_dummy			0.0005	3.63E-05	32.95882
interaction*OO_dummy			-0.0004	2.46E-05	-17.4999

The base model predicts correlations for the Peak-Peak case, which tends to be primarily controlled by distance as the strongest parameter. In Off-peak-Peak case, the model indicates an increase of the influence of both distance and time difference. The fact that time difference parameter is the most significant in the Off-peak-Peak model agrees with the observed correlations plot in Figure 0.5 (indicating negative correlations in far downstream links). In contrast, the Off-peak-Off-peak case parameters tend to weaken the base model: all the variables have opposite signs as the main predictors. This interpretation is confirmed by the 7:00 AM plot on Figure 0.4, where the slope of correlations drop is not as steep as on the other plots.

Since the presented linear model is not bounded, it is valid for small distances over time and/or space only. As the distance approaches infinity, the model will go to negative infinity; thus, the work should be continued on non-linear models that would allow for more general applications. Since there indeed exists negative correlation, and the correlation should go to 0 as the distance approaches infinity, the regression function should not be monotonic. It might be in the shape of Figure 0.6. Our current linear model can be viewed as approximating the more general non-linear model for small distances.

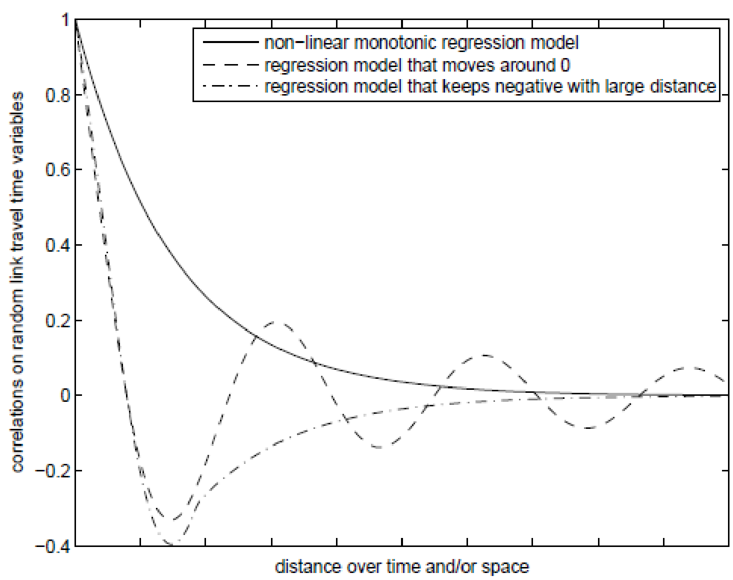


Figure 0.6 Hypothesis of Non-Linear Regression Model

3.5 Simulation

The simulation is run on the 5-link road section for 4 time intervals in peak hour (8:30-8:49:59 AM). There are 20 link travel time random variables, which are assumed to be multivariate normal distributed (distribution truncated at 0). The mean vector and variance vector are obtained from the data. The correlation coefficient matrix is calculated using the regression model for Peak-Peak case, since all 4 time intervals are in peak hour, i.e.,

$$y = 1 - 0.1591x_1 - 0.0059x_2 + 0.0011x_1x_2 \tag{0.6}$$

where y is correlation, x_1 = (0, 1, 2, 3, 4) is spatial distance between links, and x_2 = (0, 5, 10, 15) is time difference.

With mean vector, variance vector, and correlation coefficient matrix, the 20 link travel time random variables are generated for 100,000 samples. Freeway path travel time is calculated for two cases: 1) dependency is taken into account (normal case); 2) dependency is not taken into account (Miller-Hooks and Mahmassani, 2000). The distribution of freeway path travel time is obtained with mean E[Y] and standard deviation std[Y] for both cases. Stochastic choice model is applied in the simulation to calculate highway share, and systematic utility with mean and standard deviation for path travel time is employed. It is assumed c_1 = 0.2, and c_2 = 0.5. Thus, highway share is:

$$P(freeway) = \frac{1}{1 + \exp\left(-0.2\left(Z - E[Y]\right) + 0.5std[Y]\right)} \tag{0.7}$$

With the 100,000 samples of the 20 link travel time random variables, for case 1, freeway path travel time has mean E[Y] = 10.0428 and standard deviation std[Y] = 2.4206; and for case 2, freeway path travel time has mean E[Y] = 10.0018 and standard deviation std[Y] = 0.7501. Assume local path travel time is fixed Z = 15. Highway share is 44.55% for case 1 and 65.13% for case 2.

Sensitivity analysis is conducted for three parameters:

- Distance parameter: the coefficient of x_1 in the regression model is changed from -0.3182 to 0, and the coefficient of x_1x_2 is changed with the same ratio;
- Time parameter: the coefficient of x_2 in the regression model is changed from -0.0118 to 0, and the coefficient of x_1x_2 is changed with the same ratio;
- Risk attitude parameter: the coefficient of std[Y] in the stochastic choice model is changed from 0 to 1, so risk aversion parameter c_2/c_1 changes from 0 to 5.

The highway share results for both cases are shown in the following figures:

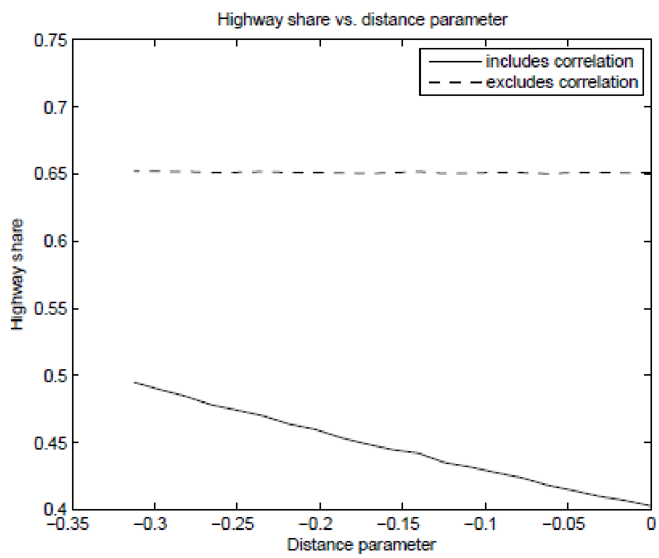


Figure 0.7 Highway Share vs. Distance Parameter

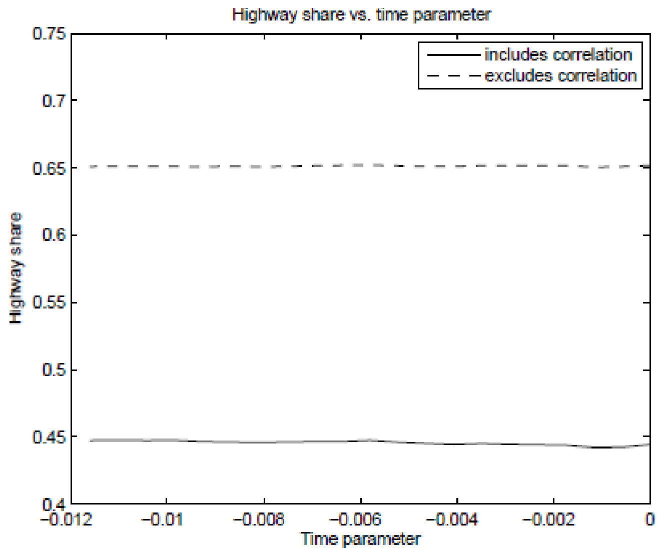


Figure 0.8 Highway Share vs. Time Parameter

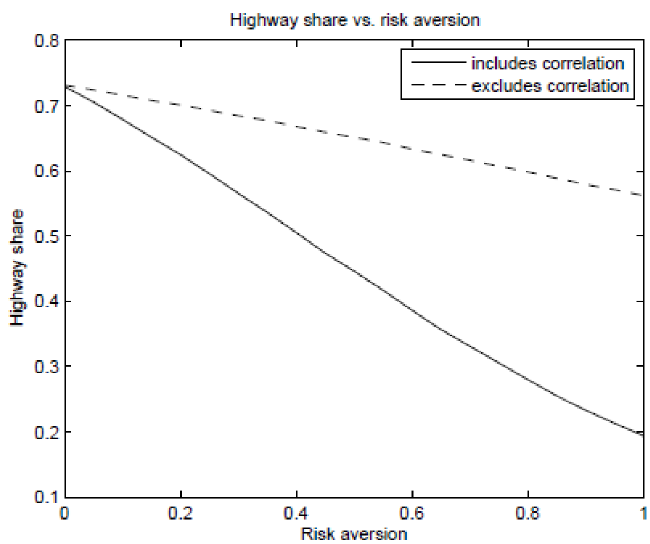


Figure 0.9 Highway Share vs. Risk Aversion Parameter

Note that all three figures show that case 2 has a higher highway share than case 1. The reason is that freeway path travel time has almost the same mean in case 1 and case 2, but a larger standard deviation in case 1 than in case 2, so with above stochastic choice model, case 1 has a smaller highway share. Intuitively, when dependency is not taken into account, the risk on the freeway path will be underestimated and thus the model will make a biased prediction that favors the freeway path.

Figure 0.7 shows that the highway share decreases with distance parameter increasing in case 1, while does not change much in case 2. The reason is that standard deviation increases with distance parameter in case 1, while does not change in case 2. In the regression model, distance x_1 has a negative coefficient, and when it increases from around -0.3 to 0, correlation increases significantly, and so standard deviation increases, which makes highway share decreases. On the other hand, in case 2, adjusting the linear regression parameters will not change standard deviation, and so highway share does not change much with it.

Figure 0.8 shows that highway share does not change much with time parameter increasing even in case 1. The reason is that the coefficient of time difference x_2 in regression model is close to 0 and its absolute value is much smaller than that of distance parameter, so adjusting it will not affect correlation and standard deviation much, and thus highway share does not change much.

Figure 0.9 shows that highway share increases with risk attitude parameter increasing (which means travelers are less risk averse) in both case 1 and 2, and the two cases get almost the same highway share when risk attitude parameter becomes 0. The reason is that the larger the coefficient of standard deviation in stochastic choice model is, the larger highway share is. When it becomes 0, there is just no standard deviation term in the Logit model, and since case 1 and 2 have almost the same mean, they will have almost the same highway share. Intuitively, when travelers are less risk averse, freeway path becomes more attractive and so highway share increases. When travelers are risk neutral, no risk is taken into account any more, so case 1 and 2 are no more different.

3.6 Conclusions and Future Directions

In this chapter, traffic data from an urban freeway segment are obtained from the PeMS database and analyzed to study the characteristics of stochastic dependencies among link travel times. It is shown that correlations between link travel times drop over temporal and spatial distances. We also show that route shares of flows are different when network stochastic dependency is taken into account and when it is not. Specifically, when dependency is not taken into account, travelers underestimate the risk of fast and risky route (i.e., freeway path), and thus are more likely to choose it. Both theoretical analysis and computational tests show that fast and risky route is more attractive when link correlation and/or risk aversion is low. It is also shown that the difference of the route shares between complete dependency case and no dependency case is larger when correlation and/or risk aversion is higher.

For future direction, we would like to continue the work on analyzing stochastic transportation networks using freeway data: 1) to investigate reasons for existence of negative correlations on downstream links at near-peak periods; 2) to perform partial correlation analysis on samples; and 3) to apply a non-linear regression model on correlations like the ones in Figure 0.6.

We would also like to make use of the correlations on the algorithm design side. For example, design a practical representation of stochastic network with the following attributes: 1) it can be efficiently stored in a computer memory; 2) it captures the essential dependencies for routing; 3) it does not overly complicate the algorithm design.

Algorithms can be designed based on the representation of stochastic network, and theoretical complexity of the developed algorithms is to be studied. Computational tests of the developed algorithms are to be performed in hypothetical and real-life networks to determine: 1) whether the consideration of stochastic dependencies significantly increase the algorithm average running time; and 2) how far off a routing algorithm is in terms of minimizing expected travel time or expected disutility, if stochastic dependencies are ignored.

CHAPTER 4

INFORMATION ON ADAPTIVE ROUTING IN STOCHASTIC NETWORKS

4.1 Introduction

In this chapter, three types of partial online information are introduced: delayed global information, global pre-trip information and radio information on a subset of links without delay. Compared with perfect online information (Gao and Chabini, 2006), the first two are limited temporally and the last spatially. The contributions of the chapter are threefold: 1) a theoretical proof that for optimal adaptive routing in a flow-independent stochastic time-dependent (STD) network, more error-free information is always better (or at least not worse); 2) an analysis of the optimal adaptive routing problem with partial and no online information indicating that Bellman's principle of optimality does not apply, and the proposal of a set of necessary conditions for optimality; and 3) a heuristic algorithm based on the necessary conditions with polynomial running time and satisfactory effectiveness tested computationally.

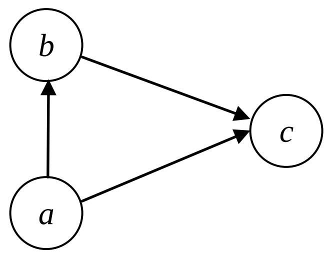
This chapter is organized as follows. In Section 4.2 , the optimal routing policy problem in an STD network is defined for partial online information situations. Section 4.3 presents a theoretical proof of the non-negative value of error-free traveler information. In Section 4.4 , Bellman's principle of optimality is shown to be invalid for the problem with partial and no online information. A set of necessary conditions for optimality is then proposed and proved. A heuristic algorithm is designed based on the necessary condition and computational test results are presented. Section 4.5 gives conclusions and future research directions.

4.2 Problem Definition

4.2.1 The Network

Let $G = (N, A, T, \tilde{C})$ denote a stochastic time-dependent network. N is the set of nodes and A is the set of links, with $|N| = n$ and $|A| = m$. It is assumed that there is at most one directional link from node j to k, and thus a link can be denoted as (j, k). T is the set of time periods $\{0, 1, \ldots, K\text{-}1\}$. A support point is defined as a distinct value (vector of values) that a discrete random variable (vector) can take. Therefore a probability mass function (PMF) of a random variable (vector) is a combination of support points and the associated probabilities. Throughout this chapter, a symbol with a ~ over it is a random variable (vector), while the same symbol without the ~ is its support point. 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 support points. The link travel time random variables are assumed to be positive integers. Beyond time period K-1 travel times are static, i.e., travel times on link (j, k) at any time $t > K\text{-}1$ is equal to that at time $K - 1$ for any given support point. The time period from 0 to K-1 is denoted as the dynamic period, while that beyond K-1 static period. It is generally possible to model the peak period as dynamic, while off-peak as static when traffic is more stable. $\{C^1, \ldots, C^R\}$ is the set of support points of the joint probability distribution of all link travel times at all times, where C^r is a vector of time-dependent link travel times with a dimension $K \times m$, $r = 1, 2, \ldots, R$. $C^r_{jk,t}$ is the travel time of link (j, k) at time t in the r-th support point, which has a probability p_r, and $\sum_{r=1}^{R} p_r = 1$.

An example network is shown in Figure 0.1 with 3 nodes, 3 links and 2 time periods. There are 3 support points, each with a probability of 1/3, for the joint distribution of 6 travel time random variables (links (*a, b*), (*b, c*) and (*a, c*) over time periods 0 and 1). A support point can be conveniently viewed as a day. Travel times beyond time 1 are the same as those at time 1 for each of the 3 support points.



Time	Link	C^1	C^2	C^3
0	(a, b)	1	1	1
	(b, c)	2	2	1
	(a, c)	3	3	2
1	(a, b)	1	1	2
	(b, c)	1	2	1
	(a, c)	3	2	2

$$p_1 = p_2 = p_3 = 1/3$$

Figure 0.1 A Small Network

The framework and methods developed in this chapter can be extended to a network with turn penalties by augmenting the network with additional links corresponding to turning movements. As the focus of this chapter is on imperfect information, we limit our discussion to a basic network without turn penalties.

The discrete distributions of link travel times are assumed for the convenience of defining routing policies (Section 3.4), which are based on realized travel times. Even if the underlying travel time distribution is continuous, in order to define a routing policy with a finite number of states, one has to discretize the distribution. The extension of the routing policy definition to a continuous travel time distribution is a challenging task and will be included in the future work.

4.2.2 Online Information

Let H be a trajectory of (node, time) pairs a traveler could experience in the network to the current node j and time t: $H = \{(j_0, t_0), \ldots, (j, t)\}$, where j_0 is the origin, t_0 is the departure time, j is the current node and t is the current time. Denote the information coverage on links and time periods as $Q \subseteq A \times T$. Information is represented as the travel time realizations on time-dependent links in Q. It is assumed there is no error in revealing the true travel times, i.e., a 1 minute travel time will be revealed as 1 minute, not any other value. An information scheme is defined as a mapping from trajectory H to coverage Q, that is, information depends on traversed locations and times. Here are examples of online information schemes with trajectory $H = \{(j_0, t_0), \ldots, (j, t)\}$:

- Perfect online information (Gao and Chabini, 2006): $Q^{\mathrm{POI}}(H) = A \times \{0,1,\ldots,t\}$ (all links up to the current time)
- Global information with time lag Δ: $Q^{\mathrm{LAG}}(H) = A \times \{0,1,\ldots,t - \Delta\}$ (all links up to Δ time ago)
- Global pre-trip information with departure time t_0: $Q^{\mathrm{PRE}}(H) = A \times \{0,1,\ldots,t_0\}$ (all links up to the departure time t_0)
- Radio information on $B \subseteq A$ with no time lag: $Q^{\mathrm{RADIO}}(H) = B \times \{0,1,\ldots,t\}$ (a subset of links up to the current time)
- No online information (see e.g., Gao and Chabini, 2006): $Q^{\mathrm{NOI}}(H) = \varnothing$ (no information on any link at any time)

The example in Figure 0.1 is used to illustrate the different information schemes. At time 0 and any node, a traveler with POI knows the travel time realizations of { $\tilde{C}_{ab,0}$, $\tilde{C}_{bc,0}, \tilde{C}_{ac,0}$ } which could be either {1,2,3} or {1,1,2}; a traveler with global information with a lag of 1 minute does not know any travel time realization yet; a traveler with global pre-trip information with departure time 0 has the same knowledge as with POI; a traveler with radio information on link (*a*, *b*) with no time lag knows the travel time realization of $\tilde{C}_{ab,0}$ which is always 1; and a traveler with NOI simply does not know any travel time realization.

As the time moves from 0 to 1, more information could be obtained while that from time 0 is kept. A traveler with POI knows the travel time realizations of { $\tilde{C}_{ab,0}$, $\tilde{C}_{bc,0}, \tilde{C}_{ac,0}, \tilde{C}_{ab,1}, \tilde{C}_{bc,1}, \tilde{C}_{ac,1}$ } which could be each of the 3 support points; a traveler with global information with a lag of 1 minute knows what happened at time 0: the travel time realizations of { $\tilde{C}_{ab,0}, \tilde{C}_{bc,0}, \tilde{C}_{ac,0}$ } which could be either {1,2,3} or {1,1,2}; a traveler with global pre-trip information with departure time 0 does not gain any more information *en route* and thus his/her information remains unchanged ; a traveler with radio information on link (*a*, *b*) with no time lag knows the travel time realization of { $\tilde{C}_{ab,0}, \tilde{C}_{ab,1}$ } which could be {1,1} or {1,2}; and a traveler with NOI still does not know any travel time realization.

As the time moves from 1 to 2, only the traveler with global information with a lag of 1 minute will gain more useful information, as he/she now knows what happened in time 1. A traveler with POI, pre-trip or radio information does not gain any more useful information because his/her information is always up-to-date and the information he/she had at time 1 is enough for any time periods beyond 1 due to the static period assumption. A traveler with NOI does not gain any more information by definition. Note that routing under no online information could still be adaptive to the arrival time at each decision node, which is random due to random travel times.

4.2.3 Event Collection

The concept of event collection is generalized from that defined in Gao and Chabini (2006) to the case of a general information scheme. Let $\tilde{C}_Q$ be the vector of random travel times of all time-dependent links in *Q*. For a given support point C_Q, there exists one or more support points C of the network, such that the travel time on any time-dependent link in *Q* is the same in both C_Q and C. In other words, for any possible revealed link travel times in *Q*, a set of support points of the network that are compatible with the information can be identified. Such a set is defined as an event collection, *EV*. As more information is collected, information coverage *Q* grows and the size of *EV* decreases or remains unchanged. When *EV* becomes a singleton, a deterministic network (not necessarily static) is revealed to the traveler. If a traveler has perfect online information with $Q^{\text{POI}} = A \times \{0, 1, \ldots, t\}$, the network becomes deterministic no later than the start of the static period, i.e., $K - 1$. When travelers have less than perfect online information, it is possible that the network remains stochastic beyond the dynamic period.

In the example of Figure 0.1, it is assumed that a traveler has POI. At time 0 he/she received the information that travel times on links (*a*, *b*), (*b, c*) and (*a, c*) are 1, 2 and 3 respectively. By utilizing his/her *a priori* knowledge of the joint distribution of link travel times, he/she can infer that support points C^1 or C^2 are possible as both provide compatible travel times with what is revealed, while support point C^3 is not. Therefore his/her event collection is $\{C^1, C^2\}$. As the time moves from 0 to 1, the traveler obtains more information. If the newly revealed travel times on links (*a, b*), (*b, c*) and (*a, c*) are 1, 1 and 3 respectively, the traveler knows for sure that support point C^1 will be realized and his/her event collection is $\{C^1\}$. Similarly, If the newly revealed travel times on links (*a, b*), (*b, c*) and (*a, c*) are 1, 2 and 2 respectively, the traveler knows for sure that support point C^2 will be realized and his/her event collection is $\{C^2\}$.

Similarly a traveler with global information with a lag of 1 minute has no idea which support point will be realized at time 0 and his/her event collection is $\{C^1, C^2, C^3\}$. At time 1, he/she knows link travel times realized at time 0, and is faced with the same situation as a traveler with POI did at time 0. If the revealed travel times on links (*a, b*), (*b, c*) and (*a, c*) at time 0 are 1, 2 and 3 respectively, his/her event collection is $\{C^1, C^2\}$. At time 2, he/she will have an event collection $\{C^1\}$ or $\{C^2\}$. The same logic can be applied to other information schemes. Note that for NOI, the event collection remains as $\{C^1, C^2, C^3\}$ for any time period.

All the possible event collections with information coverage *Q*, denoted as $\boldsymbol{EV}(Q)$, can be generated by performing a partition of $\{C^1, \ldots, C^R\}$ based on $\tilde{C}_Q$. $\boldsymbol{EV}(Q) = \{EV_1, EV_2, \ldots\}$, where $C^r_{jk,t}$ is invariant over $r \in EV_i$, $\forall((j,k),t) \in Q$, $\forall i$, and $\exists\ ((j,k),t) \in Q$ such that $C^r_{jk,t} \neq C^{r'}_{jk,t}$, for $r \in EV_i$, $r' \in EV_j$, $j \neq i$, $\forall i$, $\forall j$. In other words, support points in an *EV* are undistinguishable in terms of revealed travel times on links in *Q*, but are distinctive from those in another *EV*. All the possible event collections for a given information scheme can be generated in preprocessing. Here are some important facts about event collections:

- There is no overlapping among elements of $\boldsymbol{EV}(Q)$, so there are at most R event collections at any certain time and location ($|\boldsymbol{EV}(t)| \leq R$);
- Any element EV of $\boldsymbol{EV}(Q)$ is a subset of one and only one element EV' of a later $\boldsymbol{EV}(Q')$: $EV' \cap EV = \varnothing$ or EV';
- $|\boldsymbol{EV}(Q)| \geq |\boldsymbol{EV}(Q')|$;
- The conditional probability of $EV \in \boldsymbol{EV}(Q)$ given $EV' \in \boldsymbol{EV}(Q')$ can be evaluated as follows: $\Pr(EV' | EV) = \sum_{r \in EV' \cap EV} p_r \Big/ \sum_{r \in EV} p_r$

The generation of event collection can be carried out in increasing order of time, as the information coverage can only grow and later partitions can be done based on earlier ones. An example from Figure 0.1 is shown here for a traveler with up-to-date radio information on link (a, b). Since the information coverage depends only on the current time t, not the trajectory, $Q(H)$ can be simplified as $Q(t)$ and $\boldsymbol{EV}(Q)$ as $\boldsymbol{EV}(t)$. At time 0, information coverage $Q(0) = \{(a, b)\} \times \{0\}$. The travel time on link (a, b) at time 0 is 0 for all 3 support points, so the partition yields only one event collection and $\boldsymbol{EV}(0) = \{\{C^1, C^2, C^3\}\}$. At time 1, information coverage $Q(1) = \{(a,b)\} \times \{0, 1\}$ where the incremental information is on $\{(a, b)\} \times \{1\}$. The partition can then be carried out on $\boldsymbol{EV}(0)$ based on travel time realizations of link (a, b) at time 1, which can be either 1 or 2. Therefore $\boldsymbol{EV}(1) = \{\{C^1, C^2\}, \{C^3\}\}$. During the static period, no more useful information will be available, so $\boldsymbol{EV}(t) = \{\{C^1, C^2\}, \{C^3\}\}$ for all $t > 1$.

Another example is shown for a traveler with global information with a lag of 1 minute. At time 0, Q (0) =∅, and thus $\boldsymbol{EV}$ (0) = {{C^1, C^2, C^3}}. At time 1, Q (1) = {(a,b), (b,c), (a,c)} × {0}. First check link-time pair ((a,b), 0) with only 1 possible value, and {{C^1, C^2, C^3}} remains unchanged. Next check ((b,c),0) with 2 possible values and {{C^1, C^2, C^3}} is partitioned as {{C^1, C^2}, {C^3}}. Lastly check ((a,c),0) and {{C^1, C^2}, {C^3}} remains unchanged because $C^1_{ac,0}$ and $C^2_{ac,0}$ are the same, while {C^3} is already a singleton. Therefore $\boldsymbol{EV}$ (1) = {{C^1, C^2}, {C^3}}. Similarly $\boldsymbol{EV}$ ($t \geq 2$) = {{C^1}, {C^2}, {C^3}}.

4.2.4 The Decisions and the Optimal Routing Policy Problem

It is assumed that travelers 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, EV}, where j is the current node, t is the current time, and EV is the current event collection.

Definition 0.1 (Routing Policy) A routing policy μ is a mapping from state to decision, for all possible states and all possible next nodes out of a given state, $\mu : x = \{j, t, EV\} \mapsto k$.

A routing policy can be visualized as a contingence table with as many rows as the number of combinations of node, time and event collection, and for each combination, a next node is given. A path is a purely topological concept and a special case of a routing policy, such that the same next node is given regardless of the time and event collection. The travel time by following a routing policy (sometimes terms routing policy travel time) from any origin and departure time to a destination is a random variable, with one realization in each support point. The routing policy travel time then can be represented as a list of travel times in all support points with the associated probabilities. The routing policy itself can also be viewed as a collection of paths with the associated probabilities.

For a routing policy, the next state $y = \left\{ k, \tilde{t}', \tilde{EV}' \right\}$ of the traveler is uncertain. The travel time on link (*j*, *k*) at time *t* given *EV* could be uncertain, resulting in an uncertain arrival time $\tilde{t}'$ at node *k*. The next event collection $\tilde{EV}'$ is uncertain because: 1) $\tilde{t}'$ is uncertain and thus the next information coverage $\tilde{Q}'$ is uncertain, e.g., at 8:00 with a possible travel time of 1 or 2 minute(s) on the next link, $\tilde{Q}'$ could cover either 8:01 or both 8:01 and 8:02; 2) Even with a given *Q'* and a given *t'*, travel times of links in *Q'* between *t* and *t'* are uncertain. For a given current state and a given decision, probabilities of all possible next states can be evaluated.

For a traveler with up-to-date radio information on link (*a*,*b*) in Figure 0.1, let $\mu\{a, 0, \{C^1, C^2, C^3\}\} = c$. The travel time on link (*a*, *c*) could be either 3 or 2 given the event collection {C^1, C^2, C^3}, with a probability of 2/3 or 1/3. If the travel time is 3, the event collection at node *c* will be an element of ***EV***(3); if the travel time is 2, the event collection at node *c* will be an element of ***EV***(2). In this specific example, ***EV***(3) = ***EV***(2), but generally they are not equal. Referring to the results from the last section, *EV'* could be either {C^1, C^2} or {C^3}, and *P*({C^1, C^2}|{ C^1, C^2, C^3}) = 2/3, *P*({C^3}|{ C^1, C^2, C^3}) = 2/3.

The traveler makes another decision at state *y*, and continues the process until the destination node is reached. The travel time of a routing policy from any initial state to a destination is a random variable; a routing policy can be manifested as different paths in different support points.

Definition 0.2 (Optimal routing policy problem). The optimal routing policy (ORP) problem in a stochastic time-dependent network is to find the routing policy that optimizes an objective function of routing policy travel times over all support points to a given destination, from a given origin and departure time.

Note that an optimal routing policy is not necessarily *ex post* optimal for any given support point (day), but is optimal on average over all possible support points.

The objective function could be, e.g., expected travel time, travel time variance, expected travel time schedule delay, or a combination of a number of criteria. The discussions in Section 4.3 are not restricted to a particular objective functional form. It however does affect the algorithm design and as such only expected travel time is dealt within Section 4.4 .

Let $e_{\mu}(j,t)$ be the objective function (to be minimized) of following routing policy μ from origin node j at departure time t to a given destination. The optimal objective function value $e^*(j,t)=\min_{\mu} e_{\mu}(j,t)$.

Given an information scheme, a partition of the universal support point set $\{C^1,\ldots,C^R\}$ at (j, t) provides the initial set of event collections $\boldsymbol{EV}(Q(j,t))$. Note that generally the event collection will change during the trip with more information (one exception being pre-trip information), as described in Section 4.2.3 . If the objective function is additive over support points, e.g., in the case of expected travel time or expected schedule delay, an optimal routing policy for the initial universal set of support points is also optimal for any of the initial event collections. In this case, finding an optimal routing policy for the universal set of support points is equivalent to finding an optimal routing policy for each of the initial event collection, and as such Section 5 deals with optimal routing policies with regard to initial event collections. However this is not necessarily true for a non-additive objective function, e.g., variance, and in such cases, solving an optimal routing policy problem cannot be broken down to solving a number of similar problems with initial event collections.

4.3 Theoretical Analysis of the Value of Information

We compare the optimal routing outcomes under two information schemes 1 and 2 in the same network with different coverage.

Assumption 0.1 For any trajectory H, information scheme 2 has a larger coverage Q_2 than that of information scheme 1, Q_1, that is, $Q_1(H) \subseteq Q_2(H)$.

Definition 0.3 (S_1 contains S_2). Let S_1 and S_2 be two partitions of S. S_1 is said to contain S_2 if for any $y \in S_2$, there exists $z \in S_1$, such that $y \subseteq z$. In other words, any element of S_2 is a subset of one and only one element of S_1, and any element of S_1 is the union of one or more elements of S_1. See Figure 0.2 for a schematic representation.

S	a	b	c	d	e	f	g	h
S_1	a	b	c	d	e |	f	g	h
S_2	a |	b	c |	d	e |	f |	g	h

Figure 0.2 A Schematic View of S_1 Containing S_2

Lemma 0.1. With assumption A1, $\boldsymbol{EV}(Q_1)$ contains $\boldsymbol{EV}(Q_2)$ for any trajectory H.

Proof. $\boldsymbol{EV}(Q_1)$ and $\boldsymbol{EV}(Q_2)$ are partitions of the set of support points $\{C^1, \ldots, C^R\}$. For any $EV_2 \in \boldsymbol{EV}(Q_2)$, travel times on time-dependent links of Q_2 are invariant across support points in EV_2. Since $Q_1 \subseteq Q_2$, travel times on time-dependent links of Q_1 are also invariant across support points in EV_2. Therefore there must exist $EV_1 \in \boldsymbol{EV}(Q_1)$ such that $EV_2 \subseteq EV_1$. **Q.E.D.**

With Lemma 0.1, we can proceed to compare the optimal objective function values under two different information schemes. Note that two travelers with different information schemes generally do not have the same starting information coverage and thus not the same initial set of event collections, even with the same origin and departure time. For example, assume the radio only reports travel times on the highway, while a pre-trip information source (e.g. a website) reports travel times on both the highway and arterial. There are two initial event collections under radio with the highway being normal or congested, and four initial event collections under pre-trip information, with the additional combination with the arterial being normal or congested. The comparison of the two information schemes is based on all the possible initial event collections under each scheme.

Theorem 0.1. With Assumption 0.1, the optimal objective function value under information scheme 2 is no worse than that under information scheme 1, for the same origin j_0 and departure time t_0.

$$e_2^*(j_0,t_0) \le e_1^*(j_0,t_0), \forall j_0 \in N, \forall t_0 \in T.$$

Proof. Given an optimal routing policy μ_1 under information scheme 1, an equivalent feasible routing policy μ_2 under information scheme 2 can be constructed as follows. At the original node j_0 and departure time t_0, partition the universal set of support points based on the two information schemes to obtain the initial event collection sets:

$\boldsymbol{EV}(Q_1(j_0,t_0))$ and $\boldsymbol{EV}(Q_2(j_0,t_0))$. For any $EV_2 \in \boldsymbol{EV}(Q_2(j_0,t_0))$, according to Lemma 1 there must exists $EV_1 \in \boldsymbol{EV}(Q_1(j_0,t_0))$, such that $EV_2 \subseteq EV_1$. We can then set $\mu_2(j_0,t_0,EV_2)=\mu_1(j_0,t_0,EV_1)$. As μ_1 and μ_2 give exactly the same next node under any support point, they produce the same trajectory under any support point at the next decision node. Let the arrival at the next node j occur at time t, then the information coverage Q_1 is a subset of Q_2 from the same trajectory $\{(j_0, t_0), (j, t)\}$. By Lemma 1, $\boldsymbol{EV}(Q_1)$ contains $\boldsymbol{EV}(Q_2)$, therefore we can set $\mu_2(j_0,t_0,EV'_2)=\mu_1(j_0,t_0,EV'_1)$, $\forall EV'_2 \in \boldsymbol{EV}(Q_2)$, $EV'_2 \subseteq EV'_1$. The process continues and a routing policy μ_2 is constructed with exactly the same trajectory as μ_1 under any support point, and thus the same objective function value. The optimal objective function value under scheme 2 is at least as good as that from the feasible solution μ_2 by definition, and thus at least as good as the optimal objective function value under scheme 1, namely, $e_2^*(j_0,t_0) \le e_{\mu_2}(j_0,t_0) = e_{\mu_1}(j_0,t_0) = e_1^*(j_0,t_0)$. **Q.E.D.**

The intuition behind Theorem 0.1 is that with larger information coverage throughout the trip, one has more flexibility in every decision node based on a finer partition of the possible outcomes (support points). For example, instead of having to choose a next node based on whether the highway is congested, now one can make the decision based on whether both the highway and arterial are congested. One can always ignore the additional information on arterial and act as if only information on the highway was available, and this ensures that optimal actions under larger information coverage is at least as good.

Theorem 0.1 also applies when only a subset of the universal set of support points is used to evaluate routing policies. The proof is the same with the universal set replaced by the subset.

The theorem can be alternatively stated as follows: more error-free information is always better (or at least not worse) for adaptive routing in a flow-independent network. It is consistent with Marschak and Miyasawa (1968)'s Theorem 11.3 regarding noiseless information systems: if two information systems are noiseless and one is finer than (in this chapter's terminology, contained by) the other, then it is also more informative in the sense that "it can never have smaller value than the other for any payoff function defined on a given set of events". The decision problem in Marschak and Miyasawa (1968) is however single-staged, and Theorem 0.1 extends the result to a multi-staged routing decision situation in a network context.

4.4 Solutions to the Partial and No Online Information Cases

Theorem 0.1 provides a theoretical comparison between two information schemes, however it is applicable only when one coverage is larger or no smaller in both spatial and temporal dimensions. In reality an information scheme can have larger coverage in one dimension but smaller coverage in the other. In order to evaluate the value of traveler information empirically for more complicated situations, computer algorithms to solve the optimal routing policy problem with partial and no online information are needed.

Since a routing policy has a random travel time, there exist multiple optimization criteria. The expected travel time is used in the remainder of the chapter, as generally it is the primary criterion in routing choices. Other criteria regarding travel reliability, such as expected schedule delay and travel time variance will be explored in future research, yet some criteria are harder to deal with than others.

In this section, it is shown that Bellman's principle of optimality does not hold for the three partial or no online information problems. A heuristic algorithm is then designed and computationally evaluated.

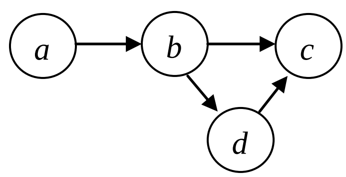
In all the studied problems, information coverage Q is determined by the current time, instead of the whole trajectory, therefore $\boldsymbol{EV}(t)$ is used instead of $\boldsymbol{EV}(Q)$. Time lag Δ in delayed information, departure time t_0 in pre-trip information and radio coverage B in radio information are treated as exogenous system parameters. In pre-trip information with departure time t_0, $\boldsymbol{EV}(t) = \boldsymbol{EV}(t_0)$, $\forall\ t \geq t_0$.

Except for delayed information, in all other four cases no more useful information is available during static period, i.e., Q does not grow beyond K–1, because either no information is provided (pre-trip and no online information), or additional information will not enlarge Q (radio and perfect online information). In the case of delayed information, a traveler continues receiving information in the static period until K-1+Δ, at which time Q=A×T. Let T* denote the time beyond which a traveler receives no more useful information and Q remains unchanged. We then have $T^*=K-1+\Delta$ for delayed information, and $T^*=K-1$ for all other four cases.

4.4.1 Bellman's Principle of Optimality

Proposition 0.1. Bellman's principle of optimality does not hold for the delayed, pre-trip, radio or no online information case. In other words, if μ^* is optimal for a given initial event collection EV_0 at (j_0,t_0), and (j,t,EV) is an intermediate state during the execution of μ^*, then the remainder of μ^* is not necessarily optimal when EV is an initial event collection at (j,t).

Proof. This can be shown through an example in Figure 0.3. Note that only relevant link travel times are shown. The travel time on link (d, c) is always 0 and not listed. No online information is assumed, such that the routing decision only depends on the arrival time at each decision node, i.e, $EV = \{C^1, C^2\}$ at any node and time. The problem is to find an optimal routing policy from node a to c for departure time 0.



Time	Link	C^1	C^2
0	(a, b)	1	2
1	(b, c)	1	10
	(b, d)	3	3
2	(b, c)	10	1
	(b, d)	3	3

$p_1 = p_2 = 1/2$

Figure 0.3 An Illustrative Small Network

Link (a, b) has two possible travel times at time 0: 1 and 2, therefore the arrival time at node b can be either 1 or 2. As there are two alternatives to go from node b to c at each of the two possible arrival times, altogether there are four routing policies, listed in Table 4 along with the corresponding expected travel times.

Table 0.1 Routing policies from node *a* at time 0

	At node *a*	At node *b*		Expected travel time
		Arrival time 1	Arrival time 2	
Routing policy 1	Node *b*	Node *c*	Node *c*	2.5
Routing policy 2	Node *b*	Node *c*	Node *d*	3.5
Routing policy 3	Node *b*	Node *d*	Node *c*	3.5
Routing policy 4	Node *b*	Node *d*	Node *d*	4.5

The optimal routing policy from node *a* to *c* at departure time 0 is therefore *a*-*b*-*c* (actually a path). However, the optimal routing policy from node *b* to *c* at either departure time 1 or 2 is not the policy *b*-*c* with mean travel time 0.5(1+10), but *b*-*d*-*c* with mean travel time 3.

The key here is the treatment of the possibly large travel time on link (*b*, *c*). The travel time of 10 on link (*b*, *c*) can never be realized if the traveler leaves node *a* at time 0, due to the stochastic dependency between link (*a*, *b*) and (*b*, *c*). However if *b* is the origin, then the travel time of 10 is possible and should be taken into account. If link travel times are time-wise and link-wise independent, Bellman's optimality principle will hold and the no online information problem reduces to the ones studied by Miller-Hooks and Mahmassani (2000), Chabini (2000) and Miller-Hooks (2001).

Examples for the three partial online information cases can be constructed similarly. If *j* is an origin with *EV*, the calculation of expected travel time from *j* is not conditional on the past and thus includes all support points in *EV*. However, if *j* is an intermediate node, the calculation must be conditional on the traversed link travel times from the origin to the current node, which are not necessarily covered by the online information. Since link travel times are stochastically dependent, the conditional expected travel time might be different from the unconditional one. Examples can be constructed so that this discrepancy will lead to different optimal policies based on whether the node is an origin. Details of these examples are not presented due to space limit. **Q.E.D.**

Bellman's principle of optimality is valid for the perfect online information case (stated formally later by combining Proposition 0.2 and Proposition **0.3**). Note that in this case the online information covers everything that happened in the past, including the traversed link travel times to any intermediate node. Therefore the expected travel time with perfect online information does not depend on whether the node is an origin.

4.4.2 Necessary Conditions for Optimality

Proposition 0.1 indicates that we cannot generate an optimal routing policy by compositing the optimal next node and the optimal policy from the next node. We then present the necessary conditions for the optimal solutions in Proposition 0.2. Any feasible solution to the optimal routing policy problem provides an upper bound on the minimal expected travel time, yet one that satisfies the necessary conditions for optimality conceivably provides a tighter upper bound than an arbitrary solution. Therefore a heuristic algorithm is proposed to solve for the necessary conditions, and its effectiveness in terms of closeness to optimal solutions evaluated computationally. The heuristic is a generalization of the algorithm for the perfect online information problem in Gao and Chabini (2006), with a distinction in the major recursive equation.

Let $e_\mu(j,t,EV)$ be the expected travel time to the destination node d by following routing policy μ, if the departure from origin node j happens at time t with the event collection EV. $S_\mu(j,t,r)$ is the travel time to the destination node d if support point r is realized with a departure from node *j* (origin or intermediate) at time *t* by following routing policy μ. The relationship between $e_\mu(j,t,EV)$ and $S_\mu(j,t,r)$ is as follows:

$$e_\mu(j,t,EV) = \sum_{r \in EV} S_\mu(j,t,r)\Pr(r \mid EV) \tag{0.1}$$

where Pr(A) is the probability of event A. Note that the algorithm in Gao and Chabini (2006) for perfect online information deals with $e_\mu(j,t,EV)$ only, while $S_\mu(j,t,r)$ is needed for partial and no online information cases to correctly calculate expected travel times.

A routing policy is defined based on event collections, not support points, where an event collection includes a number of support points compatible with revealed information at the decision node and time. Conceivably an event collection is equivalent to a support point if the traveler is omnipotent and knows exactly what will happen in each day at the beginning of the day. Generally this is impossible and one has to deal with a set of possible support points, although the set size will likely decrease over time during the trip. For each support point (at the end of a day), a routing policy is manifested as a path with a deterministic travel time. For a given time *t* and support point *r*, there is one and only one corresponding event collection $EV(t,r)$, since $\boldsymbol{EV}(t)$ is a partition of the universal set of support points. This ensures that the next node of routing policy μ at (j,t,r) can be uniquely retrieved as $\mu(j,t,EV(t,r))$, and $S_\mu(j,t,r)$ can be obtained by executing μ in support point *r*. In the example of Figure 0.1 A Small Network, for a traveler with radio information on (a,b), the routing decision at node *a* and time 0 can only be made based on the event collection $\{C^1,C^2,C^3\}$. Let $\mu\{a,0,\{C^1,C^2,C^3\}\}=c$. The travel time by following routing policy μ starting from node *a* at time 0 is a random variable with possible different outcomes in different support points: $S_\mu(a,0,C^1)=3$, $S_\mu(a,0,C^2)=3$, and $S_\mu(a,0,C^3)=2$.

The recursive relationship between S_μ at node j and the succeeding node k by following μ is critical to solving the optimal routing policy problem. $S_\mu(j,t,r)$ is defined for a trip leaving node j at time t. For all the information schemes except for pre-trip, the information coverage is not a function of departure time, and thus event collections at time t and node j are the same no matter whether j is an origin or intermediate node. In this case,

$$S_\mu(j,t,r) = C^r_{jk,t} + S_\mu(k, t + C^r_{jk,t}, r), \text{ where } k=\mu(j,t,EV(t,r)). \tag{0.2}$$

With perfect online information, the travel time on the next link (j,k) at time t, $C^r_{jk,t}$ is the same for all support points in a given EV (denoted as $\pi^{EV}_{jk,t}$), and thus taking an expectation of both sides of (2) over EV gives the following:

$$\begin{aligned}
e_\mu(j,t,EV) &= \sum_{r \in EV} S_\mu(j,t,r)\Pr(r \mid EV) \\
&= \sum_{r \in EV} \left(\pi^{EV}_{jk,t} + S_\mu(k, t + \pi^{EV}_{jk,t}, r)\right)\Pr(r \mid EV) \\
&= \pi^{EV}_{jk,t} + \sum_{EV' \in \mathbf{EV}(t+\pi^{EV}_{jk,t})} \sum_{r \in EV'} S_\mu(k, t + \pi^{EV}_{jk,t}, r)\Pr(r \mid EV')\Pr(EV' \mid EV) \\
&= \pi^{EV}_{jk,t} + \sum_{EV' \in \mathbf{EV}(t+\pi^{EV}_{jk,t})} e_\mu(k, t + \pi^{EV}_{jk,t}, EV')\Pr(EV' \mid EV)
\end{aligned} \tag{0.3}$$

where $k=\mu(j,t,EV)$. In the third equality, support points at a later time $t + \pi^{EV}_{jk,t}$ is re-partitioned into finer event collections EV'. In the fourth equality, support point travel times in each EV' are summarized as the expected travel time.

Such a relationship between expected travel times at adjacent nodes generally does not exist for partial or no online information, since the derivation in Eq. (4.3) depends on the fact that the travel time on the next link given the current EV is fixed.

For the pre-trip information, the information coverage depends on the departure time, and thus there is an ambiguity as to which event collection r belongs to at a given time t. A different variable $S_{\mu}(j,t,r;t_0)$ can be defined as the travel time from node j and time t to the destination node if support point r is realized by following routing policy μ, with a departure time t_0. Similarly $e_{\mu}(j,t,EV;t_0)$ and $\mu(j,t,EV;t_0)$ can be defined. In this case,

$$S_{\mu}(j,t,r;t_0) = C^r_{jk,t} + S_{\mu}(k,t+C^r_{jk,t},r;t_0), \text{ where } k=\mu(j,t,EV(t,r);t_0);$$

$$e_{\mu}(j,t,EV;t_0) = \sum_{r\in EV} S_{\mu}(j,t,r;t_0)\Pr(r\,|\,EV)$$

We propose the following system of recursive equations to solve for the perfect online, delayed, radio and no online information problems based on the recursive equation in Eq. (4.2).

$$e_{\mu^*}(j,t,EV) = \min_{k\in A(j)} \left\{ \sum_{r\in EV} (C^r_{jk,t} + S_{\mu^*}(k,t+C^r_{jk,t},r))\Pr(r\,|\,EV) \right\} \tag{0.4}$$

$$\mu^*(j,t,EV) = \arg\min_{k\in A(j)} \left\{ \sum_{r\in EV} (C^r_{jk,t} + S_{\mu^*}(k,t+C^r_{jk,t},r))\Pr(r\,|\,EV) \right\} \tag{0.5}$$

$$\forall j\in N\backslash\{d\},\ \forall t,\ \forall EV\in \boldsymbol{EV}(t)$$

where $A(j)$ the set of downstream nodes out of node j. The boundary conditions are:

1) At the destination: $S_{\mu^*}(d,t,r)=0$, $\mu^*(d,t,EV)=d$, $\forall t$, $\forall EV\in \boldsymbol{EV}(t)$, $\forall r\in EV$.

2) Beyond T^*: $\mu^*(j,t\geq T^*,EV)=\mu^*(j,T^*,EV)$, $\forall j$, $\forall EV\in \boldsymbol{EV}(T^*)$, $T^*=K\text{-}1+\Delta$ for delayed information, and $T^*= K–1$ for other three cases (radio, perfect and no online information).

Note that, $S_{\mu^*}(j,t,r) = C^r_{jk^*,t} + S_{\mu^*}(k^*, t + C^r_{jk^*,t}, r)$, where $k^*=\mu^*(j,t,EV(j,t))$. $S_{\mu^*}(d,t,r)$ is the travel time of the solution routing policy μ^* in support point r, not the minimum travel time calculated using a deterministic shortest path algorithm in support point r. $S_{\mu^*}(d,t,r)$ is obtained by executing μ^* after μ^* is generated.

For the pre-trip problem, a similar system of equations can be solved to obtain a solution from all nodes and all possible event collections, but with departure time t_0 only.

Proposition 0.2. Conditions in Eq. (4.4) and (4.5) are necessary for μ^* to be an optimal routing policy for all possible initial states for the perfect online, delayed, radio and no online information problems.

Proof. Trivially, if the boundary conditions at the destination node are not satisfied, μ^* is not optimal.

At time period T^* and beyond, information coverage includes all links at all time periods. Therefore there are R event collections, each with one support point. The optimal routing policy beyond T^* is not a function of time t, as travel times and event collections do not change over time. $\mu^*(j,t\geq T^*,EV)=\mu^*(j,T^*,EV)$, $\forall j$, $\forall EV\in \boldsymbol{EV}(T^*)$. Conditions in Eq. (4.4) and (4.5) become

$$e_{\mu^*}(j,T^*,\{r\}) = \min_{k\in A(j)} \{C^r_{jk,T^*} + e_{\mu^*}(k,T^*,\{r\})\} \tag{0.6}$$

$$\mu^*(j,T^*,\{r\}) = \arg\min_{k\in A(j)} \{C^r_{jk,T^*} + e_{\mu^*}(k,T^*,\{r\})\} \tag{0.7}$$

$$\forall j\in N\backslash\{d\},\ \forall r$$

plus boundary conditions. These are the optimality conditions of a static shortest path problem in a deterministic network where link travel times are C^r_{jk,T^*}, $\forall(j,k)$. If μ^* is optimal, it must manifest as a shortest path in each deterministic network defined by a support point beyond T^*, and thus Eq. (4.6) and (4.7) must be satisfied.

Assume by contradiction that Eq. (4.4) and (4.5) are not satisfied for some state with a departure time earlier than T^*. Let (j,t,EV) be such a state. Therefore there must exist an outgoing node $k \in A(j)$, such that

$$\sum_{r \in EV} (C^r_{jk,t} + S_{\mu^*}(k, t + C^r_{jk,t}, r)) \Pr(r \mid EV) < \sum_{r \in EV} (C^r_{jk^*,t} + S_{\mu^*}(k^*, t + C^r_{jk^*,t}, r)) \Pr(r \mid EV)$$

A different routing policy μ can be constructed such that $\mu(j,t,EV)=k$, and $\mu=\mu^*$ for all other states. Then the following is obtained:

$$\begin{aligned} e_\mu(j,t,EV) &= \sum_{r \in EV} S_\mu(j,t,r) \Pr(r \mid EV) = \sum_{r \in EV} (C^r_{jk,t} + S_\mu(k, t + C^r_{jk,t}, r)) \Pr(r \mid EV) \\ &= \sum_{r \in EV} (C^r_{jk,t} + S_{\mu^*}(k, t + C^r_{jk,t}, r)) \Pr(r \mid EV) \\ &< \sum_{r \in EV} (C^r_{jk^*,t} + S_{\mu^*}(k^*, t + C^r_{jk^*,t}, r)) \Pr(r \mid EV) = e_{\mu^*}(j,t,EV) \end{aligned}$$

The third equality is due to the fact that μ and μ^* are the same at all times later than t. The equation contradicts with the fact that μ^* is optimal, therefore Eq. (4.4) and (4.5) must be satisfied for $t < T^*$. **Q.E.D.**

Proposition 0.3. Conditions in Eq. (4.4) and (4.5) are sufficient for μ^* to be an optimal routing policy for all possible initial states in the perfect online information problem, and equivalent to the optimality conditions in Gao and Chabini (2006).

Proof. With perfect online information, $C^r_{jk,t}$ is the same for all support points in a given EV, and thus taking expectations of both sides of Eq. (4.4) over EV and changing Eq. (4.5) accordingly gives the optimality conditions in Gao and Chabini (2006), similar to the derivation in Eq. (4.3). The sufficiency of Eq. (4.4) and (4.5) then follows from the optimality of the conditions in Gao and Chabini (2006). **Q.E.D.**

4.4.3 Algorithm DOT-PART

In this section we design a heuristic algorithm to solve the system of equations (4)(5). The evaluation of $e_{\mu^*}(j,t,EV)$ only depends on $S_{\mu^*}(j,t',r)$ from a later time $t'>t$, due to the positive and integral link travel time assumption. Therefore the labels can be set in a decreasing order of time, making use of the acyclic property of the network along the time dimension (Chabini, 1998). At time T^* and beyond, any deterministic static shortest path algorithm can be used to compute $e_{\mu^*}(j,t,EV)$, $\forall j \in N$, $\forall t \geq T^*$, $\forall EV \in EV(T^*)$. The procedure to generate event collections carry out partitions of the universal set of support points in an increasing order of time. At time t, a partition is made on $\boldsymbol{EV}(t\text{-}1)$ based on each (link, time) pair in the incremental information coverage, $Q(t)\backslash Q(t\text{-}1)$. Note that Q is written as a function of t, because in all the five cases, Q only depends on t, not the trajectory.

Generate_Event_Collection

$D = \{C^1, \ldots, C^R\}$
If information scheme = no online, $\boldsymbol{EV}(t) \leftarrow D$, t = 0 to K-1, **STOP.**
For t = 0 to T^*
    If information scheme = perfect online, $Q(t) = A \times \{0,1,\ldots,t\}$
    If information scheme = delayed, $Q(t) = A \times \{0,1,\ldots,t - \Delta\}$
    If information scheme = pre-trip, $Q(t) = A \times \{0\}$
    If information scheme = radio, $Q(t) = B \times \{0,1,\ldots,t\}$
$Q(-1) = \varnothing$ //a proxy for convenience of representation
For t = 0 to T^*
    For each (link, time) pair $((j,k),t') \in Q(t) \setminus Q(t\text{-}1)$
        For each disjoint subset $S \in D$
            $D' \leftarrow$ A partition of S based on $\tilde{C}_{jk,t'}$
        $D \leftarrow$ Union of all D'
    $\boldsymbol{EV}(t) \leftarrow D$;

Algorithm DOT-PART

(Generic for perfect online, delayed, pre-trip, radio and no online information)

Initialization

Step 1:

If information scheme = delayed, $T^* = K - 1 + \Delta$; else $T^* = K - 1$.

Construct $\boldsymbol{EV}(t)$, $t=0,\ldots,T^*$ by calling Generate_Event_Collection.

Step 2:

Compute $e_{\mu^*}(j,T^*,EV)$ and $\mu^*(j,T^*,EV)$, $\forall j\in N$, $\forall EV\in \boldsymbol{EV}(T^*)$ with a static deterministic shortest path algorithm in a converted static deterministic network where link travel times are replaced by their means at time T^*.

Compute $S_{\mu^*}(j,T^*,r)$ by executing μ^* in the original static stochastic network, $\forall j \in N$, $\forall r\in EV$; set $S_{\mu^*}(j,t>T^*,r)=S_{\mu^*}(j,T^*,r)$

Step 3:

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

$e_{\mu^*}(d, t, EV) \leftarrow 0$, $S_{\mu^*}(d, t, r) \leftarrow 0$, $\forall t<T^*$, $\forall EV\in \boldsymbol{EV}(t)$, $\forall r\in EV$

Main Loop

For $t = T^*$-1 down to 0 and for each $EV\in \boldsymbol{EV}(t)$

For each link $(j, k)\in A$

$$temp = \sum_{r\in EV}(C^r_{jk,t} + S_{\mu^*}(k,t+C^r_{jk,t},r))\Pr(r \mid EV)$$

If $temp < e_{\mu^*}(j, t, EV)$ then

$e_{\mu^*}(j, t, EV) = temp$

$\mu^*(j, t, EV) = k$

For each $r\in EV$ and each $j\in N$

$k^* = \mu^*(j, t, EV)$

$$S_{\mu^*}(j,t,r) = C^r_{jk^*,t} + S_{\mu^*}(k^*,t+C^r_{jk^*,t},r)$$

According to Proposition 0.2 and Proposition **0.3**, Algorithm DOT-PART is exact for the perfect online information case. It generates approximate solutions with all initial states for delayed, radio and no online information, and with departure time 0 for pre-trip information. In order to solve pre-trip case with all departure times, a loop over all departure times t_0 has to be added outside the main loop, and the main loop will be executed from T^*-1 to t_0 (not shown in the algorithm statement).

Following a similar analysis as in Gao and Chabini (2006), Algorithm DOT-PART (including Generate_Event_Collection) has a time complexity of

$O(mKR\ln R+R\times\text{SSP})$ except for pre-trip information and $O(mK^2R\ln R+R\times\text{SSP})$ for pre-trip information, where SSP is the time complexity of the static deterministic shortest path algorithm. The algorithm is strongly polynomial in *R*, the number of support points. For real life applications, time-dependent travel time observations on all (random) links from each day can be viewed as one support point. Such data are available with the advent of advanced sensor and surveillance technologies, such as GPS and probe vehicles. The number of support points might seem exponential in the number of links, however, if we consider the high stochastic dependencies among link travel times and use observations from each day as a support point, we can safely have several years' data with the number of support points in the thousands, similar to the number of links in a medium-sized network and much less than its exponential.

Table 0.2 Relationship between CPU time (sec) and input variables in LAG variant

Running time of Generate_Event_Collection									
m	30			60			90		
K / *R*	600	1200	1800	600	1200	1800	600	1200	1800
50	0.23921	0.46110	0.66555	0.47334	0.92544	1.35263	0.70847	1.39290	2.00859
100	0.48257	0.91619	1.33765	0.95248	1.8222	2.70041	1.43496	2.75348	3.99936
300	1.41600	2.70133	3.95108	2.81024	5.35951	7.86103	4.20675	7.99032	11.7688
Running time of DOT-PART for LAG variant (excluding Generate_Event_Collection)									
m	30			60			90		
K / *R*	600	1200	1800	600	1200	1800	600	1200	1800
50	0.65276	1.16761	1.69362	1.45628	2.55513	3.68207	2.28277	4.04768	5.79555
100	1.43934	2.46914	3.51165	3.18501	5.38760	7.66759	4.99824	8.52984	12.0671
300	5.65247	8.85628	12.1153	12.4508	19.6269	27.1109	19.7828	31.1017	43.8705

A running time test is conducted with randomly generated networks on a Dell Optiplex with 2.40GHz Intel Core 2 CPU and 2.00GB of RAM. Details of the random network generator can be found in Gao, S. (2005). The number of nodes (*n*), the number of time periods (*K*), and the number of support points (*R*) are chosen as input variables;

the number of links (*m*) is three times as great as the number of nodes. Random numbers from multivariate normal distributions are generated for link travel times. The relationship between running time of the algorithm and the input variables for the LAG variant is shown in Table 0.2. It can be seen that the relationship between running time and each of the 3 input variables is close to linear. Similar tests are conducted for other variants and the relationships are similar.

4.4.4 Computational Tests

The objectives of the computational tests are to 1) systematically investigate the effectiveness of the heuristic, Algorithm DOT-PART in generating optimal solutions to the partial and no online information problems; and 2) study the (approximate) value of information empirically as a complement to the theoretical study in Section 4.3 .

Algorithm DOT-PART provides upper bounds of the minimal expected travel times in partial and no online information cases since it generates (conceivably good) feasible solutions. The upper bound however can be arbitrarily loose by constructing an example similar to that in Proposition 0.1. We are more interested in its effectiveness on average through a systematic test over a large number of instances. We do not have an exact solution algorithm to the partial or no online information cases. However, Theorem 0.1 states that the optimal solution under perfect online information scheme is at least as good as the optimal solution under any partial or no online information scheme, since the former coverage is larger with any given trajectory. Therefore the optimal solution with perfect online information, which can be computed exactly by Algorithm DOT-PART, provides a lower bound of the optimal solution with any partial or no online information. The error of the heuristic, which is difference between the unknown exact solution to a partial or no online information case and the heuristic solution, is then bounded above by the difference between the perfect online information solution and the heuristic solution. Furthermore, we can also view the same difference as an upper bound on the value of perfect information compared to partial or no online information. A schematic view of these relationships for any given partial or no online information case is shown in Figure 0.4.

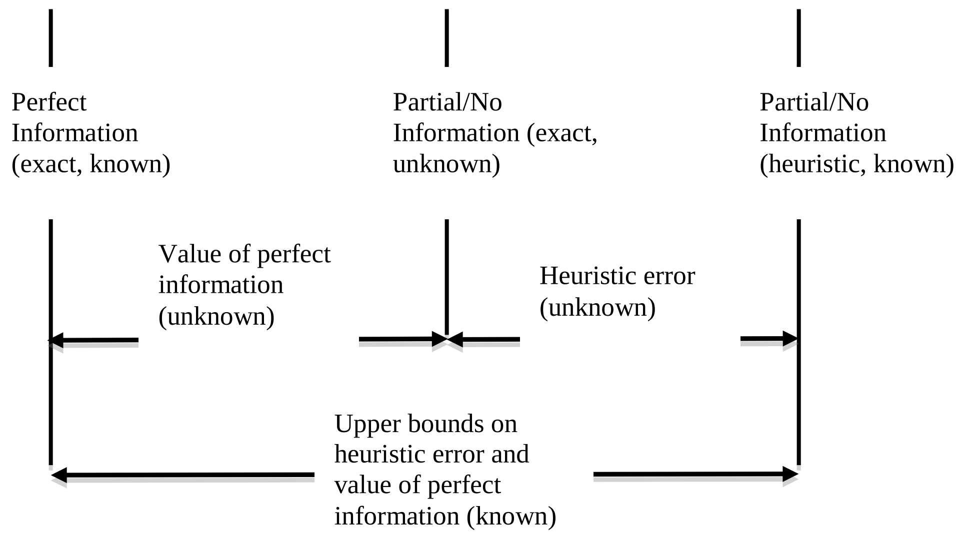


Figure 0.4 Relationships between Heuristic and Exact Solutions

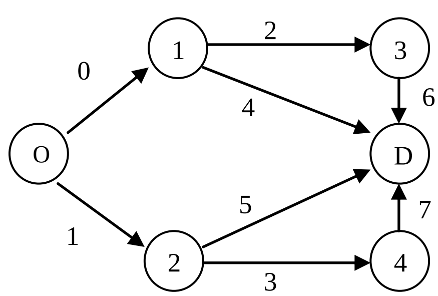


Figure 0.5 The Test Network

The first test network is shown in Fig. 5 with 6 nodes and 8 directed links. There are diversion possibilities at nodes O, 1 and 2. The study period is from 6:30am to 8:00am. The time resolution is 1 minute for departures and arrivals at intermediate nodes, and there are 90 time periods in total. The travel time is in seconds.

The link travel time distribution is generated through an exogenous simulation with the mesoscopic supply simulator of DynaMIT (Ben-Akiva et al., 2001). The demand between the origin and destination is low from 6:30am to 7:00am and higher later on. There are random incidents in the network that result in 37 support points. Details of the network can be found in Gao (2005).

Algorithm DOT-PART is run for the three partial online, no online and perfect online information cases to derive the (upper bounds of) minimum expected travel times for each of them from node O to D for all departure times and all event collections. The results are aggregated by departure time, by taking expectations over all event collections at a given time.

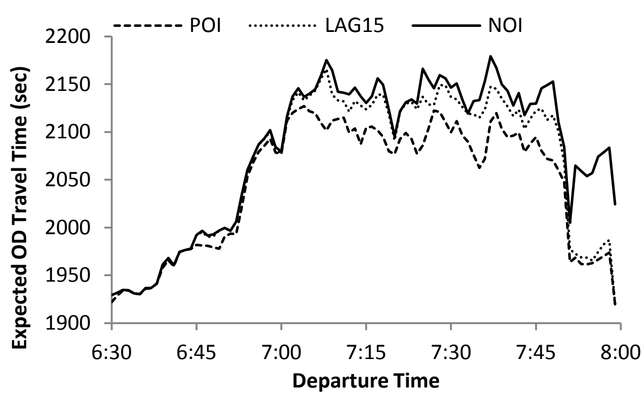


Figure 0.6 Results for the 15-min delayed (LAG15) vs. perfect (POI) and no online information (NOI)

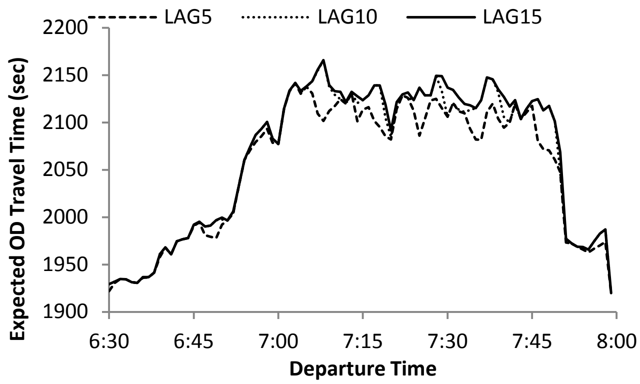


Figure 0.7 Results for delayed information with 5 (LAG5), 10 (LAG10) and 15-min lags

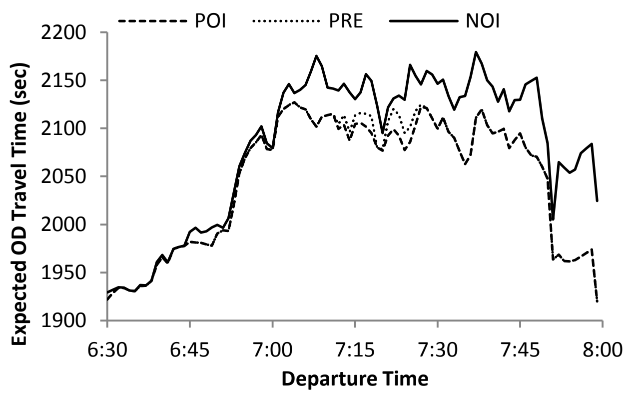


Figure 0.8 Results for pre-trip (PRE) vs. perfect and no online information

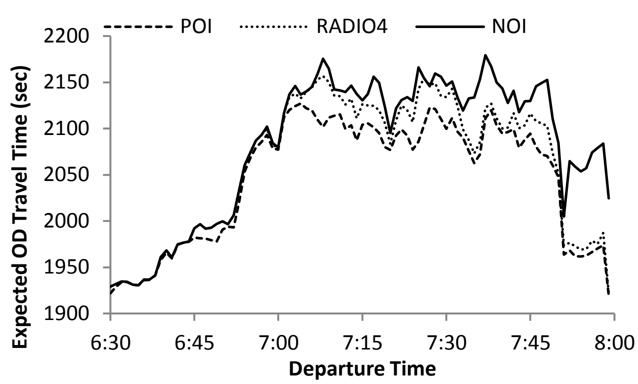


Figure 0.9 Results for radio on link 4 vs. perfect and no online information

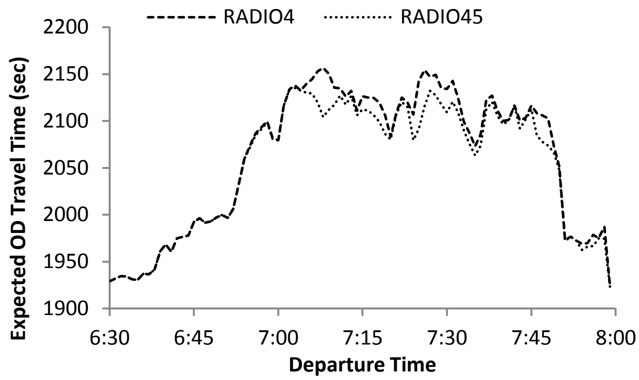


Figure 0.10 Results for radio information with different radio coverage

Figure 0.6 through Figure 0.10 show the expected OD travel times for the no online, 5-min delay, 10-min delay, 15-min delay, pre-trip, radio on link 4 and radio on links 4&5 cases. RADIO4 indicates that only traffic condition information on link 4 is available and RADIO45 on links 4 and 5. It is shown that the upper bounds generated by Algorithm DOT-PART are relatively tight: within 3% of the (unknown) exact solution. Also shown is that in the specific settings, global pre-trip information is nearly as good as perfect online information. Another interesting observation is that although the solutions to partial and no online information are not exact, they do exhibit the trend that "more error-free information is better in a flow-independent network". For example, the expected travel times with delayed information decreases when the delay decreases from 15 to 10 and from 10 to 5 minutes; and those with radio covering both links 4 and 5 are better than with radio covering only link 4. However this should not be viewed as a verification of Theorem 0.1.

Additional tests are conducted on larger randomly generated networks to investigate the effectiveness of the heuristics. The random network generator takes the following as input: 1) the number of nodes; 2) the number of links; and 3) the number of time periods. Four levels of the number of nodes are considered: 50, 100, 250, and 500. The number of links is always three times of the number of nodes, i.e., 150, 300, 750, and 1500. Three levels of the duration of the peak period are considered: 25, 50, and 100 time intervals. Other parameters include the number of support points fixed as 300, the range of link travel time fixed as [0, 10], and the maximum in-degree and out-degree fixed as 5. The topology of the network is randomly generated. The travel time on each link at each time interval for each support point is generated from a uniform distribution within the fixed range. More details on the random network generation can be found in Gao (2005).

Table 0.3 Upper bounds of heuristic errors (% difference from perfect online information)

Nodes (n)	***Links (m)***	***Time Periods (K)***	**No Online**	**Pre-trip**	**Delayed by 0.5*K***	**Delayed by 0.25*K***	**Radio on link 1**
50	150	25	40.3	0	14.9	6.1	2.2
50	150	50	26.6	0	11.2	4.2	0.5
50	150	100	22.3	0	10.5	4.9	0.3
100	300	25	13.8	0	5.3	2.3	0.9
100	300	50	24.4	0	10.5	4.1	0.6
100	300	100	26.0	0	12.8	6.1	0.4
250	750	25	31.4	0	12.0	5.1	1.8
250	750	50	33.9	0	14.3	5.6	0.8
250	750	100	27.0	0	12.4	5.6	0.3
500	1500	25	21.6	0	6.5	2.3	0.8
500	1500	50	26.5	0	11.4	4.5	0.7
500	1500	100	28.8	0	13.3	6.0	0.3
		Average	26.9	0	11.2	4.7	0.8

There are 12 different combinations of inputs, and 10 random networks are generated for each combination. Table 0.3 shows the upper bounds of heuristic errors, defined as the percentage difference of partial or no online information result from that of perfect online information. The errors are averaged over all departure times (except for pre-trip where only departure time 0 results are reported) and all origins to a single destination for each network, and then averaged over the 10 networks. The radio information covers only one link, randomly sampled 10 times for each of the 10 random networks. Thus in the radio column, the errors are averages over 100 runs.

Algorithm DOT-PART as a heuristic performs better than predicted by the theoretical worst case (arbitrarily large errors), with errors within 15% for partial online cases and 30% for most no online information. Note that these are upper bounds of errors, and the heuristic might perform better than these bounds. Future research is needed to design an exact algorithm and a more comprehensive evaluation of the effectiveness of the heuristic can then be carried out. It will also be interesting to investigate the effectiveness of the heuristic with real-world data, which is an important step towards its practical application.

We also see the same trend that "more error-free information is better in a flow-independent network". For example, information delayed for $0.25K$ unit time produces smaller expected travel time than information delayed for $0.5K$ unit time, which in turn is smaller than no online information. Pre-trip information is as good as perfect online information in all test scenarios, and radio information is almost as good. On the other hand, delayed information seems to perform not as well. This might suggest that up-to-date information is more valuable than information that covers a large area. However, again, since the solutions are not exact, these observations should be viewed with caution.

4.5 Conclusions and Future Directions

In this chapter, a generic representation of online information in a general stochastic network is developed, based on which three types of information schemes are specialized: delayed global information, global pre-trip information, and radio information on a subset of links without time lag. The scope limitations of an information system on both the temporal and spatial dimensions are taken into account. A theoretical proof of the non-negative value of error-free traveler information for adaptive routing in a flow-independent stochastic network is presented. It is shown that Bellman's principle of optimality does not apply to the optimal routing policy problem with partial or no online information. A heuristic algorithm is then designed based on a set of necessary conditions for optimality and its effectiveness is tested empirically and shown to be satisfactory.

Other interesting information schemes will be studied in the future, e.g., VMS, which is one of the most common types of ATIS. The problem with VMS is more involved than those discussed in this chapter, as the information is trajectory-based rather time-based only. This could significantly complicate the algorithm design. The noise level of the information will also be considered, such that the information is no longer error-free. Theoretical studies will be conducted to establish the conditions (if existing) under which noisy information systems are comparable.

Predictive information (Bovy and van der Zijpp, 1999; Bottom, 2000; and Dong et al., 2006) that provides estimates of future travel times is not explicitly studied under the online information framework in this chapter. Mathematically one can easily build an information scheme where the coverage $Q(t)$ contains realized travel times beyond t, and all the analyses and algorithm in this chapter apply. The more fundamental question is whether an analysis framework built upon error-free information assumption is good for predictive information. Although the error in measuring realized travel times can be reasonably assumed approaching zero with the ever-increasing accuracy of traffic surveillance, the same cannot be said for predictive information. Therefore the effort to model predictive information should be joined with that on noisy information as mentioned in the previous paragraph.

The interaction between demand and supply needs to be considered to assess the value of real-time information with a large market penetration of information. In a congested un-priced network, information could be detrimental, as shown in Gao (2005) and many other studies (e.g., Arnott et al., 1991, 1999, Levinson, 2003). The next step of the research would be studies of the value of various types of information systems in a congested network. An equilibrium dynamic traffic assignment model or a day-to-day dynamic process model is to be applied.

Another interesting direction would be a theoretical quantification of the value of traveler information as a function of an array of information system and network characteristics. This would enable the cross comparison of different types of information systems. For example, is up-to-date spatially-limited information better than delayed global information? Answers to this type of questions can be obtained computationally as shown in Section 4.4.4 , however a theoretical solution would provide valuable insights and guidelines for, e.g., optimal investment in ATIS.

CHAPTER 5

OPTIMAL *A PRIORI* PATHS IN STOCHASTIC NETWORKS

5.1 Introduction

In this chapter, we study the optimal *a priori* path problem in a stochastic and time-dependent network with complete dependencies, where all link travel times in all time periods are assumed to be correlated. The paths are evaluated by a disutility function of travel time, and the optimal paths are those with the minimum expected disutility. An exact label-correcting algorithm is designed to find the optimal paths, where the disutility function can be any increasing function of travel time, and thus the algorithm is applicable to a wide range of reliability requirements in path finding.

In CHAPTER 3, we work on a simple network, where there are only two paths between an OD pair, and investigate whether route (path) choice prediction will be biased if correlation is not taken into account and how sensitive route (path) shares are to the level of correlation and risk attitudes. In this chapter, in order to study the impact of link travel time correlations on the optimal path solution, a comparison is made with similar problems that do not consider stochastic dependencies through theoretical and computational analyses. The results show how the optimal path solution is affected by the level of correlations and the traveler's risk attitude.

This chapter is organized as follows. In Section 5.2 , the optimal path problem in an STD network is defined. A label-correcting algorithm is presented in Section 5.3 , and computational tests are conducted in Section 5.4 . A supplemental analytical solution is given in Section 5.5 to provide insights into the problem. In Section 5.6 , conclusions are made and future directions given.

5.2 Problem Statement

5.2.1 Optimal Path

This chapter addresses the problem of finding optimal paths from all origins and departure times to a single destination D. $S_\lambda(O, t, r)$ is defined as the travel time of path λ from origin node O and departure time t to the destination node D if support point r is realized. $e_\lambda(O, t)$ is the expected travel time of path λ from origin node O and departure time t to the destination node D, where the expectation is taken over all support points. Let $D_\lambda(O, t, r)$ denote the disutility of path λ from origin node O and departure time t to the destination node D in support point r, and $D(\cdot)$ is the disutility function, i.e., $D_\lambda(O, t, r) = D(S_\lambda(O, t, r))$. The disutility function $D(\cdot)$ can be linear or nonlinear, and is an increasing function of travel time. $d_\lambda(O, t)$ is the expected disutility where the expectation is taken over all support points.

The relationship between the support point travel time / disutility of a path and the expected travel time / disutility is given as follows:

$$e_\lambda(O,t) = \sum_{r=1}^{R} S_\lambda(O,t,r) \cdot p_r$$

$$d_\lambda(O,t) = \sum_{r=1}^{R} D_\lambda(O,t,r) \cdot p_r \tag{0.1}$$

The relationship between the support point travel times / disutilities of a path and of its sub-path is given as follows:

$$S_\lambda(O,t,r) = C_{Ok,t}^{r} + S_{\lambda'}\left(k, t + C_{Ok,t}^{r}, r\right)$$

$$D_\lambda(O,t,r) = D\left(C_{Ok,t}^{r} + S_{\lambda'}\left(k, t + C_{Ok,t}^{r}, r\right)\right) \tag{0.2}$$

where node k is the next node on path λ and the starting node of sub-path λ', and $t + C_{Ok,t}^{r}$ is the exit time out of node k in support point r.

The expected travel time / disutility is then re-written as follows:

$$e_\lambda(O,t) = \sum_{r=1}^{R}\left(C_{Ok,t}^{r} + S_{\lambda'}\left(k, t + C_{Ok,t}^{r}, r\right)\right)\cdot p_r$$

$$d_\lambda(O,t) = \sum_{r=1}^{R} D\left(C_{Ok,t}^{r} + S_{\lambda'}\left(k, t + C_{Ok,t}^{r}, r\right)\right)\cdot p_r \quad (0.3)$$

Note that this is different from how the expected travel time / disutility is calculated in an STD network where no stochastic dependencies are considered, where marginal distributions of link travel times are utilized, as shown below:

$$e_\lambda^{ND}(O,t) = \sum_{i=1}^{Q}\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right)\cdot p_i$$

$$d_\lambda^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right)\cdot p_i \quad (0.4)$$

where the superscript "ND" stands for "no dependency", Q is the number of support points for the marginal distribution of travel time on link (O, k) and p_i the corresponding marginal probability. Note that the equation for $e_\lambda^{ND}(O,t)$ is the same as the equation in Step 2 of Algorithm EV in Miller-Hooks and Mahmassani (2000).

If an exponential disutility function is used to represent risk aversion, i.e., $D_\lambda(O,t,r) = D\left(S_\lambda(O,t,r)\right)$ = exp($\alpha \cdot S_\lambda(O,t,r)$), the expected disutilities are given as follows:

$$d_\lambda(O,t) = \sum_{r=1}^{R} D\left(C_{Ok,t}^{r} + S_{\lambda'}\left(k, t + C_{Ok,t}^{r}, r\right)\right)\cdot p_r$$

$$d_\lambda^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right)\cdot p_i \quad (0.5)$$

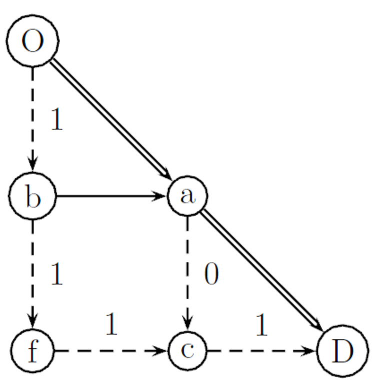
The parameter α in the exponential disutility function represents the level of risk aversion. When α is larger, the traveler is more risk-averse. When α is close to 0, the traveler is close to risk-neutral. Suppose a path has a random travel time of 10 or 20 minutes, each with probability 0.5. Table 0.1 shows the α value and the corresponding certainty equivalency value x such that a traveler who aims to minimize the expected exponential disutility is indifferent between (10, 0.5; 20, 0.5) and (x, 1.0). For a traveler with a larger α, the risky travel time is equivalent to a worse deterministic value, and thus he/she is less likely to take the risk.

Table 0.1 Traveler's Risk-Averse Attitude

A	0.01	0.1	0.2	0.5	1.0	1.5	2.0	3.0
X	15.1	16.2	17.2	18.6	19.3	19.5	19.7	19.8

In this chapter, we define the paths with minimum expected disutility (MED) as optimal paths, and the goal is to find the optimal paths from all origins to a given destination for all departure times. Note that, if the disutility is the travel time itself, we are seeking the paths with minimum expected travel time (METT).

Definition 0.1 (Path with MED for Departure Time *t*). A path λ with MED from origin O to destination D for departure time t has the minimum expected disutility evaluated over all support points among all the paths between the same OD pair and for the same departure time, i.e., $\nexists$ path μ such that $d_\mu(O,t) < d_\lambda(O,t)$.



Time	Link	C^1	C^2
0	(O,a)	1	2
	(a,D)	1	M
	(b,a)	1	M
1	(O,a)	1	2
	(a,D)	1	M
	(b,a)	1	1
2	(O,a)	1	1
	(a,D)	1	1
	(b,a)	1	1

Figure 0.1 The Illustrative Network

An illustrative network is shown in Figure 0.1 with 6 nodes and 8 links. The travel time on link (a, c) is always 0, and that on any of the other 4 dashed links is 1. Link travel times on solid links are stochastic and time-dependent. There are 2 time periods in the dynamic domain, in which the link travel time random variables are time-dependent (t = 0 and 1). There are 2 support points, each with a probability of 1/2, for the joint distribution of 6 travel time random variables on links (O, a), (a, D) and (b, a) over time periods 0 and 1. Travel times at and beyond time 2 are 1 for the 3 links in both support points (static and deterministic). M in the table is a large positive number. For the sake of simplicity, we assume the disutility function is the travel time itself, i.e., $D_\lambda(O,t,r) = S_\lambda(O,t,r)$, so we are working on an METT path problem. There are 5 paths from origin O to destination D, listed as follows:

$$\lambda_1: O \to a \to D$$

$$\lambda_2: O \to a \to c \to D$$

$$\lambda_3: O \to b \to a \to D$$

$$\lambda_4: O \to b \to a \to c \to D$$

$$\lambda_5: O \to b \to f \to c \to D$$

$D_\lambda(O,t,r)$ and $d_\lambda(O,t)$ ($S_\lambda(O,t,r)$ and $e_\lambda(O,t)$ in this case) for each path are calculated in Table 0.2 and the columns under "complete dependency" of Table 0.3, respectively. It can be observed that path $\lambda_1: O \to a \to D$ and path $\lambda_2: O \to a \to c \to D$ are optimal for all departure times.

Table 0.2 Path Support Point Travel Time

Path	$C^1, t = 0$	$C^2, t = 0$	$C^1, t = 1$	$C^2, t = 1$	$C^1, t \geq 2$	$C^2, t \geq 2$
$\boldsymbol{\lambda_1}$	2	3	2	3	2	2
$\boldsymbol{\lambda_2}$	2	3	2	3	2	2
$\boldsymbol{\lambda_3}$	3	3	3	3	3	3
$\boldsymbol{\lambda_4}$	3	3	3	3	3	3
$\boldsymbol{\lambda_5}$	4	4	4	4	4	4

Table 0.3 Path Expected Travel Time

	Complete Dependencies			No Dependencies		
Path	$t = 0$	$t = 1$	$t \geq 2$	$t = 0$	$t = 1$	$t \geq 2$
$\boldsymbol{\lambda_1}$	2.5	2.5	2	2.25+M/4	2.5	2
$\boldsymbol{\lambda_2}$	2.5	2.5	2	2.5	2.5	2
$\boldsymbol{\lambda_3}$	3	3	3	3	3	3
$\boldsymbol{\lambda_4}$	3	3	3	3	3	3
$\boldsymbol{\lambda_5}$	4	4	4	4	4	4

In general, if link travel time stochastic dependencies are ignored, some link travel times that are impossible to be realized under certain situations will be incorrectly taken into account when calculating expected travel times / disutilities, and this might affect the optimal solution. The columns under "no dependency" of Table 0.3 show the expected travel time for each path in the same network with the assumption that link travel time stochastic dependencies are ignored. In this case, each link retains the travel time marginal distribution as described in Figure 0.1, however no joint support point exists anymore and link travel times are assumed independent. For example, in the complete dependency case, if link (*O*, *a*) travel time is 1 at time 0, then link (*a*, *D*) at time 1 can only have a travel time of 1. However in the no dependency case, travel time on (*a*, *D*) at time 1 is assumed to always take its marginal distribution regardless of travel time realizations on other links, and thus can be either 1 or M. This results in a different expected travel time for path $\lambda_1 : O \to a \to D$ as shown in the right half of Table 0.3.

5.2.2 Pure Path

In this section, we first show that Bellman's principle of optimality (Bellman, 1958) that any sub-path of an optimal path must also be an optimal sub-path is no longer valid in our problem context (**Proposition 0.1**). We then show that Bellman's principle of non-dominance that any sub-path of a non-dominated path must also be a non-dominated sub-path is not valid either (**Proposition 0.2**), even though it is valid in problems studied by, e.g., Miller-Hooks and Mahmassani (2000), Opasanon and Miller-Hooks (2006), Miller-Hooks (1997), and Nie and Wu (2009b). We further define a subset of the non-dominated paths as pure paths, and purity is a property that can be maintained across path and sub-path. It is then proved (**Theorem 0.1**) that for any origin node, there always exists a pure optimal path, and an exact algorithm can be designed based on this property.

Proposition 0.1. A sub-path of a path with MED for a departure time is not necessarily with MED for every possible exit time out of the intermediate node (i.e., the starting node of the sub-path).

Proof. We prove this proposition by an example, and the general idea is given as follows. The path with MED for a departure time has the minimum expectation of disutility evaluated over all support points. However, the sub-path is not necessarily with MED over all support points for every possible exit time out of the intermediate node. It might have a large disutility in some support points which are impossible to be realized for some exit times out of the intermediate node due to the stochastic dependencies of link travel times, and this large disutility is accounted for when calculating its expected disutility over all support points.

In the illustrative network of Figure 0.1, assuming a simple disutility function of the travel time itself, we can determine that path $\lambda_1: O \rightarrow a \rightarrow D$ is optimal for departure time t = 0. However, the sub-path $a \rightarrow D$ is not optimal for exit time t_1 = 1, since $S_{a\rightarrow D}(a, 1, C^2) = C^2_{aD,1}$ = M and $d_{a\rightarrow D}(a, 1) = e_{a\rightarrow D}(a, 1)$ = (1+M)/2 , which is larger than the expected disutility of path $a \rightarrow c \rightarrow D$ that is a fixed value of 1. Note that, for exit time t_1

= 1, support point C^2 is impossible to be realized if the traveler comes from node O and time 0, i.e., the large travel time M should not be considered in the calculation of the expected travel time from origin O to destination D for departure time 0. **Q.E.D.**

Since Bellman's principle of optimality is not valid, we next define non-dominated path and see whether Bellman's principle of non-dominance will hold. Before defining a non-dominated path, we introduce the complete time-support-point set Ω as the Cartesian product of the sets of time periods T and support points C, that is, $\Omega = \{(t, r) \mid t \in T, r \in C\}$. Non-dominance is then defined over (a subset of) the universal set Ω.

Definition 0.2 (Non-Dominated Path). A path λ from origin O to destination D is non-dominated w.r.t. a subset Ω' of Ω iff $\nexists$ path μ between the same OD pair such that

$D_\mu(O,t,r) \le D_\lambda(O,t,r), \forall (t,r) \in \Omega'$ and

$\exists (t^0,r^0) \in \Omega'$ such that $D_\mu(O,t^0,r^0) < D_\lambda(O,t^0,r^0)$.

If not specified, in the remainder of this chapter, non-dominance is w.r.t. the complete set of departure time and support points Ω.

For the example of Figure 0.1, it can be determined from Table 0.2 that path $\lambda_1: O \to a \to D$ and path $\lambda_2: O \to a \to c \to D$ are non-dominated, as for every support point and departure time pair, they have the minimum support point travel time. Note that this is a special case, where non-dominated paths have the same (minimum) support point travel times for all support point and departure time pairs.

A more general example can be obtained when we check the non-dominated paths from node b to the destination node D. There are three paths between them $\mu_1: b \to a \to D$, $\mu_2: b \to a \to c \to D$, and $\mu_3: b \to f \to c \to D$. Table 0.4 shows the support point travel times for the three paths and it can be determined that all three paths are non-dominated.

Table 0.4 Path Support Point Travel Time between b and D

Path	$C^1, t = 0$	$C^2, t = 0$	$C^1, t = 1$	$C^2, t = 1$	$C^1, t \geq 2$	$C^2, t \geq 2$
$\boldsymbol{\mu_1}$	2	M+1	2	2	2	2
$\boldsymbol{\mu_2}$	2	M+1	2	2	2	2
$\boldsymbol{\mu_3}$	3	3	3	3	3	3

Note that, since the disutility function is increasing in travel time and joint distribution is utilized as complete dependencies are considered, non-dominance in terms of distuility is equivalent to non-dominance in terms of travel time. Thus, the $D_\lambda(O, t, r)$ terms in **Definition 0.2** can be changed to $S_\lambda(O, t, r)$ terms.

Also note that, in an STD network with stochastic dependencies among link travel times, the non-dominance over support point is required in order to take the dependencies into account. In Miller-Hooks and Mahmassani (2000) and Nie and Wu (2009b), the dominance is defined only over time, as they do not consider network stochastic dependencies. In the complete dependency case, the travel time on the next link of a path and that on the sub-path are dependent not only through the time-dependency of travel times from the next node, but also through stochastic dependencies. It follows that if only expected travel times are used in defining non-dominance, generating non-dominated paths from non-dominated sub-paths could result in the wrong non-dominance set. A similar treatment can be found in Nie and Wu (2009b) where local stochastic dependencies are considered and non-dominance is defined over the states of the outgoing links.

However, even with the non-dominance defined over both time and support point, Bellman's principle still does not apply, as stated formally in the following proposition.

Proposition 0.2. A sub-path of a non-dominated path w.r.t. the complete set of departure time and support points Ω is not necessarily non-dominated w.r.t. Ω.

Proof. We prove this proposition by an example and the general idea is given as follows. The non-dominated path is non-dominated w.r.t. the complete set of departure time and support points Ω. However, a sub-path might have an equal disutility as another path for a subset Ω', which is relevant in the composition of the path travel time from the sub-path, but is dominated by that path in other time periods and support points which are irrelevant in the composition. As a result, the sub-path is dominated w.r.t. the complete set Ω.

In Figure 0.1, the sub-path $a \rightarrow D$ of the non-dominated path $\lambda_1: O \rightarrow a \rightarrow D$ has the same travel time as $a \rightarrow c \rightarrow D$ in support point C^1 for all exit times, but has travel time M for exit time 0 and 1 in support point C^2, and so is dominated by $a \rightarrow c \rightarrow D$ whose travel time is always 1. Note that this large travel time M cannot be realized if the traveler comes from node O, i.e., it is not considered in the calculation of the travel time from O to D. **Q.E.D.**

Note that in **Proposition 0.1** and **Proposition 0.2**, Bellman's principle does not hold for the complete set of departure times and support points Ω at the intermediate node. This should not be confused with the fact that it will hold if the departure time and support point sets are adequately defined at the intermediate node.

The path with MED for a departure time as defined in this chapter has the minimum expected disutility evaluated over all support points. For every possible exit time out of an intermediate node, the sub-path starting from the intermediate node must have the minimum expected disutility evaluated over the compatible support points given the traversal history so far, but does not necessarily achieve the minimum when the expectation is taken over all support points. For example, in the illustrative network of Figure 0.1, we can determine from Table 0.2 that path $\lambda_1 : O \rightarrow a \rightarrow D$ is with MED for departure time $t = 0$. There are two possible exit times out of the intermediate node a: $t_1 = 1$, and $t_2 = 2$. For exit time $t_1 = 1$, the corresponding support point is C^1, and the sub-path $a \rightarrow D$ is with MED for exit time $t_1 = 1$ at C^1; for exit time $t_2 = 2$, the corresponding support point is C^2, and the sub-path $a \rightarrow D$ is with MED for exit time $t_2 = 2$ at C^2. However as shown before, $a \rightarrow D$ is not with MED at time 1 if the expectation is taken over C^1 and C^2.

Similarly, the non-dominated path is non-dominated w.r.t. the complete set of departure time and support points Ω. The sub-path at an intermediate node is non-dominated w.r.t. such a subset Ω' that contains all the possible pairs of the exit time out of the intermediate node and the corresponding support points. For example, in the illustrative network of Figure 0.1, path $\lambda_1 : O \rightarrow a \rightarrow D$ is non-dominated w.r.t. Ω. The set of possible exit times out of the intermediate node a and the corresponding support points is $\Omega' = \{(1,C^1), (2,C^1), (2,C^2), (3,C^1), (3,C^2) \cdot \cdot \cdot \}$. The sub-path $a \rightarrow D$ is non-dominated w.r.t Ω', as the travel time is always 1, the same as (in other words, it is not dominated by) $a \rightarrow c \rightarrow D$, even though $a \rightarrow D$ is dominated by $a \rightarrow c \rightarrow D$ w.r.t. Ω.

The above observations however cannot help build a tractable case. There are potentially 2^{KR} relevant time-support-point set Ω' (the power set of Ω), and generating a non-dominated path set for each of them is intractable. Fortunately we find out that a property related to non-dominance satisfy Bellman's principle for the complete set Ω as described next.

Definition 0.3 (Pure Path). A path is pure iff the path itself and all its sub-paths are non-dominated w.r.t. the complete set of departure time and support points Ω; otherwise, it is a mixed path.

For the example of Figure 0.1, path $\lambda_2: O \rightarrow a \rightarrow c \rightarrow D$ is a pure path from origin O to destination D.

Any pure path is a non-dominated path, while a mixed path could be either dominated or non-dominated. Any dominated path is a mixed path, while a non-dominated path could be either mixed or pure. The relationship can be represented by the following chart, where the outer rectangle represents the complete set of paths between a given OD pair:

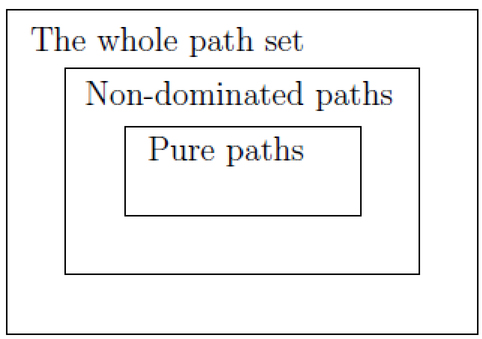


Figure 0.2 Path Category

Unlike non-dominated paths, pure paths have the property that any sub-path of a pure path must be pure by definition, i.e., Bellman's principle holds for this property. Moreover, the following proposition and theorem guarantee that there must be a pure optimal path.

Proposition 0.3. For any mixed path μ from origin node O to destination D, there exists a pure path λ such that $D_\lambda(O,t,r) \leq D_\mu(O,t,r), \forall(t,r) \in \Omega$.

Proof. We prove the proposition by induction.

Basis. At time $t \geq K-1$, link travel times become static and deterministic. Pure paths are optimal, and any mixed path (non-optimal) is dominated by a pure (optimal) path. Therefore **Proposition 0.3** holds.

Inductive step. Suppose **Proposition 0.3** holds at any time $t \geq \tau + 1$. Consider a mixed path μ at $t = \tau$ and node O. If μ is dominated, denote the non-dominated path that dominates μ as γ, and γ can be either pure or mixed. If μ is non-dominated, set $\gamma = \mu$, and then γ is mixed non-dominated. Therefore, $D_\gamma(O,\tau,r) \leq D_\mu(O,\tau,r), \forall r$.

Now consider the non-dominated path γ.

Case 1: γ is pure. Set $\lambda = \gamma$, so $D_\lambda(O,\tau,r) \leq D_\mu(O,\tau,r), \forall r$.

Case 2: γ is mixed. Denote the next node as k. If the sub-path γ' from node k to the destination is mixed, then there must exist a pure path λ' such that $D_{\lambda'}\left(k,\tau + C_{Ok,\tau}^r, r\right) \leq D_{\gamma'}\left(k,\tau + C_{Ok,\tau}^r, r\right), \forall r$ according to the inductive assumption that **Proposition 0.3** holds at any time $t \geq \tau + 1$. Note that $\tau + C_{Ok,\tau}^r \geq \tau + 1$ due to the positive and integer travel time assumption. The disutility function is an increasing function of travel time, so $S_{\lambda'}\left(k,\tau + C_{Ok,\tau}^r, r\right) \leq S_{\gamma'}\left(k,\tau + C_{Ok,\tau}^r, r\right), \forall r$. Then construct a path λ from origin node O to destination D by replacing the dominated sub-path γ' of the mixed non-dominated path γ with the pure sub-path λ'. Then for the resulting path λ, we have the following: $S_\lambda(O,\tau,r) = C_{Ok,\tau}^r + S_{\lambda'}\left(k,\tau + C_{Ok,\tau}^r, r\right) \leq C_{Ok,\tau}^r + S_{\gamma'}\left(k,\tau + C_{Ok,\tau}^r, r\right) = S_\gamma(O,\tau,r), \forall r$. The disutility function is an increasing function of travel time, so $D_\lambda(O,\tau,r) \leq D_\gamma(O,\tau,r) \leq D_\mu(O,\tau,r), \forall r$.

Since γ is non-dominated, the newly constructed path λ is also non-dominated. Furthermore, the sub-path of λ is pure, so λ is pure and **Proposition 0.3** is true at time τ.

With the basis and inductive step, **Proposition 0.3** holds $\forall (t,r) \in \Omega$. **Q.E.D.**

Note that in the basis step, the proposition also holds in the static time period without the deterministic assumption. In other words, a sub-path of a non-dominated path must be non-dominated in a static stochastic network.

A straightforward conclusion can be drawn from **Proposition 0.3** that, if a mixed path has MED for a departure time, then there must exist a pure path with the same MED for the same departure time. This leads to the following theorem:

Theorem 0.1 (Pure Optimal Path). For any origin *O* and departure time *t*, there exists a pure path with MED.

Definition 0.3 and **Theorem 0.1** show two most important properties of the pure paths: any sub-path of a pure path must be pure, and it is guaranteed that there is a pure optimal path. Therefore we can construct a pure path based on downstream pure paths, and, as long as we find all pure paths, we can find the pure optimal path(s). Moreover, the set of pure paths is the same for any disutility function as long as it is increasing with travel time, i.e., for any type of users, no matter whether they are risk-averse or risk-seeking, assuming their risk attitudes can be described by the expected utility theory (EUT). However, the final optimal path is potentially different for users with different risk attitudes.

Other properties of the pure paths are given as follows.

From **Proposition 0.3** we can draw another conclusion that, for any mixed non-dominated path γ from origin node *O* to destination *D*, there exists a pure path λ such that $D_\lambda(O, t, r) = D_\gamma(O, t, r)$, for all (t, r) in Ω, i.e., they share the same travel time distribution. However, for a pure path, it is not necessarily true that there exists a mixed non-dominated path that shares the same travel time distribution. If there does exist a mixed non-dominated path sharing the same travel time distribution with the pure path, we call the mixed non-dominated path a shadow path of the pure path. Note that, for a pure path, there might be no shadow path, and there might be multiple shadow paths. This gives a clue of the relationship between the set of non-dominated paths and the set of pure paths in Figure 0.2.

A pure path, compared with its shadow path, if there exists one, is a more robust routing choice against the errors in the travel time distribution data. If some travel time data in some support point is not correct, a traveler might arrive at an intermediate node earlier or later than he/she should according to the data, and, under the circumstance, the sub-path of a pure path is still a non-dominated path from the intermediate node to the destination, which is not guaranteed for its shadow path, i.e., a mixed non-dominated path.

5.3 Algorithm CD-Path

5.3.1 Solution Approach

Algorithm CD-Path is designed to find all pure paths and thus will find the pure paths with MED for every departure time. However it will miss all the shadow paths and thus those shadow paths with MED. Note that Algorithm CD-Path finds pure paths using support point travel times rather than support point disutilities due to the equivalence between the non-dominance/purity w.r.t. these two.

The algorithm maintains a set of pure paths for each node j, denoted as $\chi(j)$. A scan eligible (SE) list is used to identify each distinct pure path by node-path pairs (j, λ). At each iteration of the algorithm, a pair (k_0, λ_0) is selected from the SE list. Two pointers are required for each path λ at each predecessor node j to store the pure paths: $\pi^{\lambda}{}_{j}$, indicating the next node; and L^{λ}_{j}, indicating the sub-path out of next node. A new path λ is constructed (if not yet) for each possible predecessor node j by making k_0 the next node and λ_0 the sub-path, i.e., $\pi^{\lambda}_{j} = k_0$, and $L^{\lambda}_{j} = \lambda_0$. λ is then added to $\chi(j)$ and dominance among the set is checked. Dominated paths are removed from the set, and temporally non-dominated paths are maintained. Upon termination, the final solution set contains only non-dominated paths.

However, note that at this point the final solution sets might contain mixed non-dominated paths with dominated sub-paths for the following reason. In some iteration, a path λ_0 is added to the pure path set $\chi(k_0)$ and is not dominated by any path in the set at that iteration, so its node-path pair (k_0, λ_0) is added to the SE list, and a path might be constructed for k_0's predecessor nodes based on λ_0, say, λ for node j. In a later iteration, a pure path λ'_0 that dominates path λ_0 is added to the pure path set $\chi(k_0)$ and so path λ_0 is discarded. At this point, λ at the predecessor node j is determined mixed, yet still stays in the pure set $\chi(j)$. At the end of the algorithm, while λ needs to be explicitly retrieved, it will encounter the problem that its sub-path λ_0 is no longer in the pure path set at node k_0, $\chi(k_0)$. In this case, we can determine that λ is a mixed path and remove it from the pure path set $\chi(j)$. After the mixed non-dominated paths are removed, the final solution set contains only pure paths, and the pure paths with MED for every departure time can be selected from the set.

Alternatively we could remove the mixed non-dominated path λ as soon as the sub-path λ_0 is found to be dominated. This procedure has the potential advantage of reduced path set size, because mixed path is removed right away and, by having a smaller current pure path set, a newly generated candidate path at the predecessor node is less likely to be included in the set. However, it requires significant additional computation time to find all the paths that contains a particular sub-path. Therefore, we decide to remove those mixed non-dominated paths only in the end.

Note that, when checking dominance, we adapt Procedure LR-CHECK from Nie and Wu (2009b) in order to reduce the amount of work required to check dominance. We firstly determine whether the newly generated path λ will update the Pareto frontier, i.e., whether it has a smaller travel time in some support point than the current Pareto frontier. If yes, then λ must be non-dominated, and next we only need to check whether it dominates any path in the current pure path set $\chi(j)$ that does not contribute to the Pareto frontier; if not, then we still need to check whether λ is dominated by any path in $\chi(j)$.

Algorithm CD-Path can be viewed as an extension of Algorithm EV (Miller-Hooks and Mahmassani, 2000). The major difference between the two is that Algorithm CD-Path works in an STD network where both temporal and spatial dependences are considered while Algorithm EV works in an independent STD network.

As a result, the dominance rule is different. Algorithm CD-Path checks dominance of paths w.r.t. the support point travel times over the complete set of departure time and support point pairs Ω, while in Algorithm EV, the dominance is checked w.r.t. the expected travel times over the departure time set T.

The difference is also reflected in the computational demand. The path set is potentially much larger in Algorithm CD-Path as the chance of dominance is smaller with a larger dimension for checking dominance, and potentially longer computation time is required.

Algorithm EV can be extended with $d_{\lambda}^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right) \cdot p_i$ (0.4$d_{\lambda}^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right) \cdot p_i$

(0.4) to find the MED path in an independent STD network, only if the disutility is either an affine or exponential function of the travel time (Eiger et al., 1985), as for those two types of utility functions the recursive equation between expected disutilities at two adjacent nodes are valid. In contrast, in our problem context, the non-dominance / purity of a path w.r.t. disutility is equivalent to that w.r.t. travel time as long as the disutility function is increasing with travel time, and thus Algorithm CD-Path actually generates all pure paths w.r.t. any increasing disutility function. In other words, Algorithm CD-Path can be applied to any increasing disutility function of the travel time, and the algorithm is applicable to a wide range of risk attitudes in path finding.

5.3.2 Algorithm Statement

The steps of Algorithm CD-Path are described next:

Algorithm CD-Path

Step 0: Initialization

Step 0.1: Initialize labels and path pointers:

For each $j \in N\backslash\{D\}$

$$\chi(j) = \emptyset, \pi_j^c = \infty, L_j^c = \infty, S_c(j,t,r) = \infty, \forall t, \forall r, \forall c \in \{1, 2, \cdots, M\}$$

where M is a large enough number so as to permit as many pure paths at any node as might be needed.

For the destination node *D*, there is one pure path – going to itself, and the travel time is always 0:

$$\chi(D) = \{1\}, \pi_D^1 = D, L_D^1 = 1, S_1(D,t,r) = 0, \forall t, \forall r$$

Step 0.2: Initialize the scan eligible list:

Insert the node-path pair (D, 1) in the SE list.

Step 1: Check SE List and Scan Node

If the SE list is not empty, then

Select the first node-path pair (k_0, λ_0) from the list. Call the associated node k_0 the current node and λ_0 the current path.

If the list is empty, then

Go to Step 3.

Step 2: Update Labels
For each $j \in \Gamma^{-1}(k_0)$, i.e., for each $(j, k_0) \in A$

Step 2.1. Temporal Label Creation:
Set the path pointers: $\pi_j^\lambda = k_0, L_j^\lambda = \lambda_0$, construct a new path λ from node j to destination D.
Calculate $S_\lambda(j,t,r)\ \forall t, \forall r$ by Eq. $D\lambda(O,t,r) = D\left(C_{Ok,t}^r + S_{\lambda'}\left(k, t + C_{Ok,t}^r, r\right)\right)$ (0.2):

$$S_\lambda(j,t,r) = C_{jk_0,t}^r + S_{\lambda_0}\left(k_0, t + C_{jk_0,t}^r, r\right)$$

Step 2.2. Label Comparison:
Add λ to $\chi(j)$ and check dominance among the set. Remove dominated paths from $\chi(j)$. If λ is not dominated by any other path in $\chi(j)$, then add node-path pair (j, λ) to the SE list.

Step 3: Stop and Find the Paths with MED
For each $j \in N$

Retrieve each path by recursively combining the next node and next sub-path. If a path is not retrievable due to a missing sub-path, it is a mixed path and discarded. The remaining set $\chi(j)$ contains all pure paths at node j, and the path with MED can be determined for each node j and each departure time t.

Algorithm CD-Path terminates with the set of all pure paths at each node after a finite number of steps. It has exponential worst-case computational complexity, but the computational tests in Section 5.4 show that the set of pure paths in a typical transportation network is much smaller than the worst case and thus manageable. The following propositions give important facts of Algorithm CD-Path.

Proposition 0.4. Algorithm CD-Path terminates with the set of all pure paths.

Proof. Firstly, a proof is provided to show that, upon termination, for each origin node j, all paths in $\chi(j)$ are pure. This comes from the path construction principle of the algorithm. In Steps 2 and 3 of Algorithm CD-Path, not only the dominated paths are discarded, but also all paths that contain the discarded paths as sub-paths are removed from $\chi(j)$. Thus, no mixed paths can remain in $\chi(j)$.

Next, it is established that all pure paths departing from node j are in $\chi(j)$. Suppose there exists a pure path which is not in $\chi(j)$, then either 1) it is constructed and then discarded at some point, or 2) it is never constructed. Case 1 is not possible because it contradicts to the fact that a pure path and all its sub-paths are non-dominated. Case 2 is not possible because if so, either the SE list is not empty, which contradicts to the statement of termination, or the path contains at least one sub-path which is dominated, which contradicts to the definition of a pure path. **Q.E.D.**

Proposition 0.5. Algorithm CD-Path terminates after a finite number of steps.

Proof. Suppose the algorithm does not terminate after a finite number of steps, then the SE list does not become empty after a finite number of steps, thus, either 1) at least one node-path pair enters the SE list for an infinite number of times, or 2) an infinite number of node-path pairs enter the SE list.

Case 1 is not possible because any node-path pair can enter the SE list at most once when it is constructed and remains in the SE list iff it is determined pure.

Case 2 is not possible because the network is finite, and there are a finite number of time intervals and support points. **Q.E.D.**

Proposition 0.6. Algorithm CD-Path has exponential worst-case computational complexity.

Proof. It is possible that, in the worst case, all paths are pure and, thus, stay in the final solution set generated by the algorithm upon termination. Consequently, Algorithm CD-Path, which generates all pure paths, is exponential in worst-case computational complexity. **Q.E.D.**

As we store the support point travel time at each departure time, $K \times R$ labels are needed for each path and this could mount to a high memory requirement. The computational tests conducted in Section 5.4 also show that the limit of the computation comes from the memory. If we do not store the support point travel times as labels, but calculate them each time it is needed, the requirement on memory will be significantly lowered as no labels are stored. However this approach will require prohibitively longer computational time, which might render the computation practically infeasible.

One potential solution could be a heuristic that limits the size of the pure path set as a tractable number M (Miller-Hooks and Mahmassani, 2000). However, Miller-Hooks (1997) shows that such a heuristic might not find the optimal path. Masin and Bukchin (2008) proposes another algorithm based on the idea of diversity maximization so that feasible paths that are as different from each other as possible will be maintained in the final set, and Nie et al. (2010) implements the heuristic in an optimal path problem with second-order stochastic dominance. Other heuristic ideas include 1) certainty equivalent approximation, which replaces every link travel time random variable by its expected value and thus transforms the stochastic network into a deterministic one; 2) aggregating the distribution, where we check the similarity of the support points, group the similar ones, and replace every link travel time random variable by its expected value within the group, and thus the number of support points is reduced; and 3) working on a limited number of scenarios, e.g., after aggregating the distribution, we can choose a certain number of scenarios such as most-likely scenario, best scenario and worst scenario, and work on them only.

It is desirable for us to explore the actual difference between the pure path set and the non-dominated path set. Note that, since non-dominated paths could be mixed paths, i.e., they could contain dominated sub-paths, generating the non-dominated path set would require enumerating all the paths. In Section 5.4.2 we adapt Algorithm CD-Path to generate non-dominated paths and run tests to investigate the difference.

5.4 Computational Tests

The objectives of the computational tests are to: 1) investigate the average running time of Algorithm CD-Path as a function of the network size in all three types of networks; 2) investigate the size of pure path set as a function of network size in all three types of networks; 3) study computationally how the risk aversion coefficient affects the optimal path solution; and 4) study computationally how the level of stochastic dependency affects the optimal path solution.

5.4.1 Network and Link Travel Time Distribution

The computational tests in this section are conducted in three types of networks: step networks, grid networks, and random networks, the topology of which are randomly generated. Detailed information on each network type is given next.

1. Step Network

Theoretically, in an STD network all links have random travel times. However, in order to have a tractable yet still realistic model, we treat the most variable part of the network as stochastic and the rest deterministic.

In this section, we call a network as in Figure 0.3 a step network. The double-lined links on the diagonal are freeway links, and the nodes on the diagonal are freeway entrances / exits. The horizontal solid link next to each freeway entrance node is an on-ramp link, and the remaining dashed links are local links or off-ramp links. It is assumed that freeway links and on-ramp links have stochastic and time-dependent travel times, while local links and off-ramp links have static and deterministic travel times.

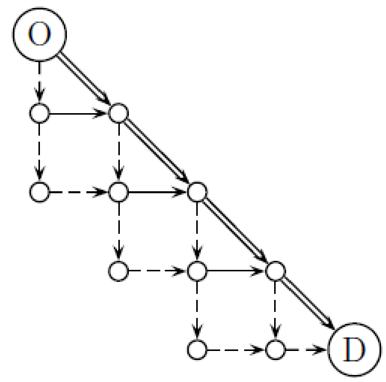


Figure 0.3 Step Network

A step network can be viewed as a representative transportation network for a typical transportation corridor with a highway and parallel arteries. The underlying rationale is that the variations of the travel times on freeway links are similar and much larger than those of the travel times on local links. The all-local path represents the shortest among all local paths that do not have much variability and can be treated as deterministic, and other all-local paths are removed from the original network. Those deterministic links could be restored to the step network without changing the optimal path solution and the complexity of the problem.

For a step network of level n, there are $3n$ nodes, $n + 1$ of which are freeway exits, and $5n - 2$ links: n freeway links, $n - 1$ on-ramp links and $3n-1$ local links or off-ramp links. The network in Figure 0.3 is a step network of level 4, and the one in Figure 0.1 is of level 2. In a step network, there is one all-freeway path and one all-local path. The other paths are mixed with freeway links, on-ramp links, local links and/or off-ramp links.

2. Grid Network

Another transportation network type is the grid network, which is often seen in an urban area such as Manhattan. In a grid network, potentially all links have similar variability and probably should be treated as random. Figure 0.4 gives an example grid network of level 4. For a grid network of level n, there are $(n + 1)^2$ nodes and $2n(n + 1)$ links.

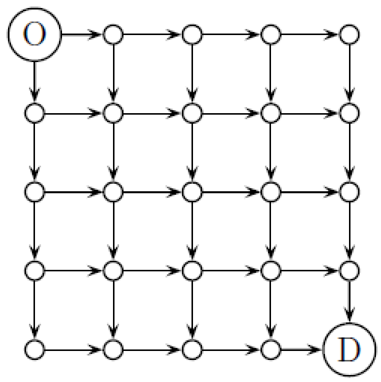


Figure 0.4 Grid Network

3. Random Network

The previous two network types represent two typical transportation networks, one as a corridor connecting two cities and the other as an urban network. We also conduct computational tests on more general networks with randomly generated topology, called random networks in this section. We take the number of nodes as input, set the number of links as three times the number of nodes and use a random network generator to construct the network topology. More details of the random network generator can be found in Gao (2005) and Gao and Huang (2011).

The computational tests are run for all three types of networks. The details are given next.

The tests are conducted on step networks of levels from 3 to 15 (3, 5, 7, 10, 12 and 15). The first freeway node is set as the origin and the last freeway node the destination (the nodes O and D in Figure 0.3).

The tests on grid networks are conducted for levels from 3 to 7 with 30 time periods. The left-top node is assumed the origin and the right-bottom node the destination (the nodes O and D in Figure 0.4). Note that, although the largest level of the tested grid network is smaller than that of the step network, the number of nodes and the number of links are not smaller at all. For a step network of level 15, the number of nodes is 45 and the number of links is 73; for a grid network of level 7, the number of nodes is 64 and the number of links is 112.

For random networks, for the purpose of comparison, we set the number of nodes the same as that of the step networks, i.e., the number of nodes are from 9 to 45. The number of links is always three times the number of nodes, i.e., from 27 to 135.

For all three types of networks, the travel times on stochastic links are sampled from truncated (at 3) multivariate normal distribution, with 3 as the original mean, 4 the original variance, and an original uniform correlation coefficient varying from 0 to 1. Note that the actual mean of the sample will be between 4 and 5. The actual variance and the actual correlation coefficient will also be different from the original. The positive uniform correlation coefficient ensures that the covariance matrix is positive-semi definite, and thus its validity. Note that the stochastic links indicate the freeway links and on-ramp links for step network, and all links for grid network and random network. Travel times on deterministic links, i.e., the local links and off-ramp links for step network, are fixed as 3.

There are 50 support points and 30 time periods for link travel time random variables. For each combination of network level and correlation coefficient, 10 networks are randomly generated. Note that, for step network and grid network, the network topology remains the same across the 10 while the link travel time distributions are different; for random network, both are different. The results shown in Section 5.4.2 are the averages over the 10 networks for each parameter combination.

An exponential disutility function of path travel time is applied, i.e., $D_\lambda(O,t,r) = D\big(S_\lambda(O,t,r)\big) = \exp(\alpha \cdot S_\lambda(O,t,r))$, and the expected disutility is given in Eq. $d_\lambda^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\big(k,t+C_{Ok,t}^{i}\big)\right) \cdot p_i$ (0.5).

Please find next Table 0.5 for a summary of the computational test parameters.

Table 0.5 Summary of the Computational Test Parameters

	Step Network	Grid Network	Random Network
Number of Levels	3, 5, 7, 10, 12, 15	3, 5, 7	N/A
Number of Nodes	9, 15, 21, 30, 36, 45	16, 36, 64	9, 15, 21, 30, 36, 45
Number of Links	13, 23, 33, 48, 58, 73	24, 60, 112	27, 45, 63, 90, 118, 135
Number of Stochastic Links	5, 9, 13, 19, 23, 29	24, 60, 112	27, 45, 63, 90, 118, 135
Number of Support Points	50	50	50
Number of Time Periods	30	30	30
Number of Generated Networks	10	10	10
Topology of Generated Networks	The Same	The Same	Different

5.4.2 Computational Tests Results

Algorithm CD-Path is coded using C++ and tested on a Windows Vista Business (64 bit) workstation with Intel Core i5 CPU 650 @ 3.20GHz and 8.00GB RAM.

Tables Table 0.6 through Table 0.11 show the average running time of Algorithm CD-Path and the average size of the pure path set for all three network types. Note that the algorithm finds optimal paths from all nodes to the destination. For step networks and grid networks, the average size of the pure path set is that of the set for the origin; for random networks, it is the average of the sizes of the sets for all nodes. We present two regressions for each of the six tables, one of which is exponential function and the other polynomial. In the regressions, RUN is the average running time over all tested correlation coefficients, SIZE is the average size of the pure path set of the origin node over all tested correlation coefficients, and *n* is the number of nodes.

Table 0.6 Average Running Time vs. Network Size for Step Network

ρ	Step Network					
0	1.145737	1.202073	2.52565	22.17755	105.5225	62.117
0.2	1.165558	1.212483	2.676304	33.05109	57.51884	1285.98
0.4	1.143493	1.207659	2.283864	41.28578	83.39659	1500.43
0.6	1.146104	1.199578	2.116415	27.0273	84.48582	1658.971
0.8	1.143877	1.178022	1.576122	5.609157	20.13407	774.196
1	1.143734	1.146031	1.162972	1.186454	1.380533	1.945746
avg.	1.148084	1.190974	2.056888	21.72289	58.73973	880.6067
$\|N\| = n$	9	15	21	30	36	45
$\|A\| = m$	13	23	33	48	58	73
Level	3	5	7	10	12	15
Regressions	$\text{RUN} = 0.0849 \cdot e^{0.1908n} (R^2 = 0.9409)$ $\text{RUN} = 0.00005 \cdot n^{3.9631} (R^2 = 0.7978)$					

Table 0.7 Average Running Time vs. Network Size for Grid Network

ρ	Grid Network		
0	1.297491	2.360869	215.6773
0.2	1.277201	2.366057	216.1953
0.4	1.277616	2.370289	217.3097
0.6	1.266753	2.372914	220.3326
0.8	1.20332	2.38388	220.7128
1	1.198285	1.29772	3.818742
avg.	1.253444	2.191955	182.3411
$\lvert N\rvert = n$	16	36	64
$\lvert A\rvert = m$	24	60	112
Level	3	5	7
Regressions	$\text{RUN} = 0.1257 \cdot e^{0.1072n}(R^2 = 0.898)$ $\text{RUN} = 0.00005 \cdot n^{3.4018}(R^2 = 0.7542)$		

Table 0.8 Average Running Time vs. Network Size for Random Network

ρ	Random Network					
0	1.162832	1.199492	1.216785	1.27384	1.319142	1.448924
0.2	1.152093	1.173776	1.200634	1.269578	1.310311	1.456919
0.4	1.1511	1.168702	1.186761	1.265214	1.267603	1.391558
0.6	1.146741	1.162835	1.169498	1.196327	1.210066	1.253251
0.8	1.145219	1.162771	1.163558	1.170083	1.188513	1.178153
1	1.144682	1.146426	1.152655	1.150763	1.155437	1.157254
avg.	1.150445	1.169	1.181649	1.220967	1.241845	1.314343
$\lvert N\rvert = n$	9	15	21	30	36	45
$\lvert A\rvert = m$	27	45	63	90	118	135
Regressions	$\text{RUN} = 1.1054 \cdot e^{0.0035n}(R^2 = 0.9587)$ $\text{RUN} = 0.9596 \cdot n^{0.0747}(R^2 = 0.842)$					

Table 0.9 Average Size of Pure Path Set vs. Network Size for Step Network

ρ	Step Network					
0	9	56	229.4	662.1	1273.18	643
0.2	9	54	246.4	929.6	923.9	3112.3
0.4	9	52.3	217.3	1106.6	1319	3720.4
0.6	9	49.3	198	859.2	1372.8	3505
0.8	8	35.3	119	295	615.6	1811.6
1	4	6	8	11	13	16
avg.	8	42.15	169.6833	643.9167	919.6833	2134.717
$\|N\| = n$	9	15	21	30	36	45
$\|A\| = m$	13	23	33	48	58	73
Level	3	5	7	10	12	15
Regressions	$\text{SIZE} = 4.0395 \cdot e^{0.1509n}(R^2 = 0.9388)$ $\text{SIZE} = 0.0036 \cdot n^{3.507}(R^2 = 0.9969)$					

Table 0.10 Average Size of Pure Path Set vs. Network Size for Grid Network

ρ	Grid Network		
0	20	252	3432
0.2	20	252	3432
0.4	20	252	3432
0.6	20	252	3432
0.8	20	252	3432
1	8.3	38.7	69.3
avg.	18.05	216.45	2871.55
$\|N\| = n$	16	36	64
$\|A\| = m$	24	60	112
Level	3	5	7
Regressions	$\text{SIZE} = 3.8973 \cdot e^{0.1048n}(R^2 = 0.9929)$ $\text{SIZE} = 0.0007 \cdot n^{3.6179}(R^2 = 0.9881)$		

Table 0.11 Average Size of Pure Path Set vs. Network Size for Random Network

ρ	Random Network					
0	4.844444	7.726667	7.961905	9.273333	10.15556	11.59556
0.2	4.066667	6.14	7.319048	9.44	9.391667	11.13778
0.4	3.688889	5.793333	6.590476	8.77	8.280556	10.16
0.6	3.555556	4.48	4.695238	5.616667	5.855556	6.711111
0.8	2.522222	3.64	3.319048	3.88	4.225	4.015556
1	1.344444	1.626667	1.633333	1.666667	1.605556	1.564444
avg.	3.337037	4.901111	5.253175	6.441111	6.585648	7.530741
$\|N\| = n$	9	15	21	30	36	45
$\|A\| = m$	27	45	63	90	118	135
Regressions	$\text{SIZE} = 3.2507 \cdot e^{0.0202n} (R^2 = 0.881)$ $\text{SIZE} = 1.236 \cdot n^{0.4775} (R^2 = 0.9701)$					

The tables show that, for step networks, the average running time of Algorithm CD-Path grows exponentially with the network size and we tend to believe that the average size of the pure path set at the origin node grows polynomially with the network size (the difference of R^2 is rather small).

The running time grows exponentially because the algorithm potentially needs to check all the paths, the number of which grows exponentially with the network size. Moreover, although the final pure path set size is polynomial, the sets in the process of label-correcting might contain a lot more paths, which are later determined dominated, and this could result in the exponential running time. We have checked the number of operations of checking dominance and it increases exponentially with the network size, which provides an evidence for the exponential running time.

The main reason that pure path set size grows polynomially is that not all the paths are potentially pure. Since the on-ramp link travel time distribution is the same as the freeway link travel time distribution, taking off- and then on-ramps is probably not as good as traveling directly on the freeway. The on-ramp link travel time is always larger than the travel time on a local link, so taking on- and then off-ramps will have a larger travel time than traveling on two consecutive local links. Therefore, frequently taking on- and off-ramps is not an attractive option and that type of path is not likely to be in the pure set. In other words, only three types of paths are potentially pure: 1) the all-freeway path; 2) the all-local path; and 3) the freeway-local paths with a small number of on-ramp and off-ramp links. For a path of type 3, if the small number is one, it would means that, once the traveler is off the freeway, he/she would better never drives back, and, in that case, the number of paths of type 3 is $O(n^2)$. Similarly, for the freeway-local paths with a small number of on-ramp and off-ramp links (not restricted to be one), the number would be polynomial with n.

Another interesting observation is that the pure path set size is relatively small when the correlation is low (e.g., $\rho = 0$) and high (e.g., $\rho = 0.8$ or 1), and so is the running time. The path travel time is the sum of link travel times, so its variance increases with link covariance. When the correlation is low (e.g., $\rho = 0$), the variance of path travel time is small, and so the all-freeway path travel time approximately equals the network level (n) times the expected freeway link travel time (between 4 and 5) in every support point, which is smaller than the all-local path travel time ($6n$). Thus, the all-freeway path is more attracting than when the correlation is a little higher, and so the pure path set size is relatively small. On the other hand, when the correlation is high (e.g., $\rho = 0.8$ or 1), the variance of path travel time is large, and, in this case, taking on- and off-ramps is a very bad choice. Thus, paths with a relatively large number of on- and off-ramp links would not be in the pure path set. For $\rho = 1$, the situation becomes extreme. Only all-freeway path, all-local path and those freeway-local paths with consecutive freeway links, then one off-ramp link, and then consecutive local links are pure, and so the number is $n + 1$.

In grid networks, the size of pure path set seems to grow exponentially (the difference of R^2 is rather small) as well as the average running time.

For the same network level, a grid network generates a lot more pure paths than a step network. One reason is that there are more nodes and links. Another more important reason is that, in a grid network, most of the paths are quite similar to each other and one path is not likely to dominate another. As a matter of fact, it can be observed from Table 0.10 that, for correlation coefficient $\rho \neq 1$, for each network size, the number of pure paths from the origin to the destination are the same across the correlation values. The number is exactly the total number of paths from the origin to the destination: $(2n)!/(n!n!)$.

As the number of pure paths grow extremely fast, the largest grid networks we can run tests on are only of level 7. Therefore, for grid networks, one might want to consider a heuristic where the size of the pure path set is bounded to a tractable number. As pointed out by Miller-Hooks (1997), such a heuristic might not find the optimal path. However, in a grid network where all paths are relatively similar, a sub-optimal path might not be too different from the optimal one and be well acceptable.

In random networks, we tend to believe that the average running time grows exponentially while the size of pure path set seems to grow polynomially with the network size.

Both the running time and the average pure path set size are extremely small compared with those in the other two types of networks. The reason is that, for a random network whose topology is randomly generated, it is quite possible that there are a small number of relatively shortest paths (in terms of the number of links in a path) connecting each node to the given destination, which dominates all others. Still it can be observed that the average size of the pure path sets decreases as the correlation coefficient grows. One possible explanation is that, with a larger correlation coefficient, the number of aforementioned relatively shortest paths is smaller.

More tests are run to compare the pure path set and the non-dominated path set for all the network types. The algorithm to generate non-dominated paths is quite similar to Algorithm CD-Path, only in Step 2.2 we mark the dominated paths as "dominated" rather than discard them. Instead, we keep all the paths in the sets, due to **Proposition 0.2**.

The tests show that, for all the networks we have generated, the non-dominated path set is the same as the pure path set, i.e., a shadow path never exists in the tests. This is probably due to the setting that all link travel time random variables are sampled from the same distribution and uniformly correlated with each other. If a pure path would have a shadow path, then they should share the same travel time distribution. Therefore the sub-paths of the two also share the same travel time distribution for all the possible arrival times at the intermediate node where the two sub-paths separate. For all other departure times for the intermediate node, the sub-path of the pure path should have no larger travel time than the sub-path of the shadow path and should have a smaller travel time for at least one departure time. This would be very rare with the current setting of the tests. Whether a path is non-dominated/pure mainly depends on the number of links in it. If a pure path would have a shadow path, then the number of links in the two paths would be the same and the number of links in the sub-path of the pure path would be smaller than that in the sub-path of its shadow path. This would not be possible in grid networks, as all paths from any node to the destination are with the same number of links and all of them are non-dominated/pure. In random networks, a small number of paths dominate all others and they very likely have the same number of links. If a sub-path of a path has a larger number of links and is dominated, then the path itself would be dominated as well. It is a little more complicated in step networks, as there exist local links. The same idea can be applied. Although we are not able to find a shadow path in the tests, Figure 0.1 gives an example of it.

We also conduct tests to study computationally how the risk aversion coefficient and the level of stochastic dependency affect the optimal path solution with an exponential disutility function. Note that the grid networks generate an extremely large number of similar paths that do not dominate each other, the random networks generate an extremely small number of dominant relatively shortest paths, and there is not much we can tell from the optimal paths of those two types of networks. Therefore, we only work on step networks to investigate how the optimal path solution is related to the risk aversion coefficient in the disutility function and the correlation coefficient of the link travel time random variables. We use the all-freeway path as a benchmark and test in what circumstances the all-freeway path is the optimal and in what circumstances it is not.

Tables Table 0.12 and Table 0.13 show the largest value of α with which the all-freeway path is with MED from the origin node to the destination node for a given link correlation coefficient in two cases, one with stochastic dependencies considered (complete dependency) and the other without (no dependency). The range of the tested α values is from 0.01 to 10 with step 0.01.

An adapted Algorithm EV (Miller-Hooks and Mahmassani, 2000) is applied to generate optimal paths in the no-dependency case. The expected disutility is calculated based on Eq. $d_{\lambda}^{ND}(O,t) = \sum_{i=1}^{Q} D\left(C_{Ok,t}^{i} + e_{\lambda'}^{ND}\left(k, t + C_{Ok,t}^{i}\right)\right) \cdot p_i$ (0.5), which replaces the equation in Step 2 of Algorithm EV. The original Algorithm EV finds the paths with the least expected travel time and thus implicitly assumes risk-neutral users. In order to make comparison between Algorithm CD-Path and Algorithm EV and show the effects of the link travel time correlations and the degree of travelers' risk-averse attitude on the optimal path solution, we need to adapt the original Algorithm EV to make it work with the exponential disutility function. Note that the same network data are used as in the complete dependency case, only a different algorithm is used that treats link travel times as independent.

Table 0.12 Largest Value of α for an Optimal All-Freeway Path (Complete Dependency)

	Network Level					
$\boldsymbol{\rho}$	3	5	7	10	12	15
0	10	9.092	1.773	3.625	1.045	0.604
0.2	2.332	9.017	2.473	0.274	1.619	0.179
0.4	1.29	0.295	0.178	0.158	0.12	0.115
0.6	0.358	0.306	0.131	0.127	0.112	0.087
0.8	0.267	0.135	0.094	0.086	0.076	0.05
1.0	0.165	0.132	0.113	0.053	0.035	0.059

Table 0.13 Largest Value of α for an Optimal All-Freeway Path (No Dependency)

	Network Level					
$\boldsymbol{\rho}$	3	5	7	10	12	15
0	10	10	4.431	3.478	2.723	1.806
0.2	10	9.603	6.187	2.986	3.076	2.673
0.4	8.938	9.231	5.734	2.947	2.961	2.289
0.6	9.062	8.826	4.58	3.721	3.765	3.095
0.8	9.23	9.043	3.908	4.149	3.112	2.487
1.0	8.467	8.918	4.186	3.231	2.637	2.576

It is shown that the all-freeway path is more attractive when the correlation and/or risk aversion is low. In the complete dependency case, the boundary value of decreases with ρ, suggesting that the all-freeway path is more attractive when the correlation is lower for a given risk aversion level. Furthermore, the boundary value of α decreases with the network size, suggesting that when the network size is larger, the all-freeway path is less likely to be optimal. This can be explained by noting that in a larger network the OD paths have larger number of links and thus the effect of link correlation on path travel time risk is more prominent, which is to the disadvantage of the most risky path – the all-freeway path. If the travel times are assumed independent, Table 0.13 shows that the boundary value of α is virtually independent with the correlation. This is understandable as the correlation is used only in the data generation and ignored by the adapted Algorithm EV. This shows that ignoring stochastic dependency would generate the same optimal path regardless of the correlation, yet in reality the optimal path changes with correlation. Comparing the α values in Tables Table 0.12 and Table 0.13, the difference between the complete dependency case and the no dependency case is small when the correlation is low. When the correlation is high, the complete dependency case shows only with a very small α, the all-freeway path is optimal, while the no dependency case shows the same values as when the correlation is low.

5.5 Supplemental Analytical Solutions

We next work on a small example with static and continuously distributed travel times where analytical solutions can be obtained. This analysis complements the computational tests with time-dependent and discrete travel time distributions. As will be shown, similar effects of link travel time correlations and the degree of travelers' risk-averse attitude on optimal path solutions are found, which demonstrates the robustness of the results to some extent.

In the network of Figure 0.1, let freeway and on-ramp link travel times be multivariate normal random variables X_1, X_2, X_3, which represent link travel times on (O, a), (a, D) and (b, a) respectively. Assume they have identical mean μ, variance σ^2 and each pair has an identical correlation coefficient ρ. Their joint distribution is written as X_1, X_2, X_3 ~ MVN(μ, σ^2, ρ). Local link travel time is fixed at μ. The travel times are static.

The distributions for the travel times of the five paths from origin O to destination D are given as follows:

$$\lambda_1: X_1 + X_2 \sim N(2\mu, 2(1+\rho)\sigma^2)$$

$$\lambda_2: X_1 + 2\mu \sim N(3\mu, \sigma^2)$$

$$\lambda_3: X_3 + X_2 + \mu \sim N(3\mu, 2(1+\rho)\sigma^2)$$

$$\lambda_4: X_3 + 3\mu \sim N(4\mu, \sigma^2)$$

$$\lambda_5: 4\mu$$

Compared to λ_3, λ_1 has the same variance, but a smaller mean; and compared to λ_4, λ_2 has the same variance, but a smaller mean. Therefore λ_3 and λ_4 are first-order dominated by λ_1 and λ_2 respectively, and can be eliminated from further analysis (Hadar and Russell, 1969). Note that in this case, the all-freeway path λ_1 is risky yet short, the all-local path λ_5 is risk-free yet long, and the freeway-and-local path λ_2 has moderate risk and a medium travel time.

Assuming exponential disutility function, the disutility functions for the paths are log-normally distributed and their expected values are given as follows:

$$\lambda_1: e^{\alpha(X_1+X_2)} \sim \log - N(2\alpha\mu, 2\alpha^2(1+\rho)\sigma^2)$$

$$E\left[e^{\alpha(X_1+X_2)}\right] = e^{2\alpha\mu+\alpha^2(1+\rho)\sigma^2}$$

$$\lambda_2: e^{\alpha(X_1+2\mu)} \sim \log - N(3\alpha\mu, \alpha^2\sigma^2)$$

$$E\left[e^{\alpha(X_1+2\mu)}\right] = e^{3\alpha\mu+\ \alpha^2\sigma^2/2}$$

$$\lambda_5: e^{4\alpha\mu}$$

If μ = 3 minutes (roughly equivalent to freeway exit spacing of 3 miles at 60 mph), and $\sigma^2 = 2^2 = 4$ minutes^2, then the expected disutilities of the paths are:

$$\lambda_1: e^{\alpha(X_1+X_2)} \sim \log - N\left(6\alpha, 8\alpha^2(1+\rho)\right)$$

$$E\left[e^{\alpha(X_1+X_2)}\right] = e^{6\alpha+4\alpha^2(1+\rho)}$$

$$\lambda_2: e^{\alpha(X_1+2\mu)} \sim \log - N(9\alpha, 4\ \alpha^2)$$

$$E\left[e^{\alpha(X_1+2\mu)}\right] = e^{9\alpha+\ 2\alpha^2}$$

$$\lambda_5: e^{12\alpha}$$

Figure 0.5 shows how the optimal path solution changes with α and ρ values. The all-freeway path is more likely to be optimal when α is smaller (suggesting an attitude closer to risk neutrality) and ρ is smaller, similar to the results from the computational tests in Section 5.4 .

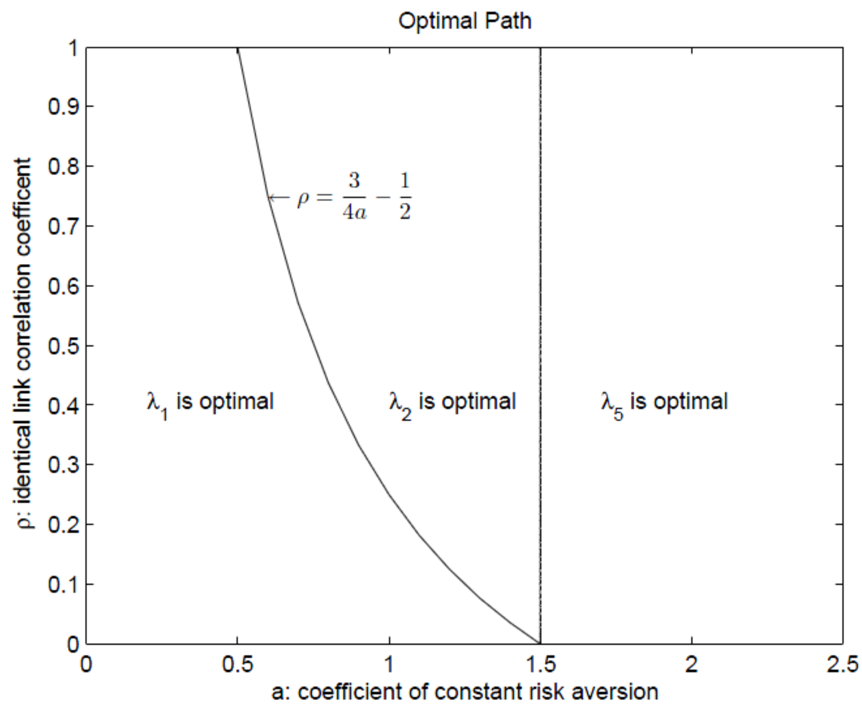


Figure 0.5 Optimal Path Solution and the Corresponding α and ρ Values

Specifically, when $\alpha < 0.5$, the all-freeway path λ_1 is optimal regardless of the correlation. When $0.5 < \alpha < 1.5$, the all-freeway path λ_1 is optimal for low correlations, and the freeway-local path λ_2 is optimal for high correlations. The boundary dividing "low" and "high" correlations changes with α as specified by the equation in Figure 0.5. Note that the boundary is derived numerically in the computational tests. When $\alpha > 1.5$, the all-local path λ_5 is optimal regardless of the correlation.

For normally distributed variables, independence is equivalent to zero correlation coefficient. If the stochastic dependencies are ignored as in most existing studies, the horizontal line in Figure 0.5 with $\rho = 0$ shows that the freeway-local path can never be an optimal path regardless of the risk aversion level, and the all-freeway path is always optimal for $\alpha < 1.5$, which can be viewed as a reasonable range for an average person's risk aversion parameter. This is due to the underestimation of the all-freeway path risk by assuming stochastic independence between links.

5.6 Conclusions and Future Directions

This chapter addresses the optimal path finding problem in a stochastic time-dependent network where all link travel times are temporally and spatially correlated. It is shown that, in such a network, Bellman's principle does not hold if the optimality or non-dominance is defined w.r.t. the complete set of departure time and support point pairs for the path and its sub-paths. A property related to non-dominance is found to satisfy Bellman's principle for the complete set, and it is proved that, for any origin node, there always exists a pure path with MED. An exact label-correcting algorithm is designed to find the optimal paths with MED, and the computational tests show that the average running time of Algorithm CD-Path grows exponentially with the network size and the average size of pure path set grows polynomially in a step network with properly defined stochastic links. Computational tests in large step networks and analytical solutions in a small step network show that all-freeway path is more attractive when link correlation and/or risk aversion is low. The difference of the optimal solution between the complete dependency case and the no dependency case is not large when the correlation of link travel times is low, and relatively large when the correlation is high.

We would like to continue the work on analyzing stochastic transportation networks using real-life freeway data. More computational tests on real-life networks will be valuable. Traffic data could be obtained (e.g., from the PeMS database) and analyzed to study the characteristics of stochastic dependencies among link travel times. Spatial and temporal correlation coefficients among link travel time random variables are to be obtained. A correlation prediction model is to be created by performing a linear or non-linear regression on the observed data. The model will show how the correlation changes over time and space, and can provide a more realistic covariance matrix for link travel time random variables in the computational tests than current identical correlation coefficient assumption.

We will also investigate the extent of spatial and temporal dependencies. For example, given the incoming link travel times at 8:05 AM, will the knowledge of those further upstream at 8:00 AM provide additional useful information about the outgoing link at 8:05 AM? In other words, is the travel time random variable of the outgoing link independent from those further upstream, given the incoming link travel times? If such conditional independence exists, the stochastic network can be represented through a set of conditional probability distributions, instead of a joint distribution of all link travel times. This will enable both efficient storage of the representation in the computer memory and the design of more efficient algorithms than when a joint distribution is used.

When working on real-life networks, we should realize that, if we assume that the link travel times are stochastic for every link and every time periods, the data set will be prohibitively huge. Therefore, we need to assume that only a limited number of links have stochastic and time-dependent (also for a limited number of time periods) travel times. The problem is then how to choose those links and time periods appropriately to strike a good trade-off between realism and tractability.

CHAPTER 6

OPTIMAL TRAJECTORY-ADAPTIVE ROUTING IN STOCHASTIC NETWORKS

6.1 Introduction

This chapter studies the problem of finding the optimal trajectory-adaptive routing policies in a stochastic time-dependent network where all link travel times are temporally and spatially correlated. Similar to CHAPTER 5, the trajectory-adaptive routing policies are evaluated by a disutility function of travel time, and the optimal trajectory-adaptive routing policies are those with the minimum expected disutility. An exact algorithm is designed to find the optimal trajectory-adaptive routing policies. We compare the computational test results with those of the optimal *a priori* path problem (CHAPTER 5) and of the optimal routing problem with perfect online information (CHAPTER 4) to investigate the benefit of traveler information and being adaptive to it.

This chapter is organized as follows. In Section 6.2 , trajectory-adaptive routing policy and optimal trajectory-adaptive routing problem are defined. An exact algorithm is presented in Section 6.3 , and computational tests are conducted in Section 6.4 . In Section 6.5 5.6 , conclusions are made and future directions given.

6.2 Problem Statement

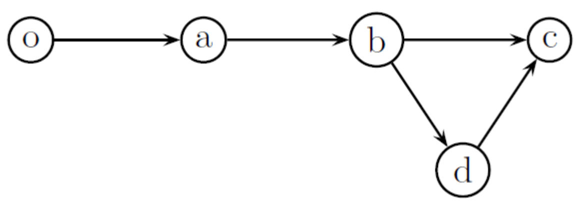
6.2.1 Trajectory Adaptive Routing Policy – Mapping

We firstly define trajectory as follows.

Definition 0.1 (Trajectory). H is a trajectory of node-time pairs a traveler has experienced from the origin j_0 and departure time t_0 up to the current node j and time t: H

$= \{(j_0, t_0), (j_1, t_1), ..., (j, t)\}$, where j_i and t_i are the intermediate nodes and the corresponding arrival times at them.

In trajectory-adaptive routing, the information contains the revealed travel time on link (j_x, j_{x+1}) at time t_x, which is $t_{x+1} - t_x$ for all (j_x, t_x) along the trajectory. The trajectory H itself contains the information, so we use H to denote the trajectory information. For a given trajectory H, we can identify a set of support points of the network as compatible with it if they contains the same link travel times as those along H. This set of such support points is defined as an event collection (CHAPTER 4), $EV(H)$. With more links traversed and more information collected, trajectory (information) H grows and the size of $EV(H)$ decreases or remains unchanged. When $EV(H)$ becomes a singleton, a deterministic network (not necessarily static) is revealed to travelers.



Time	Link	C^1	C^2
0	(o, a)	1	0
	(a, b)	1	2
	(a, b)	1	2
1	(b, c)	3	M
	(b, d)	3	3
2	(b, c)	1	M
	(b, d)	3	3

Figure 0.1 The Illustrative Network

An illustrative network is shown in Figure 0.1. The travel time on link (d, c) is always 0 and not listed. M is a large positive number. The network contains 5 nodes, 5 links and 3 time periods. There are 2 support points, each with a probability of 1/2, for the joint distribution of travel time random variables (links (o, a), (a, b), (b, c) and (b, d) over time periods 0, 1, 2 and beyond). Travel times beyond time 2 are 0, 1, 1 and 3 respectively for the 4 links in both support points. The problem is to find the optimal trajectory-adaptive routing policy to travel from the origin o at time 0 to the destination c. Note that only relevant travel times are listed.

Suppose the current node-time pair is (b, 2), and there are two possible trajectories from (o, 0): $H_1 = \{(o, 0), (a, 1), (b, 2)\}$ and $H_2 = \{(o, 0), (a, 0), (b, 2)\}$. It can be observed that support point C^1 is compatible with trajectory H_1 and support point C^2 is compatible with trajectory H_2. Thus, $EV(H_1) = \{C^1\}$ and $EV(H_2) = \{C^2\}$.

With the trajectory (information) and event collection defined, we can define trajectory-adaptive routing policy following Definition 4.1 in CHAPTER 4 as follows:

Definition 0.2 (Trajectory-Adaptive Routing Policy – Mapping) A trajectory-adaptive routing policy μ from the origin j_0 and departure time t_0 to a given destination d is a mapping from trajectories to next nodes, $\mu: H \to k, k \in \Gamma(j)$, where H is the trajectory, k is the next node, j is the current node, and $\Gamma(j)$ is the set of successive nodes of node j.

For the network in Figure 0.1, an example of a trajectory-adaptive routing policy out of (o, 0) can be given as a mapping as follows:

$$(o, 0) \to a$$

$$(o, 0; a, 1) \to b$$

$$(o, 0; a, 0) \to b$$

$$(o, 0; a, 1; b, 2) \to c$$

$$(o, 0; a, 0; b, 2) \to d$$

$$(o, 0; a, 0; b, 2; d, 5) \to c$$

A trajectory-adaptive routing policy specifies routing decisions (next nodes) based on trajectories, rather than arrival times at intermediate nodes as for a time-adaptive routing policy. In the above example, since node-time pair (b, 2) can be reached via different trajectories out of origin node o and departure time 0, the decision can be different. A representation of the trajectory-adaptive routing policy in a time-space diagram is shown in Figure 0.2.

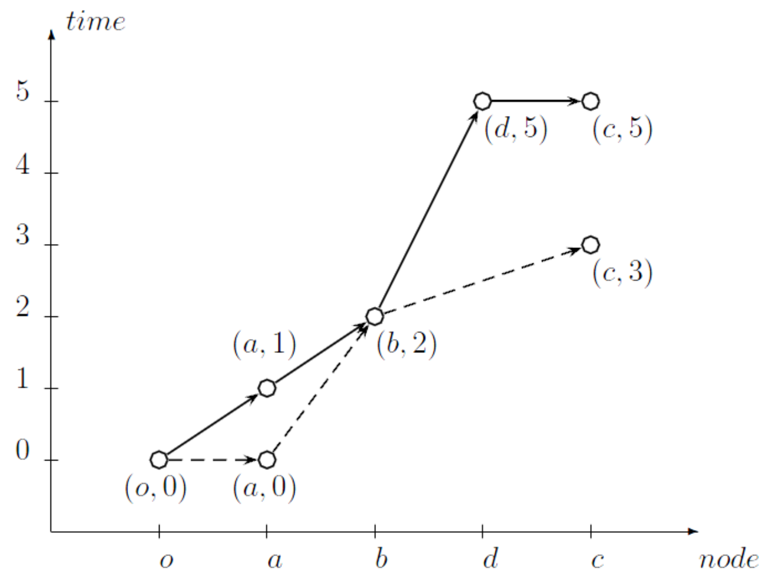


Figure 0.2 Trajectory-Adaptive Routing Policy in Time-Space Domain

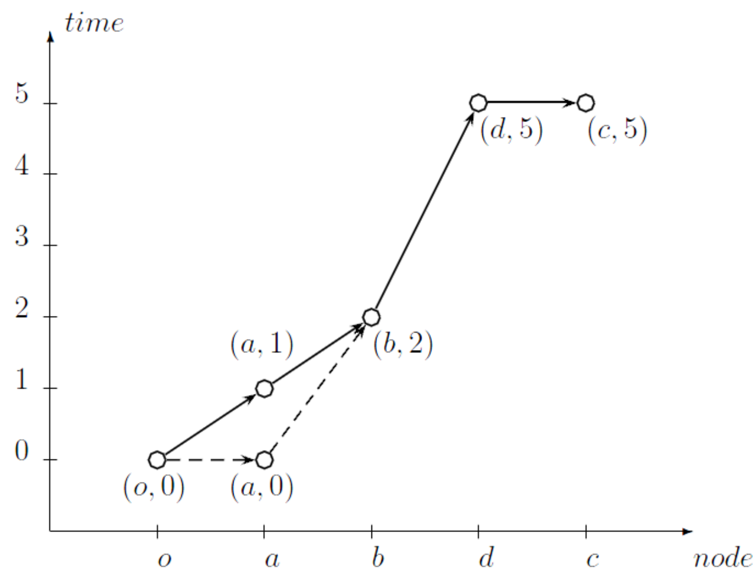


Figure 0.3 Time-Adaptive Routing Policy in Time-Space Domain

On the contrary, a time-adaptive routing policy is defined for a node-time pair, i.e., the decision at a node is a function of the arrival time only, regardless of the trajectory. For example, in Figure 0.1, if travelers arrive at node *b* at time 2, the decision will be unique, even though they may have traversed different trajectories. Figure 0.3 shows a time-adaptive routing policy in a time-space diagram.

In general, an optimal trajectory-adaptive routing policy is no worse than an optimal time adaptive routing policy, since a time-adaptive routing policy is a constrained trajectory-adaptive routing policy. The constraint is that the same next node should be taken out of a given node-time pair regardless of the trajectory. An example can be found by comparing the trajectory-adaptive routing policy in Figure 0.2 and the time-adaptive routing policy in Figure 0.3. A similar example can be found in Pretolani et al. (2009).

As stated and illustrated in CHAPTER 4, Bellman's principle of optimality does not apply to time-adaptive routing where the link travel times are stochastic time-dependent and jointly distributed. A time-adaptive routing policy with minimum expected travel time to the destination may contain a sub-policy with non-minimum expected travel time to the same destination. Good news is that, when the optimality criterion is minimum expected travel time (METT), Bellman's principle of optimality holds for trajectory-adaptive routing policies as defined in Definition 6.2, as shown in the following preposition.

Proposition 0.1. Any sub-policy of a trajectory-adaptive routing policy with METT must be with METT, where the expectation is taken over the support points compatible with the trajectory up to the starting node of the sub-policy.

Proof. Consider the problem of finding the trajectory-adaptive routing policy with METT from node-time pair (j, t) to the destination with an existing trajectory H = {(j_0, t_0), (j_1, t_1), ..., (j, t)}. Let $e_\mu(H)$ be the expected travel time of policy μ, $S_\mu(H, r)$ the support point travel time of policy μ in support point r, and $Pr(r|H)$ the conditional probability that the r-th support point will be realized given trajectory H. By definition,

$$e_\mu(H) = \sum_{r \in EV(H)} S_\mu(H, r) \cdot Pr(r|H)$$

Assume μ^* is an optimal trajectory-adaptive routing policy for this problem and the next node $k = \mu^*(H)$. Denote the i-th support point of the conditional marginal distribution of $\tilde{C}_{jk,t}|H$ as $\tau^i_{jk,t}$. The corresponding trajectory at node k is H'_i with arrival time $t + \tau^i_{jk,t}$. The corresponding sub-policy for trajectory H'_i is λ^*_i.

Assume by contradiction that μ^* has a sub-policy λ^*_1 for trajectory H'_1 that is not with METT, and the policy with METT for trajectory H'_1 is λ_1. Therefore we have $e_{\lambda_1}(H'_1) < e_{\lambda^*_1}(H'_1)$. Construct a trajectory-adaptive routing policy μ for trajectory H such that it shares with μ^* all the sub-policies λ^*_i, for i = 2, 3, ..., except that λ^*_1 is replaced by λ_1. We will then show in the following equations that μ has a lower expected travel time than μ^* for H. Note that $\tau^i_{jk,t}$ is shortened as τ_i and $C^r_{jk,t}$ as C^r, since it is clear that link (j, k) and time t are under discussion.

$$e_\mu(H) = \sum_{r \in EV(H)} S_\mu(H, r) \cdot Pr(r|H)$$

$$= \sum_i Pr(H_i'|H) \cdot \left(\tau_i + \sum_{r \in EV(H_i')} S_{\lambda_i}(H_i', r) \cdot Pr(r|H_i') \right)$$

$$= \sum_i Pr(H_i'|H) \cdot \left(\tau_i + e_{\lambda_i}(H_i') \right)$$

$$< \sum_i Pr(H_i'|H) \cdot \left(\tau_i + e_{\lambda_i^*}(H_i') \right)$$

$$= e_{\mu^*}(H)$$

The first equality is by definition. The second equality is a re-arrangement conditional on the travel time on the next link τ_i – the next trajectory H_i'. The third equality calculates the unconditional expected travel time of a sub-policy λ_i. The fourth inequality is due to the contradiction assumption. The last equality can be derived following the same logics in the first three equalities, but in a reverse order.

This contradicts the assumption that μ^* is with METT for trajectory H. Thus, all the sub-policies λ^*_i are with METT for the corresponding trajectories H'_i. **Q.E.D.**

In time-adaptive routing, the realized travel time on the next link is not included in the information at the next node, and thus the unconditional expected travel time of a sub-policy might be different from that given the next link travel time. This discrepancy could result in the failure of the Bellman's principle. We also further conjecture that trajectory information is a sufficient condition for the principle to hold for the METT routing policy problem in a stochastically dependent network. Examples include the perfect online information studied in Gao and Chabini (2006) and information on outgoing links studied in Polychronopoulos and Tsitsiklis (1996). Formal proofs will be the research topic in the future.

Preposition 6.1 suggests that the optimality conditions in CHAPTER 4 are valid in this case, and the dynamic-programming type algorithm DOT-PART can give an exact solution to the METT trajectory-adaptive routing policy problem. As a matter of fact, Algorithm DOT-PART can be extended to solve MED trajectory-adaptive routing policy problem for an affine or exponential disutility function of the travel time (Eiger et al., 1985). However, generating all the event collections according to trajectory information is conceivably a formidable task, due to the potentially exponential number of trajectories to any given node-time pair. In order to circumvent the curse of dimensionality in state space, a definition of trajectory-adaptive routing policy without the trajectory H in the state variable is given in the next section and used in the remainder of the chapter.

6.2.2 Trajectory Adaptive Routing Policy – Recursive

We firstly give a new definition to trajectory-adaptive routing policy without the trajectory as follows:

Definition 0.3 (Trajectory-Adaptive Routing Policy – Recursive) A trajectory-adaptive routing policy $\mu(j_0, t_0)$ departing node j_0 at time t_0 to a given destination d is recursively defined as a combination of the next node k and the set of sub-policies $\{\mu_i(k, t_i)\}$, where t_i is the i-th possible arrival time at node k from the marginal distribution of $\tilde{C}_{jk,t}$. If denote the i-th support point of the marginal distribution of $\tilde{C}_{jk,t}$ as $\tau^i_{jk,t}$, then $t_i = t_0 + \tau^i_{jk,t}$.

Note that this is a recursive definition. The sub-policies μ_i at all the possible next node-time pairs (k, t_i) are also defined recursively as a combination of the next node k' and sub-policies $\{\mu_i^1(k', t_i^1), \mu_i^2(k', t_i^2), \cdots\}$. The recursion stops at the destination d.

Although each policy is defined over a node-time pair only, the recursive nature allows the routing decisions dependent on the trajectory.

Consider two different possible trajectories to the current node-time pair (j, t) by following a given routing policy out of the origin and departure time (j_0, t_0). Assume they start to differ at (j_i, t_i) due to different arrival times at the next node k, and the next node-time pairs are (k, t_i^1) and (k, t_i^2) respectively. The sub-policies at the two node-time pairs can then be defined such that they will both reach (j, t) with a positive probability but contain different sub-policies from (j, t). This way, the decisions at (j, t) can differ for the two different trajectories.

For example, one trajectory-adaptive routing policy out of node-time pair (o, 0) in Figure 0.1 can be recursively written as follows.

$$\mu(o,0) = \{a; \mu_1(a,1), \mu_2(a,0)\}$$

$$\mu_1(a,1) = \{b; \mu_{1,1}(b,2), \mu_{1,2}(b,3)\}$$

$$\mu_2(a,0) = \{b; \mu_{2,1}(b,1), \mu_{2,2}(b,2)\}$$

$$\mu_{1,1}(b,2) = \{c; \mathrm{STOP}\}$$

$$\mu_{1,2}(b,3) = \{c; \mathrm{STOP}\}$$

$$\mu_{2,1}(b,1) = \{c; \mathrm{STOP}\}$$

$$\mu_{2,2}(b,2) = \{d; \mu_{2,2,1}(d,5)\}$$

$$\mu_{2,2,1}(d,5) = \{c; \mathrm{STOP}\}$$

At node-time pair (b, 2), $\mu_{1,1}$ and $\mu_{2,2}$ are two different routing policies. Which one of the two will be executed depends on the trajectory traveling from the origin and departure time pair (o, 0) to the current one (b, 2): if (a, 1) is on the trajectory, then $\mu_{1,1}$ is the next policy; if (a, 0), then $\mu_{2,2}$.

Under Definition 6.3, a sub-policy by itself does not imply any trajectory information, unlike that under Definition 6.2, where the trajectory information is included in the state variable of the routing policy. For example, one cannot tell from the sub-policy $\mu_{1,1}(b, 2) = \{c; \text{STOP}\}$ which trajectory is followed from the origin and departure time pair (o, 0) to the current one (b, 2); on the other hand, $(o, 0; a, 1; b, 2) \rightarrow c$ can tell us that the trajectory is $H_1 = \{(o, 0), (a, 1), (b, 2)\}$.

A sub-policy under Definition 6.3 treats the current node as the origin and the current time the departure time and gives all possible arrival times at the next node. For example, $\mu_1(a, 1) = \{b; \mu_{1,1}(b, 2), \mu_{1,2}(b, 3)\}$ treats (a, 1) as the origin and departure time pair and gives two possible arrival times at the next node b. However, when retrieving a trajectory-adaptive routing policy from the real origin and departure time, we might encounter the problem that its sub-policy introduces arrival time at downstream node that is not compatible with the trajectory and thus sub-policy that actually is not possible to be realized. For example, when μ is retrieved from the real origin and departure time pair (o, 0), it can be observed that $\mu_{1,2}(b, 3)$ and $\mu_{2,1}(b, 1)$ are never applied, as the arrival times at node b as 3 and 1 are not compatible with the trajectory $\{(o, 0), (a, 1)\}$ and $\{(o, 0), (a, 0)\}$ respectively.

We term anything that is not compatible with the trajectory (information) as "phantom". We call the arrival times at downstream nodes that are not compatible with the trajectory phantom arrival times. The sub-policies that are not possible to be realized due to phantom arrival times at the next nodes are called phantom sub-policies. Note that when two trajectory-adaptive routing policies differ only in phantom sub-policies, they are actually the same. For example, in the above trajectory-adaptive routing policy $\mu(o, 0)$, if we replace $\mu_{1,2}(b,3)$ as $\mu_{1,2}(b,3) = \{d; \mu_{1,2,1}(d,6)\}$ where $\mu_{1,2,1}(d,6) = \{c; \text{STOP}\}$, or replace $\mu_{2,1}(b,1)$ as $\mu_{2,1}(b,1) = \{d; \mu_{2,1,1}(d,4)\}$ where $\mu_{2,1,1}(d,4) = \{c; \text{STOP}\}$, the trajectory-adaptive routing policy $\mu(o, 0)$ is not changed.

Moreover, note that the travel time of a sub-policy is evaluated over all support points, and so the link travel times in some support points that are not possible to be realized due to the phantom arrival times (termed phantom travel times) will still be included in the evaluation of its travel time. However, when the travel time of a trajectory-adaptive routing policy is evaluated, the phantom travel times of its sub-policies will not be considered. For example, when evaluating the travel time of $\mu_1(a,1)$, we will calculate its support point travel times in both support points. However, when the travel time of $\mu(o, 0)$ is evaluated, only the support point travel time of $\mu_1(a,1)$ in support point C^1 is considered as that is the compatible support point.

In the remainder of the chapter, the phantom sub-policies will not be written in the trajectory-adaptive routing policy as they will not affect what the trajectory-adaptive routing policy really is. Thus, the above trajectory-adaptive routing policy $\mu(o, 0)$ is now written as follows.

$$\mu(o,0) = \{a; \mu_1(a,1), \mu_2(a,0)\}$$

$$\mu_1(a,1) = \{b; \mu_{1,1}(b,2), \mu_{1,2}(b,3)\}$$

$$\mu_2(a,0) = \{b; \mu_{2,1}(b,1), \mu_{2,2}(b,2)\}$$

$$\mu_{1,1}(b,2) = \{c; \text{STOP}\}$$

$$\mu_{2,2}(b,2) = \{d; \mu_{2,2,1}(d,5)\}$$

$$\mu_{2,2,1}(d,5) = \{c; \text{STOP}\}$$

It can also be written in a tree format as follows.

$$(o, 0) \to a: (a, 1) \to b: (b, 2) \to c$$

$$(a, 0) \to b: (b, 2) \to d: (d, 5) \to c$$

Next we show that Definitions 6.2 and 6.3 are equivalent.

Proposition 0.2. Definitions 6.2 and 6.3 of Trajectory-Adaptive Routing Policy are equivalent.

Proof. We prove this proposition by showing that any trajectory-adaptive routing policy under Definition 6.3 can be converted to one under Definition 6.2, and vice versa.

Suppose a trajectory-adaptive routing policy μ is defined under Definition 6.3, i.e., $\mu_i(j, t_i) = \left\{j_i; \left\{\mu_{i,j}\left(j_i, t_{i,j}\right)\right\}\right\}$. Assume ($j_0$, t_0) is the origin node and departure time. Therefore the trajectory-adaptive routing policy μ is recursively defined as follows (assume that phantom sub-policies are not included).

$$\mu(j_0, t_0) = \{j; \mu_1(j, t_1), \mu_2(j, t_2), \cdots\}$$

$$\mu_1(j, t_1) = \left\{j_1; \mu_{1,1}\left(j_1, t_{1,1}\right), \mu_{1,2}\left(j_1, t_{1,2}\right), \cdots\right\}$$

$$\mu_2(j, t_2) = \left\{j_2; \mu_{2,1}\left(j_2, t_{2,1}\right), \mu_{2,2}\left(j_2, t_{2,2}\right), \cdots\right\}$$

$$\cdots$$

$$\mu_{1,1}\left(j_1, t_{1,1}\right) = \left\{j_{1,1}; \mu_{1,1,1}\left(j_{1,1}, t_{1,1,1}\right), \mu_{1,1,2}\left(j_{1,1}, t_{1,1,2}\right), \cdots\right\}$$

$$\cdots$$

$$\mu_{2,1}\left(j_2, t_{2,1}\right) = \left\{j_{2,1}; \mu_{2,1,1}\left(j_{2,1}, t_{2,1,1}\right), \mu_{2,1,2}\left(j_{2,1}, t_{2,1,2}\right), \cdots\right\}$$

$$\cdots$$

$$\mu_i(d, t_i) = \{d; \text{STOP}\}, \forall \mu_i, t_i$$

This trajectory-adaptive routing policy μ under Definition 6.3 can be converted to one under Definition 6.2, i.e., a mapping from trajectories to next nodes, by combining node-time pairs starting from (j_0, t_0) as a trajectory and making the next node of the sub-policy for the last node-time pair as the next node corresponding to the trajectory. Since the phantom sub-policies are not included, all the trajectories generated are valid.

$$\{j_0, t_0\} \rightarrow j$$

$$\{(j_0, t_0), (j, t_1)\} \rightarrow j_1$$

$$\{(j_0, t_0), (j, t_2)\} \rightarrow j_2$$

$$\ldots$$

$$\{(j_0, t_0), (j, t_1), (j_1, t_{1,1})\} \rightarrow j_{1,1}$$

$$\ldots$$

$$\{(j_0, t_0), (j, t_1), (j_2, t_{2,1})\} \rightarrow j_{2,1}$$

$$\ldots$$

$$\{(j_0, t_0), \cdots (d, t_i)\} \rightarrow \text{STOP}, \forall H$$

We have shown that any trajectory-adaptive routing policy under Definition 6.3 can be converted to one under Definition 6.2, and next we will show the other way around.

Suppose a trajectory-adaptive routing policy μ is defined under Definition 6.2, i.e., μ is a mapping from trajectories to next nodes, μ: $H \rightarrow k, k \in \Gamma(j)$. The conversion can be conducted as follows.

For each trajectory H = {(j_0, t_0), (j_1, t_1), ..., (j, t)}, choose the last node-time pair (j, t) as the node-time pair for the current routing policy, and the next node k corresponding to the trajectory H as the next node of the sub-policy out of the current node-time pair (j,

t). Find all trajectories $H' = \{(j_0, t_0), (j_1, t_1), ..., (j, t), (k, t_i)\}$, where t_i is a possible arrival time at the next node k from the trajectory H, and choose the last node-time pairs (k, t_i) as the node-time pair for the sub-policies of the current routing policy. Thus, we have $\mu(j, t) = \{k; \{\mu_i(k, t_i)\}\}$.

When all the trajectories are visited, the recursive definition of the trajectory-adaptive routing policy for the origin and departure time pair (j_0, t_0) is complete, and no phantom sub-policy is included. **Q.E.D.**

An example of the equivalence is that the example trajectory-adaptive routing policies in Sections 6.2.1 and 6.2.2 are the same – both are the one represented by Figure 0.2.

6.2.3 Optimal Trajectory Adaptive Routing Policy

Similar to CHAPTER 5, the trajectory-adaptive routing policies are evaluated by a disutility function of travel time, which can be either linear or non-liner and is increasing with travel time. The calculations of support point travel time / disutility and expected travel time / disutility are similar to those shown by the equations in Section 5.2.1 and not listed here.

In this chapter, we define the trajectory-adaptive routing policies with minimum expected disutility (MED) as optimal trajectory-adaptive routing policies, and the goal is to find the optimal trajectory-adaptive routing policies from all origins to a given destination for all departure times. Note that, if the disutility is the travel time itself, we are seeking the trajectory-adaptive routing policies with minimum expected travel time (METT).

Definition 0.4 (Trajectory-Adaptive Routing Policy with MED for Departure Time *t*). A trajectory-adaptive routing policy λ with MED from origin *O* to destination *D* for departure time *t* has the minimum expected disutility evaluated over all support points among all the trajectory-adaptive routing policies between the same OD pair and for the same departure time, i.e., $\nexists$ trajectory-adaptive routing policy μ such that $d_\mu(O,t) < d_\lambda(O,t)$.

In the example network of Figure 0.1, suppose the disutility function is the travel time itself, i.e., we are looking for routing policies with METT, it can be observed that the optimal trajectory-adaptive routing policy is the one represented by Figure 0.2, i.e.,

$$(o,0) \to a: (a,1) \to b: (b,2) \to c$$

$$(a,0) \to b: (b,2) \to d: (d,5) \to c$$

6.2.4 Pure Trajectory Adaptive Routing Policy

In this section, we follow the procedure of Section 5.2.2 . We first show that Bellman's principle of optimality (Bellman, 1958) that any sub-policy of an optimal routing policy must also be an optimal routing policy itself is no longer valid in our problem context (Proposition 6.3). We then show that Bellman's principle of non-dominance that any sub-policy of a non-dominated routing policy must also be a non-dominated routing policy itself is not valid either (Proposition 6.4). We further define a subset of the non-dominated routing policies as pure routing policies, and purity is a property that can be maintained across routing policy and sub-policy. It is then proved (Theorem 6.1) that for any origin node, there always exists a pure routing policy with MED, and an exact algorithm can be designed based on this property.

Proposition 0.3. A sub-policy of a trajectory-adaptive routing policy with MED for a departure time is not necessarily with MED for the arrival (exit) time at the intermediate node (i.e., the starting node of the sub-policy).

Proof. We prove this proposition by an example. When the disutility function is the travel time itself, the optimal trajectory-adaptive routing policy for the origin and departure time pair (o, 0) in Figure 0.1 is as follows:

$$(o,0) \rightarrow a\colon (a,1) \rightarrow b\colon (b,2) \rightarrow c$$

$$(a,0) \rightarrow b\colon (b,2) \rightarrow d\colon (d,5) \rightarrow c$$

However, the sub-policy $\mu_{1,1}(b,2)$, i.e., $(b,2) \rightarrow c$, is not optimal as its expected travel time (over both support points) is larger than that of $b\colon (b,2) \rightarrow d\colon (d,5) \rightarrow c$. **Q.E.D.**

The key is the phantom travel times. When evaluating a routing policy out of the origin, those phantom travel times will not be considered. However, when evaluating a sub-policy out of an intermediate node, since we treat the current node as the origin and the current time the departure time, the phantom travel times will be included in the calculation of the expected disutility where the expectation is taken over all support points, ignoring the fact that there are phantom travel times in some support points.

As a matter of fact, if we do not include phantom travel times, i.e., if we consider only those support points that are compatible with the trajectory information, that is, if we define the routing policy as Definition 6.2 and define the sub-policy out of an intermediate state (j, t, H) instead of the node-time pair only, the sub-policy will also be optimal itself, i.e., Bellman's principle of optimality is valid, as presented in Proposition 6.1.

Similarly to CHAPTER 5, non-dominated routing policy is defined with the hope of finding another property that can be maintained in the way optimality is maintained from a routing policy to all its sub-policies in Bellman's principle of optimality. Unfortunately, the hope evaporates with the fact that a sub-policy of a non-dominated routing policy is not necessarily non-dominated. The reason is similar to that why Bellman's principle of optimality does not hold for trajectory-adaptive routing policy. However, good news is that pure routing policy can be defined based on non-dominated routing policy and it can be proved that for any origin-departure-time pair (j, t), there always exists an optimal routing policy which is pure.

Definition 0.5 (Non-Dominated Routing Policy w.r.t. Support Point Set B). A trajectory-adaptive routing policy λ at origin j with departure time t is non-dominated w.r.t. support point set B iff $\nexists$ routing policy μ such that

$D_\mu(j,t,r) \leq D_\lambda(j,t,r), \forall r \in B$ and

$\exists\ r^0 \in B$ such that $D_\mu(j,t^0,r^0) < D_\lambda(j,t^0,r^0)$.

If not specified, in the remainder of this chapter, B is the set of all support points.

Note that, since the disutility function is increasing in travel time and joint distribution is utilized as complete dependencies are considered, non-dominance in terms of distuility is equivalent to non-dominance in terms of travel time. Thus, the $D_\mu(j,t,r)$ terms in Definition 6.5 can be changed to $S_\mu(j,t,r)$ terms.

Proposition 0.4. A sub-policy of a non-dominated trajectory-adaptive routing policy is not necessarily non-dominated.

Proof. We prove this proposition by an example. Consider the following two routing policies departing node a at time 0 in Figure 0.1, both of which are optimal (suppose the disutility function is the travel time itself):

Policy 1: $(a, 0) \to b$: $(b, 1) \to c$

$(b, 2) \to d$: $(d, 5) \to c$

Policy 2: $(a, 0) \to b$: $(b, 1) \to d$: $(d, 4) \to c$

$(b, 2) \to d$: $(d, 5) \to c$

The support point travel times of both routing policies are calculated as follows: $S_1(a, 0, C^1) = 4$, $S_1(a, 0, C^2) = 5$; $S_2(a, 0, C^1) = 4$, $S_2(a, 0, C^2) = 5$. Thus both are non-dominated.

However, it can be observed that the sub-policy of Policy 1 $(b, 1) \to c$ is dominated by the sub-policy of Policy 2 $(b, 1) \to d$: $(d, 4) \to c$. The support point travel times of both routing policies are calculated as follows: $S_{b\text{-}c}(b, 1, C^1) = 3$, $S_{b\text{-}c}(b, 1, C^2) = M$; $S_{b\text{-}d\text{-}c}(b, 1, C^1) = 3$, $S_{b\text{-}d\text{-}c}(b, 1, C^2) = 3$.

Both routing policy 1 and 2 departing node a at time 0 are non-dominated but policy 1 contains a dominated sub-policy which departs node b at time 1. **Q.E.D.**

The key is again the phantom travel times. Non-dominance of a routing policy is defined for a given node-time pair over all support points. The sub-policy is taken at a downstream node-time pair. The specific arrival time at the downstream node already implies that only a subset of support points is possible to be realized. Ignoring this fact results in the violation of Bellman's principle. It is trivial to show that non-dominance can be maintained at any intermediate state (j, t, H). However, for the recursively defined trajectory-adaptive routing policy, non-dominance is checked at the intermediate node-time pair (j, t), i.e., w.r.t. the complete set of support points. A sub-policy μ at the intermediate node-time pair (j, t) of a non-dominated policy from the origin and departure time pair (j_0, t_0) could be dominated in such a way that it has an equal travel time as sub-policy μ' for each support point compatible with the trajectory H from (j_0, t_0) to (j, t), but is dominated by μ' in the set of support points that are not compatible with the trajectory, and thus dominated by μ' w.r.t. the complete set of support points.

Fortunately we find out that a property related to non-dominance satisfy Bellman's principle for the complete set of support points as described next.

Definition 0.6 (Pure Trajectory-Adaptive Routing Policy). A trajectory-adaptive routing policy is pure iff the trajectory-adaptive routing policy itself and all its sub-policies are non-dominated w.r.t. the complete set of support points; otherwise, it is a mixed trajectory-adaptive routing policy.

Unlike non-dominated trajectory-adaptive routing policy, pure trajectory-adaptive routing policy has the property that any sub-policy of a pure trajectory-adaptive routing policy must be pure by definition, i.e., Bellman's principle holds for this property. Moreover, the following proposition and theorem guarantee that there must be a pure optimal trajectory-adaptive routing policy.

Proposition 0.5. For any mixed trajectory-adaptive routing policy μ from origin and departure time (j, t) to destination d, there exists a pure trajectory-adaptive routing policy λ such that $D_\lambda(j,t,r) \leq D_\mu(j,t,r), \forall r$.

Proof. We prove the proposition by induction.

Basis. At time $t \geq K{-}1$, link travel times become static and deterministic. Routing policies collapse to paths. Any mixed path must be dominated by an optimal path which is pure (**Proposition 0.3**).

Inductive step. Suppose Proposition 6.5 holds at any time $t \geq \tau + 1$. Consider a mixed routing policy μ at $t = \tau$ and node j. If μ is dominated, denote the non-dominated routing policy that dominates μ as γ, and γ can be either pure or mixed. If μ is non-dominated, set $\gamma = \mu$, and then γ is mixed non-dominated. Therefore, $D_\gamma(j,\tau,r) \leq D_\mu(j,\tau,r), \forall r$.

Now consider the non-dominated routing policy γ.

Case 1: γ is pure. Set $\lambda = \gamma$, so $D_\lambda(j,\tau,r) \leq D_\mu(j,\tau,r), \forall r$.

Case 2: γ is mixed. Denote the next node as k. If the sub-policy γ' from node k to the destination is mixed, then there must exist a pure routing policy λ' such that $D_{\lambda'}\left(k,\tau + C^r_{jk,\tau},r\right) \leq D_{\gamma'}\left(k,\tau + C^r_{jk,\tau},r\right), \forall r$ according to the inductive assumption that Proposition 6.5 holds at any time $t \geq \tau + 1$. Note that $\tau + C^r_{jk,\tau} \geq \tau + 1$ due to the positive and integer travel time assumption. The disutility function is an increasing function of travel time, so $S_{\lambda'}\left(k,\tau + C^r_{jk,\tau},r\right) \leq S_{\gamma'}\left(k,\tau + C^r_{jk,\tau},r\right), \forall r$. Then construct a routing policy λ from origin node j to destination d by replacing the dominated sub-policy γ' of the mixed non-dominated routing policy γ with the pure sub-policy λ'. Then for the resulting routing policy λ, we have the following: $S_\lambda(j,\tau,r) = C^r_{jk,\tau} + S_{\lambda'}\left(k,\tau + C^r_{jk,\tau},r\right) \leq C^r_{jk,\tau} + S_{\gamma'}\left(k,\tau + C^r_{jk,\tau},r\right) = S_\gamma(j,\tau,r), \forall r$. The disutility function is an increasing function of travel time, so $D_\lambda(j,\tau,r) \leq D_\gamma(j,\tau,r) \leq D_\mu(j,\tau,r), \forall r$.

Since γ is non-dominated, the newly constructed routing policy λ is also non-dominated. Furthermore, the sub-policy of λ is pure, so λ is pure and Proposition 6.5 is true at time τ.

With the basis and inductive step, Proposition 6.5 holds $\forall t$. **Q.E.D.**

A straightforward conclusion can be drawn that, if a mixed routing policy has MED, then there must exist a pure routing policy with the same MED.

Theorem 0.1 (Pure Optimal Trajectory-Adaptive Routing Policy). For any origin *j* and departure time *t*, there exists a pure trajectory-adaptive routing policy with MED.

Proof. Assume by contradiction that all optimalouting policies are mixed. According to Preposition there exists a pure routing policy whose expected disutility is no larger than that of the optimal mixed routing policy. Therefore this pure routing policy must also be optimal. **Q.E.D.**

Definition 6.6 and Theorem 6.1 show the two most important properties of the pure routing policies: any sub-policy of a pure routing policy must be pure, and it is guaranteed that there is a pure optimal routing policy. Therefore we can construct a pure routing policy based on downstream pure routing policies, and, as long as we find all pure routing policies, we can find the pure optimal one(s). Moreover, due to the equivalence between the non-dominance in terms of disutility and that in terms of travel time, the set of pure routing policies is the same for any disutility function as long as it is increasing with travel time, i.e., for any type of users, no matter whether they are risk-averse or risk-seeking, assuming their risk attitudes can be described by the expected utility theory (EUT). However, the final optimal one(s) is potentially different for users with different risk attitudes.

6.3 Algorithm DOT-CD-Traj

6.3.1 Solution Approach

Algorithm DOT-CD-Traj is designed based on the concept of decreasing order of time (DOT). Note that the construction of routing policies at time t involves only routing policies at times later than t, due to the assumption of positive link travel times.

At time $K-1$ or beyond, the network becomes deterministic and static, and, for any node-time pair (j, t) where $t \geq K-1$, the set of pure routing policies denoted as $\chi(j, t)$ contains only one policy (the shortest path). Any deterministic static shortest path algorithm can be employed to compute the policy. Then the set of pure routing policies at time $K-1$ at any node is complete, i.e., no routing policy in the set will become mixed and no new pure routing policies will be constructed from later operations. Therefore the set of pure routing policies at time $K-2$ constructed from pure sub-policies at time $t \geq K-1$ will also be complete. This procedure is continued down to time 0, and pure routing policy sets at all times will be constructed with one pass along the time dimension.

Two pointers are required for each routing policy λ at each node j and departure time t to store the pure routing policies: $\pi_\lambda(j, t)$, indicating the next node; and $L_\lambda(j, t, t')$, indicating the sub-policy out of the next node at time t', where t' is a possible arrival time at the next node if the traversal of the next link starts at time t.

For each link (j, k) in the network, treat node j as the current node and k the next node on the routing policy. Starting from time $t = K - 1$ down to time 0, treat node-time pair (j, t) as the origin node and departure time pair for the newly constructed routing policy λ. Let $\pi_\lambda(j, t) = k$. For each possible arrival time $t'_i = t + \tau^i_{jk,t}$ at the next node k, where $\tau^i_{jk,t}$ is the i-th support point of the marginal distribution of $\tilde{C}_{jk,t}$, choose one routing policy from the set of pure routing policies $\chi(k, t'_i)$ and make it the sub-policy for the next node and arrival time pair (k, t'_i). Let P_i denote the number of pure routing policies in $\chi(k, t'_i)$, then the number of possible new routing policies for the current node-time pair (j, t) is $Q = \prod_i P_i$. After the non-dominance is checked among the Q newly generated routing policies, those which are not discarded are the pure routing policies for node-time pair (j, t) and maintained in the pure routing policy set $\chi(j, t)$.

6.3.2 Algorithm Statement

The steps of Algorithm DOT-CD-Traj are described next:

Algorithm DOT-CD-Traj

Step 1: Deterministic and Static Period

$t = K - 1$.

For each $j \in N \backslash \{D\}$

Compute $\mu^*(j, K - 1)$ with a static deterministic shortest path algorithm.

Compute $S_{\mu^*}(j, K - 1, r)$, $\forall r$; set $S_{\mu^*}(j, t > K - 1, r) = S_{\mu^*}(j, K - 1, r)$

Suppose the next node on $\mu^*(j, K - 1)$ is k, then

$\chi(j, t) = \{\mu^*(j, K - 1)\}$, $\pi_{\mu^*}(j, K - 1) = k$, $L_{\mu^*}(j, K - 1, K - 1) = \mu^*(j, K - 1)$

Step 2: Stochastic and Dynamic Periods

For $t = K - 2$ down to 0

For each link $(j, k) \in A$

For $q = 1$ to Q

Find the corresponding indices p_i of the sub-policies in their respective sets $\chi(k, t'_i)$.

Construct a new routing policy λ as follows:

$\pi_\lambda(j, t) = k$, $L_\lambda(j, t, t'_i) = p_i$, $\forall i$.

Calculate $S_\lambda(j, t, r), \forall r$ by the following equation:

$$S_\lambda(j, t, r) = C^r_{jk,t} + S_{p_i}\left(k, t + C^r_{jk,t}, r\right)$$

Note that i (support point of the marginal distribution of $\tilde{C}_{jk,t}$) and r (support point of the joint distribution) must be compatible.

Add λ to $\chi(j, t)$ and check dominance among the set. Remove dominated routing policies from $\chi(j, t)$.

Step 3: Stop and Find the Routing Policies with MED

For each node-time pair (j, t)

Calculate the expected disutility for each pure routing policy in $\chi(j, t)$ and identify the one with MED.

Algorithm DOT-CD-Traj will find all pure routing policies upon termination and thus will find the optimal pure routing policies. However it will miss the mixed non-dominated routing policies and thus the optimal mixed routing policies.

Note that Algorithm DOT-CD-Traj finds pure routing policies using support point travel times rather than support point disutilities due to the equivalence of these two.

The algorithm terminates after a finite number of steps, yet the worst-case complexity is exponential, and so heuristics might be needed to work more efficiently.

The proofs to the above facts of the algorithm are similar to those in CHAPTER 5 and are not given here.

6.4 Computational Tests

Algorithm DOT-CD-Traj is coded using C++ and tested on a Windows Vista Business (64 bit) workstation with Intel Core i5 CPU 650 @ 3.20GHz and 8.00GB RAM.

The computational tests are conducted on step networks, as described in Section 5.4.1. The objectives of the computational tests are to: 1) investigate the average running time of Algorithm DOT-CD-Traj as a function of the network size in step networks; 2) investigate the average size of pure routing policy set as a function of network size in step networks; 3) study computationally how the risk aversion efficient affects the optimal routing policy solution; and 4) study computationally how the level of stochastic dependencies affects the optimal routing policy solution.

The tests are conducted on step networks of levels from 3 to 10 with 30 time periods. The first freeway node is set as the origin and the last freeway node the destination. Travel times on freeway links and on-ramp links are sampled from truncated multivariate normal distribution, where the original multivariate normal distribution has 3 as the uniform mean, 4 the uniform variance, and a uniform correlation coefficient varying from 0 to 1, and the sample is truncated at 3. The positive uniform correlation coefficient ensures that the covariance matrix is positive-semidefinite, and thus its validity. There are 50 support points for freeway link and on-ramp link travel times. Travel times on local links are fixed as 3. For each combination of network level and correlation coefficient, 10 networks are randomly generated. The results shown are the averages over the 10 networks for each parameter combination.

An exponential disutility function of travel time is applied, i.e., $D_\lambda(O,t,r) = D\big(S_\lambda(O,t,r)\big) = \exp(\alpha \cdot S_\lambda(O,t,r))$.

Table 0.1 shows the average running time of Algorithm DOT-CD-Traj. Note that the algorithm finds optimal routing policies from all nodes to the destination. The table shows that the average running time of Algorithm DOT-CD-Traj is growing exponentially with the network size. The regression function is RUN = $0.3528 \cdot e^{0.8614n}$ (R^2 = 0.9892), where RUN is the average running time over all tested correlation coefficients and *n* is the step network level. Note that this result is related to the special setting of step network.

Table 0.1 Average Running Time vs. Network Level

	Network Level			
$\boldsymbol{\rho}$	3	5	7	10
0	5.109171	19.521787	260.516811	1619.724861
0.2	5.084598	19.576821	283.035162	2651.01106
0.4	5.083323	19.364403	232.497614	3556.979064
0.6	5.110512	19.497106	210.527767	2278.314268
0.8	5.04645	18.969142	138.495415	433.254619
1.0	5.094363	18.127895	87.613568	63.372237
Avg.	5.0880695	19.17619233	202.1143895	1767.109352

Table 0.2 Average Size of Pure Routing Policy Set vs. Network Level

	Network Level			
$\boldsymbol{\rho}$	3	5	7	10
0	18	112.5	582	1878.4
0.2	18	110	605.2	2468.8
0.4	18	104.8	527	2899
0.6	18	102.3	486.4	2298.8
0.8	18	93.2	322	865.2
1.0	14	32	43.6	54.4
Avg.	17.33333	92.46667	427.7	1744.1

Table 0.2 shows the average size of the pure path set for the origin node and departure time 0. The table shows that the average size of the pure routing policy set for the origin node and departure time 0 grows exponentially with the network size. The regression function is SIZE = $3.0415 \cdot e^{0.6581n}$ (R^2 = 0.9799) respectively, where SIZE is the average size of the pure routing policy set for the origin node and departure time 0 over all tested correlation coefficients, and n is the step network level. Note that the results are related to the special setting of step network.

With the computational test results for the optimal trajectory-adaptive routing policy problem and those for the optimal *a priori* path problem, we can compare them to investigate the benefit of being adaptive to trajectory information. Table 0.3 shows how the benefit of being adaptive to trajectory information is affected by the risk aversion efficient (i.e., the value of α in the exponential disutility function of travel time) and the level of stochastic dependencies (i.e., the uniform correlation coefficient of the truncated multivariate normal distribution for the link travel time random variables). The benefit is presented by the ratio between the expected disutility of the optimal trajectory-adaptive routing policy for the origin node and departure time 0 and that of the optimal path for the same origin node and departure time pair.

Table 0.3 Benefit of Being Adaptive to Trajectory Information

	Network Level 3							
$\rho \setminus \alpha$	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
0	0.9108	0.7893	0.6673	0.5623	0.4758	0.4036	0.3421	0.2888
0.2	0.9062	0.7615	0.6090	0.4752	0.3652	0.2780	0.2106	0.1593
0.4	0.8785	0.6527	0.4216	0.2482	0.1363	0.2695	0.2042	0.1545
0.6	0.6702	0.4979	0.3217	0.1893	0.1040	0.2056	0.1557	0.1178
0.8	0.5513	0.4096	0.2646	0.1557	0.0855	0.1691	0.1281	0.0969
1.0	0.4536	0.3370	0.2177	0.1281	0.0704	0.1391	0.1054	0.0797
	Network Level 5							
$\rho \setminus \alpha$	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
0	0.8735	0.6128	0.3970	0.2632	0.1920	0.1566	0.1393	0.1307
0.2	0.8150	0.5718	0.3704	0.2455	0.1791	0.1461	0.1299	0.1220
0.4	0.6009	0.4215	0.2731	0.1810	0.1321	0.1077	0.0958	0.0899
0.6	0.7177	0.2293	0.1485	0.0985	0.0718	0.0586	0.0521	0.0489
0.8	0.5069	0.1619	0.1049	0.0695	0.0507	0.0414	0.0368	0.0345
1.0	0.3580	0.1144	0.0741	0.0491	0.0358	0.0292	0.0260	0.0244
	Network Level 7							
$\rho \setminus \alpha$	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
0	0.6915	0.3576	0.2259	0.1270	0.0769	0.0469	0.0287	0.0176
0.2	0.7188	0.4194	0.2086	0.0977	0.0453	0.0211	0.0098	0.0046
0.4	0.6765	0.3947	0.1963	0.0919	0.0426	0.0198	0.0093	0.0043
0.6	0.6367	0.3715	0.1847	0.0865	0.0401	0.0187	0.0087	0.0041
0.8	0.5993	0.3496	0.1739	0.0814	0.0378	0.0176	0.0082	0.0038
1.0	0.5640	0.3291	0.1637	0.0766	0.0355	0.0165	0.0077	0.0036
	Network Level 10							
$\rho \setminus \alpha$	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
0	0.6380	0.3567	0.2109	0.1328	0.0846	0.0535	0.0335	0.0207
0.2	0.5138	0.2873	0.1699	0.1069	0.0682	0.0431	0.0269	0.0167
0.4	0.4137	0.2313	0.1368	0.0861	0.0549	0.0347	0.0217	0.0134
0.6	0.3332	0.1863	0.1101	0.0693	0.0442	0.0280	0.0175	0.0108
0.8	0.2683	0.1500	0.0887	0.0558	0.0356	0.0225	0.0141	0.0087
1.0	0.2161	0.1208	0.0714	0.0450	0.0287	0.0181	0.0113	0.0070

It can be observed that the benefit of being adaptive to the trajectory information increases (as the ratio decreases) with the traveler's risk-aversion (α), the correlation (ρ) and the network size (n). When a traveler is more risk-averse, he/she would like to be more adaptive to avoid the risk and gain more benefit from being adaptive. When trip travel time variance, which increases with the correlation, is larger, a traveler tends to be more adaptive in order to avoid large travel times and thus the benefit of being adaptive is also larger. When the network is larger, the risk of the trip increases, so the traveler would like to be more adaptive to neutralize the negative impact of the risk.

6.5 Conclusions and Future Directions

This chapter addresses the optimal trajectory-adaptive routing policy problem in a stochastic time-dependent network where all link travel times are temporally and spatially correlated. It is shown that, in such a network, Bellman's principle does not hold if the optimality or non-dominance is defined w.r.t. the complete set of support points for the routing policy and its sub-policies. A property related to non-dominance is found to satisfy Bellman's principle for the complete set, and it is proved that, for any origin node, there always exists a pure optimal routing policy. An exact algorithm is designed to find all the pure routing policies and thus the optimal ones, and the computational tests show that the average running time of Algorithm DOT-CD-Traj and the average size of the pure routing policy set are growing exponentially with the network size in a step network with properly defined stochastic links. Computational tests also show that the benefit of being adaptive to trajectory information is larger with a higher risk-aversion, a higher correlation and a larger network.

There remains much work to do in the future. First of all, a formal proof is needed for the conjecture that trajectory information is a sufficient condition for Bellman's principle of optimality to hold for the METT routing policy problem in a stochastically dependent network. Note that trajectory information is the least amount of information a traveler can obtain *en route* even without any external traveler information resource.

In a real-life network, it is impossible to treat every node in the network as a decision node for adaptive routing. A subset of the nodes is selected to be decision nodes, and adaptive routing can only be made at those nodes, and not on others. The routing between the decision nodes is just path. This kind of hybrid routing can allow us to solve the optimal adaptive routing problem in a real-life network with manageable running time and memory usage.

CHAPTER 7

CONCLUSIONS AND FUTURE DIRECTIONS

7.1 Research Summary

Congestion on roadways and the high level of uncertainty of travel times are major considerations for trip planning. In CHAPTER 3, traffic data from an urban freeway segment are obtained from the PeMS database and analyzed to study the characteristics of stochastic dependencies among link travel times. Spatial and temporal Pearson's correlation coefficients among traffic variables over five consecutive road links during peak and off-peak periods are obtained. A correlation prediction model is created by performing a linear regression on the observed data. The negative parameters of time and distance show that temporal and spatial distances reduce correlations. The positive parameters of the spatial and temporal distances interaction terms show that the reduction rate along the temporal (spatial) dimension slows down with farther temporal (spatial) distance. The sensitivity analysis shows that highway shares are lower when dependency is taken into account compared to models excluding correlations, and are higher when correlations and/or travelers' risk aversion are lower. This chapter sheds light on the necessity of considering link correlations in evaluating trip travel time reliability.

Real-time information is important for travelers' routing decisions in uncertain networks by enabling online adaptation to revealed traffic conditions. Usually there are spatial and/or temporal limitations in traveler information. In CHAPTER 4, a generic description of online information is provided based on which three types of partial online information and one no online information schemes are derived. A theoretical analysis shows that more error-free information is always better (or at least not worse) for optimal adaptive routing in flow-independent networks. For the empirical evaluation of information benefit in a general network, a heuristic algorithm is designed for the optimal adaptive routing problem with the three partial and no online information schemes, based on a set of necessary conditions for optimality. The effectiveness of the heuristic is shown to be satisfactory over the tested random networks. This chapter is potentially of interest for traveler information system evaluation and design.

CHAPTER 5 and CHAPTER 6 study the problem of finding the optimal *a priori* paths and the optimal trajectory-adaptive routing policies in a stochastic network. It is shown that stochastic dependencies are required to be considered in such routing problems, as whether it is considered or not will affect the optimal solutions. It is also shown that when the traveler is more risk-averse, when link travel times in the network are more correlated, and when the network is larger, being adaptive to trajectory information can gain the traveler more benefit in terms of minimizing the expected disutility of travel time.

7.2 Future Research Plan

The thesis shows that correlations exist in stochastic networks and how correlations and information affect travelers' routing. However, besides the future directions discussed in each chapter, there are more questions not answered yet:

- From the analysis in CHAPTER 3, we see that there are negative correlations on downstream links at near-peak periods, and it is shown that a linear regression model on correlation coefficients is not so realistic, as it can only reflect short-distance case. The questions that need to be answered are: what do the negative correlations tell us? And what would a more realistic regression model on correlations be?

In order to answer these questions, more traffic data are to be obtained and more in-depth study is to be carried out to investigate the reason negative correlations exist. Non-linear regression models with the shapes in Figure 0.6 are to be created and it is to be determined whether non-linear regression models can perform better. With more intensive analysis on correlations, we can get a better understanding of stochastic dependencies among traffic variables.

- An effective routing algorithm with realistic assumptions on network stochastic dependencies is not yet designed, and the question is not yet answered how far off a routing strategy will be if stochastic dependencies are ignored, compared with a more realistic case where they are taken into account, e.g., where the regression models (linear or non-linear) on correlation coefficients are applied.

In order to answer this question, an efficient routing algorithm with realistic assumptions on network stochastic dependencies is to be designed. Theoretical and computational analyses of the developed algorithms will be conducted in hypothetical and real-life networks to investigate whether the consideration of stochastic dependencies significantly increase the algorithm average running time and also to show the impact of correlations on routing in stochastic networks.

- CHAPTER 4 studies imperfect information schemes with spatial or temporal limitations (delayed, pre-trip, radio, and no online information). CHAPTER 6 studies the case where the lease amount of information is considered. There are other interesting information schemes. For example, VMS is one of the most common types of ATIS. The problem with VMS is more involved than those discussed in CHAPTER 4 and CHAPTER 6, as the information is trajectory-based rather time-based only and it contains more information than trajectory. This could significantly complicate the algorithm design. Bellman’s principle of optimality is shown invalid in the three partial information schemes and no online information scheme in CHAPTER 4 and valid when trajectory information is included. However, whether it holds and how it works for VMS case is to be confirmed.

In order to answer these questions, trajectory-adaptive routing and adaptive routing under other information scheme are to be combined. Whether Bellman’s principle of optimality holds in this case is to be determined through theoretical analysis. An efficient algorithm is to be developed and its performance is to be analyzed through theoretical and computational tests. Based on trajectory-adaptive routing analysis, VMS information scheme is to be derived and an efficient algorithm is to be designed. Theoretical and computational analyses are to be carried out to study the optimal routing in VMS information scheme.

BIBLIOGRAPHY

Ahuja, R. K., Magnanti, T. L. and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms and Applications*, Prentice-Hall, Englewood Cliffs, NJ.

Arnott, R., de Palma, A., Lindsey, R. (1991). Does providing information to drivers reduce traffic congestions? *Transportation Research Part A* 25(5):309–318.

Arnott, R., de Palma, A., Lindsey, R. (1996). Information and usage of free-access congestible facilities with stochastic capacity and demand. *International Economic Review* 37(1): 181-203.

Arnott, R., de Palma, A., Lindsey, R. (1999). Information and time-of-usage decisions in the bottleneck model with stochastic capacity and demand. *European Economic Review* 43: 525-548.

Bander, J. and White, C. C. III (2002). A heuristic search approach for a nonstationary stochastic shortest path problem with terminal cost. *Transportation Science* 36(2):218–230.

Ben-Akiva, M. E., Bierlaire, M., Burton D., Koutsopoulos, H. N., Mishalani, R. (2001). Network state estimation and prediction for real-time traffic management. *Networks and Spatial Economics* 1(3/4): 293-318.

Ben-Akiva, M., de Palma, A., Kaysi, I. (1991). Dynamic network models and driver information systems. *Transportation Research Part A* 25(5): 251-266.

Bottom, J. A. (2000). Consistent anticipatory route guidance. PhD Dissertation, Center for Transportation Studies, Massachusetts Institute of Technology, Cambridge, MA, USA.

Bovy, P. H. and Van der Zijpp, N. J. (1999). The Close Connection between Dynamic Traffic Management and Driver Information Systems. In Richard Emmerink and Peter Nijkamp, editors, *Behavioural and Network Impacts of Driver Information Systems*: 355-370. Ashgate Publishing Company.

Boyles, S. D. (2006). Reliable routing with recourse in stochastic, time-dependent transportation networks. Master's Thesis. The University of Texas, Austin, TX, USA.

Chabini, I. (1998). Discrete dynamic shortest path problems in transportation applications: complexity and algorithms with optimal run time. *Transportation Research Record* 1645, pp. 170-175.

Chabini, I. (2000). Minimum expected travel times in stochastic time-dependent networks revisited. Internal Report. MIT, Cambridge, MA.

Chorus, C. G., Arentze, T. A., Molin, E. J. E., Timmermans, H. J. P., van Wee, B. (2006). The value of travel information: Decision strategy-specific conceptualizations and numerical examples. *Transportation Research Part B* 40: 504-519.

Denant-Boemont, L. and Petiot, R. (2003). Information value and sequential decision-making in a transport setting: an experimental study. *Transportation Research Part B* 37: 365-386.

de Palma, A. and Picard, N. (2006). Equilibrium and information provision in risky networks with risk-averse drivers. *Transportation Science* 40(4): 393-408.

Dong, J., Mahmassani, H. S. and Lu, C. C. (2006). How reliable is this route?: Predictive travel time and reliability for anticipatory traveler information systems. *Transportation Research Record: Journal of the Transportation Research Board* 1980: 117-125.

Emmerink, R. H. M., Axhausen, K. W., Nijkamp, P., Rietveld, P. (1995). The potential of information provision in a simulated road transport network with non-recurrent congestion. *Transportation Research Part C* 3(5):293–309.

Emmerink, R. H. M., Verhoef, E. T., Nijkamp, P., Rietveld, P. (1998). Information policy in road transport with elastic demand: Some welfare economic considerations. *European Economic Review* 42: 71-95.

Engelson, L. (2003). On dynamics of traffic queues in a road network with route choice based on real time traffic information. *Transportation Research Part C* 11, pp:161–183.

Eom, J. K., Park, M. S., Heo, T.-Y. and Huntsinger, L. F. (2006). Improving 1 the prediction of annual average daily traffic for nonfreeway facilities by applying a spatial statistical method, *Transportation Research Record* 1968: 20–29.

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

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

Friesz, T. L., D. Bernstein, N. J. Mehta, R. L. Tobin, Ganjalizadeh, S. (1994). Day-to-day dynamic network disequilibrium and idealized traveler information systems. *Operations Research* 42(6): 1120-1136.

Gajewski, B. J. and Rilett, L. R. (2004). Estimating link travel time correlation: An application of bayesian smoothing splines, *Journal of Transportation and Statistics* **7**(2/3): 53–70.

Gao, S. (2005). Optimal adaptive routing and traffic assignment in stochastic time-dependent networks. PhD Dissertation, Massachusetts Institute of Technology, Cambridge, MA.

Gao, S. and Chabini, I. (2002). The best routing policy problem in stochastic time-dependent networks. *Transportation Research Record* 1783, pp. 188-196.

Gao, S. and Chabini, I. (2006). Optimal routing policy problems in stochastic time-dependent networks. *Transportation Research Part B* 40: 93-122.

Gao, S. and Huang, H. (2009). Is more information better for routing in an uncertain network? The 88th Annual Meeting of Transportation Research Board Compendium of Papers DVD. Paper #09-1315, 21 pages, Washington, DC, Jan. 11 – 15.

Gao, S. and Huang, H (2011). Real-Time Traveler Information for Optimal Adaptive Routing in Stochastic Time-Dependent Networks. *Transportation Research Part C* 21(1):196-213.

Goel, P. K., McCord, M. R. and Park, C. (2005). Exploiting correlations between link flows to improve estimation of average annual daily traffic on coverage count segments: Methodology and numerical study, *Transportation Research Record* **1917**: 100–107.

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

Hall, R. W. (1996). Route choice and advanced traveler information systems on a capacitated and dynamic network. *Transportation Research Part C* 4(5): 289-306.

Huang, H. (2009). Value of traveler information for adaptive routing in stochastic time-dependent networks, Master's thesis, University of Massachusetts Amherst.

Jha, M., Madanat, S., Peeta, S. (1998). Perception updating and day-to-day travel choice dynamics in traffic networks with information provision. *Transportation Research Part C* 6: 189-212.

Kim, S., Lewis, M.E., White, C.C. (2005). Optimal vehicle routing with real-time traffic information. IEEE Transactions on Intelligent *Transportation Systems* 6(2): 178-188.

Lam, T. C. and Small, K. A. (2001). The value of time and reliability: measurement from a value pricing experiment, *Transportation Research Part E* **37**: 231–251.

Levinson, D. (2003). The value of advanced traveler information systems for route choice. *Transportation Research Part C* 11: 75-87.

Lo, H. K. and Szeto, W. Y. (2004). Modeling advanced traveler information services: static versus dynamic paradigms. *Transportation Research Part B* 38: 495–515.

Mahmassani, H. S. and Jayakrishnan, R. (1991). System performance and user response under real-time information in a congested traffic corridor. *Transportation Research Part A* 25(5):293–307.

Mahmassani, H. and Liu, Y. H. (1999). Dynamics of commuting decision behaviour under advanced traveller information systems. *Transportation Research Part C* 7: 91-107.

Marschak, J. and Miyasawa, K. (1968). Economic comparability of information systems. *International Economic Review* 9(2): 137-174.

Miller-Hooks, E. D. and Mahmassani H. S. (2000). Least expected time paths in stochastic, time-varying transportation networks. *Transportation Science* 34(2):198-215.

Miller-Hooks, E. D. (2001). Adaptive least-expected time paths in stochastic, time-varying transportation and data networks. *Networks* 37(1):35-52.

Mirchandani, P. B. and Soroush, H. (1985). Optimal paths in probabilistic networks: A case with temporary preferences, *Computers and Operations Research* **3**(4): 347–355.

Nie, Y. and Wu, X. (2009a). Reliable a priori shortest path problem with limited spatial and temporal dependencies, *Proceedings of the 18th International Symposium on Transportation and Traffic Theory*, Hong Kong, China.

Nie, Y. and Wu, X. (2009b). Shortest path problem considering on-time arrival probability, *Transportation Research Part B* 43: 597–613.

Nie, Y., X. Wu and T. Homem-de-Mello (2010) Optimal path problems with second-order stochastic dominance constraints. *Networks and Spatial Economics*, tentatively accepted.

Nielsen, L. R., K. A. Andersen and D. Pretolani (2003). Bicriterion shortest hyperpaths in random time-dependent networks. *IMA Journal on Management Mathematics* 14:271-303.

Opasanon, S. and Miller-Hooks, E. D. (2006). Multicriteria Adaptive Paths in Stochastic, Time-Varying Networks. *European Journal of Operational Research* 173:72-91.

Pretolani, D. (2000). A directed hyperpath model for random time dependent shortest paths. *European Journal of Operational Research* 123:315–324.

Pretolani, D., L. R. Nielsen, K. A. Andersen and M. Ehrgott (2009). Time-adaptive and history-adaptive multicriterion routing in stochastic, time-dependent networks. *Operations Research Letters* 37:201-205.

Psaraftis, H. N. and Tsitsiklis, J. N. (1993). Dynamic shortest paths in acyclic networks with Markovian arc cost. *Operations Research* 41(1):91-101.

Schrank, D. and Lomax, T. (2003). 2003 Annual Urban Mobility Report, *Technical report*, Texas Transportation Institute.

Schrank, D. and Lomax, T. (2009). 2009 Annual Urban Mobility Report, *Technical report*, Texas Transportation Institute.

Sen, S., Pillai, R., Joshi, S. and Rathi, A. K. (2001). A mean-variance model for route guidance in advanced traveler information systems, *Transportation Science* **35**(1): 37–49.

Sivakumar, R. A. and Batta, R. (1994). The variance-constrained shortest path problem, *Transportation Science* **28**(4): 309–316.

Tam, M. L. and Lam, W. H. K. (2009). Short-term travel time prediction for congested urban road networks, *The 88th Annual Meeting of Transportation Research Board Compendium of Papers DVD*, Washington, DC.

Waller, S. T. and Ziliaskopoulos, A. K. (2002). On the online shortest path problem with limited arc cost dependencies, *Networks* **40**(4): 216–227.

Yang, B. and Miller-Hooks, E. (2004). Adaptive routing considering delays due to signal operations. *Transportation Research Part B* 38B:385–413.

Yang, H. (1998). Multiple equilibrium behaviors and advanced traveler information systems with endogenous market penetration. *Transportation Research Part B* 32(3): 205-218.

Yin, Y. and Yang, H. (2003). Simultaneous determination of the equilibrium market penetration and compliance rate of advanced traveler information systems. *Transportation Research Part A* 37: 165–181.