

# Optimal Path Problems with Second-Order Stochastic Dominance Constraints

Yu (Marco) Nie\* and Xing Wu  
Department of Civil and Environmental Engineering  
Northwestern University, 2145 Sheridan Road, Evanston, IL 60208

Tito Homem-de-Mello  
Department of Industrial Engineering and Management Sciences  
Northwestern University  
2145 Sheridan Road, Evanston IL 60208

June 29, 2009

## Abstract

This paper studies optimal path problems integrated with the concept of stochastic dominance. These problems arise from applications where travelers need trade off the risk associated with travel time against other costs when making routing decisions. Risk-averse behavior is embedded by constraining optimal paths with second-order stochastic dominance (SSD). A general linear operating cost is introduced to combine link- and path-based costs. The latter is employed to address schedule costs pertinent to late or early arrival. An equivalent linear program of the problem is constructed by transforming the SSD constraint into a finite number of linear constraints. Various solution techniques are discussed, including those based on linear programming and dynamic programming. Numerical results are provided using small- and medium-size examples.

*Keywords:* optimal path problem, stochastic dominance, random utility theory, dynamic programming, linear programming

## 1 Introduction

Travel time reliability has become an increasing concern of transportation users. Personal and business travelers who face unexpected delays may be late for work and miss important appointments. Freight carriers may lose revenue when unexpected delays strike their supply chains and disrupt just-in-time delivery. The lack of reliability often forces them to choose between running the risk of being late or budgeting a large buffer time, much of which is often wasted. Therefore, transportation users are in great need of proper route guidance to weigh in travel uncertainty. Because unexpected delays impact travel behavior in the long run, it is also important to address

---

\*Corresponding author

travel reliability in travel forecasting and network design. Optimization problems arising from these applications often involve determining optimal paths in a probabilistic network.

At the heart of stochastic optimal path problems lies the fundamental issue of comparing random path travel times. When uncertainty is involved, numerous definitions of (*a priori*) "optimal paths" exist in the literature, which are related to or formulated based on either robust optimization (e.g. Yu & Yang 1998, Bertsimas & Sim 2003), expected utility theory (e.g. Eiger, Mirchandani & Soroush 1985, Murthy & Sarkar 1996, Sivakumar & Batta 1994, Sen, Pillai, Joshi & Rathi 2001), or first-order stochastic dominance (FSD) (e.g. Loui 1983, Frank 1969, Mirchandani 1976, Miller-Hooks & Mahmassani 2003, Nie & Wu 2009b).

Robust routing model such as Yu & Yang (1998) aims at finding the best route in the worse scenario. It does not fully address risk-taking heterogeneity in route choice because, intuitively, not every traveler anticipate worse-case travel times when planning their trips. Expected utility theory postulates that an individual chooses an alternative to maximize the expected utility of returns. Since most likely an individual's utility function is not completely observable, a restricted class of functions that capture the commonalities of individuals are often considered. It is well known that utility maximization is reduced to trading off mean and variance of the return for quadratic functions or for any polynomial function if the return is normally distributed (see e.g. Loui 1983). The mean-variance rule was first proposed by Markowitz for portfolio selection (Markowitz 1952, Markowitz 1970), and has been widely used to formulate optimal path problems (e.g. Loui 1983, Eiger et al. 1985, Murthy & Sarkar 1996, Sivakumar & Batta 1994, Sen et al. 2001). Despite its popularity, however, this rule has several well known deficiencies. For one thing, quadratic utility implies increasing absolute risk aversion<sup>1</sup>, which is inconsistent with known economic phenomena (Arrow 1971, Stiglitz 1970). Moreover, the assumption of normal distribution is often violated in real-world applications (Cootner 1964, Li, Rose & Sarvi 2006). Consequently, the mean-variance rule produces reasonable approximation to reality only when the "risk" is limited (Samuelson 1970). First-order stochastic dominance (FSD) establishes that an alternative  $X$  is preferred to  $Y$  if the cumulative probability at any return levels for  $X$  is always lower. Effectively, the dominance requires the cumulative distribution function (CDF) of  $X$  never lies above that of  $Y$ . In route choice, it is interpreted as maximizing the probability of arriving on time for any given time budget (Nie & Wu 2009b). FSD is closely related to expected utility theory (see Section 2) yet offers a more general alternative to overcome the aforementioned limitations of the mean-variance rule. However, FSD is unable to distinguish routes preferred by decision-makers of different risk-taking preferences. Higher-order stochastic dominance (SD) is required for that purpose. For instance, second-order SD captures risk-averse behavior (e.g. Levy 1992).

This paper aims at formulating a more flexible approach to address risk-taking in route choice under uncertainty, by integrating the concept of second-order stochastic dominance (SSD) into optimal path problems. We focus on SSD because it captures risk-averse behavior. The proposed framework constrains the feasible path set with SSD requirement, thereby building risk-taking preference into the route choice model. If the feasible set is defined by paths that dominate a given benchmark path, the problem can be formulated as an optimization problem with stochastic dominance constraints (OP-SD, Dentcheva & Ruszczyński 2003). In the context of route choices, such a problem may arise from applications where travelers or shippers are not only *risk-averse* with respect to travel times, but also sensitive to other costs in routing. For example,

---

<sup>1</sup>Absolute risk aversion is known as Arrow-Pratt measure (Arrow 1965). An increasing Arrow-Pratt measure implies that the amount of returns someone is willing to expose to risk decrease as the total returns increases.

a freight carrier may wish to minimize operating costs (e.g. fuel consumption) and/or schedule costs (early or late arrival penalty) while insuring that the selected route is no riskier than the one currently in use. A special instance of such problems will be closely examined, which aims at finding paths that minimize a total cost while stochastically dominating a benchmark path in the second order (referred hereafter as an *SSD-constrained optimal path problem*, or SSDOP). We will present a general OP-SD formulation for the SSDOP problem, construct its LP equivalence, and examine the properties of its solutions. The SSDOP problem can be solved using standard LP solvers or specialized routing algorithms. Both approaches are implemented and experimented with numerical examples. While either approach has its advantages and disadvantages, both are intractable for real-world applications due to the complexities of the problem. Although the primary focus is to fully reveal these complexities, heuristic solutions techniques that can be incorporated into the routing algorithm are briefly discussed.

The rest of the paper is organized as follows. Section 2 reviews the theory of stochastic dominance. In particular, we redefine the second-order stochastic dominance by disutility and discuss its relationship with random utility theory. The SSDOP problem is formulated as an OP-SD problem and transformed to an equivalent LP in Section 3. Section 4 discusses solution algorithms. Section 5 presents numerical results. Section 6 summarizes the paper with some concluding remarks.

## 2 Theory of stochastic dominance

### 2.1 Preliminary

*Stochastic-Dominance* (SD) theory (Hanoch & Levy 1969, Hadar & Russell 1971, Rothschild & Stiglitz 1970, Whitmore 1970) provides a systematic framework to analyze economic behavior under uncertainty. It has found many applications in the fields of finance, statistics and economics (Muller & Stoyan 2002), but has not yet received much attention in optimization and operations research, let alone transportation. SD is consistent with expected-utility theory yet waives the restrictions imposed by the mean-variance approach, and provides flexibility to capture heterogeneity resulting from both trip purposes and risk-taking attitude.

Denoting  $F_X$  as the cumulative distribution function (CDF) of  $X$ , first- and second-order stochastic dominance (FSD and SSD) are defined as follows.

**Definition 1 (FSD (SSD))** *A random variable  $X$  dominates another random variable  $Y$  in the first (second) order, denoted as  $X \succeq^1 Y$  ( $X \succeq^2 Y$ ) if  $F_X$  (the integral of  $F_X$ ) never lies above  $F_Y$  (the integral of  $F_Y$ ).*

**Definition 2 (FSD (SSD) Admissibility)** *An alternative  $X \in \Omega$  (where  $\Omega$  is the set of feasible alternatives) is said to be FSD (SSD) admissible if no  $Y \in \Omega$  exists such that  $Y \succeq^1 X$  ( $Y \succeq^2 X$ ) and  $F_X(t) > F_Y(t)$  ( $\int_{-\infty}^t F_X(w)dw > \int_{-\infty}^t F_Y(w)dw$ ) at least for one  $t$ .*

The following result connects the SD rules with expected utility theory.

**Theorem 1**  *$X \succeq^1 Y$  ( $X \succeq^2 Y$ ) iff  $E[U(X)] \geq E[U(Y)]$  for every nondecreasing (concave nondecreasing) function  $U(\cdot)$ , where  $E[U(X)]$  denotes the expectation of  $U(X)$ .*

Proof: See e.g., Bawa (1975). □

Theorem 1, which is a classical result, implies that  $X \succeq^2 Y$  (SSD) iff any *nonsatiable* and *risk-averse* decision maker prefers  $X$  to  $Y$  (e.g. Friedman & Savage 1948). Similarly,  $X \succeq^1 Y$  (FSD)

implies that  $X$  is preferable to  $Y$  among *nonsatiabile* decision makers. The result can be extended to higher-order SD rules (Bawa, Bondurtham, Rao & Suri 1983). For example, the third-order SD (TSD) captures decreasing absolute risk aversion (Arrow 1965).

## 2.2 Optimization problem with stochastic dominance constraints

Although SD theory is a useful tool to compare random variables, it imposes only a partial order since a dominance relationship may not exist between two alternatives. Consequently, SD alone is usually unable to produce a complete ranking to support decision making. However, the set of alternatives can be refined according to an SD rule, and then ranked using additional criteria. This idea gives rise to an optimization framework in which SD is employed not to determine the best alternative but to constrain the feasible set so as to capture decision-makers' risk-taking preferences. In order to restrict alternatives using SD, two approaches are often used: 1) remove all non-*admissible* alternatives, or 2) retain only alternatives that dominate given benchmark alternatives.

By definition, an alternative that is *admissible* under an SD rule will satisfy *some* decision makers of the corresponding nature. In some cases, identifying the set of admissible alternatives itself is of practical interest. For example, the CDFs of all FSD admissible paths form a *Pareto frontier* which maximizes the probability of arriving on time for any given time budget (Nie & Wu 2009b). In other words, if the preferred probability of completing a trip within a given time budget is given, the route that requires minimum "buffer time" can be identified from the FSD admissible set. SD-admissible alternatives can be determined from cross-comparing all available alternatives using Definition 1. Miller-Hooks (1997) defines optimality for routing problems in a probabilistic and time-dependent network using FSD. An exact label-correcting algorithm was proposed (Miller-Hooks 1997, Miller-Hooks & Mahmassani 1998) to determine FSD admissible paths. Nie & Wu (2009b) studied a continuous formulation of a similar problem and proposed pseudo-polynomial approximation schemes. Recently, Nie & Wu (2009a) studied the FSD admissible path problem in a probabilistic network with limited temporal and spatial dependency. Additional objectives can be incorporated to define the "best" in a set of admissible paths, thereby providing flexibility to integrate risk preferences with other desirable routing criteria. In principle, one can first generate all SD-admissible paths using algorithms discussed in Miller-Hooks (1997) or Nie & Wu (2009b), and then find the optimal path by comparing the objective function values associated with these admissible paths. However, since the number of SD-admissible paths grow exponentially with network size, such a brutal-force approach may be prohibitively expensive in a large network.

To find a set of alternatives that dominate a given benchmark, which ensures that *all* decision makers of the corresponding nature will be satisfied, is equivalently if not more difficult. In route choice, this operation essentially requires enumerating all paths. Interestingly, when SSD is concerned, the dominance relationship can be represented with a finite number of linear constraints, which leads to the recently-introduced *optimization problems with stochastic dominance constraints* (OP-SD) (Dentcheva & Ruszczyński 2003, Dentcheva & Ruszczyński 2004). A generic form of such optimization problems reads

$$\min f(z) \text{ s.t. } G(z) \succeq^2 Y, \quad z \in \mathcal{Z} \quad (1)$$

where  $z$  is a decision vector,  $G(z)$  is a random measure of system performance as a function of design variable  $z$ , and  $\mathcal{Z} \in R^n$ . The SD constraint ensures that  $G(z)$  is preferable over the given

benchmark  $Y$ . If the random variables  $G(z)$  and  $Y$  have finite support, under mild conditions Dentcheva & Ruszczyński (2003) showed that the feasible region in (1) can be expressed using a *finite* number of variables and constraints, which are explicitly given. Homem-de-Mello & Mehrotra (2009) show a similar property for problems with multi-variate constraints, i.e., when the function  $G$  in (1) is vector-valued, which imposes significant difficulties. Many of the applications of (1) found in the literature are in the realm of finance. A typical problem is to determine a portfolio whose performance dominates that of a given benchmark such as the S&P 500 index; see Dentcheva & Ruszczyński (2003) and Roman, Darby-Dowman & Mitra (2006) for discussions. To the best of our knowledge, this is the first paper that studies an application of (OP-SD) in transportation.

### 2.3 Second-order stochastic dominance by disutility

Definition 1 assumes that the return (utility) increases with the random quantity of interest. However, since travel time is generally considered a disutility, the utility of an *a priori* path is a decreasing function of its traversal time. For random variables associated with disutility, one can develop results symmetric to Theorem 1. The second-order stochastic dominance (SSD) by disutility is defined as follows <sup>2</sup>

**Definition 3 (Stochastic dominance by disutility  $\succeq_2$ )** A random variable  $X$  dominates another random variable  $Y$  by disutility in the **second** order, denoted as  $X \succeq_2 Y$  if  $\int_t^\infty F_X(w)dw \geq \int_t^\infty F_Y(w)dw, \forall t \in R$ , where  $R$  is the set of all real numbers <sup>3</sup>.

Denote by  $\mathcal{V}_1$  a set of non-increasing and concave functions. The result symmetric to Theorem 1 is presented as follows.

**Theorem 2**  $X \succeq_2 Y$  iff  $E[U(X)] \geq E[U(Y)]$  for all  $U \in \mathcal{V}_1$ .

Proof: See pages 18 - 19 of (Muller & Stoyan 2002). For convenience an alternative proof is given in Appendix A.  $\square$

It is well-known that for any random variable  $X$ , a *risk-averse* individual prefers the expectation  $E(X)$  to  $X$  itself (Friedman & Savage 1948). Introduce  $\delta_X$  as a random variable such that  $P[\delta_X = E(X)] = 1$ . Risk averse implies

$$\delta_X \succeq X \Leftrightarrow E[U(\delta_X)] \geq E[U(X)] \Leftrightarrow U(E(X)) \geq E[U(X)]$$

According to Jensen's inequality, the utility function  $U$  satisfies the above condition if and only if  $U$  is concave. In other words, all risk-averse individuals are similar in that their utility functions are concave. Such utility functions can be either non-decreasing or non-increasing, depending on whether the random variable is associated with utility or disutility. Thus, it follows from Theorem 2 that SSD defined herein captures individuals' risk-averse behavior.

The following result gives an equivalent yet more useful condition to specify SSD.

**Proposition 1**  $X \succeq_2 Y$  iff  $E(X - \eta)_+ \leq E(Y - \eta)_+, \forall \eta \in R$ , where  $X_+ = \max(0, X)$ .

<sup>2</sup>In Muller & Stoyan (2002), it is termed as Increasing Convex Order (ICX). We call it second-order stochastic dominance by disutility in order to highlight the connection with the classical results.

<sup>3</sup>We use  $\succeq_2$  for SSD by disutility to distinguish it from the classic SSD definition  $\succeq^2$ .

Proof: See pages 18 - 19 of (Muller & Stoyan 2002). For convenience an alternative proof is given in Appendix A.  $\square$

As discrete probability distributions are used in most practical circumstances, we now extend the equivalent SSD condition to the discrete case.

**Proposition 2** *Suppose  $Y$  has a discrete distribution with realizations  $y_i, i = 1, \dots, m$  where  $a \leq y_i \leq b$ , then  $X \succeq_2 Y$  iff  $E(X - y_i)_+ \leq E(Y - y_i)_+, i = 1, \dots, m$ .*

Proof: it follows from Proposition 1 and Proposition 3.2 in Dentcheva & Ruszczyński (2003).  $\square$

A condition symmetric to the above one has been used in Dentcheva & Ruszczyński (2003) to transform an SSD constraint into a finite number of linear constraints. In the next section, we will make use of this result to obtain LP equivalence for the SSDOP problem.

### 3 SSD-constrained optimal path problem

#### 3.1 Basic formulation

Consider a freight carrier (or a commuter) who would like to find a path that is no riskier (as far as travel time is concerned) than the one currently in use while minimizing a specific operating cost. Assuming travel takes place only between one origin-destination pair  $r - s$ , the above problem can be generally formulated as an SSD-constrained optimal path problem (SSDOP):

$$\min \alpha_1 \sum_{ij \in \mathcal{A}} z_{ij}(x_{ij}) + \alpha_2 f\left(\sum_{ij \in \mathcal{A}} \bar{c}_{ij} x_{ij}\right) \quad (2a)$$

$$\text{s.t. } \sum_{j:ji \in \mathcal{A}} x_{ji} - \sum_{j:ij \in \mathcal{A}} x_{ij} = d_i, \forall i \in \mathcal{N} \quad (2b)$$

$$\sum_{ij \in \mathcal{A}} \bar{c}_{ij} x_{ij} \succeq_2 \bar{\pi}_k^{rs} \quad (2c)$$

$$x_{ij} \geq 0, \forall ij \in \mathcal{A} \quad (2d)$$

where  $d_i$  denotes the total shipping demands at node  $i$  (note that  $d_r = -1, d_s = 1$  and  $d_i = 0, \forall i \neq s$ ).  $x_{ij}$  and  $\bar{c}_{ij}$  represent the flow and random travel time on link  $ij$  respectively;  $k^{rs}$  is the benchmark path and  $\bar{\pi}_k^{rs}$  denotes the benchmark travel time.  $z_{ij}(\cdot)$  is a link-based cost function and  $f(\cdot)$  is a cost associated with the random path travel time. Both  $z_{ij}(\cdot)$  and  $f(\cdot)$  are assumed to be linear or piecewise linear functions. However, we note that nonlinear functions may be approximated by piecewise linear functions. The objective function is a combination of link- and path-based costs, weighted by  $\alpha_1$  and  $\alpha_2$  respectively. Constraint (2b) states the flow conservation condition; Constraint (2c) ensures that no risk-averse carrier will prefer the current path to the optimal one.

Two special cases of Problem (2) may be of practical interest. First, when  $\alpha_2 = 0$  and  $z_{ij}(x_{ij}) = \kappa l_{ij} x_{ij}$  (where  $l_{ij}$  is the distance of link  $ij$  and  $\kappa$  is an operating cost associated with distance, such as fuel consumption or emission), the problem is to find a minimum-cost path which dominates the current path in the second order with respect to travel time. Second, when on-time delivery is important, the carrier may impose a penalty cost to late arrival, or to both late and early arrival. Denote  $\pi_0$  as the scheduled travel time (which may be interpreted as the desirable arrival time),

the expected penalty can be evaluated by

$$\sum_{ij \in \mathcal{A}} f_{ij}(\bar{c}_{ij} x_{ij}) = \sum_{\theta} P_{\theta} [\beta e_{\theta}^{-} + \gamma e_{\theta}^{+}] \quad (3a)$$

$$e_{\theta}^{-} = [\pi_0 - \sum_{ij \in \mathcal{A}} c_{ij}(\theta) x_{ij}]_{+} \quad ; \quad e_{\theta}^{+} = [\sum_{ij \in \mathcal{A}} c_{ij}(\theta) x_{ij} - \pi_0]_{+} \quad (3b)$$

where  $c_{ij}(\theta)$  is the travel time on link  $ij$  at realization  $\theta \in \Theta$ ,  $\Theta$  is a set of realizations of link travel times, and  $P_{\theta}$  is the probability of realizing  $\theta$ . Moreover,  $\beta \geq 0$  and  $\gamma \geq 0$  are parameters associated with early and late arrival penalty<sup>4</sup>. Finally,  $e_{\theta}^{-}$  and  $e_{\theta}^{+}$  are auxiliary random variables which measure the difference between the scheduled and realized travel times at  $\theta$ .

### 3.2 LP transformation

We only discuss path-based costs in what follows, since dealing with link-based costs is simpler and can be treated similarly. Accordingly,  $\alpha_1$  is set to zero and the carrier is assumed to minimize the expected penalty among all paths that outperform the benchmark path in the SSD sense. Noting that the stochastic dominance constraint (2c) can be relaxed and transformed to a polyhedral set (Dentcheva & Ruszczyński 2003), we reformulate the minimization problem of on-time arrival penalty as:

$$\min \sum_{\theta} P_{\theta} [\beta e_{\theta}^{-} + \gamma e_{\theta}^{+}] \quad (4a)$$

$$\text{s.t. } e_{\theta}^{-} \geq \pi_0 - \sum_{ij \in \mathcal{A}} c_{ij}(\theta) x_{ij}, \quad \forall \theta \quad (4b)$$

$$e_{\theta}^{+} \geq \sum_{ij \in \mathcal{A}} c_{ij}(\theta) x_{ij} - \pi_0, \quad \forall \theta \quad (4c)$$

$$\sum_{j:ji \in \mathcal{A}} x_{ji} - \sum_{j:ij \in \mathcal{A}} x_{ij} = d_i, \quad \forall i \in \mathcal{N} \quad (4d)$$

$$\sum_{ij \in \mathcal{A}} c_{ij}(\theta) x_{ij} - s(\theta, \eta) \leq \eta, \quad \forall \theta \in \Theta, \eta \in \Phi \quad (4e)$$

$$\sum_{\theta} P_{\theta} s(\theta, \eta) \leq \sum_{\theta} P_{\theta} [(\pi_k^{rs}(\theta) - \eta)_{+}], \quad \forall \eta \in \Phi \quad (4f)$$

$$e_{\theta}^{-}, e_{\theta}^{+} \geq 0, \quad \forall \theta \in \Theta \quad (4g)$$

$$s(\theta, \eta) \geq 0, \quad \forall \theta \in \Theta, \eta \in \Phi \quad (4h)$$

$$x_{ij} \geq 0, \quad \forall ij \in \mathcal{A} \quad (4i)$$

where  $s(\theta, \eta) : \Theta \times \Phi \rightarrow \mathbf{R}$  are dummy variables introduced to linearize the non-smooth operator  $[\cdot]_{+}$  in the SSD condition (cf. Proposition 2);  $\Phi$  is a set of a realized travel times on the benchmark path, i.e.,  $\Phi = \{\pi_k^{rs}(\theta), \theta \in \Theta\}$ ;  $\pi_k^{rs}(\theta)$  is the travel time on the current path at the realization  $\theta$ . Constraints (4b)-(4c) are introduced, together with the auxiliary variables  $e_{\theta}^{+}$  and  $e_{\theta}^{-}$ , to deal with the piecewise linear objective function. Constraints (4e) - (4f) are linear transformation of (2c) according to the SSD condition given in Proposition 2. For the right side of Constraint (4f), as  $\pi_k^{rs}(\theta)$ ,  $\eta$  and  $P_{\theta}$  are known, the values of  $\sum_{\theta} P_{\theta} [(\pi_k^{rs}(\theta) - \eta)_{+}]$  for all  $\eta$  can be calculated respectively.

<sup>4</sup>The piecewise linear penalty imposed here is similar to the ‘‘schedule cost’’ often appeared in the morning commute analysis (Vickrey 1969).

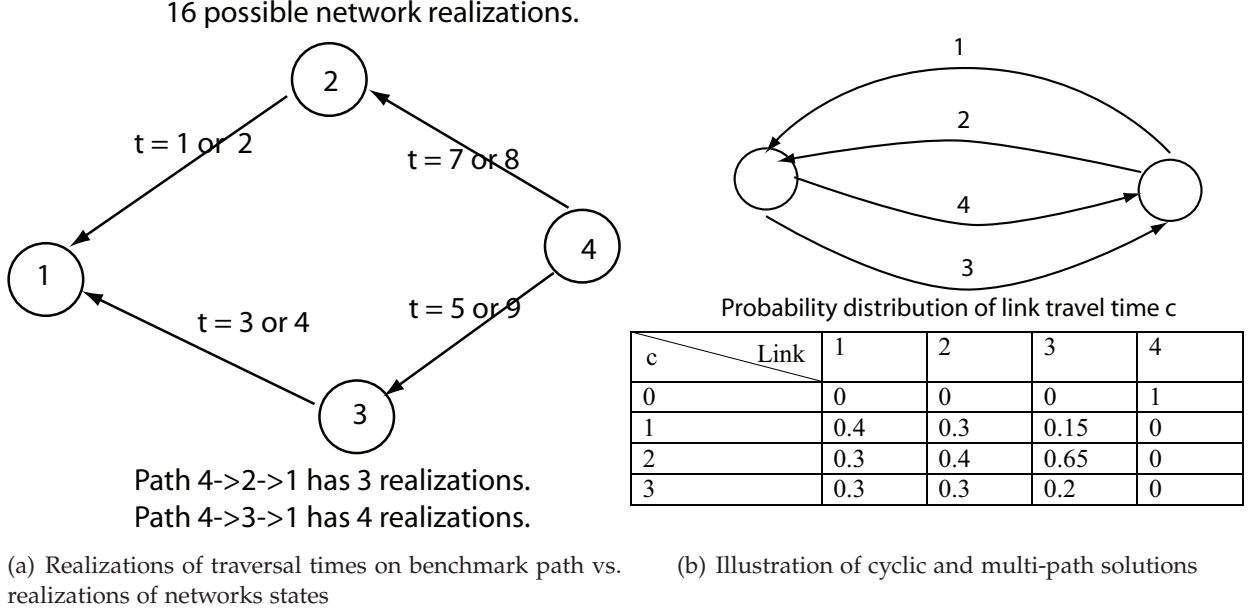


Figure 1: Illustrative examples

For random link traversal times that are discrete and independently distributed, the size of realizations  $|\Theta| = L^m$ , where  $m$  is the number of links and  $L$  is the uniform number of realizations on each link. However, the number of revealed realizations is often much smaller than this formidable bound in reality thanks to the correlations between link travel times. Moreover, we note that the number of realizations of traversal time on the benchmark path ( $\pi_k^{rs}(\theta)$ ) may be significantly smaller than  $|\Theta|$ , see Figure 1(a) for an illustration. Supposing that the number of travel time realizations of the benchmark path is  $R$ , Constraint (4f) can be replaced by

$$\sum_{\forall \theta} P_{\theta^s}(\theta, \eta) \leq \sum_{i=0}^R p_{\pi_k}(i) [(\pi_k^{rs}(i) - \eta)_+], \quad \forall \eta \in \Phi \quad (5)$$

where  $p_{\pi_k}(i)$  is the probability of the traversal time  $\pi_k(i)$ ,  $i = 0, 1, \dots, R$ , which can be obtained once  $k^{rs}$  and  $\Theta$  are known.

### 3.3 Properties of optimal solutions

Problem (4) becomes trivial when the benchmark path itself is SSD-admissible, because it would be impossible to find a path that dominates the benchmark in the second order. In this case, the only element in the feasible set would be the benchmark path itself. Therefore, we shall assume that the benchmark path is not SSD-admissible. With this assumption, let us present the following property of the optimal solution to Problem (4).

**Proposition 3** *Let  $\Gamma$  be a set of SSD-admissible paths and  $\Pi$  be a set of optimal paths that solves (4). If 1)  $\Gamma \neq \emptyset$ , 2) the benchmark path  $k \notin \Gamma$ , and 3)  $\beta = 0, \gamma > 0$ , there exists at least one path  $l \in \Pi$  such that  $l \in \Gamma$ .*

Proof: The objective (4a) seeks to minimize an expected disutility  $E(U)$ , where

$$U = \beta[\pi_0 - \sum_{ij \in \mathcal{A}} c_{ij}(\theta)x_{ij}]_+ + \gamma[\sum_{ij \in \mathcal{A}} c_{ij}(\theta)x_{ij} - \pi_0]_+$$

Suppose one optimal solution of Problem (4) corresponds to path  $\bar{k}^{rs}$  which is not SSD-admissible. Then  $\bar{k}^{rs}$  must be dominated by at least one path, say  $l^{rs}$ . Since  $\beta = 0$ ,

$$E(U(\bar{k}^{rs})) - E(U(l^{rs})) = \gamma \left( E([\sum_{ij \in \bar{k}^{rs}} c_{ij}(\theta)x_{ij} - \pi_0]_+) - [\sum_{ij \in l^{rs}} c_{ij}(\theta)x_{ij} - \pi_0]_+ \right)$$

If  $l^{rs}$  does not correspond to an optimal solution, then we have  $E(U(\bar{k}^{rs})) < E(U(l^{rs}))$ . This would imply

$$E([\sum_{ij \in \bar{k}^{rs}} c_{ij}(\theta)x_{ij} - \pi_0]_+) < [\sum_{ij \in l^{rs}} c_{ij}(\theta)x_{ij} - \pi_0]_+$$

According to Proposition 2, this contracts to the assumption that  $l^{rs} \succeq_2 \bar{k}^{rs}$ . If  $l^{rs}$  is among the optimal solutions, one can assume  $l^{rs}$  is not SSD-admissible and repeat the above analysis until an optimal solution is found that cannot be dominated by any other optimal solutions. The above process is finite as long as the number of optimal paths is finite. It follows that at least one optimal solution will be SSD-admissible.  $\square$

The above result suggests that an optimal path may not be SSD-admissible when  $\beta > 0$ . Note that the penalty function is no longer a decreasing function of random travel time if  $\beta > 0$ , although it is still concave. Thus, the SSD-dominance does not necessarily imply a better utility. In fact, an optimal solution to Problem (4) may not even be a simple path in general cases. Rather it can contain arbitrary cycles and/or multiple paths. This is demonstrated using the example given in Figure 1(b), where  $\pi_0 = 4, \beta = 1, \gamma = 2$ . First, one can verify that path  $1 \rightarrow 4 \rightarrow 1$  has a lower expected penalty cost ( $0.92 = 0.16 \times (4 - 2) + 0.24 \times (4 - 3) + 0.33 \times (4 - 4) + 0.18 \times (5 - 4) + 0.09 \times (6 - 4)$ ) than path 1 ( $2.1 = 0.4 \times 3 + 0.3 \times 2 + 0.3$ ). This is essentially because cycling around link 4 can delay the arrival, which is desirable given the early arrival penalty. If we let the benchmark path be path 2 and force the stochastic dominance constraint, then the above cyclic path is no longer optimal because it does not dominate path 2. Note that path 1  $\succeq_2$  path 2, and path 2 and 3 do not dominate each other in the second order. However, when solving the proposed formulation, path 1 is not the optimal solution. Instead, the optimal solution assigns 0.6 units of flow to link 1, and 0.4 units of flow to link 3. As such, a *hyper-path* is created with the following distribution  $P(c = 1) = 0.3, P(c = 2) = 0.44, P(c = 3) = 0.26$  ( $c$  denotes path travel time, as shown in Figure 1(b)). The reader can verify that this hyper-path dominates path 2 in the second order, and its penalty cost is  $2.04 < 2.1$ .

## 4 Solution techniques

### 4.1 Techniques based on linear programming

Problem (4) can be solved using commercial LP solvers such as CPLEX. Nonetheless, such an approach may not be fully tractable in practice owing to several reasons. First, the number of constraints of the problem grows with  $|\Theta||\Phi|$ . Thus, when the number of realizations is large, directly solving the problem can be computationally prohibitive. Note that the number of realizations can be large even for small networks — for example, if there are 20 links and travel times on each link can take three possible values (assume all travel times are independent of each other), then there are  $3^{20} \approx 3.5 \times 10^9$  realizations.

A possible way around the problem is to use sampling techniques. In the context of more traditional stochastic optimization problems — where the goal is to minimize the expected value

of a function, subject to a fixed number of deterministic or expected-value constraints — it is well known that sampling methods can often approximate well, with a small number of samples, problems that have very large number of scenarios; see, for instance, (Linderoth, Shapiro & Wright 2006) for numerical reports. The reasons for such good performance are well understood and have to do with how well-conditioned the problem is (Shapiro, Homem-de-Mello & Kim 2002). The situation is somewhat different in case of Problem (4), since the number of constraints and variables also depend on the number of realizations. Very recent research (Hu, Homem-de-Mello & Mehrotra 2009) indicates that sampling methods can indeed be extended to optimization problems with stochastic dominance constraints; a detailed study of the application of those results to the present problem is left for future research.

Another issue with Problem (4) is that, in order to avoid a hyper-path solution as shown in the above example, one may have to restrict the solution variables  $x_{ij}$  in Problem 4a - 4i to binary variables. Consequently, the problem becomes an integer programming problem, which is significantly more difficult to solve. Note that this restriction may be unnecessary in other instances, for instance when  $\alpha_2 = 0$ . Finally, it is difficult to detect and remove cycles from the solution, if they are undesirable outcomes.

## 4.2 Techniques based on dynamic programming

An alternative way to solve Problem (4) is to use a modified label-correcting algorithm presented in Nie & Wu (2009b). Let  $\mathcal{N}_b$  be a set of nodes traversed by the benchmark path  $k^{rs}$  and  $\Gamma_b^{is}$  be a set of paths at node  $i$  that can be a subpath of a path  $l^{rs} \succeq_2 k^{rs}$ . An exact label correcting algorithm for Problem (4), named as SSDOP-LC, is given below.

### Algorithm SSDOP-LC

**Step 0** Initialization. At all nodes  $i \in \mathcal{N}_b$ , preset  $\Gamma_b^{is} = \{k^{is}\}$ , where path  $k^{is}$  is a subpath of benchmark path  $k^{rs}$ . Initialize the scan list  $Q = \{s\}$  where  $s$  is the destination node. At destination  $s$ , define a dummy path  $0^{ss}$  and let  $\Gamma_b^{ss} = \{0^{ss}\}$ .

**Step 1** If  $Q = \emptyset$ , go to Step 5; otherwise, select the first node  $j$  from  $Q$ , and delete it from  $Q$ . Let  $\Gamma^{js}$  be the current path set at node  $j$  and  $B(j)$  be the predecessor node set of node  $j$ .

**Step 2** Augment paths for any combination of path  $l^{js} \in \Gamma^{js}$  and predecessor node  $i \in B(j)$ . Go to step 1 when all combinations are examined. For each combination, if path  $l^{js}$  has not traversed node  $i$  before (cyclic check), create a new path  $\hat{l}^{is}$  by augmenting path  $l^{js}$  along link  $ij$ , perform the following operations:

Step 2.1 If node  $i \notin \mathcal{N}_b$ , go to Step 2.1.1; otherwise, go to Step 2.1.2.

Step 2.1.1 Set  $\Gamma^{is} = \Gamma^{is} \cup \{\hat{l}^{is}\}$ .

Step 2.1.2 Calculate and compare  $E(\pi_i^{is} - \eta)_+$  as well as  $E(\pi_k^{is} - \eta)_+$  for all discrete points  $\eta \in \Phi$ . Here  $k^{is}$  is the subpath of the benchmark path  $k^{rs}$ . If  $E(\pi_i^{is} - \eta)_+ \geq E(\pi_k^{is} - \eta)_+$  for all  $\eta$  and  $E(\pi_i^{is} - \eta)_+ > E(\pi_k^{is} - \eta)_+$  for at least one  $\eta$ , drop  $\hat{l}^{is}$  because  $k^{is} \succeq_2 \hat{l}^{is}$ ; otherwise, set  $\Gamma_b^{is} = \Gamma_b^{is} \cup \{\hat{l}^{is}\}$ .

Step 2.2 If  $\Gamma^{is}$  is updated, set  $Q = Q \cup \{i\}$ .

**Step 3** Select the path that minimizes the objective function defined in (2a) from  $\Gamma_b^{rs}$ .

The following remarks are in order.

**Remark 1:** For any node  $i \notin \mathcal{N}_b$  the above algorithm has to enumerate all paths, and for the rest ( $i \in \mathcal{N}_b$ ), it removes paths that are dominated by the benchmark path. Note that the objective here is to identify all paths that dominate the benchmark path in the second order. However, removing all subpaths that do not dominate the corresponding subpaths of the benchmark is problematic, because  $l^{rs} \succeq_2 k^{rs}$  does not imply its sub-paths satisfy  $l^{is} \succeq_2 k^{is}$ . As a result, Algorithm SSD-LC-BP is not as efficient as the algorithms designed to determine SD-admissible paths (Miller-Hooks 1997, Nie & Wu 2009b). The latter is likely to retain a much smaller number of paths thanks to the more restrictive non-dominance relationship, albeit both are non-deterministic polynomial.

**Remark 2:** A primary challenge of using SSD-LC-BP as an alternative to solve Problem (4) lies in the difficulty of accounting for correlations embedded in a joint probability distribution. Essentially, one has to enumerate all possibilities in order to properly evaluate  $E(\pi_l^{is} - \eta)_+$ . Nevertheless, SSD-LC-BP may be preferable to a standard LP solver because it takes the network structure into consideration and thus can easily exclude cycles and hyper-paths.

Approximation techniques may be employed to improve the computational efficiency of the above algorithm. One common strategy is to cap the number of admissible paths to be enumerated with some predetermined value  $M$  (e.g. Miller-Hooks & Mahmassani 2000). However, this method seems too arbitrary to be practically useful. Instead, an algorithm based on the idea of *diversity maximization* (Masin & Bukchin 2008) is implemented and tested. The idea of diversity maximization is based on the premise that, by selecting non-dominant paths which are as different from each other as possible, a broader space of non-dominant paths can be explored. In turn, it prevents highly correlated non-dominant paths from entering the path all together. A brief description of the algorithm is given in what follows.

Recall that SSD admissibility of a path  $X$  is determined based on the shape of  $F_X^2(t) = \int_0^t F_X(w)dw$ ,  $\forall t \in [0, T]$ , where  $T = F^{-1}(1.0)$ ,  $F(\cdot)$  and  $F^{-1}(\cdot)$  are the CDF and inverse CDF of  $X$ , respectively. Our diversity maximization algorithm (DMA) attempts to find a restricted admissible set  $S \subset \Omega$  (where  $\Omega$  is the complete set of SD-admissible paths) in order to maximize a *diversity* measure, which is evaluated according to the shape of function  $F_X^2$ . Note that the shape of  $F_X^2$  may be characterized using multiple indexes. As a starting point, however, it suffices to only consider the total area under the curve  $F_X^2$ . Using the previous notation, this area can be written as  $F_X^3(T) = \int_0^T F_X^2(w)dw$ . We proceed to define the diversity of an admissible subset  $S = \{x_1, \dots, x_M\}$ . Without loss of generality, let us assume that  $F_{x_1}^3(T) \leq F_{x_2}^3(T) \leq \dots \leq F_{x_M}^3(T)$ . The diversity of  $S$ , denoted as  $\chi_S$ , is now defined as

$$\chi_S = \min_{i=1, \dots, M-1} \{F_{x_{i+1}}^3(T) - F_{x_i}^3(T)\} \quad (6)$$

Further, let  $P = \{S | S \subset \Omega, |S| = M\}$ . The diversity maximization can be defined as

$$\chi_{max} = \max\{\chi_S, S \in P\} \quad (7)$$

Figure 2 illustrate the above idea. Assume  $\Omega$  is the complete set of SD-admissible paths, and all elements (8 in this example) are sorted in an increasing order according to  $F_X^3$ . Assume that we have to pick a subset of size 4 to represent  $\Omega$ .  $S_1, S_2$  and  $S_3$  are such subsets.  $\chi_{s_1}, \chi_{s_2}$  and  $\chi_{s_3}$  are the diversity measures of the three sets. The figure shows that  $S_3$  has the largest diversity measure among the three subsets in this example.

As the above example suggests, one should enumerate all possible subsets of size  $M$  in order to solve the problem exactly. To find such an optimal solution is difficult even when  $M$  is small

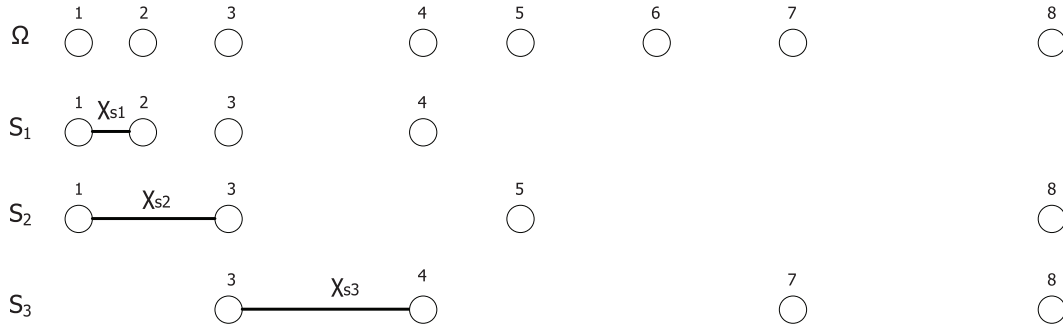


Figure 2: Demonstration of the the diversity maximization concept (total number of alternatives = 8, the number of restricted set  $M = 4$ )

and is left to subsequent studies. Instead, the problem is solved approximately in this paper as follows: whenever a new path  $Y$  attempts to enter an admissible set that has reached the limit  $M$ , we check whether or not it will increase the diversity of the set by replacing either  $x_i$  or  $x_{i+1}$  with  $Y$ , where  $F_{x_i}^3(T) \leq F_Y^3(T) \leq F_{x_{i+1}}^3(T)$ . Unless  $Y$  increases the diversity, it is discarded. The implementation details of the approximation scheme are ignored here for brevity and will be reported in a more computation-oriented work.

Algorithm SSDOP-LC can be easily revised to incorporate the above approximation schemes. For convenience the revised algorithms based on simple restriction and DMA strategy are named SSDOP-LC-Appro and SSDOP-LC-DMA, respectively. Their relative computational performance is reported in the next section.

## 5 Numerical results

### 5.1 Results based on LP techniques

For validation purpose two small examples are formulated using AMPL (Fourer, Gay & Kernighan 2003) and solved by KNITRO and MINTO<sup>5</sup>.

Table 1: Link traversal time distribution for the 4-node network

Travel time	Link 1-2	Link 1-3	Link 1-4	Link 2-3	Link 3-4
0	0.2	0 (0.1)	0	0.6	0.3
1	0.3	0.2(0.2)	0.1	0.1	0.7
2	0.3	0.3(0.3)	0.2	0.1	0
3	0.2	0.5(0.4)	0.3	0.1	0
4	0	0 (0)	0.4	0.1	0

Note: link travel time realizations on Link 1-3 have two scenarios. The data in parenthesis are Scenario 2

The first example (see Figure 3(a)) has 4 nodes and 5 links, with the distributions of random link traversal times given in Table 1. The reader can verify that  $\pi_2^{1,4}$  (travel time on path  $2^{1,4}$  :

<sup>5</sup>KNITRO is linear programming solver available at <http://www-neos.mcs.anl.gov> and MINTO is an integer programming solver available though NEOS at <http://www-neos.mcs.anl.gov/>

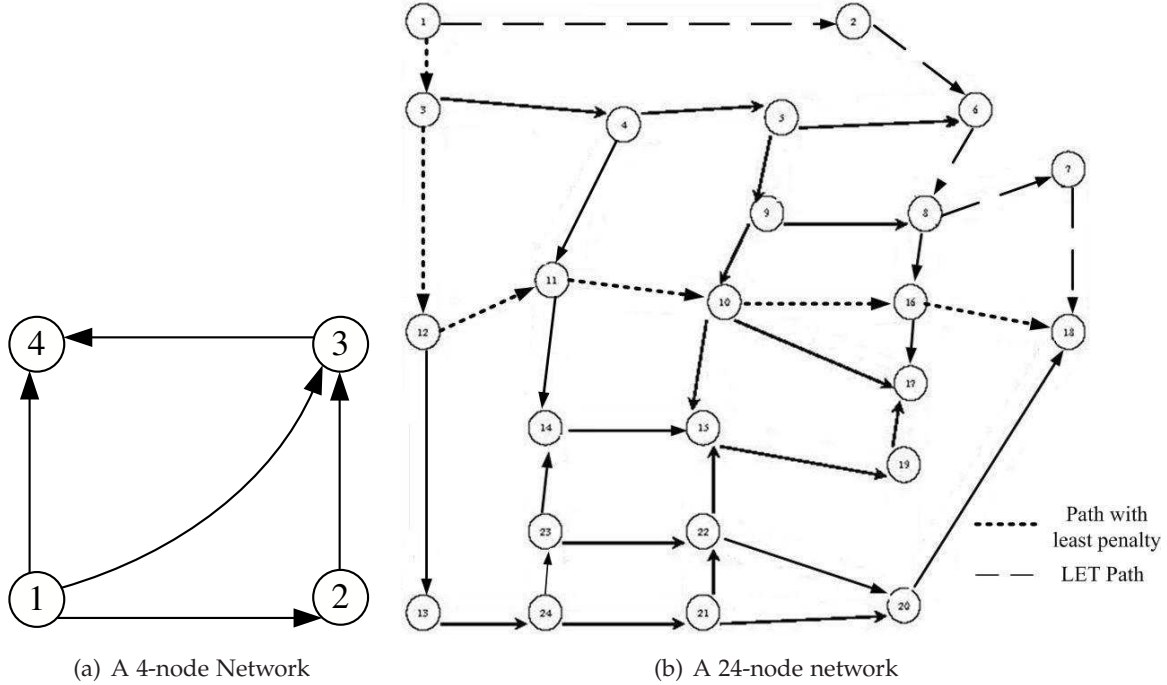


Figure 3: Test Networks

$1 \rightarrow 3 \rightarrow 4) \succeq_2 \pi_1^{1,4}(1 \rightarrow 4) \succeq_2 \pi_3^{1,4}(1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$ . Table 2 gives the expected non-punctual penalty cost of the three paths in two scenarios with different parameters. Note that  $|\Theta| = L^m = 5^5 = 3125$ . The solutions obtained from both MINTOS and KNITRO are reported in Table 3. In the first scenario, we set  $\pi_0 = 3.0, \beta = \gamma = 0.5$ . The optimal path generated by MINTOS is always path  $2^{14}$  no matter which path is set as benchmark. This is expected since  $2^{14}$  dominates the other two paths and has a lowest penalty cost for the parameters in the first scenario. In the second scenario, the travel time distribution data on link 3 are replaced with those in parenthesis (see Table 1). However,  $2^{14} \succeq_2 1^{14} \succeq_2 3^{14}$  still holds. Other parameters are  $\pi_0 = 3.0, \beta = 0.8$ , and  $\gamma = 0.2$ . The optimal path solved by MINTO is  $1^{14}$  when the benchmark path is either  $1^{14}$  or  $3^{14}$ . This is because path  $1^{14}$  not only dominates path  $3^{14}$ , but also has a lower penalty cost than it (see Table 3). However, since  $2^{14} \succeq_2 1^{1,4}$ , path  $1^{1,4}$  is not SSD-admissible. This demonstrates that an optimal path may not be an SSD-admissible path. When the benchmark path is  $2^{14}$ , the optimal solution is the path itself because the other paths do not dominate  $2^{14}$ .

In either scenario, an identical multi-path solution is yielded (which provides a lower penalty cost) from KNITRO no matter which benchmark path is applied. Thus, it is important to force  $x_{ij}$  to be binary if the desirable outcome is a single path. However, a multi-path solution may be useful when a “portfolio” of paths has to be determined for a fleet of vehicles.

We next test a larger network with 24 nodes and 37 links (see Figure 3(b)). In this experiment, 50 realizations of link traversal times are randomly drawn from a uniform distribution  $[0, 1]$ . Each realization is randomly assigned a probability  $P_\theta \in [0, 1]$  such that  $\sum_\theta P_\theta = 1.0$ . We consider only the O-D pair 1-18 and set  $\pi_0 = 3.5, \beta = 5, \gamma = 10$ . The least-expected-travel-time (LET) path for that O-D pair is path  $1^{1,18}$ :  $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 7 \rightarrow 18$ , as illustrated in Figure 3(b). For the purpose of validation, we enumerated all 12 paths between O-D pair 1-18 and calculated the objective function values. The path with the least penalty is identified as  $3^{1,18}$ :

Table 2: Expected path penalty costs in two scenarios for the 4-node network

Path	Scenario 1	Scenario 2
	$\beta = \gamma = 0.5, \pi_0 = 3.0$	$\beta = 0.8, \gamma = 0.2, \pi_0 = 3.0$
$1^{14} : 1 \rightarrow 4$	0.4	0.2
$2^{14} : 1 \rightarrow 3 \rightarrow 4$	0.35	0.29
$3^{14} : 1 \rightarrow 2 \rightarrow 3 \rightarrow 4$	0.67	0.42

Table 3: Optimal path flows solved by KNITRO and MINTO for the 4-node network

	Benchmark path	$x$ not binary (KNITRO)				$x$ binary (MINTO)			
		Path 1	Path 2	Path 3	Obj	Path 1	Path 2	Path 3	Obj
Scenario 1	Path 1	0.5	0.5	0	0.265		1		0.35
	Path 2	0.5	0.5	0	0.265		1		0.35
	Path 3	0.5	0.5	0	0.265		1		0.35
Scenario 2	Path 1	0.5	0.3	0.2	0.304	1			0.40
	Path 2	0.5	0.3	0.2	0.304		1		0.52
	Path 3	0.5	0.3	0.2	0.304	1			0.40

$1 \rightarrow 3 \rightarrow 12 \rightarrow 11 \rightarrow 10 \rightarrow 16 \rightarrow 18$  (highlighted in Figure 3(b)). As shown in Table 4 when the LET path  $1^{1,18}$  is the benchmark, the optimal solution obtained by MINTO is path  $1^{1,18}$  itself. This is because  $1^{1,18}$  is the only LET path, and thus must also be SSD-admissible. The optimal path remains to be path  $1^{1,18}$  for the benchmark path  $2^{1,18}$ . Note that path  $3^{1,18}$  does not dominate  $2^{1,18}$  in the second order although it has the least penalty cost. Finally, when path  $4^{1,18}$  is set as the benchmark path, the least penalty cost path  $3^{1,18}$  is identified as the optimal solution.

Table 5 reports the optimal solutions obtained by KNITRO for the same set of benchmark paths. When the benchmark path is  $1^{1,18}$ , the optimal solution from KNITRO is identical to that from MINTO, indicating that no hyper-path solution can dominate the LET path in the second order. For the benchmark path  $2^{1,18}$ , the optimal path is a combination of paths  $1^{1,18}$ ,  $2^{1,18}$  and  $3^{1,18}$ , with path  $1^{1,18}$  receiving largest flow (73.2%). As expected, the penalty cost of this hyper-path

Table 4: Optimal solution obtained by MINTO for the 24-node network

Benchmark Path	Optimal Path Solved by MINTO	Objective Function Value
Path $1^{1,18}$ : $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 7 \rightarrow 18$	Path $1^{1,18}$ : $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 7 \rightarrow 18$	5.848
Path $2^{1,18}$ : $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 16 \rightarrow 18$	Path $1^{1,18}$ : $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 7 \rightarrow 18$	5.848
Path $4^{1,18}$ : $1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 8 \rightarrow 16 \rightarrow 18$	Path $3^{1,18}$ : $1 \rightarrow 3 \rightarrow 12 \rightarrow 11 \rightarrow 10 \rightarrow 16 \rightarrow 18$	4.307

Table 5: Optimal solution obtained by KNITRO for the 24-node network (non-integral solution)

Benchmark Path	Optimal Path Solved by KNITRO		Objective
	Path ID	Flow	Function Value
Path 1 <sup>1,18</sup> :	Path 1 <sup>1,18</sup> :	1	5.848
Path 2 <sup>1,18</sup>	Path 1 <sup>1,18</sup>	0.732	5.446
	Path 2 <sup>1,18</sup>	0.168	
	Path 3 <sup>1,18</sup>	0.100	
Path 4 <sup>1,18</sup>	Path 1 <sup>1,18</sup> + Path 2 <sup>1,18</sup>	0.660	2.660
	Path 4 <sup>1,18</sup> + Path 5 <sup>1,18</sup>	0.051	
	Path 3 <sup>1,18</sup>	0.110	
	Path 6 <sup>1,18</sup>	0.179	

Path 5<sup>1,18</sup> : 1 → 3 → 4 → 5 → 6 → 8 → 7 → 18

Path 6<sup>1,18</sup> : 1 → 3 → 12 → 13 → 24 → 21 → 20 → 18.

solution is lower than that of path 1<sup>1,18</sup> itself. A more complicated hyper-path solution, which involves 6 paths and corresponds to an even lower penalty cost, is obtained if the benchmark path is 4<sup>1,18</sup>. Note that the optimal path 3<sup>1,18</sup> suggested by MINTO only receives a small portion of flows (11%) in the KNITRO solution.

## 5.2 Results based on DP techniques

We first re-solved the four-node problem from the last section (Figure 3(a)) using the algorithm presented in Section 4.2 and found that Algorithm SSDOP-LC produced identical results as those obtained by MINTOS. For the case when the benchmark path is 1<sup>14</sup> : 1 → 4, the solution process is detailed in Figure 4.

Note that the present DP algorithms cannot solve the second problem in the last section because they are unable to deal with correlated travel times. For independently distributed link travel times, however, much larger instances of Problem 4 can be solved using the DP techniques. In the following example, the sketch of Chicago network, which has 933 nodes and 2950 links, is employed (for the network topology, see Nie & Wu 2009b). Two scenarios are considered: in the first, the network is reduced into an acyclic network (which includes 1230 links) such that the exact solution can be obtained easily to verify the correctness of the algorithm; the second scenario employs the original network to examine difference in efficiency of the two approximation schemes, SSDOP-LC-DMA and SSDOP-LC-Appro (for simplicity, they will be referred to as DMA and Appro in this section).

In both tests,  $\beta$  (early arrival penalty) and  $\gamma$  (late arrival penalty) are set to 1 and 2, respectively. Link travel times are assumed to follow Gamma distributions, whose probability density function is defined as

$$p_{ij}(x) = \frac{1}{\theta^\kappa \Gamma(\kappa)} x^{\kappa-1} e^{-x/\theta} \quad (8)$$

where  $\theta$  and  $\kappa$  are parameters and  $\Gamma(\cdot)$  is the Gamma function. The mean and variance of a Gamma distribution are  $\kappa\theta$  and  $\kappa\theta^2$ , respectively. In our experiment, parameters  $\kappa$  and  $\theta$  are generated randomly using a uniform distribution for all links. Specifically,  $\theta \propto U(0.8, 3.5)$ , and

Set Path  $1^{14}$ :  $1 \rightarrow 4$  as the benchmark path. Scanlist:  $\mathcal{Q}=\{4\}$ . At Node 1, initialize  $\Gamma^{14}=\{1^{14}\}$ ,  $\Gamma^{24}=\emptyset$ ,  $\Gamma^{34}=\emptyset$ , and  $\Gamma^{44}=\{0^{44}\}$  where  $0^{44}$  is a virtual path.

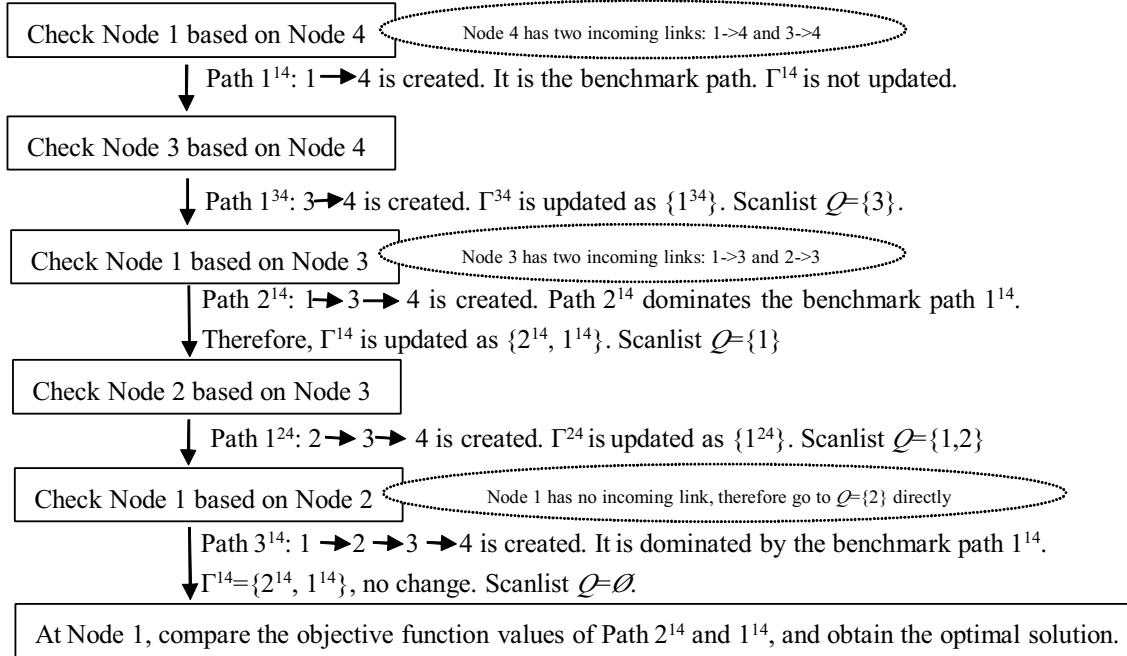


Figure 4: The solution process when applying Algorithm SSDOP-LC to the four-node network

$\kappa \propto U(1.0, 2.5)$ . Thus, the mean and standard deviation of link traversal times are within the ranges  $[0.80, 8.75]$  and  $[0.80, 5.53]$ , respectively.

A pre-defined benchmark path is selected for each tested network and its distribution of travel time is calculated. For comparison, two values of the scheduled travel time  $\pi_0$  are considered: it is set either close to the expected travel time on the benchmark path or significantly larger than it. For the acyclic network,  $\pi_0 = 60$  and  $\pi_0 = 200$  respectively; for the original network,  $\pi_0 = 80$  and  $\pi_0 = 300$  respectively.

Figure 5 compares the performance of the two approximation algorithms at different values of  $M$  for the acyclic network. As expected, both algorithms yield better objective function values (OFV) as  $M$  increases. DMA attained the exact optimal OFV (by means of path enumeration) when  $M = 140$ , whereas Appro had not reached the exact optimality until  $M$  was increased to 190 (which is not shown in the figure). Nevertheless, DMA does not demonstrate a clear superiority in this case: it not only ran much slower, but also experienced oscillations in that it did not always produce a better OFV as  $M$  increases. Note that DMA updates its admissible set much more frequently than Appro, which, in our implementation, will stop accepting any new paths once  $M$  is reached. Therefore, DMA is more computational intensive mainly because each update of the admissible set will be followed by a visit to the affected node in the labeling operation. Consequently, DMA needs more iterations (i.e., each node has to be visited more times) to solve the problem. While the calculation of diversity measure also contributes to the extra computation overhead, its impact is almost negligible. Experiments indicate that, when  $\pi_0 = 300$  and  $M = 50$ , the CPU time spent to calculate diversity measures is only 3.63 seconds out of a 241.69-second total. The number of iterations is 20996 in that case, indicating that each node was visited on average more than 20 times. In contrast, the number of iterations is 1247 for

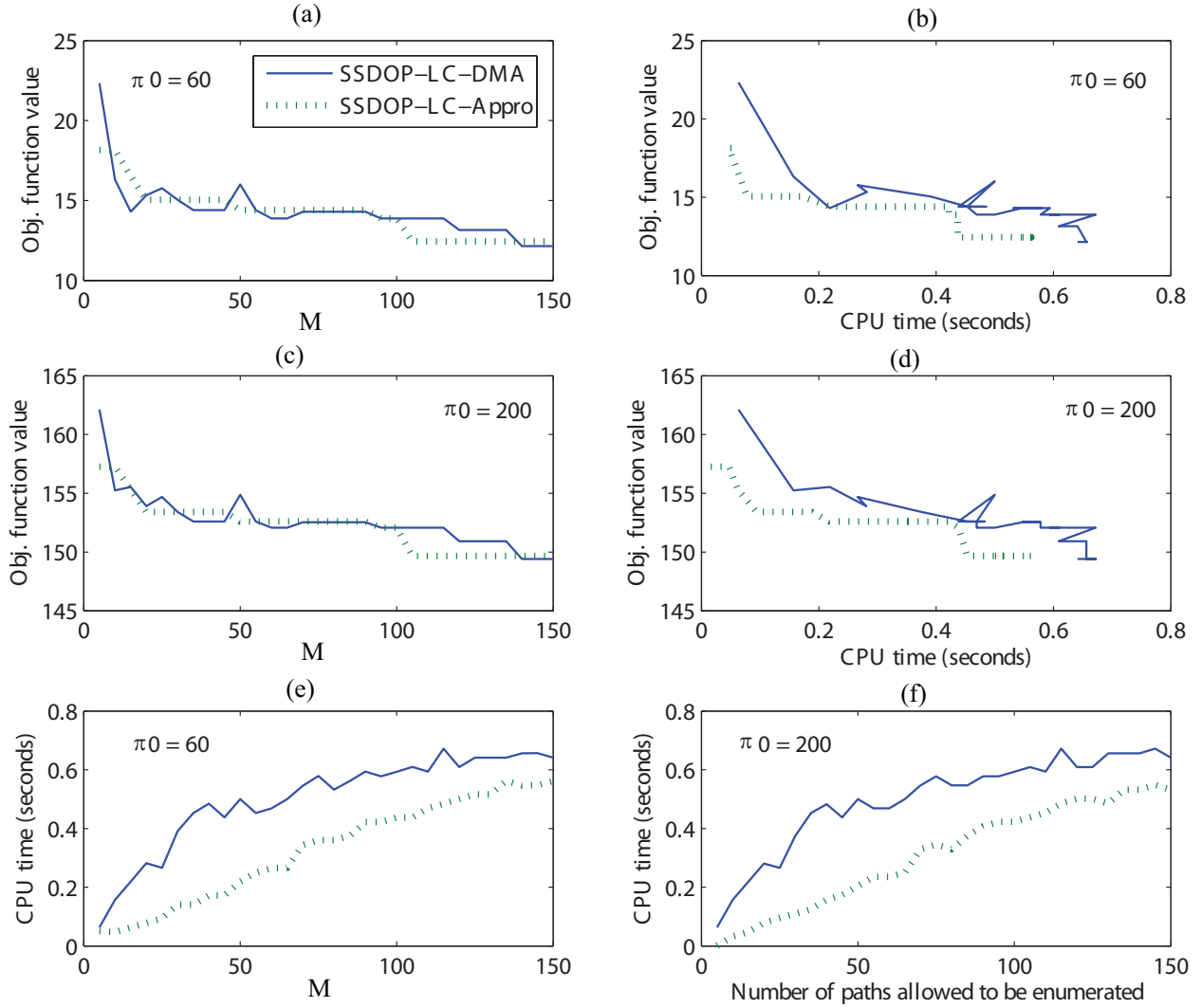


Figure 5: Performance comparison of Algorithm DMA and Appro when two algorithms are applied to Acyclic Chicago Sketch network (933 nodes, 1230 links), given OD pair (1,933),  $\beta = 1$ ,  $\gamma = 2$  and  $\pi_0 = 60$  or 200.

Appro.

The comparison results on the original network are reported in Figure 6. While the plots look similar to those in Figure 5 in general, there are a few noticeable differences. First, the increase of network size (number of links) seems to have a larger impact on the computational cost of DMA. When  $M = 150$ , DMA spent more than 2200 seconds to solve the problem; while Appro took only 30 seconds. In this case ( $M = 150$ ), DMA spent 67787 iterations to solve the problem (CPU time for diversity calculation is 33.60 seconds), while Appro only took 1410 iterations. Second, DMA consistently produced better OFV than Appro at any level of  $M$ , although the fluctuations still exist. Finally, it is worthy noting that DMA provides OFV comparable to Appro at much smaller  $M$ . Consequently, although the computational cost of DMA is generally much higher for the same  $M$  value, it often outperformed Appro in CPU time when the same value OFV is pursued (see (b) and (d) in Figure 6).

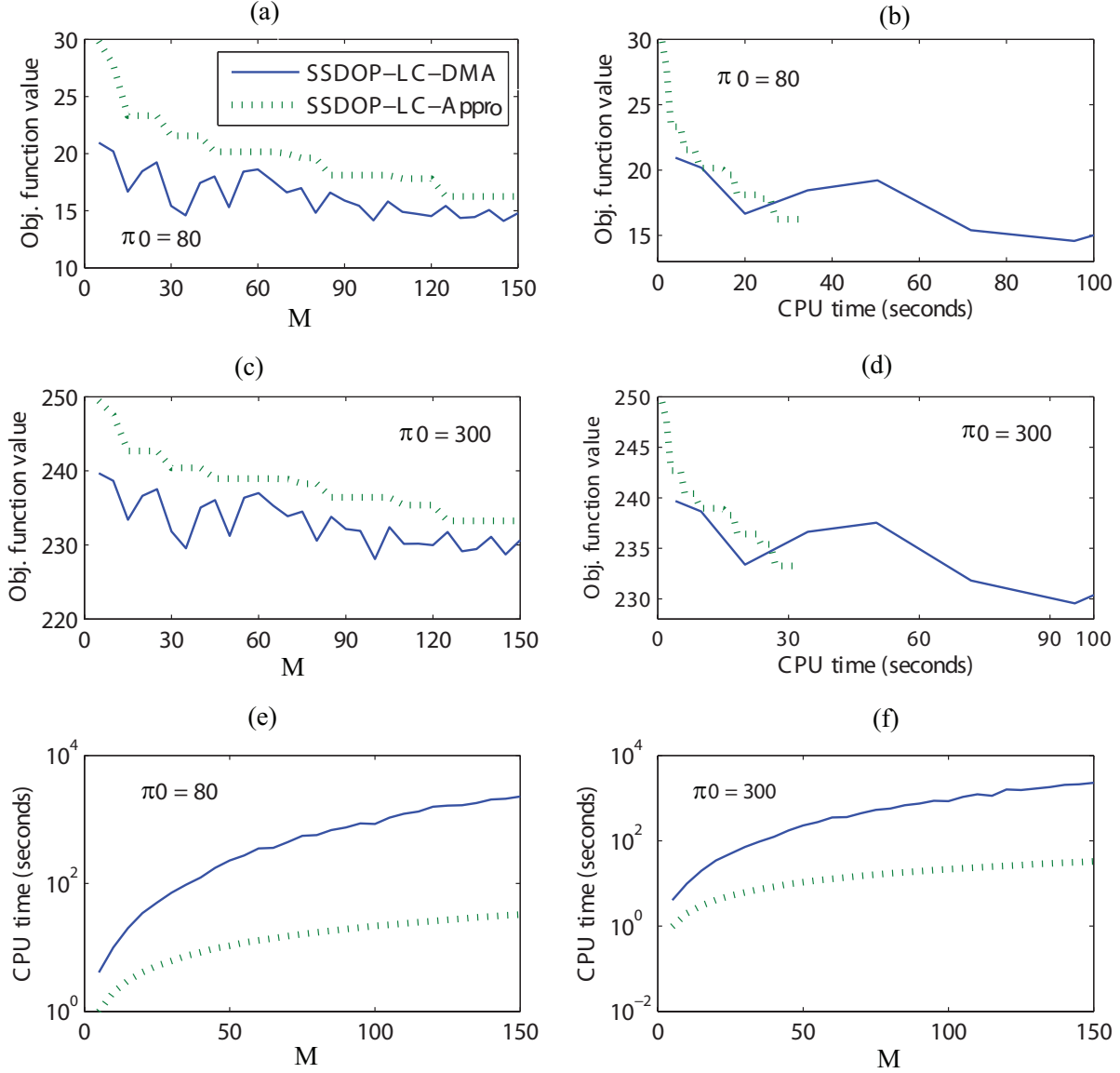


Figure 6: Performance comparison of Algorithm DMA and Appro when two algorithms are applied to Chicago Sketch network (933 nodes, 2950 links), given OD pair (1,933),  $\beta = 1$ ,  $\gamma = 2$  and  $\pi_0 = 80$  or 300

We close this section by noting that the above results are primitive, and that the purpose of the presentation is mainly to demonstrate that the SSD-constrained optimal path problem can be solved using specialized algorithms in addition to LP. We leave a more in-depth investigation on algorithmic issues to a further research.

## 6 Concluding remarks

This paper studies a class of optimal path problems which aim at helping risk-averse travelers trade off the risk associated with travel time against other costs when making routing decisions.

Risk-averse behavior is captured using second-order stochastic dominance (SSD), which has a well-established and intuitive interpretation in the expected utility theory. Accordingly, we propose an SSD-constrained optimal path (SSDOP) model formulated as a mathematical program with SSD constraints. In this model, optimal paths are determined according to a general operating cost, from a set of paths that are no riskier than a benchmark path (in the SSD sense). The operating cost is treated as a linear combination of link- and path-based costs. The latter is used to address the schedule cost pertinent to late or early arrival. We provide LP equivalence of the problem by transforming the SSD constraint into a finite number of linear constraints, using the method proposed in Dentcheva & Ruszczyński (2003). It is shown that an optimal solution to the problem may not be SSD-admissible unless the form of the objective function meets certain conditions. Moreover, cycles and hyper-paths may occur in the solution to the LP transformation.

Developing efficient solution procedures for SSDOP is a considerable challenge. Although the aforementioned transformation allows one to solve the problem using LP solvers (as demonstrated in Section 5), the transformed problem often becomes prohibitively large. One way to overcome this difficulty is to adopt sampling techniques, which warrants further investigation. The existence of hyper-paths and cycles further complicates the issue. The former forces solution variables to be integers, and the latter calls for an iterative procedure to detect cycles and add cycle-preventing constraints into the LP problem (e.g. (Current, ReVelle & Cohon 1985)). Further research is needed to search for efficient approximation schemes and implementation strategies.

The label-correcting algorithm discussed in Section 4 provides an alternative to avoid the difficulties associated with cycles and hyper-paths, which is particularly attractive when correlations among link travel times can be ignored. However, the algorithm is also intractable for large problems because it has to enumerate a very large set of paths. Approximation schemes were explored to make the algorithm computationally feasible by restricting the number of paths to be enumerated. In particular, an algorithm based on the idea of diversity maximization was implemented. Our primary results indicate that this algorithm generates sub-optimal solutions which other approximation strategies could never achieve. To make the algorithm a computationally competitive alternative for solving SSDOP is an on-going research effort.

The SSDOP problem can be extended to accommodate other SD rules such as the first-order SD (FSD). However, since an FSD-constrained optimization problem is non-convex (Noyan & Ruszczyński 2007), solving it may be even more difficult. The proposed model can also be extended to tackle assignment problems, in which a freight carrier (or a network operator) wishes to spread its shipments (or traffic) over various routes to reduce risks and minimize operating costs for the system. The formulation of an assignment problem is similar to Problem (4) but may involve multiple SD constraints and O-D pairs.

## Acknowledgements

Professor Richard A. Waltz of the Department of Electrical Engineering and Computer Science at Northwestern University kindly provides a free copy of KNITRO to be used with AMPL. The authors would also like to thank Dr. Pattharin Sarutipand from Northwestern University for her assistance in building AMPL models.

## References

- Arrow, K. (1965), *Aspects of the Theory of Risk-Bearing*, Helsinki: Yrj Hahnsson Foundation.
- Arrow, K. J. (1971), *Essays in the theory of risk bearing*, Markham, chapter Theory of risk aversion.
- Bawa, V. S. (1975), 'Optimal rules for ordering uncertain prospects', *Financial Economics* **2**, 95–121.
- Bawa, V. S., Bondurtham, J., Rao, R. & Suri, H. L. (1983), 'On determination of stochastic dominance optimal set', *Journal of Finance* **40**, 417–431.
- Bertsimas, D. & Sim, M. (2003), 'Robust discrete optimization and network flow', *Mathematical Programming Series B* **98**, 49–71.
- Cootner, P. (1964), *The random character of stock market prices*, MIT press, Cambridge, Massachusetts.
- Current, J. R., ReVelle, C. S. & Cohon, J. (1985), 'The maximum covering/shortest path problem: A multiobjective network design and routing formulation', *European Journal of Operational Research* **21**, 189–199.
- Dentcheva, D. & Ruszczyński, A. (2003), 'Optimization with stochastic dominance constraints', *SIAM Journal of Optimization* **14**(2), 548–566.
- Dentcheva, D. & Ruszczyński, A. (2004), 'Optimality and duality theory for stochastic optimization problems with nonlinear dominance constraints', *Mathematical Programming* **99**, 329–350.
- Eiger, A., Mirchandani, P. B. & Soroush, H. (1985), 'Path preferences and optimal paths in probabilistic networks', *Transportation Science* **19**(1), 75–84.
- Fourer, R., Gay, D. M. & Kernighan, B. W. (2003), *AMPL: A Modelling Language for Mathematical Programming*, 2 edn, Duxbury Press.
- Frank, H. (1969), 'Shortest paths in probabilistic graphs', *Operations Research* **17**(4), 583–599.
- Friedman, M. & Savage, L. (1948), 'The utility analysis of choices involving risk', *Journal of Political Economy* **56**, 279–304.
- Hadar, J. & Russell, W. R. (1971), 'Stochastic dominance and diversification', *Journal of Economic Theory* **3**, 288–305.
- Hanoch, G. & Levy, H. (1969), 'The efficiency analysis of choices involving risk', *Review of Economics studies* **36**, 335–346.
- Homem-de-Mello, T. & Mehrotra, S. (2009), 'A cutting surface method for uncertain linear programs with polyhedral stochastic dominance constraints', *SIAM Journal on Optimization*. Forthcoming.
- Hu, J., Homem-de-Mello, T. & Mehrotra, S. (2009), 'Sample average approximation for stochastic dominance constrained problems'. Manuscript, Northwestern University.
- Levy, H. (1992), 'Stochastic dominance and expected utility: Survey and analysis', *Management Science* **38**, 555 – 592.

- Li, R., Rose, G. & Sarvi, M. (2006), 'Using automatic vehicle identification data to gain insight into travel time variability and its causes', *Transportation Research Record* **1945**, 24–32.
- Linderoth, J. T., Shapiro, A. & Wright, S. J. (2006), 'The empirical behavior of sampling methods for stochastic programming', *Annals of Operations Research* **142**(1), 215–241.
- Loui, R. P. (1983), 'Optimal paths in graphs with stochastic or multidimensional weights', *Communications of the ACM* **26**(9), 670–676.
- Markowitz, H. (1952), 'Portfolio selection', *Journal of Finance* **7**, 77–91.
- Markowitz, H. (1970), *Portfolio selection: efficient diversification of investments*, Wiley.
- Masin, M. & Bukchin, Y. (2008), 'Diversity maximization approach for multiobjective optimization', *Operations Research* **56**(2), 411–424.
- Miller-Hooks, E. (1997), *Optimal Routing in Time-Varying, Stochastic Networks: Algorithms and Implementations*, PhD thesis, Department of Civil Engineering, University of Texas at Austin.
- Miller-Hooks, E. D. & Mahmassani, H. S. (2000), 'Least expected time paths in stochastic, time-varying transportation networks', *Transportation Science* **34**(2), 198–215.
- Miller-Hooks, E. D. & Mahmassani, H. S. (2003), 'Path comparisons for a priori and time-adaptive decisions in stochastic, time-varying networks', *European Journal of Operational Research* **146**(2), 67–82.
- Miller-Hooks, E. & Mahmassani, H. (1998), On the generation of nondominated paths in stochastic, time-varying networks, in 'Proceedings of TRISTAN III (Triennial Symposium on Transportation Analysis)', San Juan.
- Mirchandani, P. B. (1976), 'Shortest distance and reliability of probabilistic networks', *Computers and Operations Research* **3**(4), 347–355.
- Muller, A. & Stoyan, D. (2002), *Comparison Methods for Stochastic Models and Risks*, John Wiley & Sons, Chichester, UK.
- Murthy, I. & Sarkar, S. (1996), 'A relaxation-based pruning technique for a class of stochastic shortest path problems', *Transportation Science* **30**(3), 220–236.
- Nie, Y. & Wu, X. (2009a), 'Reliable a priori shortest path problem with limited spatial and temporal dependencies', In *Proceedings of the 18th International Symposium on Transportation and Traffic Theory*, accepted .
- Nie, Y. & Wu, X. (2009b), 'Shortest path problem considering on-time arrival probability', *Transportation Research Part B* **43**, 597–613.
- Noyan, N. & Ruszczyński, A. (2007), 'Valid inequalities and restrictions for stochastic programming problems with first order stochastic dominance constraints', *Mathematical Programming On-line*, DOI: [10.1007/s10107-007-0100-1](https://doi.org/10.1007/s10107-007-0100-1).
- Roman, D., Darby-Dowman, K. & Mitra, G. (2006), 'Portfolio construction based on stochastic dominance and target return distributions', *Mathematical Programming* **108**, 541–569.

- Rothschild, M. & Stiglitz, J. E. (1970), 'Increasing risk. I. a definition', *Journal of Economic Theory* **2**, 225–243.
- Samuelson, P. (1970), 'The fundamental approximation theorem of portfolio analysis in terms of means, variances and higher moments', *Review of Economic Studies* **37**, 537–542.
- Sen, S., Pillai, R., Joshi, S. & Rathi, A. (2001), 'A mean-variance model for route guidance in advanced traveler information systems', *Transportation Science* **35**(1), 37–49.
- Shapiro, A. (2003), *Monte Carlo sampling methods*, *Handbook of Stochastic Optimization*, Elsevier Science Publishers B.V., Amsterdam, Netherlands.
- Shapiro, A., Homem-de-Mello, T. & Kim, J. C. (2002), 'Conditioning of convex piecewise linear stochastic programs', *Mathematical Programming* **94**, 1–19.
- Sivakumar, R. & Batta, R. (1994), 'The variance-constrained shortest path problem', *Transportation Science* **28**(4), 309–316.
- Stiglitz, J. E. (1970), 'Review of some aspects of theory of risk bearing by k. j. arrow', *Econometrica* **38**.
- Vickrey, W. S. (1969), 'Congestion theory and transport investment', *American Economic Association* **59**, 251–261.
- Whitmore, G. A. (1970), 'Third degree stochastic dominance', *American Economic Review* **60**, 457–459.
- Yu, G. & Yang, J. (1998), 'On the robust shortest path problem', *Computers and Operations Research* **25**(6), 457–468.

## Appendix A: Proof

**Theorem 2**  $X \succeq_2 Y$  iff  $E[U(X)] \geq E[U(Y)]$  for all  $U \in \mathcal{V}_1$ .

Proof: Let  $W = -X, Z = -Y$ . So  $X \succeq_2 Y$  implies  $W \succeq^2 Z$ . Note that  $F_W(t) = P(W \leq t) = P(-X \leq t) = P(X \geq -t) = 1 - F_X(-t)$ . According to Theorem 1,  $W \succeq^2 Z$  implies

$$\begin{aligned} \int_{-\infty}^t F_W(w)dw &\leq \int_{-\infty}^t F_Z(w)dw \Leftrightarrow \int_{-\infty}^t [1 - F_X(-w)]dw \leq \int_{-\infty}^t [1 - F_Y(-w)]dw \\ &\Leftrightarrow \int_{-\infty}^t F_W(-w)dw \geq \int_{-\infty}^t F_Z(-w)dw \end{aligned}$$

Let  $\gamma = -w$  and note that  $d\gamma = -dw$ , we have

$$\int_{-t}^{\infty} F_X(\gamma)d\gamma \geq \int_{-t}^{\infty} F_Y(\gamma)d\gamma$$

Therefore, according to Definition 3,  $E[U(X)] \geq E[U(Y)]$  iff  $X \succeq_2 Y$ .  $\square$

**Proposition 1**  $X \succeq_2 Y$  iff  $E(X - \eta)_+ \leq E(Y - \eta)_+, \forall \eta \in R$ , where  $X_+ = \max(0, X)$ .

Proof:

$$\begin{aligned} E(X - \eta)_+ &= \int_{\eta}^{\infty} (x - \eta)p_X(x)dx = \int_{\eta}^{\infty} x dF_X(x) - \eta \int_{\eta}^{\infty} p_X(x)dx \\ &= xF_X(x) \Big|_{\eta}^{\infty} - \int_{\eta}^{\infty} F_X(x)dx - \eta_i F_X(x) \Big|_{\eta}^{\infty} = \infty - \eta - \int_{\eta}^{\infty} F_X(x)dx \end{aligned}$$

Therefore,  $E(X - \eta)_+ - E(Y - \eta)_+ = - \int_{\eta}^{\infty} F_X(x)dx + \int_{\eta}^{\infty} F_Y(x)dx$ . Thus, the above conditions are consistent with those given in Definition 3.  $\square$