Path determination under stochastic travel times using target-oriented robust optimization

Chen Wang
SAP Research & Innovation
#14-01, 1 Create Way, Singapore
Email: chen.wang02@sap.com

Bingshui Da
School of Computer Engineering
Nanyang Technological University
Email: bingshui.da@sap.com

***Abstract*—This paper addresses an optimal path problem in which the travel time is subject to uncertainty. Many relevant works in the literature model the uncertainty using a random variable, however in many cases the underlying distribution of the uncertainty is not precisely and a solution obtained under a presumed distribution can perform poorly in practice. In this work we assume only the lower bound and upper bound of the uncertainty is known, and by only making use of such information we determine a solution using robust optimization techniques. The solution is robust in the sense that a pre-specified travel time target can be guaranteed for a uncertainty set that is maximized. Additionally, the robust optimization problem is not necessarily to be solved, and its solution can be obtained by solving and updating a deterministic problem using existing approaches such as Dijkstra approach several times. This makes the proposed approach applicable to large problem instances. The performance and advantages of the proposed approach is demonstrated by numerical experiments.**

I. Introduction

Shortest path problem is important in applications in many fields of engineering, logistics and science. Usually, a network consisting of a number of nodes and arcs connecting nodes is considered. Certain costs associated arcs are used to describe distance, travel time or other measurement of the arcs. Given a starting node and a destination node, the general problem is to find a path connecting the two nodes while at same time minimizing the cost of the path. There is a rich literature in the research of such problem and its variants. Initially, everything including the structure of the network and costs of arcs are assumed fixed and known, and such problem is referred to as a Deterministic Problem. A number of well-known results were developed for deterministic shortest path problem over fifty years ago [1], [2], [3].

Although the algorithms for the determination of the shortest path in a deterministic problem is quite efficient, the real-world problems are always far from deterministic. For example, the travel time within a transportation network is always stochastic and/or time-dependent. Hence, the solution of a deterministic problem may have poor performance in a stochastic setting. To address this issue, one option is to minimize the mean travel time and the resultant path is often referred to as path with least expected time (LET) [4]. For time invariant stochastic network, the LET path can be found simply by solving a deterministic problem with costs of arcs set to the mean values. However, to know the travel time distribution is of more interest than a known mean. Towards this end, the research work of [5] proposed an approach to determine the probability distribution function of the minimum travel time path in a stochastic stationary network, the research work of [6] proposed a optimality index for comparison of different paths in the network, and a number of other works also consider similar problems [7], [8], [9]. In some case, the mean of the travel time distribution is not of top concerns of users. In this case, it is reasonable to use various utility functions to describe the preferences of users and optimize the value of the utility function. Typical works in this direction include [10], [11], [12], [13].

In contrast to stationary stochastic network, it is more challenging to determine a shortest path in a time-varying stochastic network. The work of [14] considers such a problem and shows that traditional method for deterministic network may not work in such time-varying stochastic network. It also argues that the optimal path is an adaptive decision rule and proposes an approach for the determination of such a rule. A similar problem is also considered in [15] where the focus is one a dynamic stochastic traffic network. A heuristic algorithm based on the k-shortest path algorithm is proposed to determine a shortest path. The work of [16] proposes two algorithms for the determination of least possible travel time and the probability of the occurrence of such travel time for the travel between any two nodes in a traffic network. Other works concentrating on stochastic time-varying networks include [17], [18], [19], [20]. Specifically for stochastic vehicle routing problem which is a special type of problem, there is a rich literature [21], [22], [23] and still remains active.

In some works in the literature, the concept of travel time target or reliability was proposed as preference for paths. For example, the works [4], [5] and [6] all try to determine a path that maximize the probability of achieving a pre-specified travel time target. The major challenge of the proposed approaches is that multi-dimensional integral has to be conducted to evaluate and compare feasible paths, which requires a tremendous computation resources and makes the approaches only applicable to only small-scale instances. To overcome this difficulty, the work of [24], [25] propose a dynamic programming approach which decomposes the large problem into multi-stage small problems to reduce computational complexity. Another trick is to use appropriate utility functions to reflect the preference for a travel time target or travel time reliability [10].

One common assumption in most of the works in the literature of optimal paths in stochastic networks is that the distribution function of the underlying uncertain factor, for example travel times, is known no matter whether is stationary or time-varying. The distribution function plays an important role in the computation of the distribution of total travel time and probability of fulfilling certain travel time target. However, in reality the distribution function may not be precise as it is obtained from limited historical data, and sometimes the distribution is totally unavailable. In such cases, the performance of a solution that is tuned according to a presumed distribution can deteriorate a lot in practice. To handle problems with little or unreliable distribution information of the underlying uncertain factors, this paper propose an approach based on Robust Optimization (RO).

Robust Optimization, originally addressing optimization under uncertainties, experienced explosive growth in the last decade. Initially, the purpose of robust optimization is to immunize uncertain mathematical optimization problems against in-feasibility while preserving the tractability of models, see [26], [27], [28], [29], [30], [31], [32], [33]. Most robust optimization approaches share the following two merits: (1) Only limited knowledge of underlying uncertain parameters is used. In most of the early works in the literature, only support set of the uncertain parameters is assumed and used and later in some application support set and mean are assumed to achieve additional theoretical results [34]. (2) The tractability of the original optimization problem can be well preserved by robust optimization techniques, i.e. the robust counterpart of a LP problem remains a LP problem if the uncertainty variable is characterized by linearly constrained support set and remains a second-order cone optimization problem (SOCP) if the original optimization and the uncertain support set is second-order cone describable.

This paper proposes an approach using Robust Optimization techniques for the determination of a path with enhanced travel time reliability in a stochastic traffic network. The proposed approach makes use of limited information of the uncertain factors, and hence provides a solution that is robust against distributional uncertainty. A travel time target is also taken into consideration, and the travel time reliability is enhanced in term of that the possibility of fulfilling the travel time target is optimized. Although the approach is presented in the context of a traffic network in this paper, the idea is applicable to path determination problem in any weighted stochastic network with minor changes.

The rest of this paper is organized as follows. Section II describes the problem we consider and also the notations used in through out the paper. Section III introduces the mathematical formulations, including deterministic, stochastic and robust ones, for path determination. Section IV demonstrate the proposed approach and show its advantages using numerical experiments. The last section concludes the paper and also suggests future research directions.

II. Problem description

We consider a typical directed network with N nodes indexed by $1, 2, \ldots, N$. We use $\mathcal{N}$ to represent the index set of nodes, i.e. $\mathcal{N} := \{1, 2, \ldots, N\}$. Directed arcs connect two different nodes and we let $\mathcal{A} := \{(i, j) | i, j \in \mathcal{N}\}$ represent the set of node-pairs that has a arc from node i pointing to node j. Associated with each arc (i, j) in $\mathcal{A}$, there is a travel time, denoted by $\tilde{c}_{ij}$. In general, each $\tilde{c}_{ij}$ is a random variable and does not change with time, i.e. stationary. In practice or earlier literature, $\tilde{c}_{ij}$ can be modeled in the following ways.

M1 a deterministic reference point $\hat{c}_{ij}$.
M2 a lower bound $\underline{c}_{ij}$, an upper bound $\bar{c}_{ij}$, and a reference point $\hat{c}_{ij}$.
M3 a number of samples c^z_{ij}, $z = 1, 2, 3, \cdots$.
M4 a known distribution function.

Most of the works in the literature use M4 as the model of $\tilde{c}_{ij}$, or use M3 to estimate a M4 model for $\tilde{c}_{ij}$. In this paper, we rely mainly on M2 to determine an optimal path. In most cases in practice, a number of samples of $\tilde{c}_{ij}$ is available, but the sample size is not big enough for a good estimation of the distribution function or the sample set has certain bias due to time-varying nature of the uncertainty. Therefore, it is a good choice to use M2 to describe the uncertainty. The reference point $\bar{c}_{ij}$ could be the mean or the mode of the sample set. The lower and upper bound could be the estimated using reference point plus and minus triple sample standard deviation, or simply using the minimum or maximum value of the sample set.

Given a starting node s and a destination node d in the network together with its structure and travel time parameters (reference points and bounds), our goal is to determine an path that is optimal in certain way. We will discuss how to measure the optimality of a path in details in the next section.

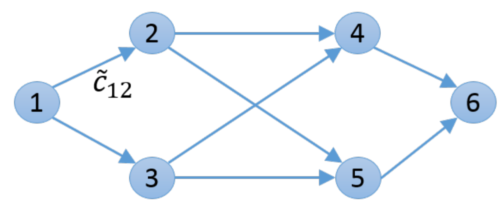


Fig. 1: Example of a routing problem

The model and the path determination problem is illustrated by an example shown in Figure 1. For this network, $\mathcal{N} = \{1, 2, 3, 4, 5, 6\}$, $\mathcal{A} = \{(1, 2), (1, 3), (2, 4), (2, 5), (3, 4), (3, 5), (4, 6), (5, 6)\}$. Suppose for $\tilde{c}_{12}$ we have 5 samples: $\{1, 1.1, 1.1, 1.3, 1.5\}$. The reference point $\hat{c}_{12}$ could be either the mean 1.2 or the mode 1.1 of the sample set. The support set could use the minimal and maximal values as bouds, $[1,\ 1,5]$, or use mean plus and minus triple sample standard deviation, $[0.6,\ 1.8]$. It depends on user's confidence level about the data set and preference of uncertainty.

Based on the description in this section, notations of parameters and sets thorough out this paper is summarized in Table I.

III. Mathematical formulation

In this section, we discuss mathematical formulations that determine a path that is optimal according to different criteria. In order to formulation the optimization problems, we define binary decision variable x_{ij} for each arc $(i, j) \in \mathcal{A}$. $x_{ij} = 1$ means the associated arc is in the path, and 0 otherwise.

TABLE I: Notation

Parameter:	N:	number of nodes.
	$\tilde{c}_{ij}$:	travel time between Node i and j, described by 1) simply a deterministic reference point $\hat{c}_{ij}$, or 2) support set $[\underline{c}_{ij},\ \bar{c}_{ij}]$ and a reference point $\hat{c}_{ij}$, or 3) a number of samples c^s_{ij}, $s = 1, 2, 3, \cdots$, or 4) a distribution function.
	s:	index of starting node
	d:	index of destination node
Set:	$\mathcal{N}$:	index set of nodes $\{1, 2, \ldots, N\}$
	$\mathcal{A}$:	index pairs (i, j) for all arcs in the network, with Node i being the starting node.
Variable:	x_{ij}:	binary variable, $x_{ij} = 1$ if the Arc (i, j) is in the route.

For the decision variables $\{x_{ij}\}$ to describe a route from the starting node to the destination node, the decision variables must satisfy the following constraints

$$\sum_{j:(s,j)\in\mathcal{A}} x_{sj} = 1, \tag{1a}$$

$$\sum_{i:(i,d)\in\mathcal{A}} x_{id} = 1, \tag{1b}$$

$$\sum_{j:(i,j)\in\mathcal{A}} x_{ij} \le 1, \quad i \in \mathcal{N}, \tag{1c}$$

$$\sum_{i:(i,j)\in\mathcal{A}} x_{ij} = \sum_{k:(j,k)\in\mathcal{A}} x_{jk}, \quad j \in \mathcal{N}/\{s, d\}. \tag{1d}$$

Constraint (1a) ensures that Node s is the starting point of the path and multiple revisit of the same node is filtered out as it is not an optimal path. Constraint (1b) ensures that Node d is the end of the path and the node is visited only once. Constraint (1c) ensures that each node is visited at most once. Constraint (1d) ensures the continuity of the path. In short, $\{x_{ij}\}$ satisfying Constraint (1) defines a path connecting Node s and Node d, and we collect all such decision variables in the following set

$$\mathbb{F} := \{\{x_{ij}\}|\ x_{ij},\ (i, j) \in \mathcal{A} \text{ satisfy (1)}\}. \tag{2}$$

Given a set of decision variables $\{x_{ij}\} \in \mathbb{F}$ and $\tilde{c}_{ij}$ of arcs, the travel time of the associated route is

$$\tilde{T} := \sum_{(i,j)\in\mathcal{A}} \tilde{c}_{ij} x_{ij}. \tag{3}$$

A. Existing formulations

If $\tilde{c}_{ij}$ is modeled by M1, then $\tilde{T}$ is deterministic and a path can be determined by solving a deterministic optimization problem. Additionally if $\hat{c}_{ij}$ is the mean of $\tilde{c}_{ij}$, then a path with the minimum mean travel time can be determined by solving the same optimization problem, which is given below.

Deterministic Formulation:

$$DP(\{\hat{c}_{ij}\}):$$

$$\min \quad \sum_{(i,j)\in\mathcal{A}} \hat{c}_{ij} x_{ij} \tag{4a}$$

$$s.t. \quad \{x_{ij}\} \in \mathbb{F} \tag{4b}$$

In other cases, the travel time is modeled by M4, then $\tilde{T}$ is a random variable. A pre-specified travel time target τ is of the most interest and we want the probability of $\tilde{T}$ being less than the travel time target to be as much as possible, i.e. a reliable travel time. The solution can be determined by solving the following optimization problem.

Stochastic Formulation:

$$SP(\{\tilde{c}_{ij}\}):$$

$$\max \quad \alpha \tag{5a}$$

$$s.t. \quad \{x_{ij}\} \in \mathbb{F} \tag{5b}$$

$$\Pr(\tilde{T} < \tau) \ge \alpha \tag{5c}$$

For discrete distribution of $\{\tilde{c}_{ij}\}$ and small-size problem instance, Problem (5) can be solved to optimality. But for large-size problems or $\{\tilde{c}_{ij}\}$ with continuous distribution, Problem (5) is hard to solve to optimality and is usually solved approximately. Even to solve an approximation of Problem (5) requires the knowledge of distribution functions of uncertain factors which is not always available. To ease the dependence on the distribution function, we propose an approach using robust optimization techniques in the rest of this section.

B. Robust Optimization formulation

In this section, we describe a Robust Optimization formulation for path determination. There are two major benefits of this new formulation. First, it is computationally amiable and its solution can be obtained efficiently by leverage on existing algorithms. Second, it does not require the distribution information of the uncertainty, and only make use of its reference point and bounds. Therefore, the performance of the solution is quite robust against distributional ambiguity.

First we define an adjustable uncertainty set for each $\tilde{c}_{ij}$ as follows,

$$C_{ij}(\gamma) := \{c_{ij}|\gamma(\underline{c}_{ij} - \hat{c}_{ij}) \le c_{ij} - \hat{c}_{ij} \le \gamma(\bar{c}_{ij} - \hat{c}_{ij})\}, \tag{6}$$

where $\gamma \in [0, 1]$. Clearly, $C_{ij}(0)$ is a singleton containing the reference point only, $C_{ij}(1)$ is the full support set of $\tilde{c}_{ij}$, and $C_{ij}(\gamma)$ is a set in between when $0 < \gamma < 1$.

To achieve a reliable travel time that is less than the target τ, we would naturally like to solve the following optimization problem.

$$\max_{0\leq\gamma\leq 1} \quad \gamma \tag{7a}$$
$$s.t. \quad \{x_{ij}\} \in \mathbb{F} \tag{7b}$$
$$\tilde{T} < \tau, \quad \forall \tilde{c}_{ij} \in C_{ij}(\gamma), \quad \forall (i,j) \in \mathcal{A} \tag{7c}$$

Given the definition of $\tilde{T}$ in (3) and $C_{ij}(\gamma)$ in (6), it can be observed that $\tilde{T}$ achieve its largest value when $\tilde{c}_{ij} = \hat{c}_{ij} + \gamma(\bar{c}_{ij} - \hat{c}_{ij})$. Therefore, Constraint (7c) is equivalent to $\sum_{(i,j)\in\mathcal{A}} (\hat{c}_{ij} + \gamma(\bar{c}_{ij} - \hat{c}_{ij}))\, x_{ij} < \tau$, and the deterministic equivalence of Problem (7) is the following.

Robust Formulation

$$RO(\{\hat{c}_{ij}, \bar{c}_{ij}\}):$$
$$\gamma^* = \max_{0\leq\gamma\leq 1} \quad \gamma \tag{8a}$$
$$s.t. \quad \{x_{ij}\} \in \mathbb{F} \tag{8b}$$
$$\sum_{(i,j)\in\mathcal{A}} (\hat{c}_{ij} + \gamma(\bar{c}_{ij} - \hat{c}_{ij}))\, x_{ij} < \tau \tag{8c}$$

Problem (8) is a Mixed Integer Programming (MIP) problem with quadratic constraints and the number of binary variables is equivalent to the number of arcs in the network. And the Mixed Integer Linear Programming (MILP) equivalence of Problem (8) is shown in Formulation (9). In this formulation, M is a sufficiently large number.

Mixed Integer Linear Programming

$$MILP(\{\hat{c}_{ij}, \bar{c}_{ij}\}):$$
$$\gamma^* = \max_{0\leq\gamma\leq 1} \quad \gamma \tag{9a}$$
$$s.t. \quad \{x_{ij}\} \in \mathbb{F} \tag{9b}$$
$$\sum_{(i,j)\in\mathcal{A}} \lambda_{ij} < \tau \tag{9c}$$
$$\lambda_{ij} \geq \hat{c}_{ij} + \gamma\,(\bar{c}_{ij} - \hat{c}_{ij}) + M\,(x_{ij} - 1) \tag{9d}$$
$$\lambda_{ij} \geq 0, \quad \forall (i,j) \in \mathcal{A} \tag{9e}$$

It may take a while to solve the problem if the network is relatively large since Integer Programming is an NP-complete problem. However, it is worthy pointing out that Problem (8) can be solved by performing a binary search over γ and solving DP problem (4) repeatedly. The procedure is summarized in Algorithm 1.

In Algorithm 1, the travel time target τ and the maximal number of iteration i_{max} are the input parameters, and the procedure $TRO_{BS}(\tau, i_{max})$ returns the value of γ^* with arbitrary accuracy. For example, when $i_{max} = 10$ the accuracy is $1/2^{10}$ which is less than 0.001. But in practice, 7 is a good enough choice for i_{max}.

Algorithm 1 Binary search for solution of Problem (8)

procedure $TRO_{BS}(\tau,\, i_{max})$
  $LB = 0$, $UB = 1$, $i = 1$
  if optimum of Problem (4) is less than τ **then** Return infeasible
  end if
  if optimum of Problem (4) with $\hat{c}_{ij}$ replaced by $\hat{c}_{ij} + \bar{c}_{ij} - \hat{c}_{ij}$ is no less than τ **then** Return 1
  end if
  while $i \leq i_{max}$ **do**
    if optimum of Problem (4) with $\hat{c}_{ij}$ replaced by $\hat{c}_{ij} + (\bar{c}_{ij} - \hat{c}_{ij}) * (UB - LB)/2$ is no less than τ **then** LB =(UB+LB)/2
    else UB =(UB+LB)/2
    end if
    $i = i + 1$
  end while
  Return LB
end procedure

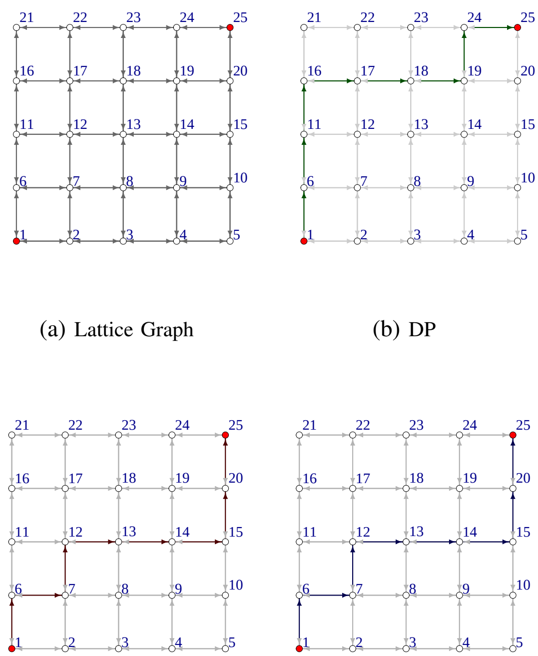
IV. Numerical Experiment

In this section, the proposed approach and algorithm is illustrated by numerical examples. Its advantage over other approaches is also shown by numerical experiments. Throughout this section, all the codes are implemented using R 3.1.3 and all the experiments are run on a laptop with an Intel Core i5-4300M CPU, 8GB ram and 64-bit Windows 8 OS.

In order to have a deep insight of our proposed method, numerical experiments is employed using R language. We generate a directed lattice graph with randomly generated weights from 10 to 25. And the uncertainty for every edge is 20% to 50% of the weight for corresponding edge, which means the upper bound and the lower bound are simply the weight plus or minus the corresponding uncertainty. So it is worth noting that the uncertainty is not the same as the definition of variance. The generated graph is shown in Fig. 2a. Vertex 1 is the departure point, and Vertex 25 is the destination point. In this specific instance, if we formulate this routing problem simply as a deterministic optimization problem (i.e., ignoring the weight uncertainty), the route obtained is shown in Fig 2b. If we formulate this VRP as a robust optimization problem defined in Problem 8, the problem should be much more complicated. The results using Algorithm 1 and Mixed Integer Linear Programming (depicted in Problem 9) are listed in Fig 2c and Fig 2d, respectively.

A. *Performance of binary search algorithm*

Based on this instance, the routes calculated by DP and TRO are totally different from each other. During numerical experiments, the routes obtained through binary search and MILP are always keeping the same, indicating that N = 7 is sufficient enough to find a route which is the exact the same route as the one calculated by MILP. But for problems in real world, the problem size is fairly big, and it is not practical to use MILP method to tackle these problems since the time complexity of MILP is NP-Complete. However, the time complexity of binary search, is simply several times using Dijkstra's algorithm (and the time complexity for Dijstra's



(a) Lattice Graph (b) DP

(c) TRO (d) LP

Fig. 2: Routes

algorithm is $\mathcal{O}(|E| + |V| \log |V|)$ based on min-priority queue implemented by a Fibonacci heap).

B. Robustness of travel time

1. distribution of travel times under TRO and DP In order to simulate real world routing problems, we generate 10 samples for each edge according to symmetric beta distribution, with randomly generated parameter $\alpha = \beta \in [0, 2]$. Following Fig. 4 shows a simulation of the distribution of the routing result. The green curve shows the route obtained by solving Problem 4, and the red curve represents the distribution calculated using binary search. From the figure, we could see that, although the mean travel time of red route is smaller than the one of the green route, the probability of not exceeding the travel time target for the red route is apparently bigger than the one for the green route, showing the robustness of the route calculated by our proposed method.

V. Conclusion

The conclusion goes here.

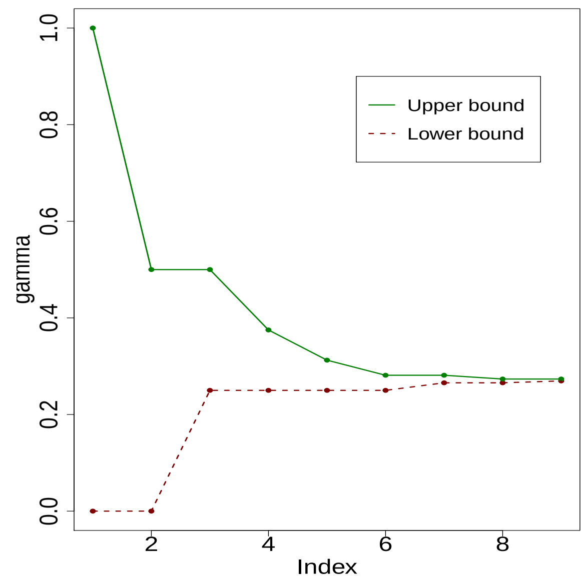


Fig. 3: Performance of binary search

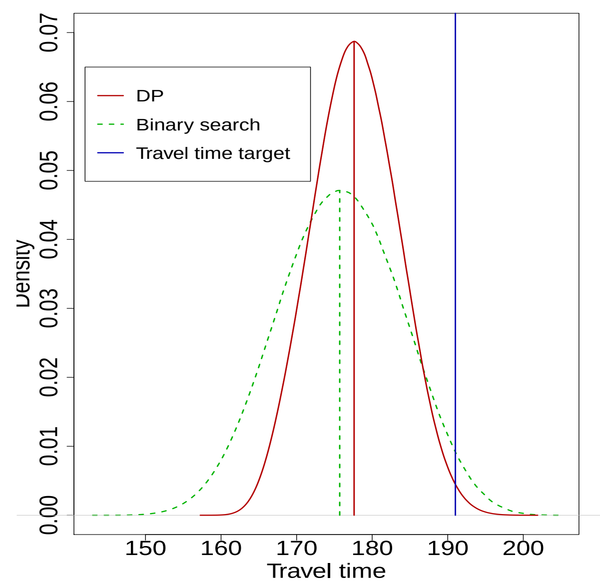


Fig. 4: Distribution

References

[1] R. Bellman, "On a routing problem," *Quarterly of Applied Mathematics*, vol. 16, pp. 87–90, 1958.

[2] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[3] S. E. Dreyfus, "An appraisal of some shortest-path algorithms," *Operations research*, vol. 17, no. 3, pp. 395–412, 1969.

[4] P. B. Mirchandani, "Shortest distance and reliability of probabilistic networks," *Computers & Operations Research*, vol. 3, no. 4, pp. 347–355, 1976.

[5] H. Frank, "Shortest paths in probabilistic graphs," *Operations Research*, vol. 17, no. 4, pp. 583–599, 1969.

[6] C. E. Sigal, A. A. B. Pritsker, and J. J. Solberg, "The stochastic shortest route problem," *Operations Research*, vol. 28, no. 5, pp. 1122–1129, 1980.

[7] V. G. Kulkarni, "Shortest paths in networks with exponentially distributed arc lengths," *Networks*, vol. 16, no. 3, pp. 255–274, 1986.

[8] K. J. Hayhurst and D. R. Shier, "A factoring approach for the stochastic shortest path problem," *Operations research letters*, vol. 10, no. 6,

pp. 329–334, 1991.

[9] G. A. Corea and V. G. Kulkarni, "Shortest paths in stochastic networks with arc lengths having discrete distributions," *Networks*, vol. 23, no. 3, pp. 175–183, 1993.

[10] R. P. Loui, "Optimal paths in graphs with stochastic or multidimensional weights," *Communications of the ACM*, vol. 26, no. 9, pp. 670–676, 1983.

[11] A. Eiger, P. B. Mirchandani, and H. Soroush, "Path preferences and optimal paths in probabilistic networks," *Transportation Science*, vol. 19, no. 1, pp. 75–84, 1985.

[12] I. Murthy and S. Sarkar, "A relaxation-based pruning technique for a class of stochastic shortest path problems," *Transportation science*, vol. 30, no. 3, pp. 220–236, 1996.

[13] I. Murthy and S. Sarkar, "Stochastic shortest path problems with piecewise-linear concave utility functions," *Management Science*, vol. 44, no. 11-part-2, pp. S125–S136, 1998.

[14] R. W. Hall, "The fastest path through a network with random time-dependent travel times," *Transportation science*, vol. 20, no. 3, pp. 182–188, 1986.

[15] L. Fu and L. R. Rilett, "Expected shortest paths in dynamic and stochastic traffic networks," *Transportation Research Part B: Methodological*, vol. 32, no. 7, pp. 499–516, 1998.

[16] E. D. Miller-Hooks and H. S. Mahmassani, "Least possible time paths in stochastic, time-varying networks," *Computers & operations research*, vol. 25, no. 12, pp. 1107–1125, 1998.

[17] E. D. Miller-Hooks and H. S. Mahmassani, "Least expected time paths in stochastic, time-varying transportation networks," *Transportation Science*, vol. 34, no. 2, pp. 198–215, 2000.

[18] B. C. Dean, "Algorithms for minimum-cost paths in time-dependent networks with waiting policies," *Networks*, vol. 44, no. 1, pp. 41–46, 2004.

[19] S. T. Waller and A. K. Ziliaskopoulos, "On the online shortest path problem with limited arc cost dependencies," *Networks*, vol. 40, no. 4, pp. 216–227, 2002.

[20] Y. Y. Fan, R. E. Kalaba, and J. E. Moore II, "Shortest paths in stochastic networks with correlated link costs," *Computers & Mathematics with Applications*, vol. 49, no. 9, pp. 1549–1564, 2005.

[21] M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing," *European Journal of Operational Research*, vol. 88, no. 1, pp. 3–12, 1996.

[22] X. Li, P. Tian, and S. C. H. Leung, "Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm," *International Journal of Production Economics*, vol. 125, no. 1, pp. 137–145, 2010.

[23] D. Taş, N. Dellaert, T. Van Woensel, and T. De Kok, "Vehicle routing problem with stochastic travel times including soft time windows and service costs," *Computers & Operations Research*, vol. 40, no. 1, pp. 214–224, 2013.

[24] Y. Y. Fan, R. E. Kalaba, and J. E. Moore II, "Arriving on time," *Journal of Optimization Theory and Applications*, vol. 127, no. 3, pp. 497–513, 2005.

[25] Y. Y. Fan and Y. Nie, "Optimal routing for maximizing the travel time reliability," *Networks and Spatial Economics*, vol. 6, no. 3, pp. 333–344, 2006.

[26] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operations Research*, vol. 23, no. 4, pp. 769–805, 1998.

[27] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operations Research Letters*, vol. 25, no. 1, pp. 1–13, 1999.

[28] A. Ben-Tal and A. Nemirovski, "Robust solutions of linear programming problems contaminated with uncertain data," *Mathematical Programming*, vol. 88, no. 3, pp. 411–424, 2000.

[29] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, no. 1, pp. 35–53, 2004.

[30] D. Bertsimas and M. Sim, "Tractable approximations to robust conic optimization problems," *Mathematical Programming*, vol. 107, no. 1, pp. 5–36, 2006.

[31] D. Bertsimas, D. Pachamanova, and M. Sim, "Robust linear optimization under general norms," *Operations Research Letters*, vol. 32, no. 6, pp. 510–516, 2004.

[32] L. El-Ghaoui and H. Lebret, "Robust solutions to least-squares problems with uncertain data," *SIAM Journal on Matrix Analysis and Applications*, vol. 18, no. 4, pp. 1035–1064, 1997.

[33] L. El-Ghaoui, F. Oustry, and H. Lebret, "Robust solutions to uncertain semidefinite programs," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 33–52, 1998.

[34] C. Wang, C.-J. Ong, and M. Sim, "Convergence properties of constrained linear system under MPC control law using affine disturbance feedback," *Automatica*, vol. 45, no. 7, pp. 1715–1720, 2009.