Submitted to *INFORMS Journal on Computing*
manuscript (Please, provide the mansucript number!)

Risk averse shortest paths: A computational study

Renaud Chicoisne
Universidad de Chile, renaud.chicoisne@gmail.com

Fernando Ordóñez
Universidad de Chile, elfordon@gmail.com

Daniel Espinoza
Universidad de Chile, daespino@gmail.com

In this work we consider the shortest path problem with uncertainty in arc lengths and convex risk measure objective. We explore efficient implementations of sample average approximation (SAA) methods to solve shortest path problems when the Conditional Value at Risk (CVaR) and Entropic risk measures (E) are used and there is correlation present in the uncertain arc lengths. Our work uses decomposition techniques to achieve an efficient implementation of SAA methods for these non-linear convex integer optimization problems. A computational study shows the effect of geometry, uncertainty correlation and dispersion, and risk measure parameters on efficiency and accuracy of the methods developed.

Key words: Stochastic Shortest Path, Sample Average Approximation, Conditional Value at Risk, Entropic Risk Measure

1. Introduction

Real life problems such as the vehicle routing problem (Laporte (1992)), the facility location problem (Drezner (1996)) or the lot sizing problem (Drexl and Kimms (1997), Jans and Degraeve (2008)) contain inherent shortest path problems as an important component. Given that deterministic shortest path sub-problems are solvable in polynomial time (Ahuja et al. (1993)), decomposition techniques are a natural option to tackle these problems in a fast way. Nevertheless, assuming deterministic data is unreasonable in a number of applications and doing so can lead to poor solutions in practice (Bertsimas and Sim (2004)). Consequently, other ways to evaluate the cost of a path that take into account its uncertainty are necessary. Unfortunately, doing so spoils the special properties of the classical shortest path problem and leads to problems that are more difficult to solve. For

instance, the shortest path problem satisfies total unimodularity, which guarantees that when we use a linear objective, the linear programming relaxation reaches an integer optimal solution. Nevertheless, taking into account uncertainty leads to a convex objective that cannot make use of this property. Previous work has put a lot of effort into methods to incorporate uncertainty in optimization problems (Avriel and Williams (1970), Tintner (1955), Wets (1966)). For an introduction to stochastic programming in general, the reader may refer to Birge and Louveaux (2011), Shapiro et al. (2014) or Ben-Tal et al. (2009) for an introduction to robust optimization. Specific methods have been developed for optimization problems under uncertainty exploiting particular problem structure. For instance, the lot sizing problem with uncertain costs or lead times in Atamtürk and Zhang (2007), the vehicle routing problem with time windows and uncertain data in Ghiani et al. (2003) and the facility location problem with uncertainty in Snyder (2006).

Taking the expected value of the cost of a path appears as a natural way to quantify the uncertain travel time of the path but ignores the dispersion of the underlying distributions, which can be significant. For example, consider two nodes s and t connected by two paths. The first path has a fixed travel time of 6 minutes and the second one has an uncertain travel time, it can be 1 minute or 9 minutes, each with probability 0.5. Then, in expected value, the best choice is the second path with an expected travel time of 4.5 minutes, but it has the highest variability between the two possibilities. Taking the second path is riskier with a 50% delay half of the time. This amount of delay can be unacceptable in certain applications, such as emergency dispatching systems, where increases in delay can lead to increases in fatalities. This notion of risk aversion was mathematically defined in the early work of Von Neumann and Morgenstern (1944) where the risk perceived by a rational agent is modeled by the expected value of their convex disutility function. Such a way to define risk is also called certainty equivalent under some utility function (Yaari (1969, 1987)).

A risk measure $\rho(\cdot)$ is a real valued mapping of a set of random variables used to quantify the degree of risk aversion of a random variable. There are some desirable features a risk measure could have. It should be normalized, i.e.: the zero random variable has zero risk, invariant by translation, i.e.: when minimizing a random cost X, the risk associated to the random variable $X + a$ for any real a is exactly $\rho(X) + a$, and finally should be monotone, i.e.: if a random variable X dominates stochastically another random variable Y (Levy

(2006)) which we write $X \succeq Y$, then we have $\rho(X) \leqslant \rho(Y)$ in the case of a minimization problem, and $\rho(X) \geqslant \rho(Y)$ in a maximization context. Convexity is also a desired property of a risk measure. For example, when optimizing a portfolio mixing different assets is a good way to reduce the dispersion of a portfolio in practice. The latter can be modeled via the convexity of the risk measure considered. The expected value or the CVaR of a random variable are typical examples of convex risk measures. The concept of coherent risk measures axiomatically defines the set of properties that a risk measure should satisfy (Artzner et al. (1999)), and shows that CVaR is a coherent risk measure. In Kusuoka (2001) they prove that CVaR is central amongst coherent risk measures, showing that any coherent, co-monotone and law invariant risk measure - or distortion risk measures - is equivalently representable as a convex combination of CVaRs at different levels of security.

An early work presenting a solution approach for a risk averse problem appears in Markowitz (1952), where they use the variance to represent the dispersion of the data, giving birth to the mean-risk model. Nikolova et al. (2006), Lim et al. (2011) used this latter model to solve CVaR minimization shortest path problems with uncorrelated and normal distributions on edge travel times. Rockafellar and Uryasev (2000) proved that minimizing a CVaR objective is a standard convex optimization problem. Nevertheless, this requires the minimization of an expectation, which can even be difficult to compute for a large number of correlated random variables. Sample average approximation (SAA) methods (Kleywegt et al. (2002)) provide a standard framework to approximately solve these difficult problems. To the best of our knowledge, all the literature on risk averse shortest path problems assumes uncorrelated uncertain travel times (or costs). Ignoring the correlation that may be present in random travel times might return shortest paths solutions failing to consider the path travel time distribution (Agrawal et al. (2012, 2010)). As shown in Yang et al. (2013) the correlation in uncertain travel times is indeed present in transportation networks.

In figures 2 and 1 we see an illustration of the influence of correlations on the shortest path problem. Let $X, Y, Z \sim \mathcal{B}(1, 3)$ be Bernoulli random variables that can take the values 1 or 3 with the same probability 0.5. The risk associated to a path P in this example will be the mean-risk objective, given by the sum of the expected value and variance over path P, explicitly $\rho(P) = \mathbb{E}[P] + \sigma^2[P]$. Let P_1 and P_2 be the paths passing respectively through the upper edge and the lower edge. In a first case (figure 1) there is no correlation between

the three edges. We have $\mathbb{E}(P_1) = 4$, $\mathbb{E}(P_2) = 3$ and $\sigma^2(P_1) = \sigma^2(P_2) = 2$. The risk associated to the path P_1 is then $\rho(P_1) = 6$ while the risk of the lower path is $\rho(P_2) = 5$, making the second one preferable. In the second case (figure 2) the distributions remain the same, but

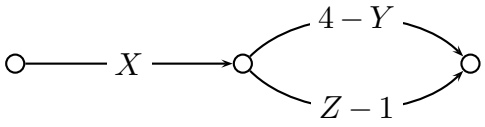


Figure 1 **Without considering correlations**

we assume there is full correlation between the edges' travel times, i.e.: $X = Y = Z$. We have $\mathbb{E}(P_1) = 4$, $\mathbb{E}(P_2) = 3$, $\sigma^2(P_1) = 0$ and $\sigma^2(P_2) = 4$. The risk associated to the path P_1 passing through the upper edge is $\rho(P_1) = 4$ while the risk of the lower path is $\rho(P_2) = 7$ meaning that the first one is chosen. In this example we can see that ignoring correlations

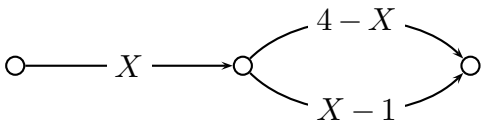


Figure 2 **Considering correlations**

can lead to take a path that is 75% worse than the optimal one. SAA methods can handle correlations between the random variables. Nevertheless, it may sometimes require large samples to guarantee the quality of the solution obtained (Lim et al. (2011)) at the cost of solving larger optimization problems.

In this chapter, we assume a given directed network $G = (V, E)$ with $m = |E|$ edges, $n = |V|$ nodes, a source node $s \in V$, and a sink node $t \in V$. We define the set of paths from s to t on G with the standard st-path polytope (Ahuja et al. (1993)) below

$$x \in X \;=\; \left\{ x \in \{0,1\}^m \;:\; \sum_{k:(j,k)\in E} x_{jk} - \sum_{i:(i,j)\in E} x_{ij} = b_j \quad \forall j \in V \right\}, \tag{1}$$

where the right hand side vector b satisfies $b_s = 1$, $b_t = -1$, and $b_j = 0$ for all $j \in V \setminus \{s, t\}$. The uncertain shortest path problem considers uncertain costs c_{ij} on every edge $(i, j) \in E$ such that the cost vector $c = (c_{ij})_{(i,j)\in E}$ follows some distribution that we can sample. We

make no further assumptions regarding the uncertain cost vector, in particular we do not assume the uncertainty between arcs is independent.

Given an *st*-path solution $x \in X$ and an uncertain cost vector c, its total cost $c^\top x$ is a random variable that we can evaluate with a convex risk measure with $\rho\left(c^\top x\right)$. The problem considered in this work is to find the *st*-path which minimizes this risk aversion measure, namely

$$(P_\rho) \qquad w^* = \min_{x \in X} \rho\left(c^\top x\right) \ . \tag{2}$$

In this chapter, we will only consider risk measures of the form $\rho\left(c^\top x\right) = \mathbb{E}\left[u\left(c^\top x\right)\right]$ with u some positive, nondecreasing and strictly convex disutility function such that $\rho(0) = 0$. At least with this type of risk measures, we can prove that problem (2) is NP-hard by reducing it to the set partition problem: Given a set of N positive integers $(a_i)_{i \in \{1,...,N\}}$, we want to find a partition $\{N_1, N_2\}$, $N_1 \cap N_2 = \emptyset$, $N_1 \cup N_2 = \{1, ..., N\}$ such that $\sum_{i \in N_1} a_i = \sum_{i \in N_2} a_i$. Assuming that the a_i are "deterministic random variables", finding the shortest path in

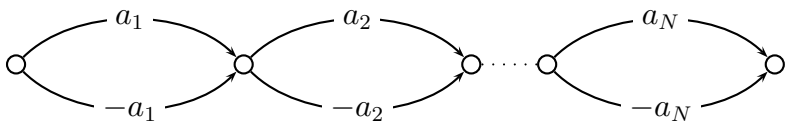


Figure 3 Reduction to Set Partition Problem

the network depicted in figure 3 with respect to the quadratic risk measure $\left(c^\top x\right)^2$ solves the set partition problem. Indeed, for this construction we have that this shortest path problem has an optimal solution of value 0 if and only if the parameters $(a_i)_{i \in \{1,...,N\}}$ define a solvable instance of the partition problem. Further, if the instance is not solvable then the optimal path defines the partition minimizing the difference of weights between N_1 and N_2.

In general, even the evaluation $\rho\left(c^\top x\right)$ of some solution $x \in X$ can be challenging. We can approximate this optimization problem using SAA methods if we can sample the uncertain total cost. In this chapter we propose efficient implementations of the SAA methodology to solve large instances of the risk averse shortest path problem (P_ρ) in (2). In particular we focus on algorithms to solve problem (P_ρ) when using the Conditional Value at Risk measure and the Entropic risk measure.

In the next section we present basic definitions and the general framework of the SAA methodology. We present the different formulations and algorithms that we implemented to solve (P_ρ) under each risk measure in section 2. This section states the SAA problems that we have to solve and introduces the decomposition algorithms implemented for each risk measure. We discuss simplifications that are possible when considering the entropic risk measure with uncorrelated uncertainty. We present the computational results in section 4 and conclude in section 5.

2. Solution Approaches Proposed

In the following, we assume that we can generate samples of the random variable c. We focus our attention on SAA methods to solve these problems. Given a sample of S realizations of c, $(c^s)_{s\in\{1,\dots,S\}}$ of respective probabilities $(p_s)_{s\in\{1,\dots,S\}}$, we can approximate the risk of a solution $x \in X$ as follows:

$$\mathbb{E}\left[u\left(c^\top x\right)\right] \approx \sum_{s=1}^{S} p_s u\left(\left(c^s\right)^\top x\right) \tag{3}$$

Consequently, the sample average approximation of (2) can be written:

$$(P_S) \quad w_S := \min_{x\in X}\left\{ w_S(x) := \sum_{s=1}^{S} p_s u\left(\left(c^s\right)^\top x\right)\right\} \tag{4}$$

In our study, we will consider two specific risk measures for the random cost $c^\top x$: the Conditional Value at Risk which for a given tolerance $\epsilon \in]0,1]$ can be written as described in Rockafellar and Uryasev (2000):

$$\mathrm{CVaR}_\epsilon(c^\top x) = \min_{z\in\mathbb{R}}\left\{ z + \epsilon^{-1}\mathbb{E}\left[\left(c^\top x - z\right)_+\right]\right\} \tag{5}$$

and the Entropic Risk Measure (Pratt (1964), Arrow (1965)) at level $\alpha \in \mathbb{R}$ that can be defined as:

$$\mathrm{E}_\alpha\left(c^\top x\right) = \alpha \ln \mathbb{E}\left[e^{\frac{c^\top x}{\alpha}}\right] \tag{6}$$

Notice that the entropic risk measure is a risk aversion measure whenever $\alpha > 0$, a risk seeking measure when $\alpha < 0$ and models a risk neutrality as $\mathrm{E}_\alpha\left(c^\top x\right) \underset{\alpha\to 0}{\longrightarrow} \mathbb{E}\left[c^\top x\right]$. We remark that minimizing risk measure (6) is equivalent to minimizing $\mathbb{E}\left[e^{\frac{c^\top x}{\alpha}}\right]$. This allows us to use the SAA formulation (4) to solve the optimization problem. The CVaR risk measure is equivalent to the minimization problem (5). This shows that it is the minimization of the expectation of some function.

2.1. Conditional Value at Risk

Monolithic form (CM). A first way to solve the CVaR minimization problem is to directly solve its associated sampled problem (7):

$$(\text{CVaR}_\epsilon^S) \quad w_S := \quad \min_{x\in X}\left\{\text{CVaR}_\epsilon^S(c^\top x) := \min_{z\in\mathbb{R}} z + \epsilon^{-1}\sum_{s=1}^{S} p_s\left((c^s)^\top x - z\right)_+\right\} \tag{7}$$

which is equivalent to the following MIP formulation:

$$w_S := \quad \min_{z,\eta,x} z + \epsilon^{-1}\sum_{s=1}^{S}\eta_s \tag{8a}$$

$$(\text{CVaR}_\epsilon^S) \qquad \text{s.t. } : x \in X \tag{8b}$$

$$\eta_s \geqslant p_s\left((c^s)^\top x - z\right) \qquad \forall s\in\{1,...S\} \tag{8c}$$

$$\eta \geqslant 0 \tag{8d}$$

Given some feasible solution $x \in X$, an important remark done in Espinoza and Moreno (2014) provides a quick way to compute an estimated value $\text{CVaR}_\epsilon^S(c^\top x)$ of $\text{CVaR}_\epsilon(c^\top x)$. Indeed, given a sample $(c^s, p_s)_{s\in\{1,...,S\}}$ we have:

$$\text{CVaR}_\epsilon^S(c^\top x) := \min_{\eta,z}\left\{z + \epsilon^{-1}\sum_{s=1}^{S} p_s\eta_s : \eta_s \geqslant (c^s)^\top x - t, \eta_s \geqslant 0, t\in\mathbb{R}\right\} \tag{9}$$

when taking the dual and changing variables we obtain:

$$\text{CVaR}_\epsilon^S(c^\top x) := \max_{\pi}\left\{\epsilon^{-1}\sum_{s=1}^{S} p_s\pi_s\,(c^s)^\top x : \sum_{s=1}^{S} p_s\pi_s = \epsilon, \pi\in[0,1]^S\right\} \tag{10}$$

We can solve this second problem sorting the scenarios by decreasing $c^s x$ and fill the equality constraint with the best scenarios. Let suppose we have such an ordering $c^{(1)}x \geqslant c^{(2)}x \geqslant ... \geqslant c^{(S)}$ and let $s^* := \max\left\{s' \in \{1,...,S\} : \sum_{s=1}^{s'} p_{(s)} \leqslant \epsilon\right\}$. Then it is easy to see that an optimal solution for the dualized problem (10) is:

$$\pi_s^* = \begin{cases} 1 & \forall s\in\{(1),...,(s^*)\} \\ \epsilon - \sum_{s'=1}^{s^*} p_{(s')} & s = (s^*+1) \\ 0 & \forall s\in\{(s^*+2),...,(S)\} \end{cases} \tag{11}$$

We can compute this optimal solution in $O(S \ln S)$ time and extract an upper bound on the optimal value of problem (8) from it. Based on the solution given by (11) we define the partition induced by x as:

$$\mathcal{Q}_x := \{\{(1), ..., (s^*)\}, \{(s^*+1)\}, \{(s^*+2), ..., (S)\}\} \tag{12}$$

Although we are able to cast the problem directly as a MIP and evaluate its estimated value quickly, we expect this monolithic form to be slow to solve in practice due to the large number of scenarios we must potentially generate to have a good approximation. To tackle this, we will present in the following two algorithms tailored to solve the monolithic formulation: a subgradient algorithm and a decomposition based method.

Subgradient (CS). As suggested in Künzi-Bay and Mayer (2006), given that the functions $\rho(c^\top \cdot) : x \longmapsto \rho\left(c^\top x\right)$ are convex, we can represent them as the point-wise maximum of its subgradients. Replacing in problem (2) we obtain formulation (13):

$$w^* := \min_{\eta, x} \quad \eta \tag{13a}$$

$$(\partial P_\rho) \quad \text{s.t.} : x \in X \tag{13b}$$

$$\eta \geqslant \rho\left(c^\top x^0\right) + d^\top (x - x^0) \qquad \forall x^0 \in X, \forall d \in \partial[\rho(c^\top \cdot)](x^0) \tag{13c}$$

Which has an infinite number of constraints in general. Nevertheless, the piecewise linearity of CVaR allows to cast the latter problem (13) into the following formulation (14):

$$w_S := \min_{z, x, \eta} \quad z + \epsilon^{-1}\eta \tag{14a}$$

$$\left(\partial \text{CVaR}^{\mathcal{S}}\right) \quad \text{s.t.} : x \in X \tag{14b}$$

$$\eta \geqslant \sum_{s \in C} p_s \left(\left(c^s\right)^\top x - z\right) \qquad \forall C \in \mathcal{S} \tag{14c}$$

Where $\mathcal{S}$ is the set of all subsets of $\{1, ..., S\}$. This suggests an iterative algorithm solving relaxed problems and generating cuts whenever constraints (14c) are violated. We first solve a relaxation of problem (14) containing only one constraint of type (14c) with $\mathcal{S} = \{\{1, ..., S\}\}$. Let $(\widetilde{x}, \widetilde{z}, \widetilde{\eta})$ be the current optimal solution of problem (14) of objective value $\widetilde{w}$. We remark that $(\widetilde{x}, \widetilde{z})$ is feasible for (8), so we have $\widetilde{w} \leqslant \widetilde{z} + \epsilon^{-1} \sum_{s=1}^{S} \left(\left(c^s\right)^\top \widetilde{x} - \widetilde{z}\right)_+$. Let us define $\widetilde{C} = \left\{s \in \{1, ..., S\} : \left(c^s\right)^\top \widetilde{x} - \widetilde{z} > 0\right\}$. If $\widetilde{w} \geqslant \sum_{s \in \widetilde{C}} p_s \left[\left(c^s\right)^\top \widetilde{x} - \widetilde{z}\right]$ then we can easily see that the current solution $(\widetilde{x}, \widetilde{z})$ is feasible and optimal for the complete problem (7). In

the other case we add the constraint corresponding to $\widetilde{C}$ and we solve again the resulting problem with $\mathcal{S} \leftarrow \mathcal{S} \cup \widetilde{C}$. We remark that the first step with $\mathcal{S} = \{\{1, ..., S\}\}$ is equivalent to solve the original problem (2) with $\rho \equiv \mathbb{E}$, which is as hard as solving the deterministic version. In our context of a shortest path problem, any existing polynomial algorithm like Dijkstra's solves it efficiently. We show a pseudo code of this method in Algorithm 1.

Algorithm 1: Pseudocode of CS algorithm

Data: A problem (CVaR_ϵ^S)

Result: A δ-optimal solution $\widetilde{x}$ of (CVaR_ϵ^S)

Generate S equiprobable samples c^s;

$\mathcal{S} \leftarrow \{1, ..., S\}$;

repeat

  compute the optimal solution $\widetilde{x}$ of $(\partial\text{CVaR}_\epsilon^{\mathcal{S}})$ with objective value $\widetilde{w}$;

  $\widetilde{C} = \{s \in \{1, ..., S\} : c^s \widetilde{x} - \widetilde{z} > 0\}$;

  $\mathcal{S} \leftarrow \mathcal{S} \cup \widetilde{C}$;

until $\text{CVaR}_\epsilon^S(c^\top \widetilde{x}) - \widetilde{w} \leqslant \delta$;

return $\widetilde{x}$

Aggregation scheme (CA). In Espinoza and Moreno (2014) the authors developed a computationally fast aggregation technique to solve the CVaR minimization problem in the case of continuous linear programming problems (i.e.: when X is a polyhedron). We show that their approach is valid as well for any feasible set $X \subseteq \mathbb{R}^m$. Let $\mathcal{Q} = \left\{\mathcal{Q}^1, ..., \mathcal{Q}^Q\right\}$ be a partition of $\{1, ..., S\}$. Summing of all type (8c) constraints corresponding to the scenarios of a single bundle Q^q we obtain the following aggregated constraint:

$$\sum_{s \in \mathcal{Q}^q} \eta_s \geqslant \sum_{s \in \mathcal{Q}^q} p_s \left((c^s)^\top x - z \right) \qquad \forall q \in \{1, ..., Q\}$$

Defining $\widetilde{\eta}_q = \sum_{s \in \mathcal{Q}^q} \eta_s$, $\widetilde{p}_q = \sum_{s \in \mathcal{Q}^q} p_s$ and $\widetilde{c}^q = \widetilde{p}_q^{-1} \sum_{s \in \mathcal{Q}^q} p_s c^s$, we can write this last constraint as follows:

$$\widetilde{\eta}_q \geqslant \widetilde{p}_q \left[(\widetilde{c}^q)^\top x - z \right] \qquad \forall q \in \{1, ..., Q\}$$

Given that the $\widetilde{\eta}_s$ are built upon a partition the following problem is a relaxation of (8):

$$\widetilde{w}_{\mathcal{Q}} := \quad \min_{z,\widetilde{\eta},x} z + \epsilon^{-1} \sum_{q=1}^{Q} \widetilde{\eta}_q \tag{15a}$$

$$(\mathrm{CVaR}_{\epsilon}^{\mathcal{Q}}) \quad \text{s.t.} : x \in X \tag{15b}$$

$$\widetilde{\eta}_q \geqslant \widetilde{p}_q \left((\widetilde{c}^q)^\top x - z \right) \qquad \forall q \in \{1, ..., Q\} \tag{15c}$$

$$\widetilde{\eta} \geqslant 0 \tag{15d}$$

Consequently we have $\widetilde{w}_{\mathcal{Q}} \leqslant w_S$, which is tight when the optimal solution $\widetilde{x}$ of (15) is optimal as well for (8). On the contrary, when $\widetilde{w}_{\mathcal{Q}} < w_S$, we can refine the constraints (15c) when evaluating the optimal solution of problem (15). Espinoza and Moreno (2014) propose the following refinement scheme: start with $\mathcal{Q} = \{\{1, ..., S\}\}$ and solve $(\mathrm{CVaR}_{\epsilon}^{\mathcal{Q}})$, then get its optimal solution $\widetilde{x}$'s estimated objective value $\mathrm{CVaR}_{\epsilon}^{S}(c^\top \widetilde{x})$ and its induced partition $\mathcal{Q}_{\widetilde{x}}$. If $\mathrm{CVaR}_{\epsilon}^{S}(c^\top \widetilde{x}) - \widetilde{w}_{\mathcal{Q}} \leqslant 10^{-6}$, we stop because $\widetilde{x}$ is optimal with respect to machine precision. Otherwise we refine the partition $\mathcal{Q} \leftarrow \{\mathcal{Q}_{\widetilde{x}}^i \cap \mathcal{Q}^j : \mathcal{Q}_{\widetilde{x}}^i \in \mathcal{Q}_{\widetilde{x}}, \mathcal{Q}^j \in \mathcal{Q}\}$ and iterate. We remark that when refining $\mathcal{Q}$ we always obtain another partition of $\{1, ..., S\}$ with a greater or equal number of subsets. Furthermore, if refining the partition does not change its cardinality we can deduce that for each $\mathcal{Q}_j \in \mathcal{Q}$ we have $\exists i \in \{1, 2, 3\} : \mathcal{Q}^j \subset \mathcal{Q}_{\widetilde{x}}^i$. This implies that the solution $(\widetilde{x}, \widetilde{z}, \bar{\eta})$ with $\bar{\eta}_s := \widetilde{\eta}_q$, $\forall s \in \mathcal{Q}_q$ is feasible for (8) and has the same objective value as the relaxed problem (15), implying its optimality.

The algorithm solves at most S aggregated problems until $\mathcal{Q} = \{1, ..., S\}$, where the problem becomes exactly the one in (8). Aside from this worst case, we expect from this procedure to solve only a few problems with reduced size instead of a brute force resolution of the complete problem. In Espinoza and Moreno (2014) they show that for LP problems this procedure can be several orders of magnitude better than standard methods when using LP-tailored approaches. We show a pseudo code of this algorithm in Algorithm 2.

Our implementation of these two methods (CA and CS) considers the case when X has integer solutions, given by the shortest path polytope. We tested two different frameworks: one solving directly the MIP, and another one solving first the LP relaxation - obtaining a partition for CA and a set of subgradient cuts for CS - and then solve the MIP with the same algorithm but starting with the constraints added during the first step for CS or the partition returned by the LP for CA. We will denote CA/IP(CS/IP) the direct resolution of the integer programming problem and CA/LP+IP(CS/LP+IP) the latter way.

Algorithm 2: Pseudocode of CA algorithm

Data: A problem (CVaR_ϵ^S)

Result: A δ-optimal solution $\widetilde{x}$ of (CVaR_ϵ^S)

Generate S equiprobable samples c^s;

$\mathcal{Q} \leftarrow \{\{1, ..., S\}\}$;

repeat

  compute the optimal solution $\widetilde{x}$ of $(\text{CVaR}^{\mathcal{Q}})$ with objective value $\widetilde{w}$;

  $\mathcal{Q} \leftarrow \{\mathcal{Q}_x^i \cap \mathcal{Q}^j : \mathcal{Q}_x^i \in \mathcal{Q}_x, \mathcal{Q}^j \in \mathcal{Q}\}$;

until $\text{CVaR}_\epsilon^S(c^\top \widetilde{x}) - \widetilde{w} \leqslant \delta$;

return $\widetilde{x}$

2.2. Entropic Risk Measure

We want to solve the following optimization problem:

$$\min_{x \in X} \text{E}_\alpha \left(c^\top x\right) := \alpha \ln \mathbb{E}\left[e^{\frac{c^\top x}{\alpha}}\right] \tag{16}$$

Uncorrelated case In the case where the variables are uncorrelated, the problem (16) can be solved with a single run of any shortest path algorithm. Indeed, if random variables are independent, the expected value of their product is the product of their expected values and we have:

$$\text{E}_\alpha \left(c^\top x\right) = \alpha \sum_{i=1}^{m} \ln \mathbb{E}\left[e^{\frac{c_i x_i}{\alpha}}\right]$$

Noticing that:

$$\ln \mathbb{E}\left[e^{\frac{c_i x_i}{\alpha}}\right] = \begin{cases} 0 & \text{If } x_i = 0 \\ \ln \mathbb{E}\left[e^{\frac{c_i}{\alpha}}\right] & \text{If } x_i = 1 \end{cases}$$

we can solve (16) by solving the equivalent following problem:

$$\min_{x \in X} \alpha \sum_{i=1}^{m} x_i \ln \mathbb{E}\left[e^{\frac{c_i}{\alpha}}\right]$$

which is a standard shortest path problem with arc costs given by $\alpha \ln \mathbb{E}\left[e^{\frac{c_i}{\alpha}}\right]$. This expected value can be approximated by the following SAA formulation:

$$\min_{x \in X} \left\{ \text{E}_\alpha^S \left(c^\top \widetilde{x}\right) := \alpha \sum_{i=1}^{m} x_i \ln \sum_{s=1}^{S} p_s e^{\frac{c_i^s}{\alpha}} \right\} \tag{17}$$

In practice, C programming language's 'double' type accepts numbers up to 10^{308} so the sums $\sum_{s=1}^{S} p_s e^{\frac{c_i^s}{\alpha}}$ cannot be allowed to be greater than 10^{308}. In order to give more slack to the values of α used, i.e.: be able to tackle cases with smaller values of α, for each edge $i \in \{1,...,m\}$ we compute $\bar{c}_i = \max_{s\in\{1,...,S\}} c_i^s$ which can be used to write the following equivalence:

$$\mathrm{E}_\alpha^S\left(c^\top \widetilde{x}\right) = \alpha \ln\left[\sum_{s=1}^{S} p_s e^{\frac{c_i^s}{\alpha}}\right] = \bar{c}_i + \alpha \ln\left[\sum_{s=1}^{S} p_s e^{\frac{c_i^s - \bar{c}_i}{\alpha}}\right] \tag{18}$$

Noticing that $\frac{c_i^s - \bar{c}_i}{\alpha} \leqslant 0$, the exponential values will then always be below one, turning the problems numerically tractable.

We will not consider this formulation further in our work as it heavily relies on the absence of correlation between the random variables. Nevertheless, the scaling method we just described will be extensively used in the next algorithms.

Subgradient (ES). Similar to CVaR, given that the function $\mathrm{E}_\alpha(c^\top \cdot) : x \longmapsto \alpha \ln \mathbb{E}\left[e^{\frac{c^\top x}{\alpha}}\right]$ is convex, so we can represent it as the point-wise maximum of its tangent planes. We can write the subgradient-cuts formulation of the SAA version of (16) as follows:

$$\min_{x\in X, t\in\mathbb{R}} \left\{ t : t \geqslant \alpha \ln\left[\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^0}{\alpha}}\right] + \left(\frac{\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^0}{\alpha}} c^s}{\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^0}{\alpha}}}\right)^\top \left(x - x^0\right), \quad \forall x^0 \in X \right\} \tag{19}$$

In contrast with CVaR, formulation 19 has an infinite number of constraints. We can use the iterative Algorithm 1 to solve the problem with a given tolerance. Indeed, at each iteration k we can add the subgradient cut corresponding to the current incumbent solution x^k and compare the lower bound it returns with the sampled value of the incumbent solution to evaluate the optimality gap. Again, there are scaling issues with the computation of the entropy and its gradient values so the cut we add at each iteration can be equivalently transformed to:

$$t \geqslant m_k + \alpha \ln\left[\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^k - m_k}{\alpha}}\right] + \left(\frac{\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^k - m_k}{\alpha}} c^s}{\sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x^k - m_k}{\alpha}}}\right)^\top \left(x - x^k\right) \tag{20}$$

With $m_k := \max_{s\in\{1,...,S\}} \left\{ (c^s)^\top x^k \right\}$. As for CVaR we notice that the first step with only one gradient cut is equivalent to solving a deterministic version of the original problem and

we can solve it with Dijkstra's algorithm because of the positiveness of the gradient of the entropic risk measure. As for algorithms CS and CA, we tested the IP and LP+IP frameworks for ES.

Exponential approximation (EEA). Since $\alpha \ln(\cdot)$ is nondecreasing we have that minimizing the entropic risk measure is equivalent to solving the following problem:

$$\min_{x \in X} \sum_{s=1}^{S} p_s e^{\frac{(c^s)^\top x}{\alpha}} \tag{21}$$

This last method consists in approximating each of the exponentials rather than the entire function. The classical way to approximate a convex function is to define *a priori* a set $(t_k)_{k \in \{0,\dots,K\}}$ of $K+1$ points on the domain $[l,u]$ of the function to approximate such that $l = t_0 < t_1 < \dots < t_{K-1} < t_K = u$. Adding an extra continuous variable for each scenario, we approximate each exponential by the point-wise maximum of its tangents in the discretization points as follows:

$$\widetilde{w}_K := \quad \min_{z,x} \sum_{s=1}^{S} p_s z_s \tag{22a}$$

$$(E_K^S) \quad \text{s.t. } : x \in X \tag{22b}$$

$$z_s \geqslant e^{t_k} \left[1 + \frac{(c^s)^\top x}{\alpha} - t_k \right] \qquad \forall s \in \{1,\dots,S\}, \forall k \in \{1,\dots,K\} \tag{22c}$$

Which is a relaxation of the original problem since tangents are always under the function for convex functions. One can choose the discretization such that the error of approximation is arbitrarily small at the cost of the number of breakpoints. Nevertheless, there are several reasons to limit the number of breakpoints for our discretization: 1) each scenario must have its own discretization, adding together $S(K+1)$ extra constraints and S extra continuous variables, 2) the original path polyhedron can already be large, 3) the interval $[l,u]$ can be particularly large if the random variables have a high variability and 4) scaling issues can be prohibitive: as u grows e^{t_k} can be out of the acceptable range of modern computing abilities.

Our approach consists in premultiplying the whole objective function by $e^{-\frac{M}{\alpha}}$ with $(c^s)^\top x \leqslant M, \forall x \in X, \forall s \in \{1,...,S\}$ and use an approximation of $t \longmapsto e^t$ on $]-\infty, 0]$ resulting in the following formulation:

$$\widetilde{w}_K(M) := \min_{z,x} \sum_{s=1}^{S} p_s z_s \tag{23a}$$

$$(E_K^S(M)) \quad \text{s.t.} : x \in X \tag{23b}$$

$$z_s \geqslant e^{t_k}\left[1 + \frac{(c^s)^\top x - M}{\alpha} - t_k\right] \qquad \forall s \in \{1,...,S\} \qquad \forall k \in \{1,...,K\} \tag{23c}$$

We notice that the exponential can be well approximated over $]-\infty, 0]$ with a fairly small number of points. Using the points $t_k := 2\ln\left(\frac{k}{K}\right)$ with $K = 10$ gives an approximation of maximum absolute error of $6.15 \cdot 10^{-3}$. Due to the structure of the exponential function, we have that the optimal solution of formulation (23) gives a lower bound for problem (16). Indeed, we have that:

$$e^{\frac{M}{\alpha}}\widetilde{w}_K(M) = \min_x \sum_{s=1}^{S} p_s \max_{k\in\{1,...,K\}} \left\{ e^{t_k + \frac{M}{\alpha}}\left[1 + \frac{(c^s)^\top x}{\alpha} - \left(t_k + \frac{M}{\alpha}\right)\right]\right\} \tag{24}$$

Which is a piecewise linear lower approximation of the exponential function with points $\left(t_k + \frac{M}{\alpha}\right)_{k\in\{1,...,K\}}$. Further, given that $\sum_{s=1}^{S} p_s = 1$ the absolute error of the approximated objective function is the same as the individual absolute error across the scenarios. Figure 4 illustrates the approximations done here.

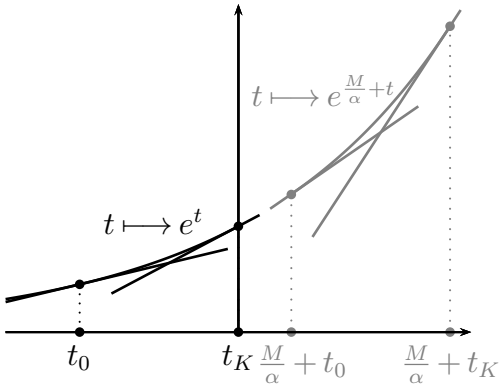


Figure 4 **Lower piecewise linear approximations of $t \longmapsto e^t$ and $t \longmapsto e^{\frac{M}{\alpha}+t}$**

Given that $e^{-M} > 0$, the resulting problem to solve is equivalent to (22). Still, although this approach has no scaling issues it loses numerical precision in practice because of big

values of M. We can see that taking $M^* := \max_{s\in\{1,...,S\}} (c^s)^\top x^*$ with x^* the optimal solution of the real problem (16), then solving (23) must return x^* as every $\frac{(c^s)^\top x - M^*}{\alpha}$ is negative, i.e.: where the exponential is correctly approximated. This suggests an iterative search of the value of M^*, successively solving (23) and updating $M := \max_{s\in\{1,...,S\}} (c^s)^\top \widetilde{x}$ with $\widetilde{x}$ the last optimal solution. We repeat the process until the gap between the estimated entropic value of $\widetilde{x}$, $\mathrm{E}_\alpha^S\left(c^\top \widetilde{x}\right)$ and $M + \alpha \ln \widetilde{w}_K(M)$ is smaller than a given tolerance δ. In practice, there is a last computational issue with the lower bound returned by the approximated problem (23). Actually, CPLEX solver considers zero every value smaller than 10^{-6} so in reality the solver stops whenever the reduced costs are smaller than this tolerance. Hence, there are two issues with the solution returned: 1) The absolute gap of the solution returned by the relaxed problem is a $10^{-6}e^{\frac{M}{\alpha}}$ approximation of the linear approximation problem and hence, can be suboptimal and can fail to provide a lower bound. and 2) The approximation provides a solution of objective value zero hence providing a $-\infty$ lower bound. This can be tackled considering that the value $e^{\frac{M}{\alpha}}\widetilde{w}_K(M)$ is a genuine lower bound whenever $\widetilde{w}_K(M) > 10^{-6}$. Because of these computational problems, there is no guarantee that EEA returns an optimal solution for (E_α^S). We show a pseudo code in Algorithm 3.

Algorithm 3: Pseudocode of EEA algorithm

Data: A problem (E_α^S)

Result: A feasible solution $\widetilde{x}$ of (E_α^S)

Generate S equiprobable samples c^s;

$\widetilde{x} \in \arg\min\left\{\sum_{s=1}^{S} p_s (c^s)^\top x : x \in X\right\}$;

$M \leftarrow \max\left\{(c^s)^\top \widetilde{x} : s \in \{1,...,S\}\right\}$;

repeat

  Solve $(\mathrm{E}_K^S(M))$ of 'optimal' solution $\widetilde{x}$ and objective value $\widetilde{w}$;

  $M \leftarrow \max_{s\in\{1,...,S\}} (c^s)^\top \widetilde{x}$;

until $(\widetilde{w} > 10^{-6}) \wedge \left(E_\alpha^S\left(c^\top \widetilde{x}\right) - \widetilde{w} \leqslant \delta\right)$;

return $\widetilde{x}$ is a feasible solution of (E_α^S).

3. General framework

Now that we can formulate the sample average approximation of our problems, we want to guarantee how close the solution to this approximate problem is to the real solution. In this goal, Kleywegt et al. (2002) developed a framework successively solving SAA problems for discrete stochastic optimization problems giving computational bounds on the real (not sampled) optimization problem (2). In what follows we assume that the scenarios are uniformly distributed, i.e.: $p_s = \frac{1}{S}$ for any scenario $s \in S$.

3.1. Stochastic lower bound

The optimal sampled value w_S is a random variable whose realizations depend on the samples that were selected. Its relation to the optimal value of (2), w^*, is given by (25) due to the convexity of the expected value:

$$\mathbb{E}\left[w_S\right] \leqslant w^* \tag{25}$$

We can estimate this expected value averaging the optimal values of problem P_S generating several samples of size S and solving the associated problem. Let T be the number of times we generate a different set of samples and solve P_S. Then, given a confidence level $\gamma \in [0,1]$ and the objective values $(w_S^t)_{t\in\{1,\dots,T\}}$ of the T different problems, by the central limit theorem we have:

$$\mathbb{P}\left[L_{N,T} := \frac{1}{T}\sum_{t=1}^{T} w_S^t - \phi^{-1}\left(1-\frac{\gamma}{2}\right)\frac{1}{\sqrt{T}}\sigma_S \leqslant w^*\right] \geqslant 1-\frac{\gamma}{2} \tag{26}$$

With:

$$\sigma_S^2 = \frac{1}{T-1}\sum_{t=1}^{T}\left(w_S^t - \frac{1}{T}\sum_{t'=1}^{T} w_S^{t'}\right)^2 \tag{27}$$

Namely, the resolution of several sampled problems gives a lower bound of the real optimal value with confidence at least $1-\frac{\gamma}{2}$. When using suboptimal resolutions with known relative gaps $(g_t)_{t\in\{1,\dots,T\}}$ and objective values $(w_S^t)_{t\in\{1,\dots,T\}}$, notice that we cannot directly multiply each sampled objective value w_S^t by $\frac{1}{1+g_t}$ because of the correcting term containing the standard deviation. One way to tackle this issue is to adjust the sampled objective values by a factor of $\frac{1}{1+\max\{g_t : t\in\{1,\dots,T\}\}}$.

3.2. Stochastic upper bound

When solving the T sampled problems we can keep some feasible solution $x \in X$ and we know that $w^* \leqslant \mathbb{E}\left[u\left(c^\top x\right)\right]$. Now, we can have an estimation of $\mathbb{E}\left[u\left(c^\top x\right)\right]$ generating another sample of size $S' >> S$. It is common practice to refer to the sample S used for each of the sampled problems solved for the lower bound as the "in-sample" and this larger sample S' as the "out-of-sample". Again, by the central limit theorem we have that:

$$\mathbb{P}\left[w^* \leqslant U_{S'}(x) := w_{S'}(x) + \phi^{-1}\left(1-\frac{\gamma}{2}\right)\frac{1}{\sqrt{S'}}\sigma_{S'}(x)\right] \geqslant 1-\frac{\gamma}{2} \tag{28}$$

With:

$$\sigma^2_{S'}(x) := \frac{1}{S'-1}\sum_{s=1}^{S'}\left[u\left(\left(c^s\right)^\top x\right) - w_{S'}(x)\right]^2 \tag{29}$$

In other words, it is possible to obtain an upper bound of the real optimal value by computing an accurate estimation of the objective value of some feasible solution. Now that we have lower and upper bounds we can compute the stochastic gap, i.e.: the real gap of our solution $x \in X$ with the real optimal value with confidence at least $1-\gamma$:

$$\mathbb{P}\left[L_{N,T} \leqslant w^* \leqslant U_{S'}(x)\right] \geqslant 1-\gamma \tag{30}$$

A common practice is to store the solution corresponding to the lowest objective value w^t_S returned by one of the T deterministic equivalent problems and use it to build the stochastic upper bound.

4. Computational Experiments

4.1. Experimental Set-up

Network generation. We generated grid networks of $r \times r$ nodes with $r \in \{5, 7, 10, 13, 15\}$. Each network has exactly the same geographical dimensions (a square with sides of 1.5 kilometer on each side), the only difference residing in the topology of the network. In addition to the grid roads, each instance includes a 'highway' that can be of three types as depicted in figure 5: straight crosses (dashed), tilted crosses (dotted) or ring-shaped (solid).

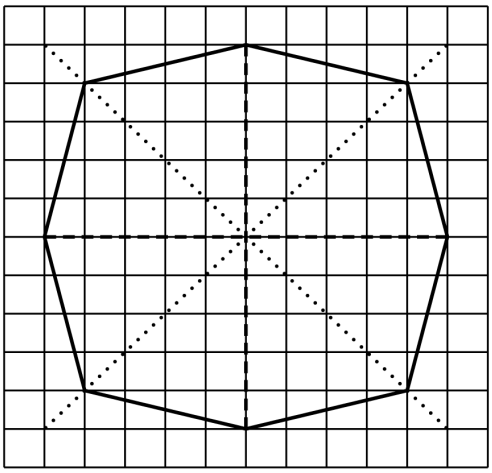


Figure 5 **Example of Grid networks generated for** $r = 13$

Distributions generation. For each edge $i \in E$ we considered a constant mean speed μ_i of 50 km (80 km) for grid edges (highway edges). To generate more instances we perturbed these reference values with an uniform noise $u_i \sim \mathcal{U}[0.5, 1.5]$ such that the resulting mean speed on edge i is $\mu'_i := \mu_i u_i$. For each instance, we generated ten times a perturbation with a different random seed. We also generated randomly special covariance matrices $\Sigma = (\sigma_{ij})_{(i,j)\in\{1,...,m\}^2}$, where $\sigma_{ii} = \sigma_i^2$ is the variance of the random variable associated with edge $i \in \{1, ..., m\}$. Given a vector of expected values $\mu' = (\mu'_i)_{i\in\{1,...,m\}}$ we considered instances with different coefficients of variation $c \in \{0.5, 2\}$ for normal streets and $c \in \{1, 4\}$ for highways (riskier) such that the standard deviations are defined as $\sigma_i = c\mu'_i$. We considered the uncorrelated and correlated cases. We built the correlations in a way that all the highway edges (street edges) are positively correlated with each other, but there is negative correlation between highway and street edges. This will highlight the risk-hedging effect obtained by forcefully passing through both types of edges in spite that one type is riskier than the other. We generated correlation matrices C defined by a product $C = LL^\top$ with $L = (l_{ij})_{(i,j)\in\{1,...,m\}^2}$ a lower triangular matrix whose rows are unitary vectors. Given a correlation matrix C and standard deviations $(\sigma_i)_{i\in\{1,...,m\}}$ we can then compute a covariance matrix Σ as follows:

$$\Sigma = \text{Diag}\left((\sigma_i)_{i\in\{1,...,m\}}\right) \cdot C \cdot \text{Diag}\left((\sigma_i)_{i\in\{1,...,m\}}\right) \tag{31}$$

Which has all the required properties to be a correlation matrix. We considered that travel times on graph edges are random following Log-normal distributions as indicated by Westgate et al. (2013) in the case of transportation networks, of joint-parameters (p, S)

with $p = (p_i)_{i\in\{1,\dots,m\}}$ and $S = (s_{ij})_{(i,j)\in\{1,\dots,m\}^2}$. Given some expected value vector μ and a covariance matrix Σ we wanted to generate samples from the associated log-normal distribution. We know that such a distribution is a log-normal of parameters (p, S) with:

$$\begin{cases} p_i &= \ln\left(\frac{\mu_i^2}{\sqrt{\mu_i^2+\sigma_{ii}}}\right) \\ s_{ij} &= \ln\left(1+\frac{\sigma_{ij}}{\mu_i\mu_j}\right) \end{cases} \tag{32}$$

Moreover, given some decomposition $S = L^\top L$ we have $LX + p \sim \mathcal{N}(p, S)$ with $X_i \sim \mathcal{N}(0, 1)$. Putting everything together, we can sample multivariate log-normal distributions of mean vector μ and covariance matrix Σ by generating univariate standard normal samples $x = (x_i)_i$ and applying the transformation $z = \exp(Lx + p)$. We generated the standard normal distribution samples via Box-Muller transform: given u and u' two independent draws of an uniform law $\mathcal{U}[0, 1]$ then:

$$x = \sqrt{-2\ln u}\cos(2\pi u')$$
$$x' = \sqrt{-2\ln u}\sin(2\pi u')$$

Are distributed as independent $\mathcal{N}(0, 1)$ random variables.

Parameters used. We chose the sample sizes used for each deterministic equivalent problem (in-sample) to be in $S \in \{0.5, 1, 2, 5, 10\} \cdot 10^3$ and the sample used for upper bounding (out of sample) in $S' \in \left\{(2k)_{k\in\{1,\dots,10\}}\right\} \cdot 10^4$. The base problem we want to solve is to find the less risky path between the north-western and the south-eastern corners of each grid network. We tested CVaR_ϵ for $\epsilon \in \{1, 5, 10, 50, 90, 95, 99\}\%$. For the entropic risk measure, the parameter α has units (like travel time) so we had to carefully choose its values. We chose a reference path inside the grid network of 15×15 nodes with circular highway and computed the sampled cumulative distribution of the travel time on that path, $F(\cdot)$, with correlated travel times across edges and a high dispersion factor ($c = 2$). We then chose $\alpha \in F^{-1}(\{0.01, 0.05, 0.1, 0.5, 0.9, 0.95, 0.99\})$, whose rounded computed values are: $\alpha \in \{10, 15, 20, 50, 200, 250, 400\}\,[s]$. The parameter α captures an absolute risk aversion and penalizes greatly paths whose random travel time exceeds α. This is why we kept the same geographical dimensions for the grid networks to keep the same set of parameters for testing all the instances.

Experiments. To evaluate the influence of each parameter on the efficiency of the proposed algorithms, we defined a base case and conducted experiments that let varied one parameter at a time. First, we wanted to compare the computational efficiency of each solution algorithm on deterministic equivalent problems. We took as a base case the grid network of 10×10 nodes with circular highway, presence of correlation, high dispersion and a sample size of $S = 2000$. If an algorithm does not reach the optimal solution within 250 minutes, we return the incumbent solution with its respective optimality gap. We then chose the algorithm with best computational behavior for each risk measure and used it to solve $T = 50$ experiments with samples of size $S = 2000$ with a time limit of 250 minutes for the whole run and 50 minutes for each deterministic equivalent problem solution. We computed the stochastic bounds with 95% confidence (i.e. with $\gamma = 0.05$).

Risk measure	Algorithm	Framework	Abbreviation
CVaR	Monolithical resolution	-	CM
	Subgradient Algorithm	LP+IP	CS/LP+IP
		IP	CS/IP
	Aggregation method	LP+IP	CA/LP+IP
		IP	CA/IP
Entropy	Subgradient algorithm	LP+IP	ES/LP+IP
		IP	ES/IP
	Exponential approximation method	-	EEA

Table 1 Frameworks abbreviations

4.2. Computational results

Deterministic Equivalent problems. In this section, we will present computational results comparing the efficiency of the different methods we presented earlier. In this goal, we compared the runtimes and the optimality gaps of the solutions returned by each of the algorithms over a single deterministic equivalent problem. This determined which method is most fit to compute stochastic bounds later. In table 2 we show averaged results over ten random instances for each cell. We see that the monolithic version of CVaR has worse solution times than the other algorithms, except when there are no correlations. We see that CA/IP outperforms greatly all the other algorithms in terms of execution time. In the general correlated case, we can rank - on average - the CVaR algorithms proposed here as follows: [CA/IP] $\succeq$ [CA/LP+IP] $\succeq$ [CS/IP] $\succeq$ [CS/LP+IP] $\succeq$ [CM]. For the entropic risk measure, we can see that ES/IP is always much faster than the other algorithms. EEA behaves slowly in practice due to the number of iterations needed to find an adequate M^*

Param	Value	CM	CA/LP+IP	CA/IP	CS/LP+IP	CS/IP	EEA	ES/LP+IP	ES/IP
Σ	I	**59.1**	195.6	154.7	1392(8)	1313(10.9)	1658.7(1.8)	1318.7(5.6)	**1312.5(4.4)**
	$\neq I$	15.3	2.2	**1.9**	4.2	4.1	1243.7(3.6)	4.35	**3.1**
c	0.5	10.9	1.8	**1.5**	2	2	849.1(2.3)	2.1	**1.7**
	2	15.3	2.2	**1.9**	4.2	4.1	1135.3(3.9)	4.4	**3.1**
Shape	$+$	13.6	2.2	**1.9**	6	5	1173.8(2.9)	7	**4**
	$\times$	15.2	2.2	**2**	6.9	6.6	1287.5(3.3)	9.3	**5.7**
	O	15.3	2.2	**1.9**	4.2	4.1	1135.3(3.9)	4.4	**3.1**
r	5	3.6	0.3	**0.1**	0.5	0.3	104.9(6.4)	0.3	**0.2**
	7	7.5	0.7	**0.4**	1.1	0.9	329.6(3.8)	1	**0.6**
	10	15.3	2.2	**1.9**	4.2	4.1	1135.3(3.9)	4.4	**3.1**
	13	24.8	4.5	**4.4**	6.9	6.3	2043.1(5.2)	9.9	**8.7**
	15	33.9	**7.7**	**7.7**	10.4	9.9	1890.8(6.3)	17	**13.2**
S	500	2.1	**0.6**	0.7	2.4	2.8	156.9(1.9)	4.3	**2.7**
	1000	6	1.3	**1.2**	3.3	3.7	379(3.6)	2.5	**1.9**
	2000	15.3	2.2	**1.9**	4.2	4.1	1135.32(3.9)	4.4	**3.1**
	5000	58.3	5	**4.2**	6.6	6.3	cutoff(2)	12.8	**8.3**
	10000	177.5	9.5	**7.7**	10.7	9.6	cutoff(7.3)	19.6	**10.5**

Table 2 Average solution time for different instance parameters (time$[s]$(gap$[\%]$))

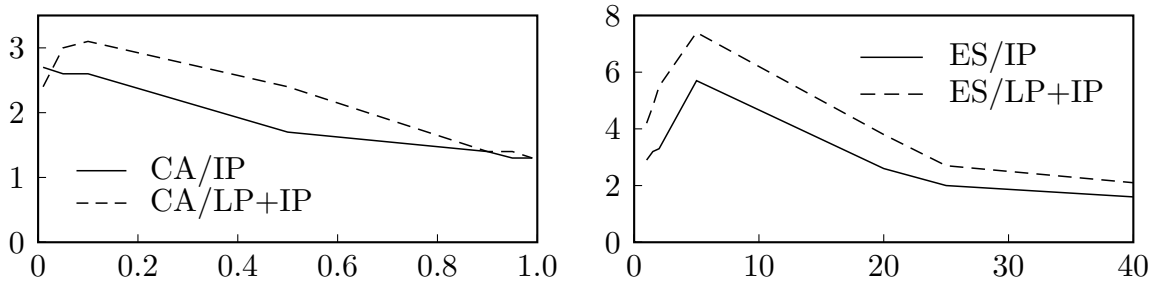
and then approximate correctly the exponentials. Furthermore, the optimality gaps are often strictly positive because there is no guarantee that the stopping criterion ensures that EEA returns an optimal solution. The cutoff symbol in this table means that for all the instances tested, the time limit was reached before the stopping criterion is met. On average, we can rank the Entropy frameworks as follows: [ES/IP] $\succeq$ [ES/LP+IP] $\succeq$ [EEA]. In the end, first solving the LP relaxation and then the original problem with the constraints generated is always slower. Although the number of iterations is smaller for LP+IP frameworks, the fact they begin the integer resolution with a larger problem instead of starting with a small one slows down significantly the algorithm. The correlations have a great influence over the execution time. Indeed, we designed Σ such that alternating between both types of streets helps to hedge against variability and thus reduces the optimal region, speeding up the procedure. The correlation also groups several edges together in their behaviors, having the solver aggregating variables and making branching more efficiently. As expected, having more dispersion makes the problem harder as we are getting further from an easy expected value minimization problem. The instance parameters that are the most influential to the solution times are the network size and the number of samples we used for the SAA. The network size being directly linked to the number of integer variables, the execution time grows significantly fast with r. In a similar way, each sample used adds an extra constraint and an extra continuous variable to CVaR problems. The execution time for CA/IP grows linearly with S. For the entropy, although the number of samples does not enlarge the problem solved by ES, it does substantially increase the

solution time. In table 3, we can see once more than CA/IP (ES/IP) is the best scheme for

Param	Value	CM	CA/LP+IP	CA/IP	CS/LP+IP	CS/IP	Param	Value	EEA	ES/LP+IP	ES/IP
	0.01	16.6	**2.5**	2.6	9.7	11.4		1000	1733.5	4.2	**2.9**
	0.05	19.9	3.0	**2.6**	7.0	6.5		1500	1511.5	4.8	**3.2**
	0.1	20.4	3.1	**2.6**	5.8	4.4		2000	1435	5.5	**3.3**
ϵ	0.5	19.3	2.4	**1.7**	2.3	1.9	α	5000	1117.2(1)	7.4	**5.7**
	0.9	11.0	1.4	**1.4**	1.5	1.5		20000	802.1(12.2)	3.8	**2.6**
	0.95	10.2	1.4	**1.3**	1.4	1.4		25000	728.4(8.3)	2.7	**2.0**
	0.99	9.9	1.3	**1.3**	1.4	1.4		40000	619.6(5.4)	2.1	**1.6**

Table 3 Algorithms comparison Vs. risk measure parameters (time$[s]$(gap$[\%]$))

CVaR (Entropy) resolution as we vary the risk measure parameters. In the following we used these two algorithms to solve the repeated procedure with stochastic bounds computation. When loosening the risk aversion, we observe that the difficulty lowers significantly for the less conservative parameters (see figure 6). Indeed, when the parameter changes to reduce risk aversion the problem to solve tends to a determininistic problem which - in our particular case - has a particularly nice structure. Nevertheless, we see that for both risk measures(see figures 6a and 6b) the most difficult problems to solve are not necessarily the most risk averse. We remark that entropy minimization problems are slightly slower to solve than CVaR.



(a) CA time $[s]$ Vs. $\epsilon \in]0,1]$ (b) ES time $[s]$ Vs. $\alpha > 0$

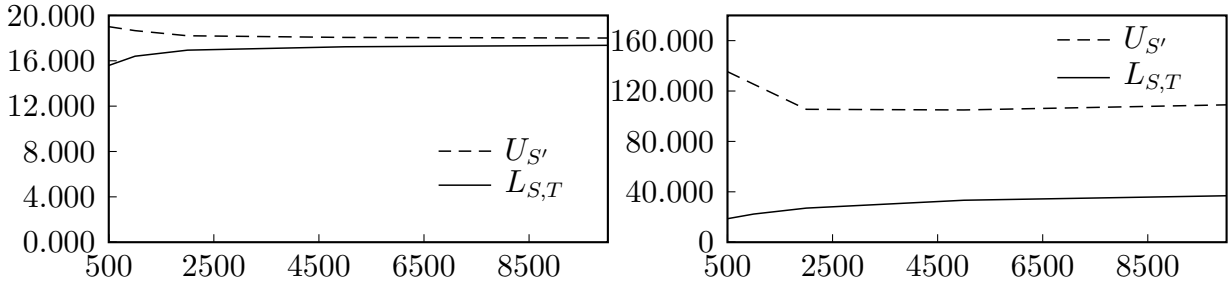
Figure 6 Execution time $[s]$ Vs. parameter and LP+IP or IP frameworks.

Stochastic bounds. In this section, we will present computational results comparing the quality of the solutions returned by the entire SAA framework. We ran the T experiments for both measures and confirmed that solving the problems without considering correlations can lead to solutions of poor quality. In table 4 we can see that for a random instance of the base case, an uncorrelated solution can have a difference of optimality gap between [17;43]%([5;53]%) when minimizing CVaR(Entropy). In figure 7, we can see that the sample

Param	Value	gap[%] co	gap[%] unco	Param	Value	gap[%] co	gap[%] unco
	0.01	11.87	46.24		1000	68.26	81.04
	0.05	4.91	38.78		1500	69.23	87
	0.1	4.12	34.53		2000	71.05	86.31
ϵ	0.5	2.71	29	α	20000	90.58	96
	0.9	1.8	19.28		25000	50.31	93.38
	0.95	1.77	19.1		5000	56.92	94.56
	0.99	1.75	18.93		40000	13.74	67.66

Table 4 Price of correlation for one instance of the base case.

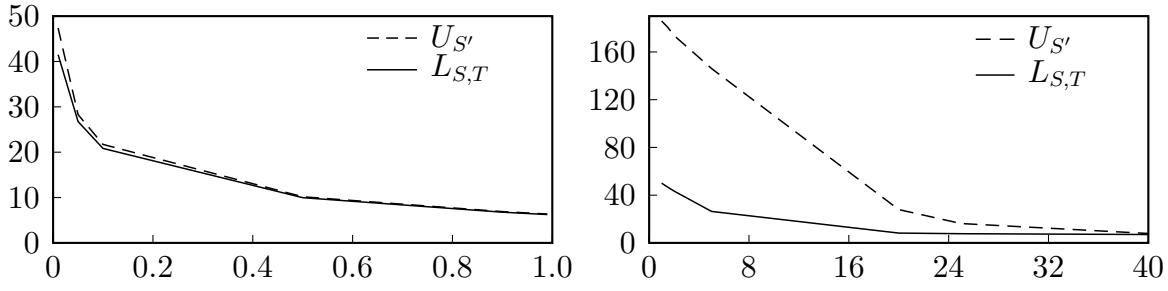
size S is important for two reasons: it approximates better the original problem, improving the lower bound and returns better candidate solutions, improving the upper bound. In



(a) Bounds for CVaR Vs. S (b) Bounds for Entropy Vs. S

Figure 7 Stochastic optimality bounds Vs. number of samples S.

figure 8, we observe that the risk aversion parameter plays a critical role in the difficulty of our problem. In particular for the entropy, the gap is high for almost all the values of α we considered, closing slowly when the risk aversion lowers. For CVaR, the optimality gap becomes acceptable ($< 5\%$) starting from $\epsilon \geqslant 0.1$.



(a) Bounds For CVaR Vs. $\epsilon \in]0, 1]$ (b) Bounds for Entropy Vs. α

Figure 8 Stochastic optimality bounds Vs. Risk Aversion parameter

Increasing the sample size S' improves (reduces) the quality of the upper bound of CVaR and is computationally cheap. Nevertheless, for the Entropy the size S' of the out

of sample needed to have accurate upper bounds can be potentially large. For conservative parameters α, we observe that the upper bound is still unstable for the values of S' we considered. In figure 9, the curve represents the upper bound in function of S' and the dots the contribution of each scenario.

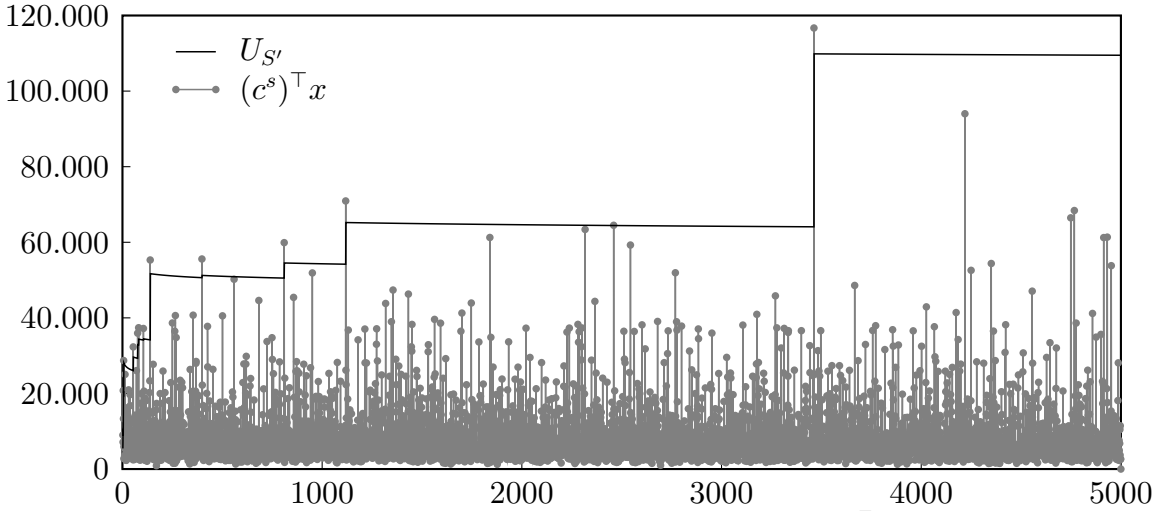


Figure 9 **Example of upper bound for entropy and scenario realizations $(c^s)^\top x$ Vs. S'**

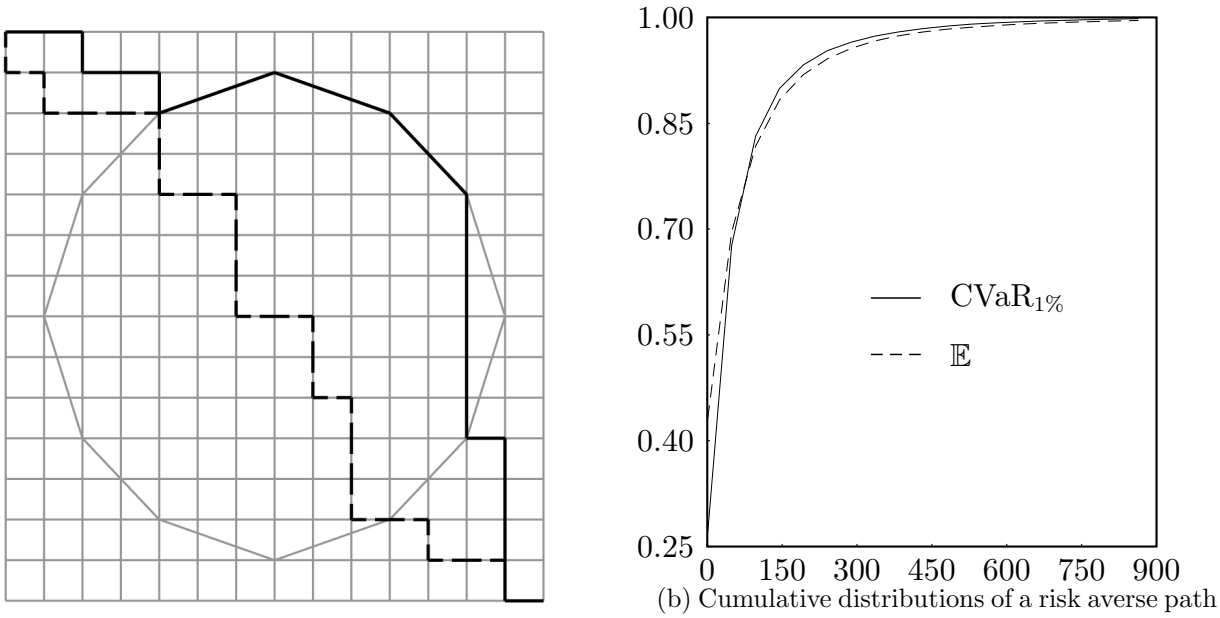
We can see jumps in the upper bound curve between - expected - lowering periods. These jumps appear because the entropic risk measure can be highly sensitive to worst cases. We can show that we have:

$$\max_{s\in\{1,\dots,S'\}} (c^s)^\top x - \alpha \ln S' \leqslant \alpha \ln \sum_{s=1}^{S'} \frac{1}{S'} e^{\frac{(c^s)^\top x}{\alpha}} \leqslant \max_{s\in\{1,\dots,S'\}} (c^s)^\top x$$

These bounds show that the entropic risk measure highly sensitive to big realizations $(c^s)^\top x$. Instead of having a smoothing effect of big values of S' we can observe the upper bound $U_{S'}$ worsening. We ran a couple of small instances with very large out of samples and could observe that for $S' \approx 10^6$ an expected nonincreasing smooth curve appears.

In figure 10a we see an example where the risk averse response (solid) of $\text{CVaR}_{1\%}$ passes trough "highway" edges to hedge against risk, using the negative correlations to its advantage, while the risk neutral (dashed) - minimizing the expected value - passes right where the mean value is minimal, without taking the volatility of travel times into account.

As such, in figure 10b we can see their respective cumulative distributions obtained with a large out of sample of size $S' = 100000$. Without surprise, we can notice that the risk averse path (solid) stochastically dominates the risk neutral one (dashed) in the worst



(a) Risk averse path Vs. Risk neutral path

(b) Cumulative distributions of a risk averse path Vs. risk neutral path.

Figure 10 Comparison of risk averse optimal path for $\text{CVaR}_{1\%}$ and a risk neutral optimal path

20% of the cases (for both solutions) lowering in a 2% the probability to have a travel time greater than two minutes. Moreover, the expected travel time of a risk averse traveler is about 1 minute 41 seconds against 1 minute 38 for the risk neutral traveler, but the variances are respectively 1 minute 46 seconds for the risk averse traveler and 2 minutes 17 seconds for the risk neutral one.

5. Conclusions

In this work, we developed a complete framework to solve shortest path problems with uncertainty in the travel times. We developed efficient techniques to solve SAA equivalent problems for the CVaR and the Entropic risk measures. We extended a result from Espinoza and Moreno (2014) of linearly constrained CVaR minimization, to any type of feasible set, be it convex or not. We showed that our algorithms for CVaR minimization return provably decent solutions within a reasonable CPU time. We empirically saw that Entropic risk measures could generate prohibitive numerical issues when considering travel time correlations. Conservative parameters can indeed cause either solving difficulties or the impossibility to obtain good stochastic bounds. A natural way to tackle the difficulty inherent to big samples and extend these results to real sized networks could come with the use of importance sampling methods, where less samples but more representative are used. In the particular case of CVaR frameworks, some recent works used a smoothing

technique allowing to use state of the art algorithms to solve CVaR objective continuous minimization problems. Indeed, the nondifferentiability of CVaR could be the cause of the slow convergence of the subgradient framework in practice.

Acknowledgments

Funded by Conicyt through grant FONDEF No. D10I1002

References

Agrawal, S., Y. Ding, A. Saberi, Y. Ye. 2010. Correlation robust stochastic optimization. *Proceedings of the 21st annual ACM-SIAM Symposium on discrete algorithms, Austin, TX*. 1087–1096.

Agrawal, S., Y. Ding, A. Saberi, Y. Ye. 2012. Price of correlations in stochastic optimization. *Operations Research* **60** 150–162.

Ahuja, R.K., T.L. Magnanti, J.B. Orlin. 1993. *Network flows: theory, algorithms, and applications*. Prentice hall.

Arrow, K. J. 1965. *Aspects of the theory of risk-bearing*. Yrjö Jahnssonin Säätiö.

Artzner, P., F. Delbaen, J.-M. Eber, D. Heath. 1999. Coherent measures of risk. *Mathematical finance* **9** 203–228.

Atamtürk, A., M. Zhang. 2007. Two-stage robust network flow and design under demand uncertainty. *Operations Research* **55** 662–673.

Avriel, M., A.C. Williams. 1970. The value of information and stochastic programming. *Operations Research* **18** 947–954.

Ben-Tal, A., L. El Ghaoui, A. Nemirovski. 2009. *Robust optimization*. Princeton University Press.

Bertsimas, D., M. Sim. 2004. The price of robustness. *Operations research* **52** 35–53.

Birge, J. R., F. Louveaux. 2011. *Introduction to stochastic programming*. Springer Science & Business Media.

Drexl, A., A. Kimms. 1997. Lot sizing and scheduling, survey and extensions. *European Journal of Operational Research* **99** 221–235.

Drezner, E. 1996. Facility location: A survey of applications and methods. *Journal of the Operational Research Society* **47** 1421–1421.

Espinoza, D., E. Moreno. 2014. A primal-dual aggregation algorithm for minimizing conditional value-at-risk in linear programs. *Computational Optimization and Applications* **59** 617–638.

Ghiani, G., F. Guerriero, G. Laporte, R. Musmanno. 2003. Real-time vehicle routing: Solution concepts, algorithms and parallel computing strategies. *European Journal of Operational Research* **151** 1–11.

Jans, R., Z. Degraeve. 2008. Modeling industrial lot sizing problems: a review. *International Journal of Production Research* **46** 1619–1643.

Kleywegt, A.J., A. Shapiro, T. Homem-de Mello. 2002. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* **12** 479–502.

Künzi-Bay, A., J. Mayer. 2006. Computational aspects of minimizing conditional value-at-risk. *Computational Management Science* **3** 3–27.

Kusuoka, S. 2001. On law invariant coherent risk measures. *Advances in mathematical economics* **3** 83–95.

Laporte, G. 1992. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research* **59** 345–358.

Levy, H. 2006. *Stochastic dominance: Investment decision making under uncertainty*, vol. 12. Springer Science & Business Media.

Lim, A. E.-B., J.G. Shanthikumar, G.-Y. Vahn. 2011. Conditional value-at-risk in portfolio optimization: Coherent but fragile. *Operations Research Letters* **39** 163–171.

Markowitz, H. 1952. Portfolio selection. *The journal of finance* **7** 77–91.

Nikolova, E., M. Brand, D.R. Karger. 2006. Optimal route planning under uncertainty. *Proceedings of International Conference on Automated Planning and Scheduling, Ambleside, U.K.*, vol. 6. 131–141.

Pratt, J. W. 1964. Risk aversion in the small and in the large. *Econometrica: Journal of the Econometric Society* **32** 122–136.

Rockafellar, R.T., S. Uryasev. 2000. Optimization of conditional value-at-risk. *Journal of risk* **2** 21–42.

Shapiro, A., D. Dentcheva, A. Ruszczyński. 2014. *Lectures on stochastic programming: modeling and theory*, vol. 16. SIAM.

Snyder, L.V. 2006. Facility location under uncertainty: a review. *IIE Transactions* **38** 547–564.

Tintner, G. 1955. Stochastic linear programming with applications to agricultural economics. *Proceedings of the Second Symposium in Linear Programming, Washington, DC*, vol. 1. National Bureau of Standards Washington, D. C, 197–228.

Von Neumann, J., O. Morgenstern. 1944. Theory of games and economic behavior. *Princeton University Press, Princeton* .

Westgate, B. S., D. B. Woodward, D.S. Matteson, S. G. Henderson. 2013. Large-network travel time distribution estimation, with application to ambulance fleet management. *Under review* .

Wets, R. J.-B. 1966. Programming under uncertainty: the equivalent convex program. *SIAM Journal on Applied Mathematics* **14** 89–105.

Yaari, M. E. 1987. The dual theory of choice under risk. *Econometrica* **55** 95–115.

Yaari, M.E. 1969. Some remarks on measures of risk aversion and on their uses. *Journal of Economic theory* **1** 315–329.

Yang, B., C. Guo, C.S. Jensen. 2013. Travel cost inference from sparse, spatio temporally correlated time series using markov models. *Proceedings of the VLDB Endowment, Riva del Garda, Italy*, vol. 6. VLDB Endowment, 769–780.