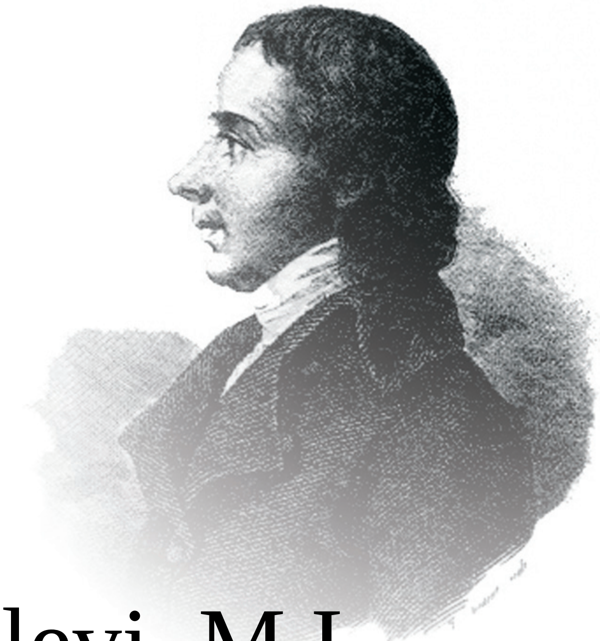
Rapporto n. 6 2008

dmsia unibg.it



F. Maggioni, E. Allevi, M.I. Bertocchi, and F. A. Potra

Mobile ad-hoc networks: a new stochastic second-order cone programming approach

Serie Ricerca

Dipartimento
di Matematica, Statistica, Informatica e Applicazioni "Lorenzo Mascheroni"

UNIVERSITÀ DEGLI STUDI DI BERGAMO

Mobile ad-hoc networks: a new stochastic second-order cone programming approach

F. Maggioni[1], E. Allevi[2], M.I. Bertocchi[1], and F. A. Potra[3]

Abstract. We study the semidefinite stochastic location-aided routing (SLAR) model described in Ariyawansa and Zhu (2006) [2] and in Zhu, Zhang, and Patel (2007) [16]. We propose a modification of their model to exploit the stochasticity inherent in the destination node movements. We formulate the problem as a two-stage stochastic second-order cone programming (SSOCP), see Alizadeh and Goldfarb (2003) [1], where the first-stage decision variables include both the position of the destination node and its distance from the sender node. Destination node movements are represented by ellipsoid scenarios defined in a neighborhood of the starting position and generated by uniform and normal disturbances. The MOSEK solver (under GAMS environment) allows to solve problems with a large number of scenarios (say 20250) versus the DSDP (under MATLAB framework) solver, see Benson, Ye and Zhang (2000) [4], adapted to stochastic programming framework with 500 scenarios. Stability results for the optimal first-stage solutions and for the optimal function value are obtained.

1 Introduction

Wireless mobile hosts, characterized by communicating each other in absence of a fixed infrastructure, have become an important tool of our daily life. The Mobile Ad hoc NETworks (MANET), based on wireless mobile nodes, and the related routing protocols have been studied extensively in the last 15 years, see Ko and Vaidya (2000) [10] and the references therein, and Vyas (2000) [15]. Routing protocols usually differ on the assumptions governing to search for a new route. To decrease the overhead of route discovery, Ko and Vaidya (2000) [10], suggest a special approach based on the use of local information. Their algorithm, known as Location-aided Routing (LAR) protocol, tries to reduce the number of nodes to whom the requested route is propagated by use of local information given, for example, by the Global Positioning System (GPS). The main concepts behind the algorithm are the expected zone and the requested zone. The first is the region that the sender node S expects to contain the destination node D in the elapsed time t_1 from the original position of the two nodes, and the second is the region defined by the sender node to include the destination node. The route being requested spreads over only if a neibourgh node belongs to the requested zone. A protocol that makes use of the requested zone should consider to start with a large requested

[1]Department of Mathematics, Statistic, Computer Science and Applications, Bergamo University, Via dei Caniana 2, Bergamo 24127, Italy.

[2]Department of Quantitative Methods, Brescia University, Contrada S. Chiara, 50, Brescia 25122, Italy.

[3]Department of Mathematics & Statistic, University of Maryland, Baltimore County, U.S.A.

zone. On the other hand, the requested zone is expected to include the expected zone and this suggests that one may first try to identify the expected zone. This is the approach proposed by Ariyawansa and Zhu (2006) [2] who formulated the expected zone identification problem of the as a stochastic two-stage model in which the first decision variables include the radius of the expected region which can be further influenced by the realization of random movements (different scenarios represented by random ellipsoids). Their model is a particular two-stage stochastic model because not all second-stage variables are scenario dependent (the enlarged radius represents the worst case containing all the possible random movements).

In Zhu, Zhang and Patel (2007) [16], this approach has been formulated as a stochastic semidefinite programming model and it has been solved for a limited number of scenarios. In the same paper Zhu, Zhang and Patel estimate the solution sensitivity to alternative scenarios probability structure and cost coefficients in the decision problem. The performance measure used by Zhu, Zhang and Patel is the percentage savings defined as $(r_2^2 - r_1^2)/r_2^2$ where r_1 and r_2 are respectively the radius of the first expected zone and the radius of the enlarged expected zone. In such a model, the radius of the enlarged expected zone includes all the possible realizations of the random movements.

We extend this model on two important aspects:

- we look for various second-stage circles, each of them covering a realization of the random movements (an ellipsoid); this allows a better representation of reality using the second-stage variables as a recourse action on the first-stage ones;
- we successfully transform the stochastic semidefinite programming model in a second-order cone programming model allowing for a substantial reduction in time execution.

Section 2 of our paper is devoted to introduction of notations for stochastic semidefinite programming and second-order cone programming. Section 3 describes the formulation of SLAR protocol for mobile ad hoc network problems as a stochastic two-stage semidefinite programming problem. Section 4 contains the formulation of SLAR protocol as stochastic two-stage second order cone programming problem. In Section 5 we describe the strategies used for scenarios generation and finally, Section 6 contains the results of experiments and their validation.

2 Basic facts and notation

Semidefinite programming problems define a class of optimization problems that have been studied extensively during the past 15 years. Semidefinite programming is naturally related to linear programming, and both are based on deterministic coefficient matrices. Semidefinite programming is primarily concerned with the selection with a symmetric matrix to minimize a linear function subject to linear constraints. The matrix is constrained throughout to be positive semidefinite. Deterministic semidefinite programming (DSDP) generalizes deterministic linear programming (DLP). DLP has nonnegative decision variables while the decision variable in DSDP is a positive semidefinite matrix.

We use the following notation: $\Re^{n\times n}$ for the vector space of real $n \times n$ matrices, lower case boldface letters $\mathbf{x}, \mathbf{c}$ etc. for column vectors, and uppercase letters A, X etc. for matrices. Subscripted vectors such as $\mathbf{x_i}$ represent the i^{th} block of $\mathbf{x}$. The j^{th} component of the vectors $\mathbf{x}$ and $\mathbf{x_i}$ are indicated by x_j and x_{ij} respectively. We use $\mathbf{0}$ and $\mathbf{1}$ for the zero vector and vectors of all ones, respectively, and 0 and I for the zero and identity matrices.

A deterministic linear programming problem (DLP) in primal standard form is

$$\begin{aligned} &\min_{\mathbf{x}\in\Re^n} \mathbf{c}^T\mathbf{x} \\ &\text{subject to } A\mathbf{x} = \mathbf{b}, \\ &\qquad\qquad\quad \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{1}$$

and its dual

$$\begin{aligned} &\max_{\mathbf{y}\in\Re^m} \mathbf{b^T}\mathbf{y} \\ &\text{subject to } \; A^T\mathbf{y} \leq \mathbf{c}, \end{aligned} \tag{2}$$

where $A \in \Re^{m\times n}$, $\mathbf{b} \in \Re^{\mathbf{m}}$ and $\mathbf{c} \in \Re^{\mathbf{n}}$ constitute given data, and $\mathbf{x} \in \Re^{\mathbf{n}}$ is the primal variable and $\mathbf{y} \in \Re^{\mathbf{m}}$ is the dual variable.
Let $\Re^{n\times n}_s$ denotes the vector space of real $n \times n$ symmetric matrices, for $A, B \in \Re^{n\times n}_s$ we write $A \succeq 0$ ($A \succ 0$) to mean that A is positive semidefinite (positive definite) and $A \succeq B$ ($A \succ B$) to mean that $A - B \succeq 0$ ($A - B \succ 0$). For $A, B \in \Re^{n\times n}$ we denote by $A \bullet B$ the Frobenius inner product between A and B: $A \bullet B = trace(A^T B)$.
A DSDP in primal standard form is

$$\begin{aligned} &\min_{X\in\Re^{n\times n}_s} \; C \bullet X \\ &\text{subject to } \; A_i \bullet X = b_i, \qquad i = 1, 2, \ldots, m \\ &\qquad\qquad\quad X \succeq 0 \end{aligned} \tag{3}$$

where $A_i \in \Re^{n\times n}_s$ for $i = 1, 2, \ldots, m$, $\mathbf{b} \in \Re^m$ and $C \in \Re^{n\times n}_s$ are given and $X \in \Re^{n\times n}_s$ is the variable.

A DSDP in dual standard form is

$$\begin{aligned} &\max_{\mathbf{y}\in\Re^m} \mathbf{b}^T\mathbf{y} \\ &\text{subject to } \; \sum_{i=1}^{m} \mathbf{y}_i A_i \preceq C \end{aligned} \tag{4}$$

where $A_i \in \Re_s^{n\times n}$ for $i = 1, 2, \ldots, m$, $\mathbf{b} \in \Re^m$ and $C \in \Re_s^{n\times n}$ are given data, and $\mathbf{y} \in \Re^m$ is the dual vector. Notice that it is always possible to convert a problem in form (4) to an equivalent problem in form (4) and vice versa.

Stochastic programming was introduced in the 1950s as a paradigm for dealing with uncertainty in the data related to linear programming. Ariyawansa and Zhu (2006), [2] introduced stochastic semidefinite programs as a paradigm for dealing with uncertainty in data related to semidefinite programs.

We recall the structure of a two stage stochastic linear programming problem with recourse (SLPs): a SLPs in primal standard form is

$$\begin{aligned} &\min_{\mathbf{x}\in R^{n_1}} \ \mathbf{c}^T\mathbf{x} + E[Q(\mathbf{x},\omega)] \\ &\text{subject to} \ \ A\mathbf{x} = \mathbf{b} \\ &\qquad\qquad\quad \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{5}$$

where $\mathbf{x} \in R^{n_1}$is the first-stage decision variable, $\mathbf{c} \in R^{n_1}$ is a given vector, frequently called cost vector, $\mathbf{b} \in R^{m_1}$ an other given vector, $A \in \Re^{m_1\times n_1}$, $\mathbf{c}$, $\mathbf{b}$ and A are deterministic data. $Q(\mathbf{x}, \omega)$ is the minimum of the problem

$$\begin{aligned} &\min_{\mathbf{y}(\omega)\in R^{n_2}} \ \mathbf{q}(\omega)^T\mathbf{y} \\ &\text{subject to} \ \ T(\omega)\mathbf{x} + W(\omega)\mathbf{y} = \mathbf{h}(\omega) \\ &\qquad\qquad\quad \mathbf{y} \geq \mathbf{0} \end{aligned} \tag{6}$$

and

$$E[Q(\mathbf{x},\omega)] = \textstyle\int_\Omega Q(\mathbf{x},\omega)P(\mathrm{d}\omega) \tag{7}$$

where $\mathbf{y}(\omega) \in R^{n_2}$ is the second-stage decision vector, $\mathbf{q} \in R^{n_2}$, $T(\omega) \in \Re^{m_2\times n_1}$ is the technology matrix, $W(\omega) \in \Re^{m_2\times n_2}$ is the recourse matrix, $\mathbf{h} \in R^{m_2}$ and $\omega \in \Omega$ is a random outcome with a known probability distribution P.

The stochastic semidefinite programming problem with recourse (SSDP) in standard primal form, introduced by Ariyawansa and Zhu (2006) [2], is given by

$$\begin{aligned} &\min_{X\in R_s^{n_1\times n_1}} \ C \bullet X + E[Q(X,\omega)] \\ &\text{subject to} \ \ A_i \bullet X = b_i, \quad i = 1, 2, \ldots, m_1 \\ &\qquad\qquad\quad X \succeq 0 \end{aligned} \tag{8}$$

where $X \in R_s^{n_1\times n_1}$ is the first-stage decision variable, $C \in R_s^{n_1\times n_1}$ is a given matrix, $\mathbf{b} \in R^{m_1}$ another given vector, $A \in R_s^{n_1\times n_1}$, $\mathbf{c}$, $\mathbf{b}$ and A are deterministic data. $Q(X, \omega)$ is the minimum of the second stage problem

$$\begin{aligned} \min_{Y(\omega)\in R_s^{n_2\times n_2}} \quad & Q(\omega)\bullet Y \\ \text{subject to} \quad & T_i(\omega)\bullet X + W_i(\omega)\bullet Y = h_i(\omega) \qquad i=1,2,\ldots,m_2 \\ & Y \succeq 0 \end{aligned} \tag{9}$$

and

$$E[Q(X,\omega)] = \textstyle\int_\Omega Q(X,\omega)P(\mathrm{d}\omega) \tag{10}$$

where $Y(\omega) \in R_s^{n_2\times n_2}$ is the second-stage decision vector, $Q \in R_s^{n_2\times n_2}$, $T_i(\omega) \in R_s^{n_1\times n_1}$, $W_i(\omega) \in \Re_s^{n_2\times n_2}$, $\mathbf{h} \in R^{m_2}$ and $\omega \in \Omega$ is a random outcome with known probability distribution P, whose realizations will affect the coefficient matrices of the problem.

A special case of semidefinite programming (SDP) is given by second-order cone programming (SOCP). SOCP problems consist in convex optimization problems in which a linear function is minimized over the intersection of an affine set and the product of second-order (Lorentz) cones:

$$\mathscr{K}_n := \{\mathbf{x} = (x_0; \bar{\mathbf{x}}) \in \Re^n : x_0 \geq \|\bar{\mathbf{x}}\|\} \ , \tag{11}$$

where $\|\cdot\|$ refers to the standard Euclidean norm and n the dimension of $\mathscr{K}_n$ (see Alizadeh and Goldfarb, (2003) [1]) .
A second-order cone can be embedded in the cone of positive semidefinite matrices since a second-order cone constraint is equivalent to a linear matrix inequality according to the following relation:

$$Arw(\mathbf{x}) := \begin{pmatrix} x_0 & -\bar{\mathbf{x}}^T \\ -\bar{\mathbf{x}} & x_0 I \end{pmatrix} \succeq 0 \ \Leftrightarrow \ x_0 \geq \|\bar{\mathbf{x}}\| \ . \tag{12}$$

In fact $Arw(\mathbf{x}) \succeq 0$ if and only if either $\mathbf{x} = \mathbf{0}$, or $x_0 > 0$ and it holds true the Shur Complement $x_0 - \bar{\mathbf{x}}^T(x_0 I)^{-1}\bar{\mathbf{x}} \geq 0$.
Notice that the computational effort per iteration required by interior point methods to solve SOCP problems is less of that required to solve SDP's problems of similar size and structure. In fact the number of iterations to decrease the duality gap to a constant fraction of itself using the primal dual method, is bounded above by $O(\sqrt{N})$, where N is the number of second-order constraints, for the SOCP algorithm, and by $O(\sqrt{\sum_{i=1}^N n_i})$, where n_i is the dimension of each second-order cone constraint $i = 1,\ldots,N$, for the SDP algorithm (see Nesterov and Nemirovsky (1994) [11]). Furthermore, each iteration is much faster: in the SOCP algorithm is $O(n^2 \sum_{i=1}^N n_i)$ and in the SDP $O(n^2 \sum_{i=1}^N n_i^2)$ where n is the dimension of the optimization variable $\mathbf{x}$.

3 Stochastic semidefinite program for modeling networks with moving nodes

In this section we recall the semidefinite stochastic location-aided routing (SLAR) model described in Ariyawansa and Zhu (2006) [3] and we propose a new version with a modified second stage. The semidefinite stochastic location-aided routing (SLAR) model can be summarized as follows: a sender node S needs to find a route to a destination node D through broadcasting a route request to its neighbours. Once D receives the signal (and this should happen within a time-out interval t_1, otherwise the route request has to be restarted), it will respond by reversing the path followed by the request just received. We assume S static and D moving at a random speed. Notice that the communication is successful when the reply message is sent back to the source node.
Consider an origin node S that needs to find a route to another destination node D where:

- The source node S knows the location $\boldsymbol{l}, \boldsymbol{l} \in \Re^n$ of the destination node D at time t_0 and viceversa the node D knows node S location ar the same time. We suppose S positioned at the origin $\mathbf{0}$ and analyze the problem in relative terms;
- The nodes in the network are uniformly distributed;
- The node D moves at a random speed $v(\omega_1)$, which depends on an underlying outcome ω_1 in an event space Ω_1 with known probability distribution P_1;
- The node D moves towards a (normalized) random direction $\mathbf{d}(\omega_2)$, which depends on an underlying outcome ω_2 in an event space Ω_2 with a known probability distribution P_2;
- P_1 and P_2 are both discrete;
- Let $\left\{\left(v^{(k)}, \mathbf{d}^{(k)}\right) : k=1,\ldots,K'\right\}$ be the possible realizations of the speed, direction couple $(v(\omega_1), \mathbf{d}(\omega_2))$ with probability $p_k := P\left(\left(v\left(\omega_1\right), \mathbf{d}\left(\omega_2\right)\right) = \left(v^k, \mathbf{d}^k\right)\right)$, $k=1,\ldots,K'$;
- At time $t_1 > t_0$ the node D will be at location $\boldsymbol{l} + (t_1 - t_0)v^{(k)}\mathbf{d}^{(k)}$ with probability p_k;
- The K ellipsoids

  $$E_k = \{\mathbf{u} \in \Re^n : \mathbf{u}^T H_k \mathbf{u} + 2{\mathbf{g}_k}^T \mathbf{u} + \nu_k \leq 0\}, \qquad k = 1, 2, \ldots, K \tag{13}$$

  are the realizations of the random ellipsoid $\tilde{E} = \{\mathbf{u} \in \Re^n : \mathbf{u}^T \tilde{H} \mathbf{u} + 2\tilde{\mathbf{g}}^T \mathbf{u} + \tilde{\nu} \leq 0\}$, where $\tilde{H} \in \Re_s^{n\times n}$, $\tilde{H}_k \succ 0$, $\tilde{\mathbf{g}}(\omega) \in \Re^n$ and $\tilde{\nu}(\omega) \in \Re$, for $k = 1, \ldots, K$ are random data depending on the outcome ω, and $H_k \in \Re_s^{n\times n}$, $H_k \succ 0$, $\mathbf{g}_k \in \Re^n$ and $\nu_k \in \Re$ for $k = 1, 2, \ldots, K$;
- At time t_1 the node D is in E_k with probability p_k for $k = 1, 2, \ldots, K$.

Given the location of D at time t_0, to determine the new location of D at time t_1 Ariyawansa and Zhu use the following procedure:

Stage 1 Pick a disk C

$$C = \{\mathbf{u} \in \Re^n : \mathbf{u}^T\mathbf{u} - 2\tilde{\mathbf{u}}^T\mathbf{u} + \gamma \leq 0\} \tag{14}$$

with center in $\tilde{\mathbf{u}}$ and radius $\sqrt{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}} - \gamma}$ which contains the disk C_0 centered in $\boldsymbol{l}$ with radius $v(t_1 - t_0)$, where v is the minimum speed the node D is supposed to move.

Stage 2 If happens that node D is in C, no further action is needed; otherwise D is in E_k for some k, thus we pick a new disk C^*

$$C^* = \{\mathbf{u} \in \Re^n : \mathbf{u}^T\mathbf{u} - 2\tilde{\mathbf{u}}^T\mathbf{u} + \tilde{\gamma} \leq 0\} \tag{15}$$

with center in $\tilde{\mathbf{u}}$ and radius $\sqrt{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}} - \tilde{\gamma}}$ which contains the ellipsoids E_k for each $k = 1, 2, \ldots, K$.

On the other hand, to take advantage of the problem two stage formulation, we propose a second stage action as a recourse decision contingent on the realized scenario. A new circle is in this way generated conditionally on the realized ellipsoid. We suggest the following modified Stage 2:

Stage 2m For each scenario $k = 1, \ldots, K$, if happens that node D is in C, no further action is needed; otherwise D is in E_k and we pick a new disk C_k^*

$$C_k^* = \{\mathbf{u} \in \Re^n : \mathbf{u}^T\mathbf{u} - 2\tilde{\mathbf{u}}^T\mathbf{u} + \tilde{\gamma}_k \leq 0\} \tag{16}$$

with center in $\tilde{\mathbf{u}}$ and radius $\sqrt{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}} - \tilde{\gamma}_k}$ which contains the ellipsoid E_k. To be consistent with practical requirements, we fix an upper bound on the difference $\gamma - \tilde{\gamma}_k$.
In this way we are sure that at the cost of enlarging the radius we can pick up the new position of D.

The decision variables are given by:

$$\mathbf{x} = [d_1, d_2, \tilde{\mathbf{u}}, \gamma, \tau]^T , \tag{17}$$
$$\mathbf{y} = [\mathbf{z}, \tilde{\boldsymbol{\gamma}} , \boldsymbol{\delta}]^T , \tag{18}$$

where $\mathbf{x}$ is the first stage decision variable with components

- d_1: is an upper bound on the distance between the center of the disk

$$C = \{\mathbf{u} \in \Re^n : \mathbf{u}^T\mathbf{u} - 2\tilde{\mathbf{u}}^T\mathbf{u} + \gamma \leq 0\}$$

  and the source node ($S = \mathbf{0}$);

- d_2: is an upper bound on square of the radius of the disk C;

- $\tilde{\mathbf{u}} \in \Re^n$: is the center of disk C;
- γ: is a coefficient in the equation of disk C;
- τ: is a nonnegative parameter (see Vandenberghe and Boyed (1996), [14]; Sun and Freund (2004) [13]).

and $\mathbf{y}$ is the second stage decision vector whose components are

- $\mathbf{z} \in \Re^K$: is the vector of the upper bounds at each scenario k on the distance between the coefficients γ and $\tilde{\gamma}_k$ respectively in C and C_k^*;
- $\tilde{\boldsymbol{\gamma}} \in \Re^K$: is the vector of the coefficients $\tilde{\gamma}_k$ of the second stage circles C_k^*, $k = 1, \ldots, K$.
- $\boldsymbol{\delta} \in \Re^K$: is a vector of nonnegative parameters (see Vandenberghe and Boyed (1996) [14]; Sun and Freund (2004), [13]).

The unit cost vectors are given by:

$$\mathbf{c} = [\tilde{c}, \alpha, \mathbf{0}, 0, 0]^T , \qquad (19)$$

$$\mathbf{q} = [\boldsymbol{\beta}, \mathbf{0}, \mathbf{0}]^T , \qquad (20)$$

where $\tilde{c}$ denotes the cost per unit of the Euclidean distance between the center of the disk C and the source node, $\alpha > 0$ is the cost per unit of the square of the radius of C, and $\beta > 0$ is the cost per unit increase of the square of the radius after the realization of the random ellipsoids. Then our modified SLAR model is given by

$$\begin{aligned} \min_{\mathbf{x}\in\Re^{n+4}} \quad & \mathbf{c}^T\mathbf{x} + E[Q(\mathbf{x},\omega)] \\ \text{subject to} \quad & \begin{pmatrix} I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & \gamma \end{pmatrix} \preceq \tau \begin{pmatrix} I & -\boldsymbol{l} \\ -\boldsymbol{l}^T & \|\boldsymbol{l}\|^2 - (t_1 - t_0)^2 v^2 \end{pmatrix} , \\ & 0 \leq \tau , \\ & 0 \preceq \begin{pmatrix} d_1 I & \tilde{\mathbf{u}} \\ \tilde{\mathbf{u}}^T & d_1 \end{pmatrix} , \\ & 0 \preceq \begin{pmatrix} I & \tilde{\mathbf{u}} \\ \tilde{\mathbf{u}}^T & d_2 + \gamma \end{pmatrix} , \end{aligned} \qquad (21)$$

where $Q(\mathbf{x}, \omega)$ is the minimum of the problem

$$\begin{array}{ll} \min\limits_{\mathbf{y} \in \Re^{3n}} & \mathbf{q}^T \mathbf{y} \\ \text{subject to} & \begin{pmatrix} I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & \tilde{\gamma}_k \end{pmatrix} \preceq \delta_k \begin{pmatrix} H_k & \mathbf{g}_k \\ \mathbf{g}_k{}^T & v_k \end{pmatrix} , \quad k = 1, \ldots, K , \\ & 0 \leq \delta_k , \quad k = 1, \ldots, K , \\ & 0 \leq \gamma - \tilde{\gamma}_k \leq z_k , \quad k = 1, \ldots, K . \end{array} \tag{22}$$

On the contrary of the Ariyawansa and Zhu (2006) [2] formulation, according to our Stage 2m, every scenario indexed by $k \in K$ is weighed for its probability p_k. In our formulation the second stage variable $\mathbf{z} \in \Re^K$ associated to the non-zero cost $\mathbf{q}$ depends by the scenarios considered, while in [2] the corresponding second stage variable appears with probability 1.

4 Stochastic Second-order cone model for SLAR

From a computational point of view, the effort per iteration required by interior-point method to solve SOCP problems is lower than the one required to solve SDP's of similar size and structure. The aim of this section is to formulate the semidefinite stochastic location-aided routing (SLAR) problem presented in the previous section as a stochastic second-order cone SSOCP problem. We rewrite each semidefinite constraint as a second order cone one. We start with the constraint

$$\begin{pmatrix} I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & \gamma \end{pmatrix} \preceq \tau \begin{pmatrix} I & -\boldsymbol{l} \\ -\boldsymbol{l}^T & \| \boldsymbol{l}\|^2 - (t_1 - t_0)^2 v^2 \end{pmatrix} , \tag{23}$$

which represents the condition of inclusion of the disk C_0 in the first stage disk C, is equivalent to

$$0 \preceq \begin{pmatrix} \tau I - I & -\tau \, \boldsymbol{l} + \tilde{\mathbf{u}} \\ -\tau \, \boldsymbol{l}^T + \tilde{\mathbf{u}}^T & \tau \, \| \boldsymbol{l}\|^2 - \tau \,(t_1 - t_0)^2 \, v^2 - \gamma \end{pmatrix} \tag{24}$$

and it holds if and only if, by Schur Complements, $\tau I - I > 0$, i.e. $\tau > 1$ (or if $\tau = 1$, $-\tau \, \boldsymbol{l}^T + \tilde{\mathbf{u}}^T = 0$), and

$$\tau \, \| \boldsymbol{l}\|^2 - \tau \,(t_1 - t_0)^2 \, v^2 - \gamma - \left(-\tau \, \boldsymbol{l}^T + \tilde{\mathbf{u}}^T\right) (\tau I - I)^{-1} \left(-\tau \, \boldsymbol{l} + \tilde{\mathbf{u}}\right) \geq 0 \ , \tag{25}$$

or equivalently

$$\left(\tau \, \| \boldsymbol{l}\|^2 - \tau \,(t_1 - t_0)^2 \, v^2 - \gamma\right) (\tau - 1) - \left(-\tau \, \boldsymbol{l}^T + \tilde{\mathbf{u}}^T\right) \left(-\tau \, \boldsymbol{l} + \tilde{\mathbf{u}}\right) \geq 0 \ , \tag{26}$$

that is

$$\tau \, \| \boldsymbol{l}\|^2 - \tau \,(t_1 - t_0)^2 \, v^2 - \gamma - \sum_{j=1}^{n} \frac{(-\tau l_j + \tilde{u}_j)^2}{(\tau - 1)} \geq 0 \ . \tag{27}$$

If we define $\boldsymbol{r} = (r_1, \ldots, r_n)$, where $r_j = \dfrac{(-\tau l_j + \tilde{u}_j)^2}{(\tau - 1)}$ for all j such that $\tau > 1$ and $r_j = 0$ otherwise (see Alizadeh and Goldfarb (2003), [1]), then (27) is equivalent to

$$\tau \, \| \, \boldsymbol{l} \|^2 - \tau \, (t_1 - t_0)^2 \, v^2 - \mathbf{1}^T \, \boldsymbol{r} \geq \gamma \; . \tag{28}$$

Since we are minimizing the radius of the circle C, $\sqrt{\tilde{\mathbf{u}}^T \tilde{\mathbf{u}} - \gamma}$, we can relax the definition of r_j replacing it by $(\tau l_j - \tilde{u}_j)^2 \leq r_j \, (\tau - 1)$, $j = 1, \ldots, n$. Combining all the above constraints, (23) is equivalent to the following formulation involving only linear and restricted hyperbolic first-stage constraints:

$$(\tau l_j - \tilde{u}_j)^2 \;\leq\; r_j \, (\tau - 1) \;, \qquad j = 1, \ldots, n \;, \tag{29}$$

$$\gamma \;\leq\; \tau \, \| \, \boldsymbol{l} \|^2 - \tau \, (t_1 - t_0)^2 \, v^2 - \mathbf{1}^T \, \boldsymbol{r} \;, \tag{30}$$

$$\tau \;\geq\; 1 \; . \tag{31}$$

Notice that the restricted hyperbolic constraint (29) is equivalent to the following n 3-dimensional second-order cone inequalities:

$$\left\| \begin{pmatrix} 2 \, (\tau l_j - \tilde{u}_j) \\ r_j - \tau + 1 \end{pmatrix} \right\| \leq r_j + \tau - 1 \Leftrightarrow \begin{pmatrix} r_j + \tau - 1 \\ 2 \, (\tau l_j - \tilde{u}_j) \\ r_j - \tau + 1 \end{pmatrix} \in \mathscr{K}_3 \;, \quad j = 1, \ldots, n; \tag{32}$$

and each of the linear constraints (30) and (31) are 1-dimensional second-order cone constraints.

On the other hand

$$0 \preceq \begin{pmatrix} d_1 I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & d_1 \end{pmatrix} \Leftrightarrow d_1 \geq \sqrt{\tilde{\mathbf{u}}^T \tilde{\mathbf{u}}} \Leftrightarrow \begin{pmatrix} d_1 \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1} \;, \tag{33}$$

and

$$0 \preceq \begin{pmatrix} I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & d_2 + \gamma \end{pmatrix} \Leftrightarrow d_2 + \gamma \geq \tilde{\mathbf{u}}^T \tilde{\mathbf{u}} \Leftrightarrow \begin{pmatrix} \sqrt{d_2 + \gamma} \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1} \;; \tag{34}$$

the second stage constraint

$$\begin{pmatrix} I & -\tilde{\mathbf{u}} \\ -\tilde{\mathbf{u}}^T & \tilde{\gamma}_k \end{pmatrix} \preceq \delta_k \begin{pmatrix} H_k & \mathbf{g}_k \\ \mathbf{g}_k{}^T & v_k \end{pmatrix} \;, \quad k = 1, \ldots, K \;, \tag{35}$$

which represents the condition of inclusion, at each scenario k, of the ellipsoid into the disk C_k^*, is equivalent to

$$M_k := \begin{pmatrix} \delta_k H_k - I & \delta_k \mathbf{g}_k + \tilde{\mathbf{u}} \\ \delta_k \mathbf{g}_k^T + \tilde{\mathbf{u}}^T & \delta_k v_k - \tilde{\gamma}_k \end{pmatrix} \succeq 0 \;, \qquad k = 1, \ldots, K \; . \tag{36}$$

Following Alizadeh and Goldfarb (2003), [1], let $H_k = Q_k \Lambda_k Q_k^T$ be the spectral decomposition of H_k, $\Lambda_k = Diag\left(\lambda_{k1};\ldots;\lambda_{kn}\right)$ and $\mathbf{h}_k = Q_k^T\left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}}\right)$, for $k = 1,\ldots,K$. Then

$$\bar{M}_k := \begin{pmatrix} Q_k^T & \mathbf{0} \\ \mathbf{0} & 1 \end{pmatrix} M_k \begin{pmatrix} Q_k & \mathbf{0} \\ \mathbf{0} & 1 \end{pmatrix} = \begin{pmatrix} \delta_k \Lambda_k - I & \mathbf{h}_k \\ \mathbf{h}_k^T & \delta_k v_k - \tilde{\gamma}_k \end{pmatrix} \succeq 0 \ , \tag{37}$$

for $k = 1\ldots,K$, and $M_k \succeq 0$ if and only if $\bar{M}_k \succeq 0$. It holds if and only if $\delta_k \geq \frac{1}{\lambda_{min}(\Lambda_k)}$, i.e. $\delta_k \lambda_{kj} - 1 \geq 0 \ \forall k,j$, $h_{kj} = 0$ if $\delta_k \lambda_{kj} - 1 = 0$ and the Shur complement of the columns and rows of $\bar{M}_i$ that are not zero

$$\delta_k v_k - \tilde{\gamma}_k - \sum_{\delta_k \lambda_{kj} > 1} \frac{h_{kj}^2}{\delta_k \lambda_{kj} - 1} \geq 0 \ . \tag{38}$$

If we define $\boldsymbol{s_k} = \left(s_{k1};\ldots;s_{kn}\right)$, where $s_{kj} = \frac{h_{kj}^2}{\delta_k \lambda_{kj} - 1}$, for all j such that $\delta_k \lambda_{kj} > 1$ and $s_{kj} = 0$, otherwise, then (38) is equivalent to

$$\tilde{\gamma}_k \leq \delta_k v_k - \mathbf{1}^T \ \boldsymbol{s_k} \ . \tag{39}$$

Since we are minimizing the radius of the circle C_k^*, $\sqrt{\tilde{\mathbf{u}}^T \tilde{\mathbf{u}} - \tilde{\gamma}_k}$, we can relax the definition of s_{kj} replacing it by $h_{kj}^2 \leq s_{kj}\left(\delta_k \lambda_{kj} - 1\right)$, $k = 1,\ldots,K$, $j = 1,\ldots,n$. Combining all of the above constraints (35) is equivalent to the following formulation involving only linear and restricted hyperbolic second-stage constraints:

$$\begin{aligned} \mathbf{h}_k &= Q_k^T\left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}}\right) \ , \qquad k = 1,\ldots,K \ , & (40)\\ h_{kj}^2 &\leq s_{kj}\left(\delta_k \lambda_{kj} - 1\right) \ , \qquad k = 1,\ldots,K, \quad j = 1,\ldots,n \ , & (41)\\ \tilde{\gamma}_k &\leq \delta_k v_k - \mathbf{1}^T \ \boldsymbol{s_k} \ , \qquad k = 1,\ldots,K, & (42)\\ \delta_k &\geq \frac{1}{\lambda_{min}\left(\Lambda_k\right)} \ , \qquad k = 1,\ldots,K \ . & (43) \end{aligned}$$

Notice that the linear constraint (40) is equivalent to $2nK$ 1-dimensional second-order cone inequalities given by

$$\begin{aligned} Q_k^T\left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}}\right) - \mathbf{h}_k &\geq 0 \ , \qquad k = 1,\ldots,K \ , & (44)\\ -Q_k^T\left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}}\right) + \mathbf{h}_k &\geq 0 \ , \qquad k = 1,\ldots,K \ ; & (45) \end{aligned}$$

the restricted hyperbolic constraint (41) is equivalent to the following nK 3-dimensional second-order cone inequalities:

$$\left\| \begin{pmatrix} 2h_{kj} \\ s_{kj} - \delta_k \lambda_{kj} + 1 \end{pmatrix} \right\| \leq s_{kj} + \delta_k \lambda_{kj} - 1 \Leftrightarrow \begin{pmatrix} s_{kj} + \delta_k \lambda_{kj} - 1 \\ 2h_{kj} \\ s_{kj} - \delta_k \lambda_{kj} + 1 \end{pmatrix} \in \mathscr{K}_3 \ , \quad k = 1,\ldots K, \quad j = 1,\ldots,n, \tag{46}$$

and each of the linear constraints (42) and (43) are K 1-dimensional second-order cone constraints. In conclusion the SLAR model (22) and (23) can be formulate as a stochastic second-order cone SSOCP problem with two (n+1)-dimensional second-order cone constraints (see eqs. (33), (34)), $n(K+1)$ 3-dimensional second-order cone constraints (see eqs. (32) and (46)) and with all the other constraints linear, in the following way:

$$\begin{aligned}
&\min_{\mathbf{x}\in\Re^{n+4}} \mathbf{c}^T\mathbf{x} + E[Q(\mathbf{x},\omega)] \\
&\text{subject to} \quad \begin{pmatrix} r_j+\tau-1 \\ 2\,(\tau l_j-\tilde{u}_j) \\ r_j-\tau+1 \end{pmatrix} \in \mathscr{K}_3\ , \quad j=1,\ldots,n\ , \\
&\qquad\qquad\quad \gamma \le \tau\,\|\,\boldsymbol{l}\|^2 - \tau\,(t_1-t_0)^2\,v^2 - \mathbf{1}^T\ \boldsymbol{r}\ , \\
&\qquad\qquad\quad 1\le\tau\ , \\
&\qquad\qquad\quad \begin{pmatrix} d_1 \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1}\ , \\
&\qquad\qquad\quad \begin{pmatrix} \sqrt{d_2+\gamma} \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1}\ ,
\end{aligned} \tag{47}$$

where $Q(\mathbf{x},\omega)$ is the minimum of the problem

$$\begin{aligned}
&\min_{\mathbf{y}\in\Re^{3n}} \mathbf{q}^T\mathbf{y} \\
&\text{subject to} \quad \begin{pmatrix} s_{kj}+\delta_k\lambda_{kj}-1 \\ 2h_{kj} \\ s_{kj}-\delta_k\lambda_{kj}+1 \end{pmatrix} \in \mathscr{K}_3\ , \quad k=1,\ldots K, \quad j=1,\ldots,n\ , \\
&\mathbf{h}_k = Q_k^T\,(\delta_k\mathbf{g}_k+\tilde{\mathbf{u}})\ , \quad k=1,\ldots,K\ , \\
&\tilde{\gamma}_k \le \delta_k v_k - \mathbf{1}^T\ \boldsymbol{s_k}\ , \quad k=1,\ldots,K\ , \\
&\delta_k \ge \frac{1}{\lambda_{min}\,(\Lambda_k)}\ , \quad k=1,\ldots,K\ , \\
&0\le\delta_k\ , \quad k=1,\ldots,K\ , \\
&0\le\gamma-\tilde{\gamma}_k\le z_k\ , \quad k=1,\ldots,K\ ,
\end{aligned} \tag{48}$$

or equivalently, by denoting by p_k the probability of scenario k:

$$\begin{aligned}
&\min_{\mathbf{x}\in\Re^{n+4},\ \mathbf{y}\in\Re^{3n}} \mathbf{c}^T\mathbf{x} + \sum_{k=1}^{K} p_k\mathbf{q}^T\mathbf{y} \\
&\text{subject to} \quad \begin{pmatrix} r_j+\tau-1 \\ 2\,(\tau l_j-\tilde{u}_j) \\ r_j-\tau+1 \end{pmatrix} \in \mathscr{K}_3\ , \quad j=1,\ldots,n\ , \\
&\qquad\qquad\quad \gamma \le \tau\,\|\,\boldsymbol{l}\|^2 - \tau\,(t_1-t_0)^2\,v^2 - \mathbf{1}^T\ \boldsymbol{r}\ ,
\end{aligned}$$

$$
\begin{aligned}
& 1 \le \tau \ , \\
& \begin{pmatrix} d_1 \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1} \ , \\
& \begin{pmatrix} \sqrt{d_2 + \gamma} \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1} \ ,
\end{aligned} \tag{49}
$$

$$
\begin{pmatrix} s_{kj} + \delta_k \lambda_{kj} - 1 \\ 2h_{kj} \\ s_{kj} - \delta_k \lambda_{kj} + 1 \end{pmatrix} \in \mathscr{K}_3 \ , \quad k = 1, \dots K, \quad j = 1, \dots, n \ ,
$$

$$
\begin{aligned}
\mathbf{h}_k &= Q_k^T \left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}} \right) \ , \quad k = 1, \dots, K \ , \\
\tilde{\gamma}_k &\le \delta_k v_k - \mathbf{1}^T \ \boldsymbol{s_k} \ , \quad k = 1, \dots, K \ , \\
\delta_k &\ge \frac{1}{\lambda_{min} \left(\Lambda_k \right)} \ , \quad k = 1, \dots, K \ , \\
0 &\le \delta_k \ , \quad k = 1, \dots, K \ , \\
0 &\le \gamma - \tilde{\gamma}_k \le z_k \ , \quad k = 1, \dots, K \ .
\end{aligned}
$$

We observe that in the implementation the constraint

$$
\begin{pmatrix} \sqrt{d_2 + \gamma} \\ \tilde{\mathbf{u}} \end{pmatrix} \in \mathscr{K}_{n+1} \ , \tag{50}
$$

has been treated as a rotated quadratic cone (or hyperbolic constraint)

$$
\mathscr{K}_{n+2} = \left\{ \mathbf{u} \in \Re^{n+2} : 2u_1 u_2 \ge \sum_{j=3}^{n+2} u_j^2, u_1, u_2 \ge 0 \right\} \tag{51}
$$

with $u_2 = d_2 + \gamma$, $u_j = x_{j-2}$, $j = 3, \dots, n+2$ intersected with the hyperplane $u_1 = 1/2$.
By unraveling each second order constraint, the model (49) can be formulated also in the following equivalent way:

$$
\begin{aligned}
\min_{\mathbf{x} \in \Re^{n+4}, \ \mathbf{y} \in \Re^{3n}} \quad & \mathbf{c}^T \mathbf{x} + \sum_{k=1}^{K} p_k \mathbf{q}^T \mathbf{y} \\
\text{subject to} \quad & \left(\tau \, \| \, \boldsymbol{l} \|^2 - \tau \left(t_1 - t_0 \right)^2 v^2 - \gamma \right) \left(\tau - 1 \right) \ge \| \tau \ \boldsymbol{l} + \tilde{\mathbf{u}} \|^2 \ , \\
& \tau \ge 1 \ , \\
& d_1^2 \ge \| \tilde{\mathbf{u}} \|^2 \ , \\
& d_2 + \gamma \ge \| \tilde{\mathbf{u}} \|^2 \ , \\
& \left(s_{kj} + \delta_k \lambda_{kj} - 1 \right)^2 \ge \left\| \begin{pmatrix} 2h_{kj} \\ s_{kj} - \delta_k \lambda_{kj} + 1 \end{pmatrix} \right\|^2 \ ,
\end{aligned} \tag{52}
$$

$$
\begin{aligned}
&\mathbf{h}_k = Q_k^T \left(\delta_k \mathbf{g}_k + \tilde{\mathbf{u}}\right) \ , \quad k = 1, \ldots, K \ , \\
&\tilde{\gamma} \leq \delta_k v_k - \mathbf{1}^T \mathbf{s}_k \ , \quad k = 1, \ldots, K \ , \\
&\delta_k \geq \frac{1}{\lambda_{min}\left(\Lambda_k\right)} \ , \quad k = 1, \ldots, K \ , \\
&\delta_k \geq 0 \ , \quad k = 1, \ldots, K \ , \\
&\gamma - \tilde{\gamma}_k \geq 0 \ , \quad k = 1, \ldots, K \ , \\
&\gamma - \tilde{\gamma}_k \leq z_k \ , \quad k = 1, \ldots, K \ , \\
&d_1 \geq 0 \ , \\
&d_2 \geq 0 \ .
\end{aligned}
$$

5 Ellipsoid scenario generation

In this section we consider the generation of random ellipsoids

$$E_k = \{\mathbf{u} \in \Re^n : \mathbf{u}^T H_k \mathbf{u} + 2{\mathbf{g}_k}^T \mathbf{u} + \nu_k \leq 0\}, \quad k = 1, 2, \ldots, K \ . \tag{53}$$

The computational experiment is limited to the case $n = 2$, of real ellipses in the plane $\Re^2$. The algebraic equation for a second-order curve is of the type:

$$e_{11}u_1^2 + 2e_{12}u_1u_2 + e_{22}u_2^2 + 2e_{13}u_1 + 2e_{23}u_2 + e_{33} = 0 \ , \tag{54}$$

or equivalently, in matricial notation

$$\begin{pmatrix} u_1 & u_2 \end{pmatrix} \begin{pmatrix} e_{11} & e_{12} \\ e_{12} & e_{22} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + 2 \begin{pmatrix} e_{13} & e_{23} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + e_{33} = 0$$

For an ellipse, the coefficients of the simmetrix matrix E associated to eq. (54):

$$E = \begin{pmatrix} e_{11} & e_{12} & e_{13} \\ e_{12} & e_{22} & e_{23} \\ e_{13} & e_{23} & e_{33} \end{pmatrix} \ ,$$

have to satisfy the following conditions on the sign of the invariants (see e.g. Ilyin and Poznyak (1981), [7]):

$$I_2 = \begin{vmatrix} e_{11} & e_{12} \\ e_{12} & e_{22} \end{vmatrix} > 0 \ , \quad I_3 = \begin{vmatrix} e_{11} & e_{12} & e_{13} \\ e_{12} & e_{22} & e_{23} \\ e_{13} & e_{23} & e_{33} \end{vmatrix} < 0 \ . \tag{55}$$

We note that for each scenario k the coefficients H, $\mathbf{g}$ and ν of eq. (53) correspond to

$$H = \begin{pmatrix} e_{11} & e_{12} \\ e_{12} & e_{22} \end{pmatrix} \ , \quad \mathbf{g} = \begin{pmatrix} e_{13} & e_{23} \end{pmatrix} \ , \quad \nu = e_{33} \ ,$$

and in order to satisfy the condition $H \succ 0$ we have to consider also

$$e_{11} > 0 \ . \tag{56}$$

The center O, $\mathbf{u}^0 = (u_1^0, u_2^0)$ of the second-order curve (54) is obtained as solution of the following system:

$$\begin{cases} e_{11}u_1^0 + e_{12}u_2^0 + e_{13} = 0 \ , \\ e_{12}u_1^0 + e_{22}u_2^0 + e_{23} = 0 \ , \end{cases} \tag{57}$$

the angle φ between the u_1-axis and the main axis of the conic is such that

$$\cot 2\varphi = \frac{e_{11} - e_{12}}{2e_{12}} \ , \tag{58}$$

or equivalently is given by

$$\varphi = (\frac{\pi}{4} - \frac{1}{2}\arctan(\frac{e_{11} - e_{12}}{2e_{12}})) \ , \tag{59}$$

and the semiaxes s_{u_1} and s_{u_2} of the ellipse are equal to

$$\begin{cases} s_{u_1} = \sqrt{\dfrac{-I_3}{I_2\left(e_{12}\sin 2\varphi + \frac{1}{2}\left(e_{11} - e_{22}\right)\cos 2\varphi + \frac{1}{2}\left(e_{11} + e_{22}\right)\right)}} \ , \\ s_{u_2} = \sqrt{\dfrac{-I_3}{I_2\left(-e_{12}\sin 2\varphi - \frac{1}{2}\left(e_{11} - e_{22}\right)\cos 2\varphi + \frac{1}{2}\left(e_{11} + e_{22}\right)\right)}} \ . \end{cases} \tag{60}$$

Thus the parametric equation of the ellipse is given by

$$\begin{cases} u_1 = s_{u_1}\cos\varphi\cos\vartheta - s_{u_2}\sin\varphi\sin\vartheta + u_1^0 \ , \quad \vartheta \in [0, 2\pi] \ , \\ u_2 = s_{u_1}\sin\varphi\cos\vartheta + s_{u_2}\cos\varphi\sin\vartheta + u_2^0 \ , \quad \vartheta \in [0, 2\pi] \ . \end{cases} \tag{61}$$

Another equivalent formulation of an ellipse can be expressed as function of the center $\mathbf{u}^0$ and of the semiaxes s_{u_1} and s_{u_2} as follow:

$$\left\| Q^{-1}\left(\mathbf{u} - \mathbf{u}^0\right)\right\|^2 \leq 1 \ , \tag{62}$$

where $Q = K^{-1}\bar{Q}K$, with the matrix of rotation K given by

$$K = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} \ , \tag{63}$$

and

$$\bar{Q} = \begin{pmatrix} s_{u_1} & 0 \\ 0 & s_{u_2} \end{pmatrix} \ . \tag{64}$$

It is easy to show that the equation (53) is obtained from (62) by taking

$$H = Q^{-2} \ , \quad \mathbf{g} = -\mathbf{u}^{0^T}Q^{-2} \ , \quad \nu = \mathbf{x}^{0^T}Q^{-2}\mathbf{u}^0 - 1 \ . \tag{65}$$

The random ellipsoids E_k $k = 1, \ldots, K$ are generated in the following way:

- the coordinates $\left(u_1^{0,k}, u_2^{0,k}\right)$ of center O^k, $k = 1, \ldots, K$ are respectively extracted by an uniform distribution in the interval $\left(\sqrt{8}-1, 1+\sqrt{8}\right)$ and by a normal distribution $\mathcal{N}(0, 0.5)$
- the semiaxis $s_{u_1}^k$ $k = 1, \ldots, K$ is extracted by a normal distribution $\mathcal{N}(2, 1)$;
- the semiaxis $s_{u_2}^k$ $k = 1, \ldots, K$ is extracted by a normal distribution $\mathcal{N}(1, 0.5)$;
- the angle φ^k $k = 1, \ldots, K$ is extracted by an uniform distribution in the interval $\left[0, \frac{\pi}{2}\right]$.

Furthermore, E_k are generated by imposing upper and lower bounds on the u_2-coordinate of the centre O^k and on the length of the main semiaxis $s_{u_1}^k$ and $s_{u_2}^k$ according to the following conditions:

$$\begin{cases} u_2^{0,min} \leq u_2^{0,k} \leq u_2^{0,max} , & \forall\, k = 1, \ldots, K \ , \\ s_{u_1}^{min} < s_{u_1}^k \leq \ s_{u_1}^{max} , & \forall\, k = 1, \ldots, K \ , \\ s_{u_1}^{min} < s_{u_2}^k \leq \ s_{u_2}^{max} , & \forall\, k = 1, \ldots, K \ . \end{cases} \tag{66}$$

The way we have chosen for generating the ellipsoids, corresponds to a typical real situation in which people are moving along preferred directions (different motorways), identified by the length of main semiaxis $s_{u_1}^k$ and the size of the angle φ^k, with the possibility to exit from the motorway for short distances (length of the second semiaxis $s_{u_2}^k$). The center position O^k represents how far one can move from the original starting position.

6 Numerical results

In this section we present numerical results obtained for the semidefinite stochastic location-aided routing (SLAR) problem presented in section 3. The SLAR model is compared in terms of performance with respect to the stochastic second order cone model SSOCP presented in section 4. The simulation is based on the scenarios randomly generated under MATLAB 7.4.0 framework, according to the method described in the previous section with $u_2^{0,min} = -1$, $u_2^{0,max} = 1$, $s_{u_1}^{min} = s_{u_2}^{min} = 0.1$ and $s_{u_1}^{max} = s_{u_2}^{max} = 3$. The stochastic second order cone programming approach (SSOCP) (49) was implemented in GAMS 22.5 by using the Mosek package. The stochastic semidefinite programming approach (SSDP) (22) and (23) was implemented in MATLAB 7.4.0 using the software package "DSDP" developed by Benson, Ye and Zhang [4].

In our computational experiments we supposed equiprobable scenarios; furthermore we have fixed the location $\boldsymbol{l} = (1, 1)$ of the node D at initial time $t_0 = 0$, its lowest speed $v = 1$, and the final time $t_1 = 1$ so that the disk C_0 is described by the equation:

$$C_0 = \{\mathbf{u} \in \Re^2 : u_1^2 + u_2^2 - 2u_1 - 2u_2 + 1 = 0\} \ , \tag{67}$$

with center in $\boldsymbol{l} = (1, 1)$ and unitary radius.

The first and second stage costs $\mathbf{c}$ and $\mathbf{q}$ are given by:

$$\mathbf{c} \ = \ [0.1, 0.5, \mathbf{0}, 0, 0]^T \ , \tag{68}$$

$$\mathbf{q} \quad = \quad [\mathbf{0.5}, \mathbf{0}, \mathbf{0}]^T \; ; \tag{69}$$

this assumption implies that the cost of choosing the radius of the first stage circle C is five times more expensive with respect to choose its center $\tilde{\mathbf{u}}$; because of the low value of $\tilde{c}$, we expect that the center $\tilde{\mathbf{u}}$ of the circle is not necessarily closed to the origin $\mathbf{0}$. Moreover, we suppose the cost of changing, at each scenario, the radius of the second stage circle, is the same of the first stage circle. For a detailed sensitivity analysis approach according to different values of the costs $\mathbf{c}$ and $\mathbf{q}$ see Zhu, Zhang and Patel (2008), [16].

The purpose of the first test is to compare, on the same set of ellipsoid scenarios, the total cost and solutions obtained by solving our model with respect to that proposed by Ariyawansa and Zhu (2006) [2].

Table 1 refers to the particular case of five ellipsoids E_k, $k = 1, \ldots, 5$ randomly generated according to the procedure described in the previous section.

k	u_1^0	u_2^0	φ	s_{u_1}	s_{u_2}
1	2.1332	-0.7902	1.2972	1.9214	0.6592
2	2.9051	-0.5123	1.5647	0.7656	1.1444
3	1.9848	-0.2146	0.6954	2.0558	0.8161
4	2.0417	-0.2325	1.5109	2.3710	1.3641
5	3.4630	0.5189	1.3645	1.6104	0.3094

Table 1: Centre (u_1^0, u_2^0), angle φ between the u_1-axis and the main axis of the conic and semiaxes s_{u_1} and s_{u_2} of five ellipsoids E_k, $k = 1, \ldots, 5$ randomly generated according to the procedure described in section 5.

Table 2 and Figure 1 refer to the solution obtained by the model proposed by Ariyawansa and Zhu (2006) [2] in the case of the five scenarios $k = 1, \ldots, 5$ reported in Table 1. We note that the first stage disk C coincides with the second stage one C^*.

K	$\tilde{u}_1$	$\tilde{u}_2$	d_1	d_2	γ	τ	obj. value
5	2.26	-0.07	2.26	7.04	-1.91	2.65	3.75
500	2.74	0.17	2.75	13.51	-5.97	4.72	8

Table 2: First stage decision solutions and total cost obtained by the model proposed by Ariyawansa and Zhu (2006) [2] in the case of 5 and 500 scenarios.

We have solved on the same set of 5 scenarios the model with the proposed modified Stage 2m; first line of Table 3 shows the first stage decision variables and objective function. The second-stage decision variables $\tilde{\gamma}_k$, $k = 1, \ldots, 5$, related to the radius of the second-stage disks C_k^* are: $\tilde{\gamma}_1 = -3.51$, $\tilde{\gamma}_2 = \tilde{\gamma}_3 = -2.19$, $\tilde{\gamma}_4 = -3.29$ and $\tilde{\gamma}_5 = -5.53$. Looking at the results we can deduce that each second-stage disk C_k^*, $k = 1, \ldots, 5$ contains the disks C_0, C and the ellipse E_k of the corresponding scenario k. We can also deduce that the use of a second stage

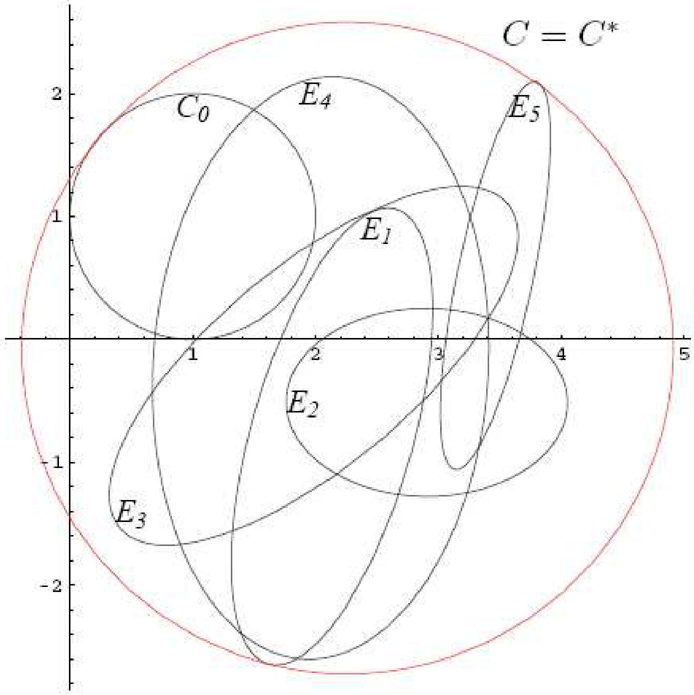


Figure 1: Solution of the model proposed by Ariyawansa and Zhu (2006) [2] in the case of five scenarios $k = 1, \ldots, 5$.

action as a recourse on the first stage decision, for each scenario, allows a saving of about 8% of the total costs. The saving drastically increases by considering an higher number of scenarios as shown in Table 2 in the case of $K = 500$, where the total cost becomes 8 instead of 4.20 of the corresponding case reported in Table 3.

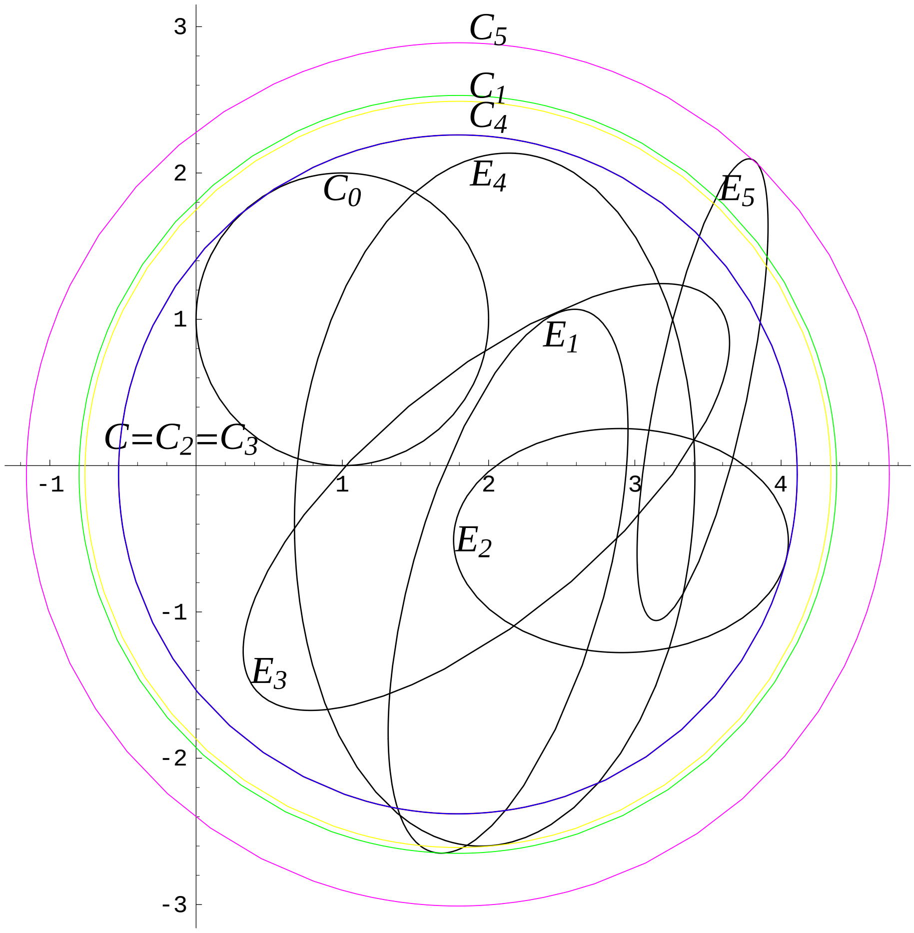


Figure 2: Solution of our model in the case of five scenarios $k = 1, \ldots, 5$.

The next purpose of our computational tests is to compare the solutions of SSOCP and SSDP models. The optimal values of the objective function and the decision variables are the same for both models. Since this SSDP model requires a lot of memory, we were able to solve the case only with 500 scenarios. Instead, with the other model we reached a larger number of scenarios (20250).

To validate the SSOCP model, at first we analyze the sensitivity of solutions to different number of scenarios and we report the relative results in Table 3. We deduce that the model gives an *in-sample stability*, i.e. whichever number of scenarios we consider, the optimal objective values are approximately the same (for a definition of in-sample stability see Kaut and Wallace, (2007) [9]). In particular, Figure 3 shows the convergence of the optimal profit value as the number of scenarios increases. The execution time for the largest case of 20250 scenarios is of 13.359 seconds. It is composed of 25 blocks of equations, 303769 single equations, 19 blocks of variables and 263269 single variables, and 44 iterations.

The values in Table 3 represent the *in-sample* costs. To estimate the impact of a richer scenario tree (see Dupačová et al. (2000), [6]), we compare the *out-of-sample* costs (see again

K	$\tilde{u}_1$	$\tilde{u}_2$	d_1	d_2	γ	τ	obj. value
5	1.79	-0.06	1.79	5.38	-2.19	2.32	3.45
10	2.26	0.53	2.32	5.51	-0.11	2.35	3.69
50	2.37	0.36	2.40	6.31	-0.56	2.51	4.18
100	2.32	0.42	2.36	5.98	-0.41	2.45	4.40
200	2.18	0.38	2.22	5.44	-0.53	2.33	4.06
300	2.28	0.39	2.31	5.85	-0.50	2.42	4.16
400	2.24	0.31	2.26	5.86	-0.74	2.42	4.25
500	2.25	0.38	2.28	5.73	-0.54	2.39	4.20
1000	2.23	0.37	2.26	5.67	-0.55	2.38	4.10
1480	2.23	0.36	2.26	5.70	-0.59	2.39	4.13
2015	2.24	0.36	2.27	5.74	-0.60	2.40	4.16
3010	2.23	0.37	2.26	5.66	-0.55	2.38	4.11
4160	2.24	0.37	2.27	5.72	-0.57	2.39	4.15
5240	2.24	0.38	2.27	5.71	-0.54	2.39	4.16
6015	2.23	0.36	2.26	5.70	-0.59	2.39	4.17
7440	2.23	0.37	2.26	5.67	-0.57	2.38	4.14
8015	2.23	0.36	2.26	5.70	-0.60	2.39	4.15
9051	2.23	0.37	2.26	5.67	-0.55	2.38	4.14
10040	2.23	0.36	2.26	5.71	-0.59	2.39	4.16
11268	2.22	0.36	2.25	5.67	-0.59	2.38	4.15
20250	2.23	0.36	2.26	5.68	-0.58	2.38	4.15

Table 3: First stage decision solutions and optimal profit value for increasing K.

Kaut and Wallace, (2007), [9]). For this purpose, we claim the 7440 scenario tree to represent real world model description, and use it as a *benchmark*. We report in Table 4 some of the results of the out-of-sample analysis relying on the benchmark tree.

To check the importance of modelling the randomness of the parameters, we compare the optimal solutions and objective value of the stochastic model with those obtained from the corresponding *deterministic* model, where we consider an unique scenario represented by an ellipse with the center $(u_1^{mean}, u_2^{mean}) = (2.8289, 0.010142)$, the angle $\varphi^{mean} = 0.79322$, the semiaxis $s_{u_1}^{mean} = 1.7814$, $s_{u_2}^{mean} = 1.0371$ given respectively by the mean of the centers, of the angles and of the semiaxes of the ellipses E_k, $k = 1, \dots, 7440$; its parametric equation is given by

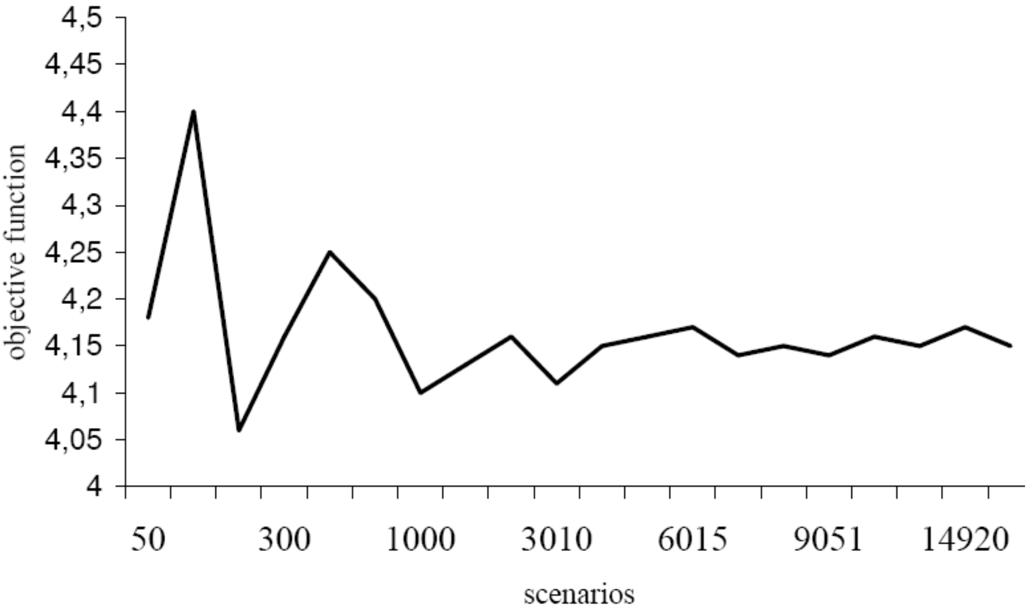


Figure 3: Convergence of the optimal function value for an increasing number of ellipsoid scenarios.

K in 7440	*obj. value*
50 in 7440	4.19
100 in 7440	4.17
200 in 7440	4.15
300 in 7440	4.15
400 in 7440	4.15
1000 in 7440	4.14

Table 4: Out-of-sample objective value for the cases of $K = 50, 100, 200, 300, 400, 1000$ scenarios, with respect to the benchmark tree composed by 7440 branches.

$$\begin{cases} u_1 = 1.7814 \cos(0.79322) \cos\vartheta - 0.7771 \sin(0.79322) \sin\vartheta + 2.8289 \ , & \vartheta \in [0, 2\pi] \ , \\ u_2 = 1.7814 \sin(0.79322) \cos\vartheta + 0.7771 \cos(0.79322) \sin\vartheta + 0.010142 \ , & \vartheta \in [0, 2\pi] \ . \end{cases} \tag{70}$$

In literature, this kind of problem is called *Expected value problem* or *Mean value problem*, (see Birge and Louveaux, (1997) [5] and Kall and Wallace (1994) [8]).

Solutions to the deterministic model are reported in Table 5 and shown in Figure 4.

Because in a deterministic problem the future is completely known, the first stage disk C coincides with the second stage one C_1^* ($\gamma = \tilde{\gamma} = 0.34$), and consequently the total cost is much smaller than in the stochastic case. We have to remember that this is an *in-sample*

K	$\tilde{u}_1$	$\tilde{u}_2$	d_1	d_2	γ	τ	obj. value
1	2.12	0.71	2.24	4.66	0.34	2.16	2.56

Table 5: First stage decision solutions and optimal profit value in the case of *Expected value problem* with one scenario given by eq. (70).

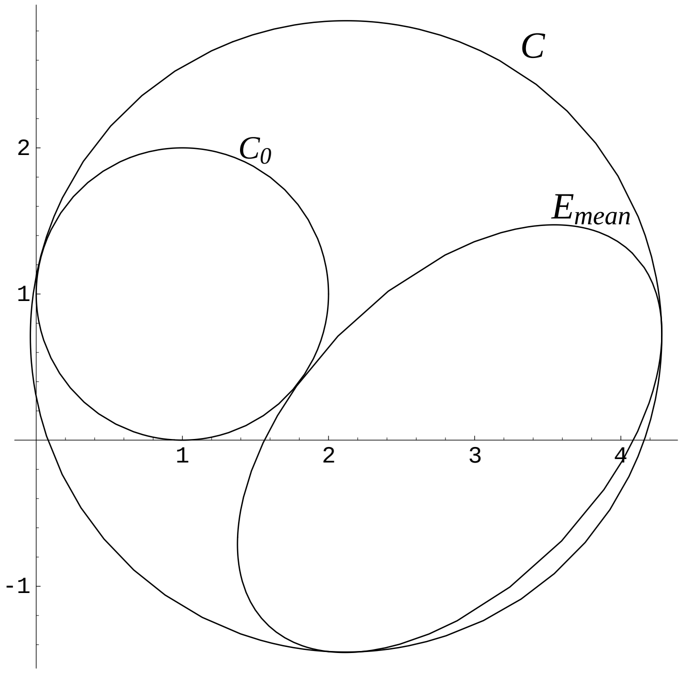


Figure 4: Solution in the case of *Expected value problem* with one scenario described by the ellipse given by eq. (70).

objective value (using the terminology from Kaut and Wallace, (2007) [9]) and the true cost of the solution—or the *out-of-sample* objective value—is likely to be higher. To see how much we can solve the stochastic model with 7440 scenarios and the first-stage variables fixed to the deterministic solution. The result is a total cost of 4.36, much higher than the predicted (in-sample) cost of 2.56. We see that the resulting total cost is higher than the optimal solution for the benchmark tree with 7440 branches. The difference is known as the *Value of stochastic solution* (VSS), (see e.g. Birge and Louveaux, (1997) [5]). In our case, it is:

$$\begin{aligned} VSS &= \textit{obj. val. (det. sol. on benchmark tree)} - \textit{obj. val.(opt. sol. of benchmark tree)} \\ &= 4.36 - 4.14 = 0.22 \ . \end{aligned}$$

This shows that one can save about 5% of the cost by using the stochastic model, compared to the deterministic one.

Another measure of the role of the randomness of the parameters in the model is given by the *Expected value of perfect information* (EVPI) (see again e.g. Birge and Louveaux,

(1997) [5]). EVPI is defined as the difference between the optimal objective value of the stochastic model with 7440 scenarios, also called *here-and-now* solution, and the expected value of the *wait-and-see* solution (WS), calculated by finding the optimal solution for each possible realization of the random variables, as follows:

$$\begin{aligned} EVPI &= \textit{obj. val. (opt. sol. of benchmark tree)} - \textit{obj. val.(WS)} \\ &= 4.14 - 3.12 = 1.02\ . \end{aligned}$$

This means that we should be ready to pay 1.02 in return for complete information before, about the direction and velocity of the destination node D. The large value obtained for EVPI means that the randomness plays an important role in the problem.

References

[1] F. Alizadeh, D. Goldfarb, Second-order cone programming, *Math. Program.*, Ser. B, 95, 3–51 (2003).

[2] K.A. Ariyawansa, Y. Zhu, Stochastic semidefinite programming a new paradigm for stochastic optimization, *Online*, Springer (2006).

[3] K.A. Ariyawansa, Y. Zhu, A preliminary set of apllications leading to stochastic semidefinite programs and chance-constrained semidefinite programs, *Tecnical report* (2006).

[4] S.J. Benson, Y. Ye and X. Zhang, Solving large-scale sparse semidefinite programs for combinatorial optimization, *SIAM Journal of Optimization*, 10(2), 443–461,(2000).

[5] J. R. Birge, F. Louveaux, *Introduction to Stochastic Programming*, Springer Series in Operations Research (1997).

[6] J. Dupačová, G. Consigli, S.W. Wallace, Scenarios for multistage stochastic programs, Annals of Operations Research, 100(1-4), 25–53, (2000).

[7] V.A. Ilyin E.G. Poznyak *Analytic Geometry*, Mir Publishers Moscow (1980).

[8] P. Kall, S.W. Wallace, *Stochastic Programming*, John Wiley Sons (1994).

[9] M. Kaut, S., Wallace, *Evaluation of Scenario-generation Methods for Stochastic Programming*, Pacific Journal of Optimization, 3(2), pp. 257–271 (2007).

[10] Y.B. Ko and N.H. Vaidya, Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, Report, Wireless Network, 6, 307–321, (2000).

[11] Yu. Nesterov, A. Nemirovsky, *Interior-point polynomial methods in convex programming*, volume 13 of Studies in Applied Mathematics, SIAM, Philadelphia, PA (1994).

[12] A. Prékopa, *Stochastic Programming*, Kluwer Academic Publishers (1995).

[13] P.Sun, R.M. Freund, Computation of minimum-costcovering elliposoids, *Oper. Res.*, 52(5), 690-706 (2004).

[14] L. Vandenberghe, S. Boyd, Semidefinite programming, *SIAM Rev.*, 38, 49-95 (1996).

[15] N. Vyas, Mobility pattern based routing algorithm for mobile ad hoc wireless networks, *Master Thesis*, CSE Dept, Florida Atlantic University, August, (2000).

[16] Y. Zhu, J. Zhang and K. Patel, Location-Aided Routing with Uncertainty in Mobile Ad-hoc Networks: A Stochastic Semidefinite Programming Approach. ASU working paper, (2008). Available online at: http://www.public.asu.edu/ yzhu30/resources/SLAR-01022008.pdf, submitted to InfoCom 2009, Rio de Janeiro, Brazil, April 19-25, 2009.

Redazione

Dipartimento di Matematica, Statistica, Informatica ed Applicazioni
Università degli Studi di Bergamo
Via dei Caniana, 2
24127   Bergamo
Tel. 0039-035-2052536
Fax 0039-035-2052549

La Redazione ottempera agli obblighi previsti dall'art. 1 del D.L.L. 31.8.1945, n. 660 e successive modifiche

Stampato nel 2008
presso la Cooperativa
Studium Bergomense a r.l.
di Bergamo