

A NUMERICAL IMPLEMENTATION OF AN INTERIOR POINT METHOD FOR SEMIDEFINITE PROGRAMMING

Djamel Benterki

Département de Mathématiques
Faculté des Sciences
Université Ferhat Abbas
Sétif – Algérie

Jean-Pierre Crouzeix *

LIMOS – Université Blaise Pascal
Aubière Cedex – France
crouzeix@math.univ-bpclermont.fr

Bachir Merikhi

Département de Mathématiques
Faculté des Sciences
Université Ferhat Abbas
Sétif – Algérie

* *Corresponding author*/autor para quem as correspondências devem ser encaminhadas

Recebido em 02/2002, aceito em 11/2002 após 1 revisão

Abstract

This paper is concerned with an algorithm proposed by Alizadeh for linear semidefinite programming. The proof of convergence given by Alizadeh relies on a wrong inequality, we correct the proof. At each step, the algorithm uses a line search. To be efficient, such a line search needs the value of the derivative, we provide this value. Finally, a few numerical examples are treated.

Keywords: semidefinite programming; interior point methods.

Resumo

Este artigo considera um algoritmo proposto por Alizadeh para programação semidefinida linear. A prova de convergência apresentada por Alizadeh baseia-se numa inequação errada, corrigimos a demonstração. Em cada passo, o algoritmo utiliza uma busca linear. Para ser eficiente, esta busca linear precisa do valor da derivada, apresentamos este valor. Finalmente, alguns exemplos numéricos são tratados.

Palavras-chave: programação semidefinida, métodos de pontos interiores.

1. Introduction

The object of this paper is to present an implementation of an interior point method. Alizadeh (1995) has proposed an attractive extension of an interior point algorithm by Ye (1991) for linear programming to linear semidefinite programming. Actually, the proof of convergence given by Alizadeh relies on a wrong inequality, we provide a counterexample. We amend the proof to make it correct. At step k , the algorithm makes use of a line search, Goldstein-Price's methods are efficient to perform such a line search, but they need the knowledge of the derivative. We show how to obtain it, allowing an efficient implementation of the algorithm. Finally, we treat some numerical examples for solving the so-called linear semidefinite program:

$$\min [\langle C, X \rangle : X \in K, \langle A_i, X \rangle = b_i \text{ for } i = 1, \dots, m], \quad \text{SDP}$$

where $b \in R^m$, C and $A_i, i = 1, \dots, m$, belong to E , the linear space of $n \times n$ symmetric matrices. The inner product on E of two matrices A and B is the trace of their product, i.e., $\langle A, B \rangle = \text{tr}(AB) = \sum_{ij} A_{ij}B_{ij}$. Finally, K denotes the cone of positive semidefinite matrices of E and the interior of K denoted by $\text{int}(K)$ is the cone of positive definite matrices of E .

Max-cut problems in a graph, graph bisection problems, the problem of finding a largest clique in a graph, min-max eigenvalue problems can be formulated as linear semidefinite programs, see for instance Helmberg *et al.* (1996).

Linear semidefinite programming presents a great similarity with linear programming. The objective function is linear, the constraint functions are affine, the difference consists of the positivity cone. Testing the positive semidefiniteness of a $n \times n$ matrix X requires to check $\langle Xh, h \rangle \geq 0$ for all h with norm 1, an infinity of linear constraints, while testing that a vector belongs to the nonnegative orthant in an euclidean space involves, by definition, a finite number of linear constraints. The duality scheme in linear semidefinite programming is about the same than the one in linear programming; there are also complementarity slackness conditions. It turns out that interior point methods used in linear programming can be easily extended to linear semidefinite programming.

Next, we precise the notation used in the paper. We have already defined the sets $E, K, \text{int}(K)$ and the scalar product \langle, \rangle in E . The identity matrix of E is denoted by I . Given $A \in E$ its Frobenius norm is

$$\|A\| = \sqrt{\langle A, A \rangle} = \sqrt{\sum_{i=1}^n |\lambda_i|^2},$$

where $\lambda_i, i = 1, 2, \dots, n$ are the eigenvalues of A . Besides this norm, we will also use the spectral norm on $E: \|A\|_s = \max_i (|\lambda_i|)$. Given a rectangular $m \times n$ matrix B , $\text{vec}B$ is the vector of R^{mn} defined by $(\text{vec}B)_{i+(j-1)m} = B_{ij}$ for $i = 1, 2, \dots, m, j = 1, 2, \dots, n$. For instance the set of relations $\langle A_i, X \rangle = b_i$ for $i = 1, \dots, m$ may be written as $A \text{vec} X = b$ where A is a $m \times n^2$ matrix, that is row i of A is $\text{vec}^t A_i$. $F \otimes G$ is the Kronecker product of matrices:

if F is a $m \times n$ matrix and G a $p \times q$ matrix then $F \otimes G$ is the $m \times n$ block matrix whose the (i,j) block is the $p \times q$ matrix $F_{ij}G$.

Let $u, v \in \mathfrak{R}^r$, we denote by $u^t v$ their inner product and by $\|u\|_2, \|u\|_\infty$ the euclidean and the max-norm of u , respectively. Given $u \in \mathfrak{R}^r$, $Diag(u)$ is the $r \times r$ diagonal matrix with the components of u as the diagonal entries. Conversely, given X an $r \times r$ matrix (not necessarily diagonal), $diag(X)$ is the vector in \mathfrak{R}^r obtained from the diagonal elements of X .

2. Duality in SDP

Let us consider the problems:

$$m_p = \min[\langle C, X \rangle : X \in K, \langle A_i, X \rangle = b_i \text{ for } i = 1, \dots, m] \quad (\text{SDP})$$

and

$$m_d = \max_{y \in \mathfrak{R}^m} \left[b^t y : \left(C - \sum_{i=1}^m y_i A_i \right) \in K \right]. \quad (\text{DSDP})$$

X is said to be a *strictly feasible solution* of (SDP) if $X \in \text{int}(K)$ and $\langle A_i, X \rangle = b_i$ for $i = 1, \dots, m$. $y \in \mathfrak{R}^m$ is said to be a *strictly feasible solution* of (DSDP) if $\left(C - \sum_{i=1}^m y_i A_i \right) \in \text{int}(K)$. A weak duality result says that $m_d \leq m_p$. A strong duality result says that if both (SDP) and (DSDP) have strictly feasible solutions, then $m_d = m_p$ and both problems have optimal solutions. Furthermore, for such optimal solutions X and y , the following complementarity slackness condition holds

$$\left\langle X, C - \sum_{i=1}^m y_i A_i \right\rangle = 0.$$

More information on semidefinite programming can be found in Alizadeh (1995), Helmsberg *et al.* (1996) and Vanderbeghe & Boyd (1996).

3. A potential reduction method for (SDP)

The method proposed by Alizadeh (1995) is a direct extension of a method given by Ye (1991) for linear programming. It is based on the reduction of a potential function and a projective transform. Let us consider the (SDP) program and his dual (DSDP). Given $z < m_p$ and a fixed integer $r > 0$, the algorithm uses the two following functions:

$$\phi(X, z) = (n+r) \log(\langle C, X \rangle - z) - \log \det(X)$$

and

$$\psi(X, y) = (n+r) \log \left\langle X, C - \sum_{i=1}^m y_i A_i \right\rangle - \log \det \left(X \left(C - \sum_{i=1}^m y_i A_i \right) \right).$$

It is clear that ϕ is well defined for all X strictly feasible solution of (SDP), while ψ is well defined for all X positive definite and y strictly feasible solution of (DSDP).

The function ϕ (the function ψ) is called the primal (the primal-dual) potential function. The convergence of the algorithm is obtained from the decrease of the sequence $\{\psi(X_k, y_k)\}$.

This method needs the knowledge of initial strictly feasible solutions X_0 and y_0 .

3.1 The projective transform

At step k , the current $X_k \in \text{int}(K)$. The Cholesky factorization of X_k gives a lower triangular matrix L_k with positive diagonal entries such that $L_k L_k^t = X_k$. Then, we consider the projective transform

$$T_k : E \rightarrow E \times \mathfrak{R}^r$$

$$T_k(X) = (\bar{X}, \bar{x})$$

where

$$\bar{X} = \frac{(n+r)L_k^{-1}XL_k^{-t}}{r + \langle X_k^{-1}, X \rangle}, \quad \bar{x} = \frac{(n+r)}{r + \langle X_k^{-1}, X \rangle} e_r.$$

Here e_r is the vector of ones in \mathfrak{R}^r . Notice that $X = T_k^{-1}(\bar{X}, \bar{x})$. Via this projective transform, (SDP) is equivalent to the problem

$$0 = \min \left[\begin{array}{l} \bar{A} \text{vec } \bar{X} + \bar{A} \bar{x} = 0, \\ \langle \bar{C}, \bar{X} \rangle + \bar{c}^t(z) \bar{x} : \text{tr}(\bar{X}) + e_r^t \bar{x} = n+r, \\ \bar{X} \in K, \bar{x} \geq 0. \end{array} \right] \quad (TSDP)$$

where

- $\bar{C} = L_k^t C L_k \in E$.
- $\bar{c}(z) = -\begin{pmatrix} z \\ r \end{pmatrix} e_r$ is a vector of \mathfrak{R}^r .
- $\bar{A}_i = L_k^t A_i L_k \in E$.
- $\bar{A} = A(L_k \otimes L_k)$ is a $m \times n^2$ matrix.
- $\bar{A} = \begin{pmatrix} -1 \\ r \end{pmatrix} b e_r^t$ is a $m \times r$ matrix.
- $[\bar{A}, \bar{A}]$ is a $m \times (n^2 + r)$ matrix.

Remark: The potential function associated to (TSDP) is

$$\bar{\phi}(\bar{X}, \bar{x}, z) = (n+r) \log(\langle \bar{C}, \bar{X} \rangle + \bar{c}^t(z)\bar{x}) - \log(\det(\bar{X})) - \sum_{j=1}^r \log(\bar{x}_j).$$

It is easily proved (see Alizadeh, 1995) that

$$\phi(X, z) - \phi(X_0, z) = \bar{\phi}(\bar{X}, \bar{x}, z) - \bar{\phi}(I, e_r, z).$$

4. The algorithm iteration

In a similar way than in the Karmarkar's method for classical linear programming, we relax the problem (TSDP) in the problem

$$0 = \min \left[\begin{array}{l} \bar{A} \text{vec } \bar{X} + \bar{A} \bar{x} = 0, \\ \langle \bar{C}, \bar{X} \rangle + \bar{c}^t(z)\bar{x} : \text{tr}(\bar{X}) + e_r^t \bar{x} = n+r, \\ \|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2 \leq \beta^2. \end{array} \right] \quad (R)$$

It is easily seen that, for $\beta \in]0,1[$, the constraint $\|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2 \leq \beta^2$ implies that $\bar{X} \in \text{int}(K)$ and $\bar{x} > 0$. Using a matricial formulation, we have

$$0 = \min \left[\begin{array}{l} \langle \bar{C}, \bar{X} \rangle + \bar{c}^t(z)\bar{x} : \bar{\bar{A}} \begin{pmatrix} \text{vec } \bar{X} \\ \bar{x} \end{pmatrix} = \begin{pmatrix} 0 \\ n+r \end{pmatrix} \\ \|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2 \leq \beta^2 \end{array} \right] \quad (R)$$

where

$$\bar{\bar{A}} = \begin{pmatrix} \bar{A} & \bar{A} \\ (\text{vec } I)^t & e_r^t \end{pmatrix}$$

is a $(m+1) \times (n^2+r)$ matrix.

It results, from a classical optimality condition, that the optimal solution $(\bar{X}^*, \bar{x}^*)^t$ of (R) is given by

$$\begin{pmatrix} \text{vec } \bar{X}^* \\ \bar{x}^* \end{pmatrix} = \begin{pmatrix} \text{vec } I \\ e_r \end{pmatrix} - \beta \frac{p(z_k)}{\|p(z_k)\|} \quad (1)$$

where $p(z_k)$ is the projection of the vector $\begin{pmatrix} \text{vec } \bar{C} \\ \bar{c}(z) \end{pmatrix}$ on the affine subspace corresponding to the equality, i.e.,

$$\begin{aligned}
 p(z_k) &= \left(I - \frac{[(\text{vec } I)^t, e_r^t]^t [(\text{vec } I)^t, e_r^t]}{n+r} \right) \times \\
 &\quad \left(I - [\bar{A}, \bar{A}]^t \left([\bar{A}, \bar{A}] [\bar{A}, \bar{A}]^t \right)^{-1} [\bar{A}, \bar{A}] \right) \begin{pmatrix} \text{vec } \bar{C} \\ \bar{c}(z) \end{pmatrix} \\
 &= \begin{pmatrix} \text{vec } P_n(z_k) \\ p_r(z_k) \end{pmatrix},
 \end{aligned}$$

where $P_n(z_k)$ is an $n \times n$ matrix and $p_r(z_k)$ is a vector in \mathfrak{R}^r .

Let

$$y(z) = \left([\bar{A}, \bar{A}] [\bar{A}, \bar{A}]^t \right)^{-1} [\bar{A}, \bar{A}] \begin{pmatrix} \text{vec } \bar{C} \\ \bar{c}(z) \end{pmatrix}.$$

Then $y(z)$ is a solution for the linear system

$$[\bar{A}, \bar{A}] [\bar{A}, \bar{A}]^t y(z) = [\bar{A}, \bar{A}] \begin{pmatrix} \text{vec } \bar{C} \\ \bar{c}(z) \end{pmatrix}.$$

This system can be efficiently solved thanks to a Cholesky factorization.

Remark: β may stay fixed during the different iterations, but it is better to adjust β at each iteration in order to accelerate the speed of convergence. This can be done thanks to a Goldstein-Price line search. For that, it is necessary to know the derivative of the potential function at 0. This is the object of the following lemma.

Lemma 1. Let $P \in E$ and

$$g(\beta) = (n+r) \log \left\langle X - \beta LPL^t, C - \sum_{i=1}^m y_i A_i \right\rangle - \log \det \left[(X - \beta LPL^t) \left(C - \sum_{i=1}^m y_i A_i \right) \right]$$

Then

$$g'(0) = \frac{-(n+r) \text{tr} \left[(LPL^t) \left(C - \sum_{i=1}^m y_i A_i \right) \right]}{\text{tr} \left[X \left(C - \sum_{i=1}^m y_i A_i \right) \right]} + \text{tr}(P).$$

Proof. We have

$$\begin{aligned}
 g'(\beta) &= \frac{-(n+r) \text{tr} \left[(LPL^t) \left(C - \sum_{i=1}^m y_i A_i \right) \right]}{\text{tr} \left[(X - \beta LPL^t) \left(C - \sum_{i=1}^m y_i A_i \right) \right]} \\
 &\quad - \frac{\left[\det(X - \beta LPL^t) \right]'}{\det(X - \beta LPL^t)}
 \end{aligned}$$

Set

$$H(\beta) = \det(X - \beta LPL^t) = \det(LL^t - \beta LPL^t) = \det(X) \times \det(I - \beta P).$$

P being symmetric, there exist an orthogonal matrix U and a diagonal matrix D such that $P = U^t D U$. Hence,

$$\begin{aligned} H(\beta) &= \det(X) \times \det(U^t U - \beta U^t D U) \\ &= \det(X) \times \det(I - \beta D) \\ &= \det(X) \times \prod_{i=1}^n (1 - \beta d_i). \end{aligned}$$

Then

$$\begin{aligned} H'(\beta) &= -\det(X) \left(\sum_{i=1}^n \left(d_i \prod_{i \neq j} (1 - \beta d_j) \right) \right) \\ H'(0) &= -\det(X) \left(\sum_{i=1}^n d_i \right) = -\det(X) \times \text{tr}(P). \end{aligned}$$

Finally,

$$g'(0) = \frac{-(n+r) \text{tr} \left[(LPL^t) \left(C - \sum_{i=1}^m y_i A_i \right) \right]}{\text{tr} \left[X \left(C - \sum_{i=1}^m y_i A_i \right) \right]} + \text{tr}(P).$$

■

4.1 Convergence of the algorithm

The proof of convergence given by Alizadeh for his algorithm is based on the following inequality.

$$\bar{\phi}(\bar{X}, \bar{x}, z) - \bar{\phi}(I, e_r, z) \leq -(n+r)\beta \left(\frac{\|p(z)\|_2}{\text{trace}(C) + \bar{c}^t(z)e_r} \right) + \frac{\beta^2}{2(1-\beta)}.$$

Actually, this inequality is false as shown by the following counter-example:

$$\min [x_{11} \quad : \quad x_{11} + x_{22} = 1, \quad x_{11} \geq 0, \quad x_{22} \geq 0].$$

Take

$$X_0 = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix} \quad \text{and} \quad y_0 = -0.5.$$

In fact, we have:

Lemma 2. Let $\bar{X} \in K$ and $\bar{x} \in \mathfrak{R}^r, \bar{x} \geq 0$. Then

$$\bar{\phi}(\bar{X}, \bar{x}, z) - \bar{\phi}(I, e_r, z) \leq -(n+r)\beta \left(\frac{\|p(z)\|_2}{\text{trace}(C) + \bar{c}^t(z)e_r} \right) + \frac{\beta^2}{2(1-\beta\sqrt{2})}.$$

Proof. Reporting to Alizadeh (1995), the following inequality holds

$$\bar{\phi}(\bar{X}, \bar{x}, z) - \bar{\phi}(I, e_r, z) \leq (n+r) \log \left(\frac{\langle \bar{C}, \bar{X} \rangle + \bar{c}^t(z) \bar{x}}{\text{trace}(\bar{C}) + \bar{c}^t(z) e_r} \right) + \frac{\|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2}{2(1 - \|\bar{X} - I\|_s - \|\bar{x} - e_r\|_\infty)}.$$

But

$$\|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2 \leq \beta^2 \leq 1$$

and

$$\begin{aligned} \|\bar{X} - I\|_s + \|\bar{x} - e_r\|_\infty &\leq \|\bar{X} - I\| + \|\bar{x} - e_r\|_2 \\ &\leq \sqrt{2(\|\bar{X} - I\|^2 + \|\bar{x} - e_r\|_2^2)} \\ &\leq \beta\sqrt{2} \end{aligned}$$

Hence

$$1 - (\|\bar{X} - I\| + \|\bar{x} - e_r\|_2) \geq 1 - \beta\sqrt{2}.$$

Finally (1) gives

$$\bar{\phi}(\bar{X}, \bar{x}, z) - \bar{\phi}(I, e_r, z) \leq -(n+r)\beta \left(\frac{\|p(z)\|_2}{\text{trace}(\bar{C}) + \bar{c}^t(z) e_r} \right) + \frac{\beta^2}{2(1 - \beta\sqrt{2})}.$$

If we replace in the proof of Alizadeh $\delta(\beta) = \alpha\beta - \frac{\beta^2}{2(1-\beta)}$ by $\delta(\beta) = \alpha\beta - \frac{\beta^2}{2(1-\beta\sqrt{2})}$,

then his proof becomes valid. In particular, we have

$$\psi(X(z), y_k) - \psi(X_k, y_k) \leq -\delta \quad \text{when} \quad \|p(z)\|_2 \geq \alpha \frac{\Delta_k}{n+r}$$

and

$$\psi(X_k, y(z)) - \psi(X_k, y_k) \leq -\delta \quad \text{when} \quad \|p(z)\|_2 < \alpha \frac{\Delta_k}{n+r}$$

where

$$\Delta_k = \langle C, X_k \rangle - b^t y_k.$$

Furthermore, Alizadeh proves the polynomial convergence:

$$(\langle C, X_k \rangle - b^t y_k) < 2^L \varepsilon$$

after a number of iteration of order $O(\sqrt{n} |\log \varepsilon|)$ for $\varepsilon > 0$.

■

5. Implementation of the Alizadeh algorithm

In the algorithm, $\alpha \in [0,1]$ and $\varepsilon > 0$ are fixed.

The Alizadeh algorithm

- a) Initialization: Start with $X_0, y_0, \alpha \in [0,1], z_0 = b^t y_0$ and $k=0$,
- b) **If** $(\langle C, X_k \rangle - b^t y_k) \leq \varepsilon$ **STOP:** (X_k, y_k) is an ε -approximate solution
If not:
 - Compute $p(z_k)$.
 - **If** $\|p(z_k)\|_2 \geq \alpha \frac{\langle C, X_k \rangle - b^t y_k}{n+r}$ **then**
 - * Find $\beta^* = \text{Argmin}_{0 \leq \beta \leq 1} (\psi(X_k - \beta L_k P_n(z_k) L_k^t, y(z_k)))$.
 - * Take $\begin{pmatrix} \text{vec } \bar{X}^* \\ \bar{x}^* \end{pmatrix} = \begin{pmatrix} \text{vec } I \\ e_r \end{pmatrix} - \beta^* \frac{p(z_k)}{\|p(z_k)\|_2}$.
 - * Take $X_{k+1} = T_k^{-1}(\bar{X}^*, \bar{x}^*)$ and go to c).
 - **If not:**
 - * Find $z^* = \text{Argmin}_{z \leq z_k} (\psi(X_k, y(z)))$.
 - * Take $y_{k+1} = y(z^*), z_{k+1} = b^t y(z^*)$ and go to c).
- c) Do $k=k+1$ and go to back b).

6. Numerical examples

In the treated examples, the initial primal-dual feasible solutions have been given .

6.1 Example 1

$$\min[\langle C, X \rangle : X \in K, \langle A_i, X \rangle = b_i \text{ for } i=1,2] \quad (\text{SDP})$$

where:

$$C = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 2 & -2 \\ 1 & -2 & 2 \end{pmatrix}, A_1 = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, A_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

and $b = (0,1)^t$.

After 4 iterations we obtain the optimal dual solution $y^* = (1, 0)^t$ and the optimal primal solution

$$X^* = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

6.2 Example 2

In this example $n=2m$.

$$\min[\langle C, X \rangle : X \in K, \langle A_k, X \rangle = b_k \text{ for } k=1, \dots, m] \quad (\text{SDP})$$

where $C = -I$. For $k=1, \dots, m$, $b_k = 2$ and the $n \times n$ matrices A_k are defined by

$$A_k[i, j] = \begin{cases} 1 & \text{if } i = j = k & \text{or } i = j = k + m \\ \lambda^2 & \text{if } i = j = k + 1 & \text{or } i = j = k + m + 1 \\ -\lambda & \text{if } i = k, j = k + 1 & \text{or } i = k + m, j = k + m + 1 \\ -\lambda & \text{if } i = k + 1, j = k & \text{or } i = k + m + 1, j = k + m \\ 0 & \text{otherwise} \end{cases}$$

with $k + m + 1 \leq n$.

The following array gives the number of iterations needed to obtain the primal and the dual solutions for different sizes.

(m,n)	Optimal Value	Nbr. of iterations
(1,2)	-1.9955	6
(2,4)	-3.9214	19
(3,6)	-5.5146	30
(4,8)	-7.211	38
(5,10)	-9.1185	30
(6,12)	-11.818	36
(7,14)	-12.516	42

7. Conclusion

The present paper corrects a small point in the proof of the convergence given by Alizadeh (1995) and gives the value of the derivative of the potential function at 0 needed to perform an efficient line search.

References

- (1) Alizadeh, F. (1995). Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, **5**, 13-51.
- (2) Helmberg, C.; Rendl, F.; Vanderbei, R.J. & Wolkowicz, H. (1996). An interior point method for semidefinite programming. *SIAM Journal on Optimization*, **6**, 342-361.
- (3) Jansen, B.; Roos, C. & Terlaky, T. (1997). A family of polynomial affine scaling algorithms for semidefinite linear complementarity problems. *SIAM Journal on Optimization*, **7**, 126-140.
- (4) Keraghel, A. (1989). Etude adaptative et comparative des principales variantes dans l'algorithme de Karmarkar. Dissertation thesis, Université Joseph Fourier, Grenoble, France.
- (5) Kojima, M.; Shindoh, S. & Hara, S. (1997). Interior point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, **7**, 86-125.
- (6) Marcus, M. & Ming, H. (1964). *A survey of matrix theory and matrix inequalities*. University of California, Santa Barbara, Allyn and Bacon, ING, Boston.
- (7) Monteiro, R.D.C. (1997). Primal-dual path following algorithms for semidefinite programming. *SIAM Journal on Optimization*, **7**, 663-678.
- (8) Overton, M. & Wolkowicz, H. (1997). Semidefinite programming. *Mathematical Programming, Series B*, **77**, 105-109.
- (9) Rama, M.V.; Tuncel, L. & Wolkowicz, H. (1997). Strong duality for semidefinite programming. *SIAM Journal on Optimization*, **7**, 641-662.
- (10) Vanderbeghe, L. & Boyd, S. (1996). Positive definite programming. *SIAM Review*, **38**, 49-95.
- (11) Wright, S.J. (1997). *Primal-dual interior point methods*. SIAM, Philadelphia, PA.
- (12) Ye, Y. (1991). An $O(n^3L)$ - potential reduction algorithm for linear programming. *Mathematical Programming*, **50**, 239-258.