



# A feasible direction interior point algorithm for nonlinear semidefinite programming

Miguel Aroztegui, Jose Herskovits, Jean Rodolphe Roche

## ► To cite this version:

Miguel Aroztegui, Jose Herskovits, Jean Rodolphe Roche. A feasible direction interior point algorithm for nonlinear semidefinite programming. 2012. hal-00758803

**HAL Id: hal-00758803**

**<https://hal.science/hal-00758803>**

Preprint submitted on 29 Nov 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A feasible direction interior point algorithm for nonlinear semidefinite programming

Miguel Aroztegui · José Herskovits · Jean Rodolphe Roche

the date of receipt and acceptance should be inserted later

**Abstract** We present a new algorithm for nonlinear semidefinite programming. It is based on the iterative solution, in the primal and dual variables, of Karush-Kuhn-Tucker first order optimality conditions. This method generates a decreasing feasible sequence. At each iteration, two linear systems with the same coefficient matrix are solved and an inexact line search is then performed. A proof of global convergence is given in the convex case. Some numerical tests involving nonlinear programming problems as well linear and nonlinear matrix inequalities are described. We also solve structural topology optimization problems employing a mathematical model based on semidefinite programming. The results suggest efficiency and high robustness of the proposed method.

**Keywords** nonlinear, semidefinite programming, feasible directions, interior-point methods.

## 1 Introduction

This paper proposes a new technique to solve the following non-linear semidefinite programming problem (SDP),

$$\begin{array}{ll} \min_{x \in} & f(x) \\ \text{s.t.} & G(x) \preceq 0 \end{array} \quad (1)$$

---

Miguel Aroztegui

Departament of Informatics, Federal University of Paraíba, João Pessoa - Paraíba, CEP: 58.051-900, CT, Cidade Universitária, João Pessoa, Paraíba, Brazil.  
E-mail: marozteg@di.ufpb.br

José Herskovits

Mechanical Engineering Program - COPPE - Federal University of Rio de Janeiro, PO Box 68503, CEP 21945-970, CT, Cidade Universitária, Ilha do Fundão, Rio de Janeiro, Brazil.  
E-mail: jose@optimize.ufrj.br

Jean R. Roche

I.E.C.N., Université de Lorraine, CNRS, INRIA B.P. 70239, 54506 Vandoeuvre lès Nancy, France.  
E-mail: jean-rodolphe.roche@univ-lorraine.fr

where  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  and  $G : \mathbb{R}^n \rightarrow \mathbb{S}^m$  are smooth functions. We denote  $\mathbb{S}^m$  the set of real symmetric matrices of size  $m \times m$ . The constraint  $G(x) \preceq 0$  is a matrix inequality which means that  $G(x)$  is negative semidefinite. We call  $\Omega = \{x \in \mathbb{R}^n; G(x) \preceq 0\}$  the set of feasible solutions and  $\text{int}(\Omega)$ , its interior.

Applications in a wide range of areas lead to semidefinite programming problems. For example, combinatorial optimization [3], nonconvex quadratic programming [16], eigenvalue optimization [25], systems control theory [12], matrix completion problems [28], problems in statistics [15] and structural design [2]. In particular SDP was considered to solve free material problems in structural mechanical design, see for example [37], [24], [35], [34] and [38]. Some of this problems are considered in our numerical examples.

When  $f$  is linear and  $G$  depends affinely on the variables, problem (1) is convex and many efficient algorithms were developed. Duality theory and the central path concept were extended from linear programming to semidefinite programming. Nesterov and Nemirovsky [8] and Alizadeh [3] introduce interior point techniques based on path-following and potential-reduction approaches. In Todd, [36], can be found a deep historical overview.

On the other hand, when  $f$  or  $G$  are nonlinear functions, problem (1) is called a Nonlinear Semidefinite Programming problem (NSDP). In general, a NSDP problem is nonconvex. Nowadays, there are several methods for NSDP problems. We mention the extension of a primal predictor-corrector interior point method of Jarre [21], the sequentially linear SDP method of Correa and Ramirez [13], which is a generalization of the method presented by Fares, Noll and Apkarian [14] and the augmented Lagrangian approach of Kočvara and Stingl [33]. Also Kanzow, Nagel, Kato and Fukushima [22] presented a successive linearization method with a trust region-type globalization.

A crucial result for semidefinite programming is the characterization of Karush - Kuhn - Tucker (KKT) optimality conditions presented by Shapiro, [31] and [32].

In this paper, we present an interior point algorithm which extends to semidefinite programming the Feasible Direction Interior Point Algorithm, FDIPA. FDIPA is a general technique for smooth nonlinear inequality and equality constrained optimization [17] [18] [29] [19]. The present algorithm, defines first at each interior point a feasible and descent direction with respect to the semidefinite constraints. Then, makes a line search in that direction to obtain a new interior point with a lower objective. A Newton or quasi - Newton and first order versions of presented algorithm can be obtained. The present method is simple to code, does not require the solution of quadratic programs and it is not a penalty neither a barrier method. It merely requires the solution of two linear systems with the same matrix per iteration.

The paper is organized as follows. In sub-section 1.1 we introduce some notations and basic concepts about symmetric matrices, the symmetric Kronecker product and the Lagrangian of the problem. In sub-section 1.2 we present some definitions and technical results. Following [31] and [32] we recall first order optimality conditions adapted to semi-definite programming. In section 2 we describe the main ideas of the algorithm. We show how the search direction is built to give a feasible and descent direction. At the end of this section we introduce the general statement of the presented algorithm. The global convergence to a critical point of the considered optimization method is proved in section 3. Details of the imple-

mentation of the presented algorithm are shown in section 4. Numerical examples are presented in section 6. The last section is dedicated to conclusions.

### 1.1 Notation

We introduce some notation concerning the matrix spaces. Let  $\mathbb{R}^{m \times n}$  denote the space of  $m \times n$  real matrices. Let  $\mathbb{S}^n$  denote the space of real symmetric matrices. The set of symmetric positive semidefinite and positive definite matrices of size  $m \times m$  are denoted  $\mathbb{S}_+^m$  and  $\mathbb{S}_{++}^m$ , respectively. In an analog way the set of negative semidefinite and negative definite matrix of size  $m$  are denoted  $\mathbb{S}_-^m$  and  $\mathbb{S}_{--}^m$ , respectively. The symbol  $\preceq$  defines the negative semidefinite order, that is,  $A \preceq B$  means that  $A - B$  is negative semidefinite. Similarly, the symbol  $\prec$ ,  $\succ$  and  $\succcurlyeq$  defines the negative definite, positive semidefinite and positive definite order, respectively.

The symmetric part of  $M \in \mathbb{R}^{n \times n}$  is indicated by  $\text{sym}(M)$  and the skew part as  $\text{skw}(M)$ . We denote  $\ker(A)$  as the null space of the matrix  $A \in \mathbb{R}^{m \times n}$ .

The  $(i, j)$ th entry of a matrix  $A \in \mathbb{R}^{m \times n}$  is referred to by  $a_{ij}$ . The transpose of  $A$  is written as  $A^\top$ . Let  $I_n$  denote the identity matrix in  $\mathbb{R}^{n \times n}$ .

Given  $A \in \mathbb{S}^m$ , then  $\bar{m} = \frac{1}{2}m(m+1)$  is the number of upper diagonal components of  $A$ . To introduce the symmetric Kronecker product, we define the two following maps:

$$\begin{aligned} \text{svec} : \mathbb{S}^m &\rightarrow \mathbb{R}^{\bar{m}} \\ \text{svec}(A) &= [a_{11} \ \sqrt{2}a_{12} \ a_{22} \ \sqrt{2}a_{13} \ \sqrt{2}a_{23} \ a_{33} \ \dots \ a_{mm}]^\top \end{aligned}$$

and  $\text{smat} : \mathbb{R}^{\bar{m}} \rightarrow \mathbb{S}^m$  the inverse of  $\text{svec}$ , see [23].

Then, the inner product

$$\langle A, B \rangle = \text{tr}(A^\top B) = \text{svec}(A)^\top \text{svec}(B), \text{ for } A, B \in \mathbb{S}^m.$$

The symmetric Kronecker product of two matrices  $A, B \in \mathbb{R}^{m \times n}$  is represented by  $A \otimes B$  and verifies for any matrix  $C \in \mathbb{S}^n$  the following equality:

$$(A \otimes B) \text{svec}(C) = \text{svec}(\text{sym}(BCA)) \quad (2)$$

It is proved that, [23]:

1.  $(A \otimes B) \text{svec}(C) = (B \otimes C) \text{svec}(A)$
2.  $A \otimes B = B \otimes A$
3.  $(A \otimes B)(C \otimes D) = \frac{1}{2}(AC \otimes BD + AD \otimes BC)$
4.  $(P \otimes P)^{-1} = P^{-1} \otimes P^{-1}$
5. If  $A, B \succcurlyeq 0$  then,  $A \otimes B \succcurlyeq 0$ .
6. If  $A \succcurlyeq 0$  and  $B \preceq 0$  then,  $A \otimes B \preceq 0$ .

(3)

where  $A, B, C \in \mathbb{S}^m$  and  $P$  is a non singular matrix.

The partial derivatives of  $G(x)$  with respect to  $x_p$  is denoted by  $\frac{\partial G}{\partial x_p}(x)$ , with components  $\frac{\partial g_{ij}(x)}{\partial x_p}$ ,  $i, j = 1, m$ .

Then, we can define the following matrix in  $\mathbb{R}^{n \times \overline{m}}$ ,

$$\nabla G(x) = \begin{bmatrix} \text{svec} \left( \frac{\partial G}{\partial x_1}(x) \right)^\top \\ \vdots \\ \text{svec} \left( \frac{\partial G}{\partial x_n}(x) \right)^\top \end{bmatrix}. \quad (4)$$

The derivative of  $G$  in the direction  $d \in \mathbb{R}^n$  at  $x$ , denoted by  $DG(x)d$ , verifies:

$$DG(x)d = \sum_{p=1}^n d_p \frac{\partial G}{\partial x_p}(x) \quad (5)$$

In view of (4), the matrix equation (5) can be expressed in a vector format

$$\text{svec}(DG(x)d) = \nabla G(x)^\top d \quad (6)$$

The Lagrangian function of problem (1) is defined as  $L : \mathbb{R}^n \times \mathbb{S}^m \rightarrow \mathbb{R}^n$  such that

$$L(x, \Lambda) = f(x) + \langle G(x), \Lambda \rangle.$$

Alternatively, the Lagrangian can be written in the form  $L : \mathbb{R}^n \times \mathbb{R}^{\overline{m}} \rightarrow \mathbb{R}^n$  such that

$$L(x, \lambda) = f(x) + \langle \text{svec}(G(x)), \lambda \rangle$$

where  $\lambda = \text{svec}(\Lambda)$ .

Therefore the gradient of the Lagrangian with respect to  $x$  can be written as

$$\nabla_x L(x, \lambda) = \nabla f(x) + \nabla G(x)\lambda, \quad (7)$$

see [32] and [10].

## 1.2 Basic definitions and technical results

The following three definitions are related with first order optimality conditions for semidefinite programming analyzed in [31] and [32].

We call  $\{b_1(x), \dots, b_p(x)\}$  an orthonormal basis of  $\ker G(x)$  and define the matrix  $E_0(x) = [b_1(x) \dots b_p(x)] \in \mathbb{R}^{m \times p}$ .

**Definition 1** A point  $x$  is a regular point of problem (1) if the vectors of the set

$$\left\{ \begin{bmatrix} b_i^\top \frac{\partial G}{\partial x_1}(x) b_j \\ \vdots \\ b_i^\top \frac{\partial G}{\partial x_n}(x) b_j \end{bmatrix} \text{ such that } i \leq j, i, j = 1, \dots, p \right\}$$

are linearly independent.

**Definition 2** A regular point  $x$  is a stationary point of problem (1) if there exist  $\Lambda \in \mathbb{S}^m$  such that the following requirements are verified:

$$\begin{aligned}\nabla_x L(x, \Lambda) &= 0 \\ \Lambda G(x) &= 0 \\ G(x) &\preceq 0\end{aligned}\tag{8}$$

**Definition 3** A Karush-Kuhn-Tucker point of problem (1) is a stationary point and  $\Lambda \succcurlyeq 0$ .

**Definition 4** The vector  $d \in \mathbb{R}^n$  is a feasible direction of  $\Omega$  at  $x \in \Omega$  if there exists  $\tau > 0$  such that  $x + td \in \Omega$  when  $t \in [0, \tau]$ .

**Definition 5** The vector field  $d(x)$  defined on  $\Omega \subseteq \mathbb{R}^n$  is said to be a uniformly feasible direction field of  $\Omega$  if there exists  $\tau > 0$  such that,  $x + td(x) \in \Omega$  for all  $x \in \Omega$  and  $t \in [0, \tau]$ .

When the vector field  $d(x)$  is a uniformly feasible direction field of  $\Omega$ , the segment  $[x, x + \tau d(x)]$  is included in  $\Omega$  for all  $x \in \Omega$ . When  $x$  is in the interior of  $\Omega$ , all directions are feasible.

**Definition 6**  $d \in \mathbb{R}^n$  is a descent direction of a real function  $f$  at  $x \in \mathbb{R}^n$  if there exist some  $\delta > 0$  such that:  
 $f(x + td) < f(x)$  for all  $t \in (0, \delta]$ .

It is possible to prove that if  $f$  is differentiable at  $x$  and  $d^\top \nabla f(x) < 0$  then  $d$  is a descent direction of  $f$  at  $x$ .

Now we introduce some technical result useful to describe and prove the convergence of the studied algorithm.

**Lemma 1**  $A \in \mathbb{S}_-^m$  and  $B \in \mathbb{S}_+^m$ , the following equalities are equivalent

$$\begin{aligned}AB &= 0 \\ \text{tr}(AB) &= 0 \\ \text{sym}(AB) &= 0\end{aligned}\tag{9}$$

*Proof* The proof for  $AB = 0 \iff \text{tr}(AB) = 0$  can be found in [33].

Here we show  $AB = 0 \iff \text{sym}(AB) = 0$ .

( $\Rightarrow$ ) Obvious.

( $\Leftarrow$ ) By hypothesis,  $\text{sym}(AB) = 0$  and since  $\text{tr}(\text{skw}(AB)) = 0$ , the trace of  $AB$  is null. Since  $A \in \mathbb{S}_-^m$  and  $B \in \mathbb{S}_+^m$ , there exist  $P, Q \in \mathbb{S}^m$  such that  $A = -PP$  and  $B = QQ$ . Then we have

$$0 = \text{tr}(AB) = -\text{tr}(PPQQ) = -\text{tr}(QPPQ) = \tag{10}$$

$$= -\langle PQ, PQ \rangle = -\|PQ\|^2. \tag{11}$$

Then  $PQ = 0$ . As a result,  $PPQQ = 0$  and  $AB = 0$ .  $\square$

**Lemma 2** If  $A, B \in \mathbb{S}^m$ ,  $A \succ 0$  and  $AB + BA \prec 0$  then  $B \prec 0$ .

*Proof* Suppose that there exist an eigenvalue  $\lambda \geq 0$  of  $B$  with an eigenvector  $v$ . Then,

$$v^\top (AB + BA)v = v^\top ABv + v^\top BAv = 2\lambda v^\top Av$$

Since  $A$  is positive definite,  $\lambda v^\top Av \geq 0$  therefore,  $v^\top (AB + BA)v \geq 0$ , and we conclude that  $AB + BA$  is not negative definite, which is a contradiction with the hypothesis.  $\square$

The proof of following lemma is similar as above.

**Lemma 3** *If  $A, B \in \mathbb{S}^m$ ,  $A \succ 0$  and  $AB + BA \preceq 0$  then  $B \preceq 0$ .*

**Lemma 4** *If  $A, B \in \mathbb{S}^m$  and  $A \succ 0$ , the matrix  $AB$  has real eigenvalues and the same inertia as  $B$ . See [20].*

**Lemma 5** *If  $A \in \mathbb{S}_{++}^m$  and  $B \in \mathbb{S}_-^m$  and they commute, then*

$$y^\top AB y = 0 \iff B y = 0.$$

*Proof* ( $\Rightarrow$ ): In view of lemma 4, all eigenvalues of  $AB$  are non positive. Let  $\lambda_1, \dots, \lambda_m$  and  $\{b_1, \dots, b_m\}$  be the eigenvalues and the orthonormal eigenvectors of  $AB$ , respectively. Let  $y$  be a not null vector in  $\mathbb{R}^m$  such that  $y^\top (AB)y = 0$ . Then,

$$y = \sum_{j=1}^m \alpha_j b_j$$

and

$$0 = y^\top (AB)y = \langle y, (AB)y \rangle \quad (12)$$

$$= \sum_{j=1}^m \sum_{i=1}^m \alpha_i \alpha_j \lambda_i \langle b_i, b_j \rangle = \sum_{i=1}^m \alpha_i^2 \lambda_i \quad (13)$$

Since,  $y \neq 0$ , there exist  $\alpha_i \neq 0$ . In view of the last equation,  $\lambda_i$  must be null, therefore  $y \in \ker(AB)$ , but  $A \in \mathbb{S}_{++}^m$  then  $y \in \ker(B)$ .  $\square$

**Lemma 6** *If  $A \in \mathbb{S}_-^m$  and  $B \in \mathbb{S}^m$  then,*

$$AB^2 = 0 \iff \text{sym}(AB) = 0$$

*Proof* ( $\Rightarrow$ ):  $A \in \mathbb{S}_-^m$  then there exist  $Q \in \mathbb{S}^m$  such that  $A = -QQ$ .

$0 = \text{tr}(AB^2) = \text{tr}(BAB) = -\text{tr}(BQQB) = -\|BQ\|^2$ , then  $BQ = 0$  and  $-BQQ = BA = 0$ .

( $\Leftarrow$ ):  $AB + BA = 0 \Leftrightarrow AB = -BA \Rightarrow AB^2 = -BAB \Leftrightarrow A$  and  $B^2$  are simultaneously diagonalizable and  $AB^2 \preceq 0$ . On the other hand,  $-BAB = B(-A)B \succ 0$ . Then,  $0 \preceq -BAB = AB^2 \preceq 0$ . Then, we conclude that  $AB^2 = 0$ .  $\square$

**Lemma 7** *If we assume  $A \in \mathbb{S}_{++}^m$  and  $B \in \mathbb{S}_-^m$  and they commute then,*

$$(A \otimes I)^{-1}(B \otimes I) \in \mathbb{S}_-^{\overline{m}}.$$

*Proof* Since  $A \succ 0$  and  $B \preccurlyeq 0$  and commute then there exist a regular matrix  $P \in \mathbb{R}^{m \times m}$  such that

$$\begin{aligned} A &= PD_A P^{-1} \\ B &= PD_B P^{-1} \end{aligned}$$

In view of symmetric Kronecker product properties, see for example lemma E.1.2 in [23] :

$$\begin{aligned} A \otimes I &= PD_A P^{-1} \\ B \otimes I &= PD_B P^{-1} \end{aligned}$$

where  $P = P \otimes P$ ,  $D_A = D_A \otimes I$  e  $D_B = D_B \otimes I$ .  $D_A$  e  $D_B$  are diagonal matrices and  $P$  is orthonormal. Therefore

$$(A \otimes I)^{-1}(B \otimes I) = PD_A D_B P^\top \in \mathbb{S}_{-}^{\overline{m}}$$

□

**Lemma 8** Let  $B \in \mathbb{S}_{-}^m$  and  $\{b_1, \dots, b_p\}$  be an orthonormal basis of  $\ker(B)$  and  $E_0 = [b_1, \dots, b_p] \in \mathbb{R}^{m \times p}$ . The following sets are equal:

$$\begin{aligned} C_1 &= \{A \in \mathbb{S}^m : \langle A, B \rangle = 0, A \succcurlyeq 0, B \preccurlyeq 0\} \\ C_2 &= \{E_0 \theta E_0^\top : \theta \in \mathbb{S}^p, \theta \succcurlyeq 0\} \end{aligned}$$

*Proof* First we prove  $C_1 \subset C_2$ . If  $A \in C_1$ ,  $A \succcurlyeq 0$ . By hypothesis  $B \preccurlyeq 0$  and  $\langle A, B \rangle = \text{tr}(AB) = 0$ . Using proposition 1,  $AB = 0$ , consequently  $A$  and  $B$  are simultaneously diagonalizable, then

$$A = [E_0 \ E_\perp] \begin{bmatrix} D_A & 0 \\ 0 & 0 \end{bmatrix} [E_0 \ E_\perp]^\top = E_0 D_A E_0^\top$$

where  $D_A \succcurlyeq 0$  is a diagonal matrix and the columns of  $E_\perp$  are eigenvectors of  $B$  in  $\ker(B)^\perp$ , then,  $A \in C_2$ .

Now we proceed to prove  $C_2 \subset C_1$ . If  $A \in C_2$ ,  $A = E_0 \theta E_0^\top$  for some  $\theta \succcurlyeq 0$ . On the other hand,  $\langle A, B \rangle = \text{tr}(E_0 \theta E_0^\top B) = 0$  because  $E_0^\top B = 0$ . Then,  $A \in C_1$ . □

## 2 Description of the algorithm

In this section, we introduce the main ideas of the present algorithm.

For a real number  $a$ , we denote

$$\Omega_a = \{x \in \Omega \text{ such that } f(x) \leq a\}$$

and introduce the following assumptions about  $f$  and  $G$ :

**Assumption 1** There exist a real number  $a$  such that  $\Omega_a$  is compact and  $\text{int}(\Omega_a) \neq \emptyset$ .

**Assumption 2** If  $x \in \text{int}(\Omega_a)$  then  $G(x) \prec 0$ .

**Assumption 3**  $f$  and  $G$  are  $C^1$  in  $\Omega_a$  and  $\nabla f$  and  $\frac{\partial G}{\partial x_p}$  for  $p = 1, \dots, n$  are Lipschitz functions.



**Assumption 4** Any KKT point  $x$  is a regular point of problem (1).

The present algorithm makes iterations in the primal and dual variables  $(x, \Lambda)$  to solve the equalities in KKT conditions. The method is modified in such a way to satisfy the inequalities at each point. That is, the primal variables are feasible and the dual variables, positive at each iteration.

Many approaches have been proposed to handle the complementarity condition  $\Lambda G(x) = 0$ , see [4] and [36]. The complementarity condition  $\Lambda G(x) = 0$  is not appropriate for a Newton iteration since the product of two symmetric matrices is not, in general, a symmetric matrix. Instead of  $\Lambda G(x) = 0$  in our case we consider  $\text{sym}(\Lambda G(x)) = 0$ . Since our algorithm generates points  $(x, \Lambda)$  such that  $G(x) \prec 0$  and  $\Lambda \succ 0$ , Lemma 1 ensures that  $\Lambda G(x) = 0$  is equivalent to  $\text{sym}(\Lambda G(x)) = 0$ .

Then, the equalities of the stationary point conditions (8) can be rewritten in the following form:

$$\begin{aligned} \nabla f(x) + \nabla G(x)\lambda &= 0 \\ \text{svec}(\text{sym}(\Lambda G(x))) &= 0 \end{aligned} \quad (14)$$

where  $\lambda = \text{svec}(\Lambda)$ .

In order to obtain the linear system to be solved at each Newton iteration we define a vectorial function  $\psi: \mathbb{R}^{n+\overline{m}} \rightarrow \mathbb{R}^{n+\overline{m}}$ , given by the equalities in (14)

$$\psi(x, \lambda) = \begin{bmatrix} \psi_l(x, \lambda) \\ \psi_c(x, \lambda) \end{bmatrix} = \begin{bmatrix} \nabla f(x) + \nabla G(x)\lambda \\ \text{svec}(\text{sym}(\Lambda G(x))) \end{bmatrix}$$

Using the Kronecker product we observe that

$$\psi_c(x, \lambda) = \text{svec}(\text{sym}(\Lambda G(x))) = [I \otimes G(x)]\text{svec}(\Lambda)$$

and also,

$$\psi_c(x, \lambda) = \text{svec}(\text{sym}(\Lambda G(x)I)) = [\Lambda \otimes I]\text{svec}(G(x))$$

therefore, the Jacobian of  $\psi$  is:

$$\begin{aligned} \nabla \psi(x, \lambda) &= \begin{bmatrix} \nabla_x \psi_l(x, \lambda) & \nabla_\lambda \psi_l(x, \lambda) \\ \nabla_x \psi_c(x, \lambda) & \nabla_\lambda \psi_c(x, \lambda) \end{bmatrix} \\ &= \begin{bmatrix} \nabla_{xx} L(x, \lambda) & \nabla G(x) \\ (\Lambda \otimes I) \nabla G(x)^\top & I \otimes G(x) \end{bmatrix} \end{aligned} \quad (15)$$

In a similar way as in [19], instead of computing the Hessian of the Lagrangian  $\nabla_{xx} L(x, \lambda)$ , we can employ a quasi-Newton matrix or the identity matrix that we denote  $B$ .

Given a starting point  $(x, \Lambda) \in \text{int}(\Omega_a) \times \mathbb{S}_{++}^m$ , a new estimative  $(x_0, \lambda_0) \in \mathbb{R}^n \times \mathbb{R}^{\overline{m}}$  is obtained by solving the following linear system

$$\begin{aligned} \begin{bmatrix} B & \nabla G(x) \\ (\Lambda \otimes I) \nabla G(x)^\top & I \otimes G(x) \end{bmatrix} \begin{bmatrix} x_0 - x \\ \lambda_0 - \lambda \end{bmatrix} &= - \begin{bmatrix} \nabla f(x) + \nabla G(x)\lambda \\ \text{svec}(\text{sym}(\Lambda G(x))) \end{bmatrix} \end{aligned} \quad (16)$$

where  $\lambda = \text{svec}(\Lambda)$  and  $\lambda_0 = \text{svec}(\Lambda_0)$ . We denote  $W(x, B, \Lambda)$  the matrix of the system (16).

Calling  $d_0 = x_0 - x$  we have,

$$\begin{aligned} Bd_0 + \nabla G(x)\lambda_0 &= -\nabla f(x) \\ (\Lambda \otimes I) \nabla G(x)^\top d_0 + (I \otimes G(x)) \lambda_0 &= 0 \end{aligned} \quad (17)$$

Note that, if  $d_0 = 0$  equation (17) becomes,

$$\nabla f(x) + \nabla G(x)\lambda_0 = 0 \quad (18)$$

and

$$(I \otimes G(x)) \lambda_0 = 0 \quad (19)$$

Since  $G(x) \prec 0$ , the matrix  $I \otimes G(x)$  is nonsingular and then, equation (19) makes  $\lambda_0 = \text{smat}(\lambda_0) = 0$ . This proves that  $\Lambda_0 G(x) = 0$ . Now, looking at equation (18) with  $\lambda_0 = 0$  we have  $\nabla f(x) = 0$ . Then,  $x$  is an stationary point of problem (1).

On the other hand, if  $d_0 \neq 0$ , we prove in this paper that  $d_0$  is a descent direction however we can not ensure that  $d_0$  is a feasible direction when  $x$  is on the boundary of  $\Omega$ .

To obtain a feasible direction, as in [19], we define another linear system in  $(d, \bar{\lambda}) \in \mathbb{R}^n \times \mathbb{R}^m$  with the same matrix and an appropriate negative definite right hand side:

$$\begin{aligned} Bd + \nabla G(x)\bar{\lambda} &= -\nabla f(x) \\ (\Lambda \otimes I) \nabla G(x)^\top d + (I \otimes G(x)) \bar{\lambda} &= -\rho\lambda \end{aligned} \quad (20)$$

where  $\rho$  is a positive real number and  $\lambda = \text{svec}(\Lambda)$ . In view of the equation (2) and (6), the second equation of (20) is equivalent to

$$\text{sym}(\Lambda DG(x)(d) + \bar{\Lambda} G(x)) = -\rho\Lambda \quad (21)$$

where  $\bar{\Lambda} = \text{smat}(\bar{\lambda})$ .

If  $x$  is in the interior of  $\Omega$ , all direction  $d$  is a feasible direction.

In the sequel we introduce the following assumption.

**Assumption 5** At every iteration  $\Lambda$  and  $G(x)$  commute.

Thanks to this last assumption and (21) it will be able to prove that  $d$  is a feasible direction for all  $x \in \Omega_a$ .

In view of (20),  $d$  converges to  $d_0$  if  $\rho$  goes to zero. Since  $d_0$  is a descent direction, it verifies  $d_0^\top \nabla f(x) < 0$ , then imposing an upper bound for  $\rho$  such that

$$d^\top \nabla f(x) \leq \xi d_0^\top \nabla f(x) \quad \xi \in (0, 1) \quad (22)$$

the feasible direction  $d$  is also a descent direction.

The pair  $(d, \bar{\lambda})$  obtained by the linear system (20) can also be computed solving

$$\begin{aligned} Bd_1 + \nabla G(x)\lambda_1 &= 0 \\ (\Lambda \otimes I) \nabla G(x)^\top d_1 + (I \otimes G(x)) \lambda_1 &= -\lambda \end{aligned} \quad (23)$$

and taking,

$$d = d_0 + \rho d_1 \quad (24)$$

$$\bar{\lambda} = \lambda_0 + \rho \lambda_1. \quad (25)$$

Imposing (22) to the search direction  $d$  defined in (24) we have a precise upper bound for  $\rho$ . If  $d_1^\top \nabla f(x) > 0$ , take

$$\rho \leq (\xi - 1) \frac{d_0^\top \nabla f(x)}{d_1^\top \nabla f(x)}.$$

Otherwise, we choose

$$\rho \leq \varphi \|d_0\|^2,$$

for some fixed parameter  $\varphi > 0$ .

Once we have computed a descent and feasible direction  $d$ , we can determine the next point in the sequence,  $x^{k+1}$ , performing a line search along the search direction  $d$  to get feasibility and an appropriate reduction of the objective function.

Before starting a new iteration, matrix  $B$  can be updated with a new positive definite matrix. Also, matrix  $\Lambda$  must be updated with a new positive definite matrix that commutes with  $G(x^{k+1})$ .

## 2.1 The statement of the algorithm

In this section we state the algorithm under study based on the descent and feasible direction obtained in the previous section.

**Input.**  $x \in \text{int}(\Omega_a)$

**Parameters.**  $\xi \in (0, 1)$ ,  $\eta \in (0, 1)$ ,  $\varphi > 0$  and  $\nu \in (0, 1)$ .

**Data.**  $B \in \mathbb{S}_{++}^n$ ,  $\Lambda \in \mathbb{S}_{++}^m$  commuting with  $G(x)$ .

**Step 1.** Computation of the search direction  $d$ .

(i) Solve the following linear system in  $d_0 \in \mathbb{R}^n$  and  $\lambda_0 \in \mathbb{R}^{\overline{m}}$

$$\begin{bmatrix} B & \nabla G(x) \\ (\Lambda \otimes I) \nabla G(x)^\top & I \otimes G(x) \end{bmatrix} \begin{bmatrix} d_0 \\ \lambda_0 \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix} \quad (26)$$

If  $d_0 = 0$ , stop.

(ii) Solve the following linear system in  $d_1 \in \mathbb{R}^n$  and  $\lambda_1 \in \mathbb{R}^{\overline{m}}$

$$\begin{bmatrix} B & \nabla G(x) \\ (\Lambda \otimes I) \nabla G(x)^\top & I \otimes G(x) \end{bmatrix} \begin{bmatrix} d_1 \\ \lambda_1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\lambda \end{bmatrix} \quad (27)$$

(iii) Compute the parameter  $\rho$  such that

$$\rho = \min \left\{ \varphi \|d_0\|^2, (\xi - 1) \frac{d_0^\top \nabla f(x)}{d_1^\top \nabla f(x)} \right\} \quad (28)$$

if  $d_1^\top \nabla f(x) > 0$ . Otherwise:

$$\rho = \varphi \|d_0\|^2. \quad (29)$$

(iv) Compute the search direction  $d$  as

$$d = d_0 + \rho d_1. \quad (30)$$

**Step 2.** Line Search.

Find  $t$ , the first element of  $\{1, v, v^2, v^3 \dots\}$  such that

$$f(x + td) \leq f(x) + t\eta d^\top \nabla f(x) \quad (31)$$

and

$$G(x + td) \prec 0. \quad (32)$$

**Step 3.** Updates.

- (i) Take the new point  $x := x + td$ .
- (ii) Define new value for  $B \in \mathbb{S}_{++}^n$ .
- (iii) Define new value for  $\Lambda \in \mathbb{S}_{++}^m$  commuting with  $G(x)$ .
- (iv) Go to Step 1.

□

In the previous algorithm we employ Armijo's line search adapted to constraint optimization problems. Extensions of Wolfe or Goldstein line search criteria [27] could be also employed in step 2.

We must consider additional assumption on  $B$  and  $\Lambda$  defined by the previous algorithm in order to prove global convergence.

**Assumption 6** There exist positive numbers  $\lambda^I$  and  $\lambda^S$  such that

$$\lambda^I I \preccurlyeq \Lambda \preccurlyeq \lambda^S I$$

**Assumption 7** There exist positive numbers  $\sigma_1$  and  $\sigma_2$  such that

$$\sigma_1 I \preccurlyeq B \preccurlyeq \sigma_2 I$$

### 3 Global convergence

In this section we prove for any initial point  $x^0 \in \text{int}(\Omega)$ , the present algorithm generates a sequence of points converging to a stationary point of problem (1). First we prove that the algorithm is well defined and in particular that the matrix  $W(x, B, \Lambda)$  given in (16) is nonsingular. At each iteration  $d_0$  and  $d$  are descent directions of  $f$  at  $x$  and  $d(x)$  is an uniformly feasible directions field in  $\Omega_a$ . Finally we state that any sequence generated by the algorithm converges to a stationary point of (1).

**Theorem 1** Assume that  $x \in \Omega_a$  is a regular point of problem (1),  $B \in \mathbb{S}_{++}^n$ ,  $\Lambda \in \mathbb{S}_{++}^m$  and  $\Lambda$  and  $G(x)$  commute. Then, the matrix  $W(x, B, \Lambda)$  defined in (16) is nonsingular.

*Proof* We have to prove that, if for some  $v \in \mathbb{R}^{n+\overline{m}}$  it is  $W(x, B, \Lambda)v = 0$ , then  $v = 0$ . Let  $v^\top = [r^\top, y^\top]$ ,  $r \in \mathbb{R}^n$ ,  $y \in \mathbb{R}^{\overline{m}}$ , then

$$r = -B^{-1}\nabla G(x)y$$

$$My = 0$$

where

$$M = \nabla G(x)^\top B^{-1} \nabla G(x) - (\Lambda \otimes I)^{-1} (G(x) \otimes I).$$

To finish the proof, we must conclude that  $M$  is nonsingular.

Performing the product  $y^\top My$  we have

$$\begin{aligned} y^\top My &= y^\top \left( \nabla G(x)^\top B^{-1} \nabla G(x) \right) y \\ &\quad - y^\top (\Lambda \otimes I)^{-1} (G(x) \otimes I) y \end{aligned} \quad (33)$$

Since  $B^{-1}$  is positive definite,

$$y^\top \left( \nabla G(x)^\top B^{-1} \nabla G(x) \right) y \geq 0 \quad (34)$$

By hypothesis and lemma 7

$$-y^\top (\Lambda \otimes I)^{-1} (G(x) \otimes I) y \geq 0 \quad (35)$$

concluding that  $y^\top My \geq 0$ .

Now, suppose that  $y^\top My = 0$ . We must prove that  $y = 0$ . From (33), (34) and (35) we have

$$\begin{aligned} 0 &\leq y^\top \left( \nabla G(x)^\top B^{-1} \nabla G(x) \right) y \\ &= y^\top (\Lambda \otimes I)^{-1} (G(x) \otimes I) y \leq 0 \end{aligned} \quad (36)$$

Then,

$$y^\top \nabla G(x)^\top B^{-1} \nabla G(x) y = 0 \quad (37)$$

and

$$y^\top (\Lambda \otimes I)^{-1} (G(x) \otimes I) y = 0. \quad (38)$$

Since  $B$  is positive definite, from equation (37) we have

$$\nabla G(x) y = 0. \quad (39)$$

Due to lemma 5 and from equation (38),

$$(G(x) \otimes I) y = 0. \quad (40)$$

By definition of  $\nabla G(x)$ , equation (39) is equivalent to

$$\left\langle \frac{\partial G}{\partial x_i}(x), Y \right\rangle = 0, \quad i = 1, \dots, n \quad (41)$$

where  $Y = \text{smat}(y)$ .

Since  $(G(x) \otimes I) y = \text{svec}(\text{sym}(G(x)Y))$ , in view of lemma 6 and proposition 1, equation (40) can be rewritten as

$$\langle Y^2, G(x) \rangle = 0 \quad (42)$$

The matrix  $Y^2$  is positive semidefinite and matrix  $G(x)$  is negative semidefinite, then applying lemma 8 we have

$$Y^2 = E_0 \theta E_0^\top, \quad \theta \in \mathbb{S}^p, \quad \theta \succcurlyeq 0 \quad (43)$$

where  $E_0 = [b_1, \dots, b_p] \in \mathbb{R}^{m \times p}$  and the set of vectors  $\{b_1, \dots, b_p\}$  is an orthonormal base of  $\ker(G(x))$ . Taking the square root of  $Y^2$  in (43), we obtain

$$Y = E_0 \theta^{1/2} E_0^\top, \quad \theta^{1/2} \in \mathbb{S}^p \quad (44)$$

Replacing  $Y$  by  $E_0\theta^{1/2}E_0^\top$  in equation (41), follows

$$\left\langle \frac{\partial G}{\partial x_i}(x), E_0\theta^{1/2}E_0^\top \right\rangle = 0, \quad i = 1, \dots, n$$

which is equivalent to

$$\left\langle E_0^\top \frac{\partial G}{\partial x_i}(x) E_0, \theta^{1/2} \right\rangle = 0, \quad i = 1, \dots, n$$

and also equivalent to the following linear system,

$$V \text{svec}(\theta^{1/2}) = 0 \quad (45)$$

where

$$V = \begin{bmatrix} \left[ \text{svec} \left( E_0^\top \frac{\partial G}{\partial x_1}(x) E_0 \right) \right]^\top \\ \vdots \\ \left[ \text{svec} \left( E_0^\top \frac{\partial G}{\partial x_n}(x) E_0 \right) \right]^\top \end{bmatrix}.$$

We identify in the columns of matrix  $V$ , vectors of the form

$$c \left[ b_i^\top \frac{\partial G}{\partial x_i}(x) b_j \dots b_i^\top \frac{\partial G}{\partial x_n}(x) b_j \right]^\top$$

with  $c = \sqrt{2}$  when  $i = j$  and  $c = 1$  when  $i \neq j$  where  $i, j = 1, \dots, p$  and  $i \leq j$ . By hypothesis,  $x$  is a regular point of problem (1), then the columns of  $V$  are linearly independent. Then, the linear system (45) has the unique solution  $\text{svec}(\theta^{1/2}) = 0$ . Consequently,  $Y = \text{smat}(y) = E_0\theta^{1/2}E_0^\top = 0$ .  $\square$

As an additional result of the proof of theorem 1, we have that  $M$  is positive definite.

Since  $\Omega_a$ ,  $A$  and  $B$  are bounded, it follows from theorem 1 that  $d_0$ ,  $\lambda_0$ ,  $d_1$  and  $\lambda_1$  are also bounded.

When  $d_0 = 0$  is obtained in step 1, the algorithm stops. In fact, since  $G(x) \prec 0$ ,  $G(x) \otimes I \prec 0$  then the solution of (26) is  $\lambda_0 = 0$ . Thus,  $\nabla f(x) = 0$  and we have that  $x$  is a KKT point associated with a null Lagrangian multiplier matrix.

In what follows we consider the case where, at every iteration,  $d_0 \neq 0$ .

**Lemma 9** *The direction  $d_0$  computed by the algorithm satisfies*

$$d_0^\top \nabla f(x) \leq -d_0^\top B d_0.$$

*Proof* Multiplying the first equation of (17) by  $d_0^\top$ ,

$$d_0^\top \nabla f(x) = -d_0^\top B d_0 - d_0^\top \nabla G(x) \lambda_0.$$

In view of the second equation of (17),

$$-d_0^\top \nabla G(x) \lambda_0 = \lambda_0^\top (G(x) \otimes I) (A \otimes I)^{-1} \lambda_0.$$

Then,

$$d_0^\top \nabla f(x) = -d_0^\top B d_0 + \lambda_0^\top (G(x) \otimes I) (\Lambda \otimes I)^{-1} \lambda_0.$$

Thanks to assumption 5 and 7,  $(\Lambda \otimes I)^{-1} (G(x) \otimes I) \in \mathbb{S}_-^m$  and  $B \in \mathbb{S}_{++}^n$  then,

$$d_0^\top \nabla f(x) \leq -d_0^\top B d_0.$$

□

As a consequence, if the direction  $d_0 \neq 0$ , it is a descent direction of  $f$  at  $x$ .

**Lemma 10** *The search direction  $d$  computed by the algorithm satisfies*

$$d^\top \nabla f(x) \leq \xi d_0^\top \nabla f(x). \quad (46)$$

*Proof* In view of (30),

$$d^\top \nabla f(x) = d_0^\top \nabla f(x) + \rho d_1^\top \nabla f(x).$$

If  $d_1^\top \nabla f(x) > 0$ , using (28),

$$\rho d_1^\top \nabla f(x) \leq (\xi - 1) d_0^\top \nabla f(x).$$

If  $d_1^\top \nabla f(x) \leq 0$ , using (29),

$$\rho \leq \varphi \|d_0\|^2.$$

Both cases verifies (46) with  $\xi \in (0, 1)$ . □

Since  $d_0$  is a descent direction of  $f$  at  $x$ , then  $d_0^\top \nabla f(x) < 0$ . Lemma 10 implies that  $d$  is also a descent direction of  $f$  at  $x$ .

**Lemma 11** *The search direction  $d$  and the parameter  $\rho$  computed by the algorithm verifies:*

$$\varphi_0 \|d_0\|^2 \leq \rho \leq \varphi \|d_0\|^2 \quad (47)$$

and

$$\|d\| \leq \delta \|d_0\|, \quad (48)$$

for some  $\delta \geq 1$  and  $\varphi_0 > 0$ .

*Proof* Looking at (28) and (29), we see that  $\rho \leq \varphi \|d_0\|^2$ . By lemma 9 and assumption 7,

$$-d_0^\top \nabla f(x) \geq \sigma_1 \|d_0\|^2.$$

If  $d_1^\top \nabla f(x) > 0$  and in view of (28) we have,

$$\min \left\{ \varphi, \frac{(1 - \xi) \sigma_1}{d_1^\top \nabla f(x)} \right\} \|d_0\|^2 \leq \rho.$$

Since  $d_1$  is bounded and (29), there exist  $\varphi_0 > 0$  such that

$$\varphi_0 \|d_0\|^2 \leq \rho.$$

and then (47) is proved.

Now using (24) and the triangular property,

$$\|d\| = \|d_0 + \rho d_1\| \leq \|d_0\| + \rho \|d_1\|$$

and using condition (47),

$$\|d\| \leq \|d_0\| + \varphi \|d_0\|^2 \|d_1\| = \delta \|d_0\|,$$

where  $\delta = 1 + \varphi \|d_0\| \|d_1\| \geq 1$ . □

As a consequence of (47) and (48),  $\rho$  and  $\|d\|^2$  has the same order of magnitude, in particular,

$$\varphi_0 \|d(x)\|^2 \leq \rho(x) \leq \varphi \|d(x)\|^2, \quad x \in \Omega_a. \quad (49)$$

**Lemma 12** *It follows from assumption (3) that there exists a positive real number  $L$  such that*

$$G(y) \preceq G(x) + DG(x)(y - x) + L\|y - x\|^2 I \quad (50)$$

where  $x, y \in \Omega$ .

*Proof* Since  $G$  is  $C^1$  we can apply the Mean Value Theorem [26]

$$G(y) = G(x) + \sum_{i=1}^n (y_i - x_i) \frac{\partial G}{\partial x_i}(x + \xi(y - x)) \quad (51)$$

for some  $\xi \in (0, 1)$ . We have also that

$$\begin{aligned} (y_i - x_i) \frac{\partial G}{\partial x_i}(x + \xi(y - x)) &\preceq (y_i - x_i) \frac{\partial G}{\partial x_i}(x) \\ &+ \left\| \frac{\partial G}{\partial x_i}(x + \xi(y - x)) - \frac{\partial G}{\partial x_i}(x) \right\| \|y - x\| I. \end{aligned} \quad (52)$$

Since  $\frac{\partial G}{\partial x_i}$  verify the Lipschitz condition, there exist  $L_i > 0$  such that

$$\left\| \frac{\partial G}{\partial x_i}(x + \xi(y - x)) - \frac{\partial G}{\partial x_i}(x) \right\| \leq L_i \|\xi(y - x)\| = L_i \xi \|y - x\|. \quad (53)$$

Then, using (53), (52) and (51), we have,

$$G(y) \preceq G(x) + \sum_{i=1}^n (y_i - x_i) \frac{\partial G}{\partial x_i}(x) + L\|y - x\|^2 I$$

where  $L = \xi \sum_{i=1}^n L_i$ . □

Due to assumption (1) the sequence  $\{x^k\}_{k \in \mathbb{N}} \in \text{int}(\Omega_a)$  generated by the algorithm have an accumulation point  $x^* \in \Omega_a$ . It follows that there exist  $\mathbb{K}_1 \subset \mathbb{N}$  such that  $\{d(x^k), \rho(x^k), \bar{\lambda}(x^k), G(x^k)\}_{k \in \mathbb{K}_1}$  converges to  $\{d(x^*), \rho(x^*), \bar{\lambda}(x^*), G(x^*)\}$ .

In [19], was proved the existence of a lower bound  $\tau_f > 0$  such that at any  $x \in \Omega_a$ , condition (31) is verified for any  $t \in [0, \tau_f]$ .

**Proposition 1** *For all  $x \in \Omega_a$  such that  $\|d(x)\| \geq M$  there exist  $\tau > 0$  such that:*

$$G(x + td(x)) \preceq 0 \quad (54)$$

for all  $t \in [0, \tau]$ .



*Proof* Thanks to lemma 12, there exist  $L > 0$  such that

$$G(x + td(x)) \preceq G(x) + t DG(x)d + t^2 L \|d(x)\|^2 I \quad (55)$$

Let  $F(t, x)$  be the matrix in the right hand side of (55), then,

$$G(x + td(x)) \preceq F(t, x). \quad (56)$$

We are going to prove the existence of  $\tau > 0$  such that

$$\text{sym}(\Lambda F(t, x)) \preceq 0, \quad t \in [0, \tau], \quad (57)$$

and since  $\Lambda \succ 0$ , lemma 3 tell us that,

$$F(t, x) \preceq 0, \quad t \in [0, \tau], \quad (58)$$

and then condition (54) is true.

Taking the symmetric part of  $\Lambda F(t, x)$  and considering (21), we can write,

$$\begin{aligned} \text{sym}(\Lambda F(t, x)) &= \text{sym}((\Lambda - t\bar{\Lambda})G(x)) + \\ &+ t \left( tL \|d(x)\|^2 - \rho(x) \right) \Lambda \end{aligned} \quad (59)$$

where  $\bar{\Lambda} = \text{smat}(\bar{\lambda})$  is given in (25).

Since  $\rho$  verifies (49), we have,

$$\rho(x) \leq \varphi_0 \|d(x)\|^2. \quad (60)$$

Then for all  $v$  such that  $\|v\| = 1$

$$\begin{aligned} v^t (\Lambda F(t, x)) v &\leq v^t \Lambda G(x) v \\ &- t(v^t (\bar{\Lambda} G(x)) v + \varphi_0 \|d(x)\|^2 \lambda^I) + \\ &+ t^2 L \|d(x)\|^2 \lambda^S \end{aligned} \quad (61)$$

The right hand side is non-positive when  $t = 0$  then, let us consider the following second order equation:

$$\begin{aligned} v^t \Lambda G(x) v - t(v^t (\bar{\Lambda} G(x)) v + \varphi_0 \|d(x)\|^2 \lambda^I) + \\ + t^2 L \|d(x)\|^2 \lambda^S = 0 \end{aligned} \quad (62)$$

where  $x \in \Omega_a$ ,  $\|d(x)\| \geq M$  and  $v$  such that  $\|v\| = 1$ .

Let  $t(x, v)$  be the positive solution of (62):

$$\begin{aligned} t(x, v) &= \left( \frac{v^t (\bar{\Lambda} G(x)) v}{2L \|d(x)\|^2 \lambda^S} + \frac{\varphi_0 \lambda^I}{2L \lambda^S} \right) + \\ &+ \sqrt{\left( \frac{v^t (\bar{\Lambda} G(x)) v}{2L \|d(x)\|^2 \lambda^S} + \frac{\varphi_0 \lambda^I}{2L \lambda^S} \right)^2 - \frac{v^t \Lambda G(x) v}{L \|d(x)\|^2 \lambda^S}} \end{aligned} \quad (63)$$

Since  $v^t \Lambda G(x) v \leq 0$ ,  $\frac{\varphi_0 \lambda^I}{2L \lambda^S} > 0$  and considering lemma 5 it can be shown that  $t(x, v)$  is positive in the compact

$$\mathbb{S} = \Omega_a \cap \{x : \|d(x)\| \geq M\} \times \{v \in \mathbb{R}^n : \|v\| = 1\},$$

then there exist  $\tau > 0$  such that  $G(x + td(x)) \preceq 0$  for all  $t \in [0, \tau]$ .  $\square$

A consequence of proposition 1 is that if  $x^* \in \Omega_a$ , is an accumulation point of a sequence  $\{x^k\}_{k \in \mathbb{N}}$  generated by the algorithm then  $\|d(x^*)\| = 0$ .

**Proposition 2** *If  $\|d(x^*)\| = 0$  there exists  $\delta > 0$  and  $\tau > 0$  such that if  $x^k \in B(x^*, \delta) \cap \Omega_a$ ,*

$$G(x^k + td(x^k)) \preceq 0 \quad (64)$$

for all  $t \in [0, \tau]$ .

*Proof* Since  $\|d(x^*)\| = 0$  thanks to the continuity of  $d(x)$ ,  $\|d(x^k)\|$  goes to zero. It follows from (59) that if

$$\max\{v^\top (\Lambda^k - t\bar{\Lambda}^k)G(x^k)v, \quad v \in \mathbb{R}^m, \quad \|v\| = 1\} \leq 0. \quad (65)$$

then (64) is verified.

Since  $\Lambda^k$  and  $G(x^k)$  commute and are symmetric real matrices then there exist an orthonormal matrix  $P_k$  such that

$$\Lambda^k = P_k^t D_\lambda^k P_k \quad (66)$$

$$G(x^k) = P_k^t D_g^k P_k \quad (67)$$

where  $D_\lambda^k$  is the diagonal matrix of eigenvalues of  $\Lambda^k$  and  $D_g^k$  is the diagonal matrix of eigenvalues of  $G(x^k)$ .

Let  $\bar{\Lambda}^k = P_k \bar{\Lambda}^k P_k^t$  then we need to prove that there exist  $\delta > 0$  and  $\tau > 0$  such that if  $x^k \in B(x^*, \delta) \cap \Omega_a$  we have

$$v^\top ((D_\lambda^k - t\bar{\Lambda}^k)D_g^k)v \leq 0. \quad (68)$$

for all  $v \in \mathbb{R}^m$ ,  $\|v\| = 1$ .

Let  $\bar{\Lambda}^k = D_\lambda^* + M^k$ , where  $M^k = P_k \bar{\Lambda}^k P_k^t - D_\lambda^*$ . Since  $\bar{\Lambda}^k$  converges to  $D_\lambda^*$  for all  $\epsilon > 0$  there exist  $\delta_1$  such that for all  $x^k \in B(x^*, \delta_1) \cap \Omega_a$ ,  $|v^\top (M^k G(x^k))v| < \epsilon$ .

Since the eigenvalues of  $D_\lambda^k$  are between  $\lambda^I$  and  $\lambda^S$  by hypothesis and the eigenvalues of  $\Lambda^*$  are bounded, due to theorem of Weyl [20], there exist  $\tau > 0$  such that the eigenvalues of  $D_\lambda^k - tD_\lambda^*$  are positive for all  $t \in [0, \tau]$ .

Then for a sufficiently small  $\epsilon$ :

$$v^\top ((D_\lambda^k - tD_\lambda^*)D_g^k)v - t v^\top (M^k G(x^k))v \leq 0. \quad (69)$$

for all  $v \in \mathbb{R}^m$ ,  $\|v\| = 1$  and  $t \in [0, \tau]$ .  $\square$

**Theorem 2** *Any accumulation point  $x^*$  of the sequence  $\{x^k\}$  generated by the algorithm is a stationary point of problem (1).*

*Proof* Let  $\{x^k\}_{k \in \mathbb{K}}$  with  $\mathbb{K} \subset \mathbb{N}$  be a sequence converging to  $x^*$ .

Since  $\Lambda^k$ ,  $B^k$ ,  $d^k$  and  $\rho^k$  are bounded, there exist  $\mathbb{K}_1 \subset \mathbb{K}$  such that

$$\{x^k, \Lambda^k, B^k, \rho^k, d^k\}_{k \in \mathbb{K}_1}$$

converges to  $x^*, \Lambda^*, B^*, \rho^*, d^*$ .

The search direction is a continuous function of  $x$ ,  $\Lambda$ ,  $B$  and  $\rho$ . Then, the sequence  $\{d^k\}_{k \in \mathbb{K}_1}$  converges to  $d^*$  and  $d^* = d(x^*, \Lambda^*, B^*, \rho^*)$ .

Let  $\{t^k\}_{k \in \mathbb{K}_2}$  where  $\mathbb{K}_2 \subset \mathbb{K}_1$  be a sequence converging to  $t^*$ .

The limit  $\|d(x^*)\| = 0$ . If not we suppose that  $\|d^*\| > \eta_d > 0$  and thanks to proposition 1 we conclude that  $t^* > 0$ .

From the line search condition (31),

$$f(x^{fol(k)}) \leq f(x^k) + \eta \cdot t^k \cdot (d^k)^\top \nabla f(x^k).$$

where  $fol(k)$  is the element that follows  $k$  in  $\mathbb{K}_2$ . Taking the limits for  $k \rightarrow \infty$ ,

$$f(x^*) \leq f(x^*) + \eta \cdot t^* \cdot (d^*)^\top \nabla f(x^*).$$

Then,  $0 \leq (d^*)^\top \nabla f(x^*)$ .

Using lemma 10 and taking the limit for  $k \rightarrow \infty$ ,

$$(d^*)^\top \nabla f(x^*) \leq \xi(d_0^*)^\top \nabla f(x^*).$$

Using lemma 9 and taking the limit for  $k \rightarrow \infty$ ,

$$(d_0^*)^\top \nabla f(x^*) \leq -(d_0^*)^\top B^* d_0^*.$$

Since  $B^*$  is positive definite,  $d_0^* = d^* = 0$ . But this is contradictory with  $\|d^*\| > \eta_d > 0$ , then, the unique possibility is  $d^* = 0$ .

Considering equation (17), we have that  $(x^*, \lambda_0^*)$  verifies the following equations:

$$\nabla f(x^*) + \nabla G(x^*) \lambda_0^* = 0 \quad (70)$$

$$(G(x^*) \otimes I) \lambda_0^* = 0. \quad (71)$$

Equation (70) is the first equation of (2). It remains to proof the complementary condition  $G(x^*) \lambda_0^* = 0$ .

If  $x^* \in \text{int}(\Omega_a)$ , then  $G(x^*) \prec 0$ . Consequently  $G(x^*) \otimes I$  is non singular and the linear system (71) has the unique solution  $\lambda_0^* = 0$ . Then,  $\lambda_0^* = \text{smat}(\lambda_0^*) = 0$  and the complementarity condition holds.

Now consider the case when  $x^* \notin \text{int}(\Omega_a)$ . We must prove that  $G(x^*) \lambda_0^* = 0$ . Since equation (71),  $\text{sym}(G(x^*) \lambda_0^*) = 0$ . To finish the proof, we show that  $\text{skw}(G(x^*) \lambda_0^*) = 0$  or, equivalently, that all eigenvalues of  $G(x^*) \lambda_0^*$  are real.

Let  $k \in \mathbb{K}_2$ . Firstly, the matrix  $\lambda_0^k$  is symmetric. Secondly, the point  $x^k$  is in the interior of  $\Omega_a$  then,  $G(x^k) \prec 0$ . In view of lemma 4,  $G(x^k) \lambda_0^k$  has real eigenvalues. On the other hand, the sequence  $\{G(x^k) \lambda_0^k\}_{k \in \mathbb{K}_2}$  converges to  $G(x^*) \lambda_0^*$ . Since the eigenvalues are continuous functions, we conclude that  $G(x^*) \lambda_0^*$  has real eigenvalues.  $\square$

The last result proves that the presented algorithm globally converges to the solution of the problem in the convex case.

## 4 Numerical Implementation

In this section we include implementation details of the introduced algorithm.

We code the algorithm of section 2.1 in a MATLAB function. The parameters are fixed with the following values:  $\xi = 0.8$ ,  $\eta = 0.1$ ,  $\varphi = 1$ ,  $\nu = 0.7$ . Those values are commonly used in the “Feasible Direction Interior Point Algorithm” (FDIPA) [19]. Matrix  $B$  and  $A$  are initialized with the identity matrix.

Initially, the present algorithm takes as input, an interior point  $x$ . In order to mount the system matrix of (26), we compute  $f(x)$ ,  $G(x)$ ,  $\nabla f(x)$ ,  $\nabla G(x)$ ,  $A \otimes I$  and  $I \otimes G(x)$ . The symmetric Kronecker products are computed with the help of a MATLAB function used in SDPPACK [5]. Our actual implementation is appropriated for  $G(x)$  of limited size. This limitation comes from the fact that we do not explore the sparsity of  $G(x)$  and the size of the system matrix is  $n + \frac{1}{2}m(m+1)$ .

Since the linear systems (26) and (27) are the same, we performs a single factorization per iteration of the algorithm.

Solving (26) we obtain  $d_0(x) \in \mathbb{R}^n$  and  $\Lambda_0(x) \in \mathbb{S}^m$ . Let  $TOL$  a positive constant. The optimization process stops if  $\|d_0\| < TOL$  is verified. For small  $TOL$ , the condition  $\|d_0\| < TOL$  means, in view of (17) and (18), that  $x$  is near to a stationary point.

In order to accept the step length  $t$  in Armijo’s line search, we compute the maximum eigenvalue of  $G(x + td)$ .

In step 3 matrix  $B$  must be taken to verify assumption 7. We use the Broyden-Fletcher-Goldfarb-Shanno (BFGS) updating rule with Powell’s correction to ensure positive definiteness of  $B$  [30].

On the other hand matrix  $A$  must satisfy assumptions 5 and 6. In order to verify assumptions 6, we use the following “ $\Lambda_0$ -update” rule:

1. Compute  $\lambda_{0min}$ , the minimum eigenvalue of  $\Lambda_0$ .
2. If  $\lambda_{0min} \geq \lambda^I$  then, set  $A = \Lambda_0$ .
3. Else, set  $A = \Lambda_0 + (\lambda^I - \lambda_{0min}) I$ .

When  $\Lambda_0$  is not positive definite, the effect of this rule is to obtain a matrix  $A$  with all the eigenvalues of  $\Lambda_0$  shifted to the right such that the minimum eigenvalue of  $A$  be equal to  $\lambda^I$ . Since  $\Lambda_0$  is bounded, this rule generates  $A$  verifying assumption 6. Unfortunately, when  $x$  is not a stationary point,  $\Lambda_0$  do not commute with  $G(x)$ . Then,  $A$  generated with the  $\Lambda_0$ -update rule do not commutes with  $G(x)$ . However, in practice, using this approach, the system matrix (26) have a bounded condition number.

## 5 Applications in Structural Optimization

In this section an overview of three semidefinite models of truss topology optimization are presented: the minimum compliance problem, the robust truss topology design and the problem of maximizing the minimal eigenvalue. We present numerical examples of those models in section 6.

### 5.1 Minimum compliance problem

Consider a two or a three-dimensional truss with  $b$  bars. Let  $l$  be the degrees of freedom of the truss. The design variables of the our problems are the volumes of the bars denoted by  $x_j$ ,  $j = 1, \dots, b$ . We define the worst possible compliance of the structure for the set of loadings  $S \subset \mathbb{R}^l$  as,

$$\phi_S(x) = \sup\{2u^\top p - u^\top K(x)u : u \in \mathbb{R}^l, p \in S\}, \quad (72)$$

where  $K(x)$  is the reduced stiffness matrix.

A typical model is the minimum compliance problem, also called Truss Topology Design (TTD) [9]. This problem considers structures submitted to a finite set of nodal loads called “primary” loads, and looks for the volume of each bar that minimizes the structural compliance. The structural topology changes if the volume of some of the bars are zero in the solution. Let consider the set of primary load cases as,

$$P = \{p^1, \dots, p^s\}, \quad (73)$$

where  $p^i \in \mathbb{R}^l$  is the  $i$ -th load case. We can write the problem of minimum compliance as:

$$\begin{aligned} & \min_{x, \tau} \tau \\ & \text{s.t.} \quad \phi_P(x) \leq \tau, \\ & \quad \sum_{j=1}^b x_j \leq \bar{V}, \\ & \quad x \geq 0, \end{aligned} \quad (74)$$

where  $\bar{V}$  is a prescribed value for the total volume of the truss. Using the following equivalence proved in [6],

$$\phi_P(x) \leq \tau \iff \begin{bmatrix} \tau & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad p \in P, \quad (75)$$

it is possible to rewrite (74) as the following semidefinite program:

$$(TTD) \begin{cases} \min_{x, \tau} \tau \\ \text{s.t.} \quad \begin{bmatrix} \tau & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad p \in P, \\ \quad \sum_{j=1}^b x_j \leq \bar{V}, \\ \quad x \geq 0. \end{cases} \quad (76)$$

### 5.2 Robust truss topology design

The following development was proposed by Ben-Tal and Nemirovsky [7]. In the robust truss topology problem the structure is “robust” if it is reasonable rigid when any set of small possible uncertain loads act on it. In addition to the primary loads, we includes a set of “secondary” loads that are uncertain in size and direction and can act over the structure. The compliance to be minimized is the worst possible one, considering “primary” and “secondary” load cases.

Let  $M$  be the ellipsoid of loading conditions defined as follows:

$$M = \{Qe : e \in \mathbb{R}^q, e^\top e \leq 1\}, \quad (77)$$

where

$$Q = [p^1 \dots p^s \ r e^1 \dots r e^{q-s}] \in \mathbb{R}^{l \times q}. \quad (78)$$

The vectors  $re^1, \dots, re^{q-s}$  are called “secondary” load cases. The set  $\{e^1, \dots, e^{q-s}\}$  must be chosen as an orthonormal basis of a linear subspace orthogonal to the linear span of  $P$ . The value  $r$  is a prescribed magnitude.

The robust truss topology problem (*TTD*) is obtained replacing  $P$  by  $M$  in (74). In a similar way, using an equivalence similar to (75) proved in [7], it is possible to rewrite the robust truss topology model as the following semidefinite programming problem:

$$(RTT) \begin{cases} \min_{x, \tau} \tau \\ s.t. \quad \begin{bmatrix} \tau I & Q^\top \\ Q & K(x) \end{bmatrix} \succeq 0, \\ \sum_{j=1}^b x_j \leq \bar{V}, \\ x \geq 0, \end{cases} \quad (79)$$

where  $Q$  is defined in (78).

### 5.3 Structural topology optimization with eigenvalues

This subsection uses theoretical results and semidefinite programming problems presented in [1] and [2].

The eigenvalues of a truss are governed by the following equation:

$$K(x)v = \lambda M(x)v \quad (80)$$

where  $M(x)$  is the reduced mass matrix of the structure and  $(\lambda, v) \in \mathbb{R} \times \mathbb{R}^m$  is an eigenvector-eigenvalue pair. The well-defined smallest eigenvalue of problem (80), as explained in [1] and [2], is:

$$\begin{aligned} \lambda_{min} : X &\rightarrow \mathbb{R} \cup \{+\infty\} \\ \lambda_{min}(x) &= \sup\{\lambda : K(x) - \lambda M(x) \succ 0\} \end{aligned} \quad (81)$$

where  $X = \{x \in \mathbb{R}^b : x \geq 0, x \neq 0\}$ .

The constraint  $\lambda_{min}(x) \geq \lambda$  is equivalent to the semidefinite constraint  $K(x) - \lambda M(x) \succ 0$ , see [1]. Then, the topology optimization problem of minimum volume subject to constraints in the minimum eigenvalue and compliance is

$$(MV) \begin{cases} \min_{x, V} V \\ s.t. \quad \begin{aligned} &K(x) - \underline{\lambda} M(x) \succ 0, \\ &\begin{bmatrix} \bar{\gamma} & p^\top \\ p & K(x) \end{bmatrix} \succ 0, \quad p \in P, \\ &\sum_{j=1}^b x_j \leq V, \\ &x \geq 0, \end{aligned} \end{cases} \quad (82)$$

where  $\underline{\lambda}$  is a lower bound to the minimum eigenvalue and  $\bar{\gamma}$  an upper bound to the compliance.

In a similar way, the problem of minimum compliance subject to constraints in the minimum eigenvalue and volume is

$$(MC) \left\{ \begin{array}{l} \min_{x, \gamma} \gamma \\ s.t. \\ K(x) - \lambda M(x) \succcurlyeq 0, \\ \begin{bmatrix} \gamma & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad p \in P, \\ \sum_{j=1}^b x_j \leq \bar{V}, \\ x \geq 0, \end{array} \right. \quad (83)$$

where  $\bar{V}$  is an upper bound to the volume of the structure.

Finally, the problem of maximizing the minimum eigenvalue with compliance and volume constraints is

$$(MF) \left\{ \begin{array}{l} \max_{x, \lambda} \lambda \\ s.t. \\ K(x) - \lambda M(x) \succcurlyeq 0, \\ \begin{bmatrix} \bar{\gamma} & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad p \in P, \\ \sum_{j=1}^b x_j \leq \bar{V}, \\ x \geq 0. \end{array} \right. \quad (84)$$

In this paper we solve  $(MF)$ ,  $(MV)$  and  $(MC)$  and show its interrelation as in the example 4.1 of [1].

Problems  $(TTD)$ ,  $(RTT)$ ,  $(MV)$  and  $(MC)$  are convex optimization problems since the matrix constraint depends affinely on the design variables. On the other hand, problem  $(MF)$  is non convex since the constraint  $K(x) - \lambda M(x) \succcurlyeq 0$  depends linearly on the both design variables  $x$  and  $\lambda$ .

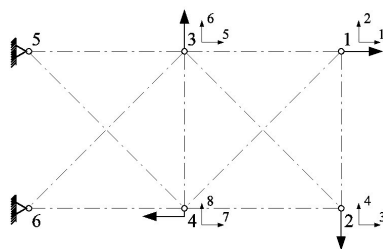
## 6 Numerical examples

Examples 1 and 2 are applications of the present algorithm to problems  $(TTD)$  and  $(RTT)$ , respectively. In example 3 we apply the present algorithm to solve the problems  $(MV)$ ,  $(MC)$  and  $(MF)$ . The structures of examples 1 and 2 are taken from [7] and the example 3 is from [1]. In all numerical tests, the optimization process stops when  $\|d_0\| < 10^{-6}$ .

Recall that all problems solved here are first converted to the general format (1). Then,  $n$  is the dimension of  $x$  and represents the number of design variables and  $m \times m$  is the size of  $G(x)$  the constraint matrix.

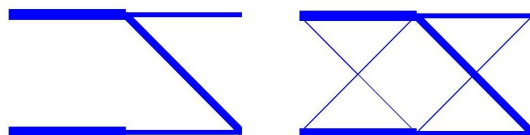
*Example 1.* Consider the ground structure as shown in figure 1. In this example a single “primary” load  $P = \{p\}$  is considered and the magnitude of each component of the load  $p$  is 2.0. The length of each of the horizontal and vertical bars is 1.0. The secondary loadings have a magnitude  $r = 0.4$  and define a basis of the orthogonal complement of the linear span of  $P$  in the linear space of all the degrees of freedoms of the structure.

Figure 2-left shows the final topology obtained with model  $(TTD)$ . The resulting truss is unstable since, for example, arbitrarily small non-horizontal force



**Fig. 1** Truss of Example 1.

applied to node 4 will produce infinite compliance. On the other hand, figure 2-right shows the final topology obtained with the robust model ( $RTT$ ).



**Fig. 2** Results of example 1. Left= $(TTD)$ . Right= $(RTT)$ .

Table 1 shows results obtained with the present algorithm. The row  $n_1 : n_2$  is the volume percentage of the bar connecting node  $n_1$  and  $n_2$ . Vertical bars 3 : 4 and 1 : 2 are omitted since their volumes are null.

**Table 1** Results of example 1. Bar volumes (%).

	$TTD$	$RTT$
3 : 5	25	24.482
1 : 3	12.5	11.954
4 : 6	25	24.483
2 : 4	12.5	11.954
4 : 5	0	1.2644
3 : 6	0	1.2644
2 : 3	25	23.679
1 : 4	0	0.9196

The results presented in figure 2 and in table 1 are similar to the results shown in [7].

*Example 2.* This example consists of a three-dimensional truss with fixed nodes on the horizontal plane  $z = 0$  and free nodes on the horizontal plane  $z = 2$ . The



structure has 8 nodes of coordinates,

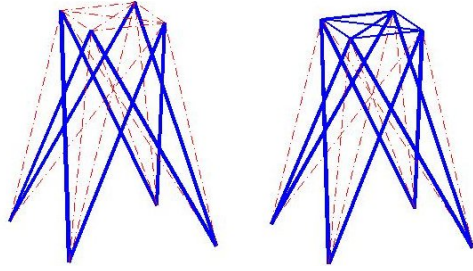
$$\begin{aligned} & \begin{bmatrix} \cos(2\pi i/4) \\ \sin(2\pi i/4) \\ 0 \end{bmatrix}, \quad i \in \{1, 2, 3, 4\}, \\ & \begin{bmatrix} \frac{1}{2} \cos(2\pi j/4) \\ \frac{1}{2} \sin(2\pi j/4) \\ 2 \end{bmatrix}, \quad j \in \{5, 6, 7, 8\}, \end{aligned} \quad (85)$$

All the possible edges between free-free or free-fixed nodes are considered. A single load case  $P = \{p\}$  is defined. The components of  $p$  acting at the nodes on the plane  $z = 2$  are given by,

$$p_j = \frac{1}{\sqrt{4(1+\rho^2)}} \begin{bmatrix} \sin(2\pi j/4) \\ -\cos(2\pi j/4) \\ -\rho \end{bmatrix}, \quad j \in \{5, 6, 7, 8\}, \quad (86)$$

where  $\rho = 0.001$ . The secondary loadings were built in the same way as in example 1.

Figure 3-left and right shows the optimal result with model  $(TTD)$  and  $(RTT)$ , respectively. The resulting structures of figure 3 differs only in the thickness of the “top” horizontal bars. The truss in figure 3-left has bars approximately 80 times smaller in volume than the right one. This result is also obtained in [7]



**Fig. 3** Results of example 2. Left= $(TTD)$ . Right= $(RTT)$ .

Table 2 summarizes the executions of the present algorithm with example 1 and 2. The rows  $n$ ,  $m$  and  $iter$  are the number of design variables, the size of the constraint matrix  $G(x)$  and the number of iterations, respectively. The resulting structure in both examples attains the maximum volume of  $\bar{V} = 1.0$ .

**Table 2** Results of example 1 and 2.

example	model	$n$	$m$	iter
1	$TTD$	11	20	22
	$RTT$	11	27	28
2	$TTD$	23	36	19
	$RTT$	23	47	21

*Example 3.*

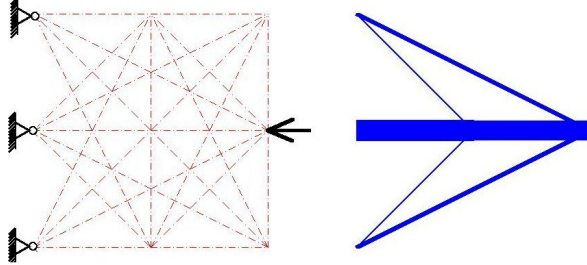
Consider the planar structure with  $3 \times 3$  nodes in figure 4-left. The nodes in the left side are fixed in all directions and a horizontal force of magnitude 1.0 is applied in the middle node of the right side. The length of each of the horizontal and vertical bars is 1.0 and the Young's module of the material is 1.0. The initial structure has all bars with sectional area of 0.01.

We consider the minimum volume problem (MV) with  $\bar{\gamma} = 1$  and  $\underline{\lambda} = 0.05$ . Our algorithm obtains the optimal design  $(x^*, V^*)$  with  $V^* = 4.73$ . The optimal design  $x^*$  shown in figure 4-right. Since the optimal structure found is symmetric, only four components of  $x^*$  are non-null and different each other. Those components are 1.58318, 1.48329,  $1.57589 \times 10^{-1}$  and  $6.75687 \times 10^{-1}$ .

Considering the minimum compliance problem (MC) with  $\bar{V} = V^* = 4.731$  and  $\underline{\lambda} = 0.05$ , we obtain the optimal design  $(x^*, \gamma^*)$  with optimal compliance  $\gamma^* = \bar{\gamma} = 1$ .

Finally, when solving the problem of maximization the minimum eigenvalue (MF) with  $\bar{V} = V^* = 4.731$  and  $\bar{\gamma} = 1$ , we obtain the optimal design  $(x^*, \lambda^*)$  with  $\lambda^* = \underline{\lambda} = 0.05$ .

This result is also obtained in example 4.1 of [1].



**Fig. 4** Truss of example 3. Our algorithm obtains the same  $x^*$  when applied to problems (MV), (MC) and (MF).

*Test problems from SDPLIB and COMPl<sub>elib</sub>*

Here we present some results with test problems of SDPLIB [11]. SDPLIB is a collection of linear semidefinite programming test problems. In this case the constraint matrix  $G(x)$  is an affine combination of symmetric matrices and the objective function  $f(x)$  is linear.

The test problems do not provide feasible points, then we first obtain a feasible point by solving the following auxiliary problem,

$$\begin{aligned} \min_{x \in \mathbb{R}^n, z \in \mathbb{R}} \quad & z \\ \text{s.t.} \quad & G(x) - zI \preceq 0 \end{aligned} \tag{87}$$

where  $z$  is a new variable. Making iteration with our algorithm to solve (87), a feasible point  $x_0$  is obtained once  $z$  becomes negative. Then,  $x_0$  is taken as the starting point to solve (1).

The results presented in table 3, were solved by our algorithm using the same default parameters indicated in section 4. The algorithm stops when  $\|d_0\| < 10^{-6}$ . In column *iter* we write, *iter1* + *iter2*, where *iter1* is the number of iterations to

solve (87) to obtain feasible point  $x_0$  and  $iter2$  is the number of iterations to solve problem (1) starting with  $x_0$ . Finally, in column  $f$  is the value of the objective function obtained.

**Table 3** Numerical results, *SDPLIB*.

<i>Problem</i>	<i>n</i>	<i>m</i>	<i>iter</i>	<i>f</i>
<i>control1</i>	21	15	5 + 18	17.7848
<i>control2</i>	66	30	6 + 30	8.3001
<i>control3</i>	136	45	5 + 38	13.6334
<i>control4</i>	231	60	6 + 27	19.7944
<i>hinf1</i>	13	14	6 + 39	2.0326
<i>hinf2</i>	13	16	15 + 57	10.9671
<i>hinf3</i>	13	16	22 + 39	56.9425
<i>hinf4</i>	13	16	17 + 22	274.766
<i>hinf5</i>	13	16	25 + 52	362.2963
<i>hinf6</i>	13	16	18 + 37	448.9428
<i>hinf7</i>	13	16	8 + 41	390.8164
<i>hinf8</i>	13	16	14 + 405	116.1638
<i>hinf9</i>	13	16	18 + 38	236.2511
<i>hinf10</i>	21	18	12 + 41	108.7538
<i>hinf11</i>	31	22	15 + 66	65.899
<i>hinf12</i>	43	24	4 + 110	2.0251
<i>hinf14</i>	73	34	18 + 137	12.9927
<i>qap5</i>	136	26	6 + 20	-435.9962
<i>qap6</i>	229	37	3 + 207	-381.4346
<i>theta1</i>	104	50	3 + 12	23.0002
<i>truss1</i>	6	13	4 + 15	-9
<i>truss3</i>	27	31	5 + 32	-9.1099
<i>truss4</i>	12	19	4 + 16	-9.0099

Looking at table 3, we can see that the numbers of iterations are comparable to the size of the problem (except for *hinf8*, *hinf14* and *qap6*) and the value of the objective function is similar to the values published in [11].

## 7 Conclusions

In this paper, a new approach for nonlinear semidefinite programming is presented and supported by strong theoretical results. In particular, global convergence was proved for the convex case.

The present technique computes feasible and descent directions based on Newton-like iterations to solve KKT optimality conditions.

To obtain a search direction only two linear system with the same coefficient matrix are solved. The present technique is very robust and does not require tuning of parameters w.r.t. the considered application.

A set of numerical tests was performed with the same set of parameters.

Tests on non-convex problems show that the algorithm is still convergent.

## 8 Acknowledgments

The authors thank the Brazilian Research Councils CAPES, CNPq, FAPERJ, the institutions supporting the program Ciência Sem Fronteiras of Brazil and the French Research Councils CNRS and INRIA for the financial support.

## References

1. Achtzinger, A., Kočvara, M.: Structural topology optimization with eigenvalues. Tech. rep., University of Dortmund (2006)
2. Achtzinger, W., Kočvara, M.: On the maximization of the fundamental eigenvalue in topology optimization. *Structural and Multidisciplinary Optimization* **34**, 181–195 (2007)
3. Alizadeh, F.: Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization* **5**, 13–51 (1995)
4. Alizadeh, F., Haberly, J.P., Nayakkankuppam, M.V., Overton, M.L., Schmieta, S.: Sdppack user's guide - version 0.9 beta. Tech. rep., Courant Institute of Mathematical Science, NYU, New York, NY (1997)
5. Alizadeh, F., Haberly, J.P.A., Nayakkankuppam, M.V., Overton, M.L., Schmieta, S.: Sdppack version 0.9 beta for matlab 5.0. <http://cs.nyu.edu/overton/software/sdppack/sdppack.html> (1997). URL <http://cs.nyu.edu/overton/software/sdppack/sdppack.html>
6. Ben-Tal, A., Nemirovski, A.: Potential reduction polynomial time method for truss topology design. *Siam Journal of Optimization* **4**, 596–612 (1994)
7. Ben-Tal, A., Nemirovski, A.: Robust truss topology design via semidefinite programming. *SIAM Journal on Optimization* **7**(4), 991–1016 (1997)
8. Ben-Tal, A., Nemirovski, A.S.: Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications. MPS/SIAM Series on Optimization (2001)
9. Bendsoe, M.P.: Optimization of structural topology, shape, and material. Springer-Verlag (1995)
10. Bonnans, F., Shapiro, A.: Perturbation Analysis of Optimization Problems. Springer Verlag (2000)
11. Borchers, B.: Sdplib 1.2, a library of semidefinite programming test problems. *Optimization Methods and Software* **11**, 683–690 (1999)
12. Boyd, S., El Ghaoui, L., Feron, E., V., B.: Linear matrix inequalities in system and control theory. SIAM Studies in Applied Mathematics, SIAM, Philadelphia, USA (1994)
13. Correa, R., Ramirez, C.H.: A global algorithm for nonlinear semidefinite programming. *SIAM Journal on Optimization* **15**, 303–318 (2004)
14. Fares B. Noll, D., Apkarian, P.: Robust control via sequential semidefinite programming. *SIAM Journal on Control and Optimization* **40**, 17911820 (2002)
15. Fletcher, R.: A nonlinear programming problem in statistics (educational testing). *SIAM Journal on Scientific Computing* **2**, 257267 (1981)
16. Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM* **79**, 11151145 (1995)
17. Herskovits, J.: Developement of a numerical method for nonlinear optimization. Ph.D. thesis, Paris IX University, INRIA Rocquencourt (1982)
18. Herskovits, J.: A two-stage feasible directions algorithm for nonlinear constrained optimization. *Mathematical Programming* **36**, 19–38 (1986)
19. Herskovits, J.: A feasible directions interior point technique for nonlinear optimization. *Journal of Optimization Theory and Applications* **99**, 121–146 (1998)
20. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press (1985)
21. Jarre, F.: An interior method for nonconvex semidefinite programs. *Optimization and Engineering* **1**, 347–372 (2000)
22. Kanzow, C., Nagel, C., Kato, H., Fukushima, M.: Successive linearization methods for nonlinear semidefinite programs. *Computational Optimization and Applications* **31**, 251–273 (2005)
23. de Klerk, E.: Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications. Kluwer (2002)

24. Kočvara, M., Stingl, M.: Free material optimization for stress constraints. *Structural and Multidisciplinary Optimization* **33**, 323–335 (2007)
25. Lewis, A.S., Overton, M.L.: Eigenvalue optimization. *Acta Numerica* **5**, 149190 (1996)
26. Lima, E.L.: *Anlise Real*, vol. 2, 3 edn. Coleo Matemtica Universitria (2007)
27. Luenberger, D.G., Ye, Y.: *Linear and Nonlinear Programming*, 3 edn. Springer (2008)
28. M. Fukuda M. Kojima, K.M., Nakata, K.: Exploiting sparsity in semidefinite programming via matrix completions i: General framework. *SIAM Journal of Optimization* **11**, 647674 (2000)
29. Panier, E.R., Tits, A.L., Herskovits, J.: A qp-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization. *SIAM Journal of Control and Optimization* **26**, 788–810 (1988)
30. Powell, M.J.D.: The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations, in *Nonlinear Programming 3*. Academic Press, London (1978)
31. Shapiro, A.: First and second order analysis of nonlinear semidefinite programs. *Mathematical Programming* **77**, 301–320 (1994)
32. Shapiro, A.: On uniqueness of lagrange multipliers in optimization problems subject to cone constraints. *SIAM Journal on Optimization* **7**, 508 518 (1997)
33. Stingl, M.: On the solution of nonlinear semidefinite programs by augmented lagrangian methods. Ph.D. thesis, University of Erlangen-Nrnberg (2005)
34. STINGL, M., KOCVARA, M., LEUGERING, G.: A sequential convex semidefinite programming algorithm with an application to multiple-load free material optimization. *SIAM Journal on Optimization* **2009**, 130–155
35. Stingl, M., Kočvara, M., Leugering, G.: Research report 319: Free material optimization with fundamental eigenfrequency constraints. Tech. rep., Institute of Applied Mathematics, University of Erlangen (2007)
36. Todd, M.J.: Semidefinite optimization. *Acta Numerica* **10**, 515–560 (2001)
37. Zowe, J., Kocvara, M., Bendsoe, M.P.: Free material optimization via mathematical programming. *Mathematical Programming* **79**, 445–466 (1997)
38. Zowe, J., Kocvara, M., Bendsoe, M.P.: Free material optimization via mathematical programming. the multi-load case with contact conditions. *SIAM,J. Optimization* **9**, 813–832 (1997)