HAL
open science

A feasible direction interior point algorithm for nonlinear semidefinite programming

Miguel Aroztegui, José Herskovits, Jean Rodolphe Roche, Elmer Bazán

To cite this version:

Miguel Aroztegui, José Herskovits, Jean Rodolphe Roche, Elmer Bazán. A feasible direction interior point algorithm for nonlinear semidefinite programming. Structural and Multidisciplinary Optimization, 2014, 50 (6), pp.1019-1035. 10.1007/s00158-014-1090-2 . hal-01274022

HAL Id: hal-01274022

https://hal.science/hal-01274022v1

Submitted on 15 Feb 2016

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 · Elmer Bazán

Abstract We present a new algorithm for nonlinear semidefinite programming, based on the iterative solution in the primal and dual variables of Karush-Kuhn-Tucker optimality conditions, which generates a feasible decreasing sequence. At each iteration, two linear systems with the same matrix are solved to compute a feasible descent direction and then an inexact line search is performed in order to determinate the new iterate. Feasible iterates are essential in applications where feasibility is required to compute some of the involved functions. A proof of global convergence to a stationary point is given. Several numerical tests involving nonlinear programming problems with linear or nonlinear matrix inequality constraints 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 optimization, semidefinite programming, feasible directions, interior-point methods.

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, Elmer Bazán
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, elmer@optimize.ufrj.br

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

1 Introduction

This paper proposes a new technique to solve the following nonlinear semidefinite programming problem (NSDP),

$$\min_x f(x) \text{ s.t. } x \in \mathbb{R}^n \text{ and } G(x) \preccurlyeq 0 \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $G : \mathbb{R}^n \to \mathbb{S}^m$ are smooth functions, not necessarily linear. We denote $\mathbb{S}^m$ the set of real symmetric matrices of size $m \times m$. $G(x) \preccurlyeq 0$ means that the matrix $G(x)$ is negative semidefinite. We call $\Omega = \{x \in \mathbb{R}^n; G(x) \preccurlyeq 0\}$ the set of feasible solutions and $int(\Omega)$, its interior.We assume that $int(\Omega_a) \neq \emptyset$ then equality constraint cannot be handled.

Applications in a wide range of disciplines lead to semidefinite programming problems, (SDP). For example, combinatorial optimization [3], nonconvex quadratic programming [16], eigenvalue optimization [28], systems control theory [12], matrix completion and statistics problems [33],[15] and applications in structural design [2]. In particular, SDP formulations were employed in free material structural mechanical design, see for example [46], [27], [40], [39] and [47].

In the case of linear SDP, when f is linear and G is an affine function, problem (1) is convex and several 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, [42], a deep overview is presented.

A crucial result for semidefinite programming is the characterization of Karush - Kuhn - Tucker (KKT) optimality conditions presented by Shapiro, [36] and [37]. Extensions to the nonlinear case of interior points methods are an important class of algorithms for NSDP

problems. We mention the primal predictor-corrector interior point method of Jarre [22] and the primal-dual interior point algorithms described in [44], [43] and [5].

The sequential linear SDP method is an another important class of techniques introduced to solve NSDP problems. We mention the approach of Correa and Ramirez [13], that is a generalization of the algorithm presented by Fares, Noll and Apkarian [14]. Also Kanzow, Nagel, Kato and Fukushima [24], presented a successive linearization method with a trust region-type globalization. In [23], successive linearization is applied to solve robust design of structures. In [29], the authors combine successive linearization with filter techniques.

A computer code based on an augmented Lagrangian approach was developed by Kočvara and Stingl [38] [26]. In [32] the authors analyze different types of augmented Lagrangian method and the convergence has been proved for ϵ-global solutions. In [41], Sun, Sun, and Zhang, have proved local convergence with a linear rate of the augmented Lagrangian method applied to NSDP problems. A new homotopy method for NSDP problems has been developed recently by Yang and Yu, [45].

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 [18] [19] [34] [20]. The proposed algorithm, at each interior point, defines first a descent direction that is also feasible with respect to the semidefinite constraints. Then, it makes a line search in that direction to obtain a new interior point with a lower objective. Feasible iterates are essential in the case when the objective function or some constraints are not defined at infeasible points. In structural optimization the stiffness and mass matrices, as well as the material matrices, must be positive definite to compute the mechanical constraints. Newton, quasi - Newton or first order versions of the present algorithm can be obtained.

The paper is organized as follows. Some notation and basic concepts are described in the next section. In section 3 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 the presented algorithm is described. Global convergence of the proposed optimization algorithm to a stationary point is proved in section 4. Details of the implementation are shown in section 5. Numerical examples are presented in section 6.4. Finally, the last section is dedicated to conclusions.

2 Notation and basic concepts

2.1 Notation

Let $\mathbb{R}^{m\times n}$ denote the space of $m \times n$ real matrices and $\mathbb{S}^n$, the space of real symmetric matrices. The sets of symmetric positive semidefinite and positive definite matrices of size $m \times m$ are denoted $\mathbb{S}^m_+$ and $\mathbb{S}^m_{++}$, respectively. Negative semidefinite and definite matrices are defined in a similar way.

The symbol $\preccurlyeq$ refers to a partial order on the negative semidefinite matrices, that is, $A \preccurlyeq B$ means that $A - B$ is negative semidefinite, [17]. Similarly, the symbol $\prec, \succcurlyeq$ and $\succ$ refers to a partial order on the negative definite, positive semidefinite and positive definite matrices, respectively.

The symmetric part of $M \in \mathbb{R}^{n\times n}$ is indicated by $sym(M)$ and the skew part as $skw(M)$. 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 called 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 $\overline{m} = \frac{1}{2}m(m + 1)$ is the number of upper diagonal elements of A. To introduce the symmetric Kronecker product, we define the two following maps:

$$svec : \mathbb{S}^m \to \mathbb{R}^{\overline{m}}$$
$$svec(A) = \left[a_{11}\ \sqrt{2}a_{12}\ a_{22}\ \sqrt{2}a_{13}\ \sqrt{2}a_{23}\ a_{33}\ ...\ a_{mm}\right]^\top$$

and $smat : \mathbb{R}^{\overline{m}} \to \mathbb{S}^m$ the inverse of $svec$, see [25].

Then, the inner product

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

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

$$(A \circledast B)svec(C) = svec\left(sym\left(BCA\right)\right) \tag{2}$$

In [25] it is proved that,

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

(3)

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

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

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

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

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

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

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

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

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

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

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

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

where $\lambda = svec(\Lambda)$.

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

$$\nabla_x L(x,\lambda) = \nabla f(x) + \nabla G(x)\lambda, \tag{7}$$

see [37] and [10].

2.2 Definitions

The following three definitions are related to the first order optimality conditions for semidefinite programming introduced in [36] and [37].

We call $\{b_1(x),\ldots,b_p(x)\}$ an orthonormal basis of $kerG(x)$ and let $E_0(x) = [b_1(x)\ldots b_p(x)]$ a matrix belonging to $\mathbb{R}^{m\times p}$.

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

$$\left\{ \begin{bmatrix} b_i^\top \dfrac{\partial G}{\partial x_1}(x) b_j \\ \vdots \\ b_i^\top \dfrac{\partial G}{\partial x_n}(x) b_j \end{bmatrix} \textit{ such that } i \leqslant j,\ i,j = 1,\ldots,p \right\}$$

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 conditions are verified:

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

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

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

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

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

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 $t \in (0,\delta]$.

2.3 Some technical results

Some results that will be employed latter are presented now.

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

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

Proof The proof for $AB = 0 \iff tr(AB) = 0$ can be found in [38].

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

($\Rightarrow$) Obvious.

($\Leftarrow$) By hypothesis, $sym(AB) = 0$ and since $tr(skw(AB)) = 0$, the trace of AB is null. Then, thanks to the first equivalence, it is $AB = 0$. □

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

The proof of the following lemma is similar as the previous one.

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

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 [21].*

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

$$y^\top ABy = 0 \Longleftrightarrow By = 0.$$

Proof It follows from lemma 4 that the matrix $AB \in S^m_{-}$. Then there exist $\lambda_1, \ldots, \lambda_m$, m non-positive eigenvalues of AB, and a complete set of orthonormal eigenvectors $\{b_1, \ldots, b_m\}$, a base of $\mathbb{R}^m$, see [21]. Let y be a not null vector in $\mathbb{R}^m$ such that $y^\top(AB)y = 0$, then there exists a set $\{\alpha_1, \ldots, \alpha_m\}$ such that:

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

In consequence,

$$0 = y^\top(AB)y = \langle y, (AB)y\rangle \tag{10}$$

$$= \sum_{j=1}^{m}\sum_{i=1}^{m} \alpha_i\alpha_j\lambda_i \langle b_i, b_j\rangle = \sum_{i=1}^{m} \alpha_i^2\lambda_i \tag{11}$$

Since, $y \neq 0$, there exist i_0 such that $\alpha_{i_0} \neq 0$. In view of the last equation, λ_{i_0} must be null. This is true for all λ_i such that $\alpha_i \neq 0$, therefore $y \in \ker(AB)$, but $A \in \mathbb{S}^m_{++}$ then $y \in \ker(B)$. □

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

$$AB^2 = 0 \Longleftrightarrow sym(AB) = 0$$

Proof **(⇒):** $A \in \mathbb{S}^m_{-}$ then there exist $Q \in \mathbb{S}^m$ such that $A = -QQ$.
Since $0 = tr(AB^2) = tr(BAB) = -tr(BQQB) = -\|BQ\|^2$, then $BQ = 0$ and $-BQQ = BA = 0$.

(⇐): We have $AB + BA = 0$, then $AB^2 = -BAB$. It follows that A and B^2 are diagonalizable simultaneously and $AB^2 \preccurlyeq 0$. On the other hand, $-BAB = B(-A)B \succcurlyeq 0$. Therefore, $0 \preccurlyeq -BAB = AB^2 \preccurlyeq 0$ and we conclude that $AB^2 = 0$. □

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

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

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

$$A = PD_AP^{-1}$$
$$B = PD_BP^{-1}$$

In view of symmetric Kronecker product properties, see for example lemma E.1.2 in [25] :

$$A \circledast I = \mathsf{P}\mathsf{D}_A\mathsf{P}^{-1}$$
$$B \circledast I = \mathsf{P}\mathsf{D}_B\mathsf{P}^{-1}$$

where $\mathsf{P} = P \circledast P$, $\mathsf{D}_A = D_A \circledast I$ and $\mathsf{D}_B = D_B \circledast I$. D_A and D_B are diagonal matrices and P is orthonormal. Therefore

$$(A \circledast I)^{-1}(B \circledast I) = \mathsf{P}\mathsf{D}_A\mathsf{D}_B\mathsf{P}^\top \in \mathbb{S}^{\overline{m}}_{-}$$

□

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

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

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

$$A = [E_0\ E_\perp]\begin{bmatrix} D_A & 0 \\ 0 & 0\end{bmatrix}[E_0\ E_\perp]^\top = E_0D_AE_0^\top$$

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, $\langle A, B\rangle = tr(E_0\theta E_0^\top B) = 0$ because $E_0^\top B = 0$. Then, $A \in C_1$. □

3 Description of the algorithm

The basic ideas involved in the present algorithm are described next. Most of the assertions are a motivation for our technique and will be proved in the following section.

For a real number a, we denote

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

and introduce the following assumptions about f and G:

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

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

Assumption 3 f and G are C^1 in Ω_a and ∇f and $\dfrac{\partial G}{\partial x_p}$ for $p = 1, ..., n$ 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, Λ) 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.

Several approaches have been proposed to handle the complementarity condition $\Lambda G(x) = 0$, see [4], [5] and [42]. Since the product of two symmetric matrices in general is not symmetric, instead of $\Lambda G(x) = 0$ we consider $sym\,(\Lambda G(x)) = 0$. In the following section we show that our algorithm generates a sequence (x^k, Λ^k) converging to (x^*, Λ^*) such that $sym\,(\Lambda^* G(x^*)) = 0$ and that this implies $\Lambda^* G(x^*) = 0$.

Then, the stationary point conditions (8) can be written in the following form:

$$\begin{aligned} \nabla f(x) + \nabla G(x)\lambda &= 0 \\ svec\,(sym\,(\Lambda G(x))) &= 0 \end{aligned} \tag{12}$$

where $\lambda = svec\,(\Lambda)$.

In order to obtain the linear system to be solved at each Newton iteration we define a vectorial function $\psi\colon \mathbb{R}^{n+\overline{m}} \to \mathbb{R}^{n+\overline{m}}$, given by the equalities in (12)

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

Using the Kronecker product we observe that

$$\psi_c(x, \lambda) = svec(sym(I\Lambda G(x)) = [I \circledast G(x)]svec(\Lambda)$$

and also,

$$\psi_c(x, \lambda) = svec(sym(\Lambda G(x)I) = [\Lambda \circledast I]svec(G(x))$$

therefore, the Jacobian of ψ is:

$$\begin{aligned} \nabla\psi(x, \lambda) &= \begin{bmatrix} \nabla_x \psi_l(x, \lambda) & \nabla_\lambda \psi_l(x, \lambda) \\ \nabla_x \psi_c(x, \lambda) & \nabla_\lambda \psi_c(x, \lambda) \end{bmatrix} \\ &= \begin{bmatrix} \nabla_{xx} L(x, \lambda) & \nabla G(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top & I \circledast G(x) \end{bmatrix} \end{aligned} \tag{13}$$

In a similar way as in [20], instead of the Hessian of the Lagrangian $\nabla_{xx} L(x, \lambda)$, we can employ a positive definite matrix denoted B. This matrix B can be a quasi-Newton approximation, or even the identity matrix.

A Newton like iteration to solve (12) is given by the following linear system

$$\begin{aligned} \begin{bmatrix} B & \nabla G(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top & I \circledast G(x) \end{bmatrix} \begin{bmatrix} x_0 - x \\ \lambda_0 - \lambda \end{bmatrix} \\ = - \begin{bmatrix} \nabla f(x) + \nabla G(x)\lambda \\ svec(sym(\Lambda G(x))) \end{bmatrix} \end{aligned} \tag{14}$$

where $(x, \Lambda) \in int(\Omega_a) \times \mathbb{S}^m_{++}$ is the current point and $(x_0, \lambda_0) \in \mathbb{R}^n \times \mathbb{R}^{\overline{m}}$ are the new estimates given by the Newton like iteration. Note that $\lambda = svec\,(\Lambda)$ and $\lambda_0 = svec\,(\Lambda_0)$. We also denote $W(x, B, \Lambda)$ the matrix of the system (14).

Instead of adopting (x_0, λ_0) we introduce a line search in the primal space and an appropriate updating rule for the dual variables.

Let be $d_0 = x_0 - x$. Then we have,

$$\begin{aligned} Bd_0 + \nabla G(x)\lambda_0 &= -\nabla f(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top d_0 + (I \circledast G(x))\,\lambda_0 &= 0 \end{aligned} \tag{15}$$

If $d_0 = 0$ equation (15) becomes,

$$\nabla f(x) + \nabla G(x)\lambda_0 = 0 \tag{16}$$

$$(I \circledast G(x))\,\lambda_0 = 0 \tag{17}$$

Since $G(x) \prec 0$, the matrix $I \circledast G(x)$ is nonsingular and we have $\Lambda_0 = smat(\lambda_0) = 0$. This proves that $\Lambda_0 G(x) = 0$.

Therefore, $\nabla f(x) = 0$ and x is a stationary point of the problem. If $d_0 \neq 0$, since B and Λ are positive definite, we prove that d_0 is a descent direction of the objective function. However we cannot ensure that d_0 is a feasible direction. In effect, when x is on the boundary of Ω, it follows from (15) that d_0 is tangent to Ω. Thus, depending on the curvature of the boundary of Ω, d_0 can point outwards of the feasible domain.

To obtain a feasible direction, as in [20], we modify the previous linear system introducing an appropriate right hand side:

$$\begin{aligned} Bd + \nabla G(x)\overline{\lambda} &= -\nabla f(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top d + (I \circledast G(x))\,\overline{\lambda} &= -\rho\lambda \end{aligned} \tag{18}$$

where ρ is a positive real number and $\lambda = svec(\Lambda)$.

In view of the equations (2) and (6), the second equation of (18) is equivalent to

$$sym\left(\Lambda DG(x)d + \overline{\Lambda}G(x)\right) = -\rho\Lambda \tag{19}$$

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

In the next section we prove that in fact d constitutes a feasible directions field in the sense of definition 5.

We introduce now the following assumption on Λ.

Assumption 5 At each iteration the current values of Λ and $G(x)$ commute.

The pair $(d, \overline{\lambda})$ obtained by the linear system (18) can also be computed solving

$$\begin{aligned} Bd_1 + \quad \nabla G(x)\lambda_1 &= 0 \\ (\Lambda \circledast I)\,\nabla G(x)^\top d_1 + (I \circledast G(x))\,\lambda_1 &= -\lambda \end{aligned} \tag{20}$$

and taking,

$$d = d_0 + \rho d_1 \tag{21}$$

$$\overline{\lambda} = \lambda_0 + \rho\lambda_1. \tag{22}$$

The descent direction d_0 verifies $d_0^\top \nabla f(x) < 0$. For a given $\xi \in (0,1)$, we get an upper bound for ρ such that

$$d^\top \nabla f(x) \leqslant \xi d_0^T \nabla f(x). \tag{23}$$

Consequently, the feasible direction d will be a descent direction also. In fact, if $d_1^\top \nabla f(x) > 0$, we take

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

Otherwise, we choose

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

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.

We shall prove global convergence to a stationary point, for any way of updating B and Λ, provided they are positive definite and Λ satisfies assumption 5.

3.1 The statement of the algorithm

In the following we state precisely the present algorithm for semidefinite programing:

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

Initial data. $x \in int(\Omega_a)$, $\Lambda \in \mathbb{S}^m_{++}$ commuting with $G(x)$ and $B \in \mathbb{S}^n_{++}$.

Step 1. Computation of the search direction d.

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

$$\begin{bmatrix} B & \nabla G(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top & I \circledast G(x) \end{bmatrix} \begin{bmatrix} d_0 \\ \lambda_0 \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix} \tag{24}$$

If $d_0 = 0$, stop.

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

$$\begin{bmatrix} B & \nabla G(x) \\ (\Lambda \circledast I)\,\nabla G(x)^\top & I \circledast G(x) \end{bmatrix} \begin{bmatrix} d_1 \\ \lambda_1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\lambda \end{bmatrix} \tag{25}$$

(iii) Compute the parameter ρ such that

$$\rho = \min\left\{\varphi\|d_0\|^2, (\xi - 1)\frac{d_0^\top \nabla f(x)}{d_1^\top \nabla f(x)}\right\} \tag{26}$$

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

$$\rho = \varphi\|d_0\|^2. \tag{27}$$

(iv) Compute the search direction d as

$$d = d_0 + \rho d_1. \tag{28}$$

Step 2. Line Search.

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

$$f(x + td) \leqslant f(x) + t\eta d^\top \nabla f(x) \tag{29}$$

and

$$G(x + td) \prec 0. \tag{30}$$

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 [31] can be also employed.

We introduce now the following assumptions on the way that B and Λ are updated in **Step 3.**

Assumption 6 There exist positive numbers λ^I and λ^S such that

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

Assumption 7 There exist positive numbers σ_1 and σ_2 such that

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

4 Global convergence

In this section we prove that for any initial point $x^0 \in int(\Omega)$, the present algorithm generates a sequence $\{(x^k, \Lambda_0^k)\}$ converging to a stationary point of problem (1), (x_0^*, Λ_0^*) . Assumptions 1 to 7 previously introduced are supposed to be satisfied.

First we prove that the algorithm is well defined and in particular that the matrix $W(x, B, \Lambda)$ given in (14) is nonsingular. Then it is shown that at each iteration d_0 and d are descent directions of f at x and $d(x)$ constitutes an uniformly feasible directions field in Ω_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 Λ and $G(x)$ commute. Then, the matrix $W(x, B, \Lambda)$ defined in (14) is nonsingular.*

Proof We have to prove that, if $W(x, B, \Lambda)v = 0$ for some $v \in \mathbb{R}^{n+\overline{m}}$, then $v = 0$.

Let $v^\top = [r^\top,\ y^\top]$, $r \in \mathbb{R}^n$, $y \in \mathbb{R}^{\overline{m}}$ such that $W(x, B, \Lambda)v = 0$. Since $B \in \mathbb{S}^n_{++}$ and using block Gaussian elimination we have:

$$r = -B^{-1}\nabla G(x)y \tag{31}$$

$$My = 0 \tag{32}$$

where M is the Schur complement:

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

To finish the proof, we must conclude that M is nonsingular. But M is symmetric, then it will be enough to show that M is positive definite.

We have

$$y^\top My = y^\top \left(\nabla G(x)^\top B^{-1}\nabla G(x)\right) y - y^\top(\Lambda \circledast I)^{-1}(G(x) \circledast I)y \tag{33}$$

Since B^{-1} is positive definite,

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

By hypothesis and lemma 7

$$-y^\top(\Lambda \circledast I)^{-1}(G(x) \circledast I)y \geqslant 0, \tag{35}$$

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

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

$$y^\top \nabla G(x)^\top B^{-1}\nabla G(x)y = 0 \tag{36}$$

and

$$y^\top(\Lambda \circledast I)^{-1}(G(x) \circledast I)y = 0. \tag{37}$$

Since B is positive definite, from equation (36) we have

$$\nabla G(x)y = 0. \tag{38}$$

Due to lemma 5 and from equation (37),

$$(G(x) \circledast I)y = 0. \tag{39}$$

By definition of $\nabla G(x)$, equation (38) is equivalent to

$$\left\langle \frac{\partial G}{\partial x_i}(x), Y \right\rangle = 0, \quad i = 1, \ldots, n \tag{40}$$

where $Y = smat(y)$.

Since $(G(x) \circledast I)y = svec(sym(G(x)Y))$, in view of lemma 6 and proposition 1, equation (39) can be rewritten as

$$\langle Y^2, G(x)\rangle = 0 \tag{41}$$

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

$$Y^2 = E_0\theta E_0^\top,\ \theta \in \mathbb{S}^p,\ \theta \succcurlyeq 0 \tag{42}$$

where $E_0 = [b_1, \ldots, b_p] \in \mathbb{R}^{m\times p}$ and the set of vectors $\{b_1, \ldots, b_p\}$ is an orthonormal base of $\ker(G(x))$. Taking the square root of Y^2 in (42), we obtain

$$Y = E_0\theta^{1/2}E_0^\top,\ \theta^{1/2} \in \mathbb{S}^p \tag{43}$$

Replacing Y by $E_0\theta^{1/2}E_0^\top$ in equation (40), follows

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

which is equivalent to

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

and also to,

$$V svec(\theta^{1/2}) = 0 \tag{44}$$

where

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

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

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

with $c = \sqrt{2}$ when $i = j$ and $c = 1$ when $i \neq j$ where $i, j = 1, \ldots, p$ and $i \leqslant j$. By hypothesis, x is a regular point of problem (1), then the columns of V are linearly independent. Therefore, the linear system (44) has the unique solution $svec(\theta^{1/2}) = 0$. Consequently $Y = smat(y) = E_0 \theta^{1/2} E_0^\top = 0$ and M is positive definite in Ω_a. □

Since Ω_a, Λ and B are bounded, it follows from theorem 1 that d_0, λ_0, d_1 and λ_1 are also bounded.

When $d_0 = 0$ is obtained in step 1, the algorithm stops. In fact, since $G(x) \prec 0$, $G(x) \circledast I \prec 0$ then the solution of (24) 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*

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

Proof Multiplying the first equation of (15) 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 (15),

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

Then,

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

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

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

□

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

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

$$d^\top \nabla f(x) \leqslant \xi d_0^\top \nabla f(x). \tag{45}$$

Proof In view of (28),

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

If $d_1^\top \nabla f(x) > 0$, using (26),

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

If $d_1^\top \nabla f(x) \leqslant 0$, using (27),

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

Both cases verifies (45) 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 ρ computed by the algorithm verifies:*

$$\varphi_0 \|d_0\|^2 \leqslant \rho \leqslant \varphi \|d_0\|^2 \tag{46}$$

and

$$\|d\| \leqslant \delta \|d_0\|, \tag{47}$$

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

Proof Looking at (26) and (27), we see that $\rho \leqslant \varphi \|d_0\|^2$. By lemma 9 and assumption 7,

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

If $d_1^\top \nabla f(x) > 0$ and in view of (26) we have,

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

Since d_1 is bounded and (27), there exist $\varphi_0 > 0$ such that

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

and then (46) is proved.

Now using (21) and the triangular property,

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

and using condition (46),

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

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

As a consequence of (46) and (47), ρ and $\|d\|^2$ have the same order of magnitude, in particular,

$$\varphi_0 \|d(x)\|^2 \leqslant \rho(x) \leqslant \varphi \|d(x)\|^2, \quad x \in \Omega_a. \tag{48}$$

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

$$G(y) \preccurlyeq G(x) + DG(x)(y - x) + L\|y - x\|^2 I \tag{49}$$

where $x, y \in \Omega$.

Proof Since G is C^1, the Mean Value Theorem [30] can be applied. Then

$$G(y) = G(x) + \sum_{i=1}^{n} (y_i - x_i) \frac{\partial G}{\partial x_i}(x + \xi(y - x)) \tag{50}$$

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

$$(y_i - x_i)\,\frac{\partial G}{\partial x_i}\,(x+\xi(y-x)) \preccurlyeq (y_i - x_i)\frac{\partial G}{\partial x_i}(x) + \left\|\frac{\partial G}{\partial x_i}(x+\xi(y-x)) - \frac{\partial G}{\partial x_i}(x)\right\| \|y-x\| I. \tag{51}$$

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

$$\left\|\frac{\partial G}{\partial x_i}(x+\xi(y-x)) - \frac{\partial G}{\partial x_i}(x)\right\| \leqslant L_i\|\xi(y-x)\| = L_i\xi\|y-x\|. \tag{52}$$

Then, using (52), (51) and (50), we have,

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

where $L = \xi\sum_{i=1}^{n} L_i$. □

Due to assumption (1), the sequence $\{x^k\}_{k\in\mathbb{N}} \in int(\Omega_a)$ generated by the algorithm have an accumulation point $x^* \in \Omega_a$. Since $\Lambda^k, B^k, G(x^k)$ and ρ^k are bounded, it follows that there exist $\mathbb{K}_1 \subset \mathbb{N}$ such that $\{d_0(x^k), d(x^k), \rho(x^k), \Lambda_0(x^k), \overline{\Lambda}(x^k), G(x^k)\}_{k\in\mathbb{K}_1}$ converges to $\{d_0(x^*), d(x^*), \rho(x^*), \Lambda_0(x^*), \overline{\Lambda}(x^*), G(x^*)\}$.

In [20], it was shown the existence of $\tau_f > 0$ such that at any $x \in \Omega_a$, condition (29) is verified for any $t \in [0, \tau_f]$.

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

$$G(x + td(x)) \preccurlyeq 0 \tag{53}$$

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

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

$$G(x+td(x)) \preccurlyeq G(x) + t\,DG(x)d + t^2 L\|d(x)\|^2 I \tag{54}$$

Let $F(t,x)$ be the matrix in the right hand side of (54), then,

$$G(x+td(x) \preccurlyeq F(t,x). \tag{55}$$

Since $\Lambda \succ 0$ and lemma 3, it is enough to show the existence of $\tau > 0$ such that

$$sym(\Lambda F(t,x)) \preccurlyeq 0, \quad t \in [0,\tau]. \tag{56}$$

Considering (19), we have,

$$sym(\Lambda F(t,x)) = sym\left(\left(\Lambda - t\overline{\Lambda}\right)G(x)\right) + + t\left(tL\|d(x)\|^2 - \rho(x)\right)\Lambda \tag{57}$$

where $\overline{\Lambda} = smat(\overline{\lambda})$ is defined in (22).

Since ρ verifies (48), for all v such that $\|v\| = 1$, we have

$$v^t(\Lambda F(t,x))v \leq v^t\Lambda G(x)v - t(v^t(\overline{\Lambda}G(x))v + \varphi_0\|d(x)\|^2\lambda^I) + + t^2\,L\|d(x)\|^2\lambda^S \tag{58}$$

It will be enough to show that the right hand side of (58) is non positive. This one is non-positive when $t = 0$. Let us consider now the following second degree equation:

$$v^t\Lambda G(x)v - t(v^t(\overline{\Lambda}G(x))v + \varphi_0\|d(x)\|^2\lambda^I) + + t^2\,L\|d(x)\|^2\lambda^S = 0 \tag{59}$$

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

Let $t(x,v)$ be the positive solution of (59):

$$t(x,v) = \left(\frac{v^t(\overline{\Lambda}G(x))v}{2\,L\,\|d(x)\|^2\lambda^S} + \frac{\varphi_0\lambda^I}{2\,L\lambda^S}\right) + + \sqrt{\left(\frac{v^t(\overline{\Lambda}G(x))v}{2\,L\,\|d(x)\|^2\lambda^S} + \frac{\varphi_0\lambda^I}{2\,L\lambda^S}\right)^2 - \frac{v^t\Lambda G(x)v}{L\,\|d(x)\|^2\lambda^S}} \tag{60}$$

Since $v^t\Lambda G(x)v \leq 0$ and $\dfrac{\varphi_0\lambda^I}{2\,L\lambda^S} > 0$, if follows from lemma 5 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 $G(x + td(x)) \preccurlyeq 0$ for all $t \in [0,\tau]$ and $x \in \Omega_a \cap \{x : \|d(x)\| \geq M\}$. □

As a consequence of proposition 1, it will be proved bellow 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 *Let be $\{x^k\}_{k\in\mathbb{N}}$ a sequence given by the algorithm converging to x^* with $\|d(x^*)\| = 0$. There exist $\delta > 0$ and $\tau > 0$ such that, if $x^k \in B(x^*,\delta) \cap \Omega_a$, then*

$$G(x^k + td(x^k)) \preccurlyeq 0 \tag{61}$$

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

Proof Since $\|d(x^*)\| = 0$, thanks to the continuity of $d(x)$, $\|d(x^k)\|$ goes to zero. As ρ verifies (48), we have

$$tL\|d(x)\|^2 - \rho(x) \leqslant (tL - \varphi_0)\,\|d(x)\|^2, \quad x \in \Omega_a.$$

It follows from (57) that if

$$\max\{v^\top(\Lambda^k - t\overline{\Lambda}^k)G(x^k)v, v \in \mathbb{R}^m, \|v\| = 1\} \leqslant 0. \tag{62}$$

then (61) is true for $t \leq \dfrac{\varphi_0}{2L}$.

Since Λ^k and $G(x^k)$ are both symmetric and commute, there exists an orthonormal matrix P_k such that

$$\Lambda^k = P_k^t D_\lambda^k P_k \tag{63}$$

$$G(x^k) = P_k^t D_g^k P_k \tag{64}$$

where D_λ^k and D_g^k are diagonal matrices.

Let $\overline{\overline{\Lambda}}^k = P_k \overline{\Lambda}^k P_k^t$. We shall prove that there exist $\delta > 0$ and $\tau > 0$ such that, if $x^k \in B(x^*, \delta) \cap \Omega_a$, then

$$v^\top((D_\lambda^k - t\overline{\overline{\Lambda}}^k)D_g^k)v \leqslant 0, \tag{65}$$

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

Let $\overline{\overline{\Lambda}}^k = D_\lambda^* + M^k$, where $M^k = P_k \overline{\Lambda}^k P_k^t - D_\lambda^*$. Since $\overline{\overline{\Lambda}}^k$ converges to D_λ^*,, then, for all $\epsilon > 0$, there exists δ such that for all $x^k \in B(x^*, \delta) \cap \Omega_a$, it is $|v^\top(M^k D_g^k)v| < \epsilon$.

Since the eigenvalues of D_λ^k are between λ^I and λ^S by hypothesis and the eigenvalues of Λ^* are bounded, due to theorem of Weyl [21], there exist $\tau_1 > 0$ such that the eigenvalues of $(D_\lambda^k - tD_\lambda^*)$ are positive for all $t \in [0, \tau_1]$.

Then for ϵ small enough:

$$v^\top((D_\lambda^k - tD_\lambda^*)D_g^k)v - t\, v^\top(M^k D_g^k)v \leqslant 0. \tag{66}$$

for all $v \in \mathbb{R}^m$, $\|v\| = 1$ and $t \in [0, \tau_1]$. The result follows for $\tau = \min\{\tau_1, \frac{\varphi_0}{2L}\}$. □

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

Proof Let be the set $\mathbb{K}_1 \subset \mathbb{K} \subset \mathbb{N}$ previously defined. It follows from propositions 1 and 2 that there exists $\mathbb{K}_2 \subset \mathbb{K}_1$ such that t^k goes to $t^* > 0$ for $k \in \mathbb{K}_2$.

We shall prove that at the limit, $\|d(x^*)\| = 0$. If this is not true, we assume that $\|d^*\| > \eta_d > 0$.

From the line search condition (29),

$$f\left(x^{fol(k)}\right) \leqslant f(x^k) + \eta\, t^k (d^k)^\top \nabla f(x^k).$$

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

$$f(x^*) \leqslant f(x^*) + \eta\, t^* (d^*)^\top \nabla f(x^*).$$

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

But, from lemma 10, when $k \to \infty$, we obtain

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

and from lemma 9 we have, for $k \to \infty$,

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

Since B^* is positive definite, $(d_0^*)^\top \nabla f(x^*) < 0$, that is a contradiction. Thus, $d^* = 0$.

Let $\lambda_0^* = svec(\Lambda_0^*)$. Now, considering equation (15), we have that (x^*, λ_0^*) verifies

$$\nabla f(x^*) + \nabla G(x^*)\lambda_0^* = 0 \tag{67}$$

$$(G(x^*) \circledast I)\lambda_0^* = 0. \tag{68}$$

To prove that (x^*, λ_0^*) is a stationary point of our problem, it remains to show that $G(x^*)\Lambda_0^* = 0$.

If $x^* \in int(\Omega_a)$, then $G(x^*) \prec 0$. Consequently $G(x^*) \circledast I$ is non singular and the linear system (68) has the unique solution $\lambda_0^* = 0$. Then, $\Lambda_0^* = smat(\lambda_0^*) = 0$ and the complementarity condition holds.

Considering now the case when x^* belongs to the boundary of Ω_a, it follows from (68) that $sym(G(x^*)\Lambda_0^*) = 0$. To finish the proof, we show that $skw(G(x^*)\Lambda_0^*) = 0$ or, equivalently, that all eigenvalues of $G(x^*)\Lambda_0^*$ are real.

Let $k \in \mathbb{K}_2$. Since Λ_0^k is symmetric and $G(x^k) \prec 0$, it follows from lemma 4, that $G(x^k)\Lambda_0^k$ has real eigenvalues. Moreover, $\{G(x^k)\Lambda_0^k\}_{k \in K_2}$ goes to $G(x^*)\Lambda_0^*$. Since the eigenvalues are continuous functions, we conclude that the eigenvalues of $G(x^*)\Lambda_0^*$ are also real. □

5 Algorithm implementation

In this section we include implementation details of the present algorithm and a set of experimental studies with linear and nonlinear SDP problems.

The quasi-Newton matrix B must verify assumption 7. We employ the Broyden-Fletcher-Goldfarb-Shanno (BFGS) updating rule with Powell's correction to ensure positive definiteness of B [35], with $B = I$ as initial value. For linear SDP problems, we have that $\nabla_{xx} L(x, \lambda) \equiv 0$. We take $B = 10^{-6} I$.

The matrix Λ must satisfy assumption 5 and 6, as required to prove global convergence. This is the case if

$$\Lambda = \mu I, \tag{69}$$

where μ is a positive real number. However, to keep good local convergence properties of Newton like methods, we should take $\Lambda^{k+1} = \Lambda_0^k$. Unfortunately, the assumptions 5 and 6 would not be always verified. We propose the following updating rule for Λ,

Rule 1:

i) Compute λ_{0min}, the minimum eigenvalue of Λ_0.
ii) If $\lambda_{0min} \geqslant \lambda^I$ then, set $\Lambda = \Lambda_0$.
iii) Else, set $\Lambda = \Lambda_0 + \left(\lambda^I - \lambda_{0min}\right) I$. □

With this rule Λ meets assumption 6, but assumption 5 is true only at the limit. To ensure global convergence with this rule, we restart with $\Lambda = I$ if the search direction is not descent or not feasible. In one of these situations the line search gives a very short step length can be obtained. In our numerical tests the line search never failed.

The stopping criterion is based on Karush-Kuhn-Tucker condition. The iterates stop when

$$\|\nabla f(x^k) + \nabla G(x^k) svec(\Lambda_0^k)\| < Tol$$

and

$$\|\Lambda_0^k G(x^k)\| < Tol,$$

where $Tol \in \mathbb{R}$.

In the numerical studies the parameters are taken with the following values: $\xi = 0.8$, $\eta = 0.1$, $\varphi = 1$, $\nu = 0.7$ and $Tol = 10^{-4}$. When a step $t < 0.01$ is obtained in the line search, in the next iteration we take $\Lambda = I$.

When an initial feasible point is not provided, the following auxiliary problem is solved,

$$\min_{x,z} z, \text{ s.t. } x \in \mathbb{R}^n, z \in \mathbb{R} \text{ and } G(x) - zI \preccurlyeq 0 \tag{70}$$

where z is an additional variable. A feasible point of Problem (1) is obtained once z becomes negative.

We shall present next the numerical results with some test problems of *SDPLIB* [11], a collection of linear semidefinite programming problems. Following, the results with some applications in structural optimization leading to linear and nonlinear SDP problems will be reported.

5.1 Numerical experiments with linear test problems

The numerical results with some test problems of *SDPLIB* are reported in table 1, where *iter* is the number of iterations to solve problem (1) with the given stopping criterion and f the computed objective function. Rule 1 was employed to define Λ and the last column shows the number of iterations with $\Lambda = I$. The last restart took place at iteration 196 in example Control4, in iteration 75 in example hinf9 and in iteration 12 in example truss1. An initial interior point for each problem was obtained, as described above.

Table 1 Numerical results, *SDPLIB*.

Problem	n	m	*iter*	$f(x^*)$	$iter(\Lambda = I)$
*control1**	21	15	37	17.7847	1
*control2**	66	30	123	8.3001	1
*control3**	136	45	95	13.6333	1
*control4**	231	60	400	19.7942	4
hinf1	13	14	24	2.0326	1
hinf2	13	16	48	10.9677	1
hinf3	13	16	30	56.9665	1
hinf4	13	16	35	274.765	1
hinf5	13	16	122	362.4300	1
hinf6	13	16	158	449.1210	1
hinf7	13	16	36	390.8300	1
hinf8	13	16	59	116.2170	1
hinf9	13	16	87	236.2490	24
hinf10	21	18	42	108.8370	1
hinf11	31	22	56	65.9161	1
qap5	136	26	16	−436.0000	1
qap6	229	37	23	−381.4360	1
theta1	104	50	20	23.0000	1
truss1	6	13	23	−8.9998	3
truss3	27	31	23	−9.1099	1
truss4	12	19	20	−9.0099	1

* $\xi = 0.4$

6 Applications of SDP to Structural Optimization

In this section an overview of some linear and nonlinear SDP models for trusses optimal design is given. The corresponding numerical results are reported in section 6.4.

6.1 The minimum compliance problem

We consider two or three-dimensional trusses with b bars and l degrees of freedom, submitted to a set of primary loading cases $P = \{p^1, \ldots, p^s\}$, where $p^i \in \mathbb{R}^l$ is the i-th loading case. The design variables are the bar volumes, denoted by x_j, $j = 1, \ldots, b$.

The worst possible compliance of the structure for the set of loadings is,

$$\phi_P(x) = \sup\{2u^\top p - u^\top K(x)u : u \in \mathbb{R}^l, p \in P\}, \tag{71}$$

where $x \in \mathbb{R}^b$ and $K(x)$ is the reduced stiffness matrix.

The so called Truss Topology Design (TTD) [9] looks for the volume of each of the bars that minimizes the structural compliance. The structural topology changes if some of the volumes are zero in the solution. Then, the minimum compliance problem can be stated as follows,

$$\begin{array}{ll} \min\limits_{x,\tau} \tau & \\ s.t. & \phi_P(x) \leqslant \tau, \\ & \sum_{j=1}^{b} x_j \leqslant \overline{V}, \\ & x \geqslant 0, \end{array} \tag{72}$$

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

$$\phi_P(x) \leqslant \tau \iff \begin{bmatrix} \tau & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad \forall p \in P, \tag{73}$$

we have that (72) is equivalent to the following semidefinite program:

$$(TTD)\begin{cases}\min\limits_{x,\tau}\tau\\ s.t.\quad \begin{bmatrix}\tau & p^\top\\ p & K(x)\end{bmatrix}\succcurlyeq 0,\quad \forall p\in P,\\ \qquad \sum_{j=1}^{b}x_j\leqslant \overline{V},\\ \qquad x\geqslant 0.\end{cases}\tag{74}$$

6.2 The robust compliance optimization problem

A structure is considered "robust" if it is reasonable rigid when any set of small possible uncertain loads acts on it. As proposed by Ben-Tal and Nemirovsky [7], is considered in addition to the primary loads, a set of "secondary" loads that are uncertain in size and direction, eventually acting on the structure. The compliance to be minimized is the worst possible one, under the simultaneous action of the "primary" and "secondary" loads.

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

$$M=\{Qe : e\in\mathbb{R}^q, e^\top e\leqslant 1\},\tag{75}$$

where

$$Q=[p^1\ \dots\ p^s\ re^1\ \dots\ re^{q-s}]\in\mathbb{R}^{l\times q}.\tag{76}$$

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 (RTT) is obtained replacing P by M in (72). Using an equivalence similar to (73) proved in [7], it is possible to rewrite the robust truss topology model as the following semidefinite programming problem:

$$(RTT)\begin{cases}\min\limits_{x,\tau}\tau\\ s.t.\quad \begin{bmatrix}\tau I & Q^\top\\ Q & K(x)\end{bmatrix}\succeq 0,\\ \qquad \sum_{j=1}^{b}x_j\leqslant \overline{V},\\ \qquad x\geqslant 0,\end{cases}\tag{77}$$

where Q is defined in (76).

6.3 Structural topology and geometry optimization with eigenvalues

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

The eigenvalues of a truss are the solution of the following equation:

$$K(x)v=\lambda M(x)v\tag{78}$$

where $K(x)$ and $M(x)$ are the reduced structural stiffness and mass matrices respectively and $(v,\lambda)\in\mathbb{R}^m\times\mathbb{R}$ is an eigenvector-eigenvalue pair.

Let $\lambda_{min}(x)$ be the smaller eigenvalue. The constraint $\lambda_{min}(x)\geqslant\underline{\lambda}$ is equivalent to the semidefinite constraint $K(x)-\underline{\lambda}M(x)\succcurlyeq 0$, see [1]. Then, the topology optimization problem of minimum volume subject to constraints on the minimum eigenvalue and the compliance is

$$(MV)\begin{cases}\min\limits_{x,V}V\\ \quad s.t.\\ \qquad K(x)-\underline{\lambda}M(x)\succcurlyeq 0,\\ \qquad \begin{bmatrix}\overline{\gamma} & p^\top\\ p & K(x)\end{bmatrix}\succcurlyeq 0,\quad \forall p\in P,\\ \qquad \sum_{j=1}^{b}x_j\leqslant V,\\ \qquad x\geqslant 0,\end{cases}\tag{79}$$

where $\underline{\lambda}$ is a lower bound of the eigenvalues and $\overline{\gamma}$ an upper bound of the compliance.

We consider now the structural geometry optimization problem of minimum volume, where the nodal coordinates y are the design variables,

$$(MVG)\begin{cases}\min\limits_{y}V(y)\\ \quad s.t.\\ \qquad K(y)-\underline{\lambda}M(y)\succcurlyeq 0,\\ \qquad \begin{bmatrix}\overline{\gamma} & p^\top\\ p & K(y)\end{bmatrix}\succcurlyeq 0,\quad \forall p\in P,\\ \qquad \underline{y}_j\leqslant y_j\leqslant \overline{y}_j; j=1,2,...,l\end{cases}\tag{80}$$

where $\underline{y}_j$ and $\overline{y}_j$ for $j=1,2,...,l$, represent lower and upper bounds on the nodal coordinates. Note that the objective function and the matrix constraints depends nonlinearly of the nodal coordinates y.

Let be the topology optimization of the compliance, subject to constraints on the lower eigenvalue and the volume,

$$(MC)\begin{cases}\min\limits_{x,\gamma}\gamma\\ \quad s.t.\\ \qquad K(x)-\underline{\lambda}M(x)\succcurlyeq 0,\\ \qquad \begin{bmatrix}\gamma & p^\top\\ p & K(x)\end{bmatrix}\succcurlyeq 0,\quad \forall p\in P,\\ \qquad \sum_{j=1}^{b}x_j\leqslant \overline{V},\\ \qquad x\geqslant 0,\end{cases}\tag{81}$$

where $\overline{V}$ is an upper bound of the structural volume.

Finally, the nonlinear problem of maximizing the minimum eigenvalue with compliance and volume constraints can be stated as,

$$(MF)\begin{cases} \max\limits_{x,\lambda} \lambda \\ \quad s.t. \\ \qquad K(x) - \lambda M(x) \succcurlyeq 0, \\ \qquad \begin{bmatrix} \overline{\gamma} & p^\top \\ p & K(x) \end{bmatrix} \succcurlyeq 0, \quad \forall p \in P, \\ \qquad \sum_{j=1}^{b} x_j \leqslant \overline{V}, \\ \qquad x \geqslant 0. \end{cases} \tag{82}$$

We solve problems (MF), (MV) and (MC) and compare the results as in [1]. Problems (TTD), (RTT), (MV) and (MC) are linear SDP problems since the matrix constraints are linear functions of the design variables. On the other hand, problems (MF) and (MVG) are nonlinear SDP problems.

6.4 Numerical tests for structural optimization problems

All the problems solved here are first converted to the general format (1). In examples 1 and 2 we apply the present algorithm to test problems (TTD) and (RTT) studied in [7]. The data of the structural optimization test problems, as well as the number of iterations required to solve them for the given stopping criterion, are shown in Table 2. The results for both updating rules for Λ are compared, with $\mu = 10^{-4}$. In all the examples, feasible descent directions were obtained without restart. We took $\Lambda = I$, only at the first iteration.

Table 2 Data and results of structural optimization examples

Example	*Model*	n	m	*Iter* ($\Lambda = rule1$)	*Iter* ($\Lambda = \mu I$)
1	TTD	11	20	25	263
	RTT	11	27	26	231
2	TTD	23	36	15	162
	RTT	23	47	17	471
3	MC	23	36	18	223
	MV	23	47	18	141
	MF	23	47	12	107
4	MVG	8	53	23	432
5	MVG	4	25	14	40

Example 1. Consider the 2D structure shown in Figure 1 submitted to a single "primary" loading case $P = 2\,[1, 0, 0, -1, 0, 1, -1, 0]$. The length of each of the horizontal and vertical bars is 1.0. The secondary loading cases 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.

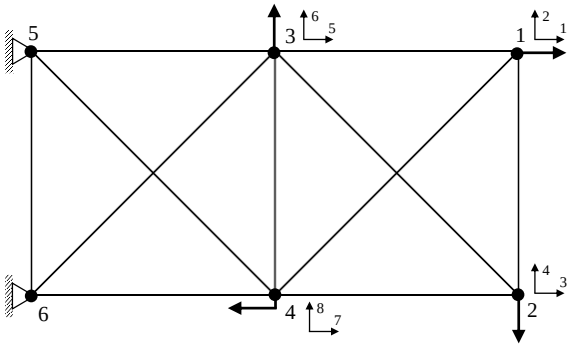

Fig. 1 Truss of Examples 1 and 5

Figure 2-left shows the final topology obtained with model (TTD). The resulting truss is unstable since, for any non-horizontal force applied to node 4 the compliance will be infinite. On the other hand, Figure 2-right represents the final topology obtained with the robust model (RTT).

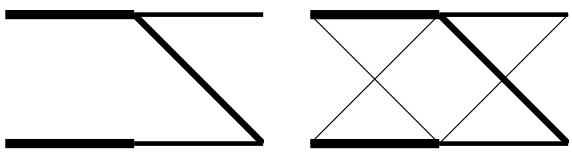

Fig. 2 Results of example 1. Left=(TTD). Right=(RTT).

Table 3 shows the deigns 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 3 Optimal designs - 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 3 are similar to those reported in [7].

Example 2. Let be a 3D truss with fixed nodes on the horizontal plane $z = 0$ and free nodes on the horizontal plane $z = 2$. The 8 nodes coordinates are,

$$\begin{bmatrix} \cos(2\pi i/4) \\ \sin(2\pi i/4) \\ 0 \end{bmatrix}, \; i \in \{1,2,3,4\},$$
$$\begin{bmatrix} \frac{1}{2}\cos(2\pi j/4) \\ \frac{1}{2}\sin(2\pi j/4) \\ 2 \end{bmatrix}, \; j \in \{5,6,7,8\}, \tag{83}$$

All the free and fixed nodes are jointed by bars. A single load case $P = \{p\}$ is defined. The components of p acting at the nodes on the plane $z = 2$ are given by,

$$p_j = \frac{1}{\sqrt{4(1+\rho^2)}} \begin{bmatrix} \sin(2\pi j/4) \\ -\cos(2\pi j/4) \\ -\rho \end{bmatrix}, \quad j \in \{5,6,7,8\}, \tag{84}$$

where $\rho = 0.001$. The secondary loading cases were defined as in example 1.

Figure 3-left and right show the optimal designs with (TTD) and (RTT) models. These results, that are given in Table 4, were also obtained in [7]

Table 4 Optimal designs of example 2. Bar volumes (%).

	TTD	RTT
1 : 5	0.0010	0.0033
1 : 6	12.500	12.459
1 : 7	0.0010	0.0027
1 : 8	12.500	12.460
5 : 6	0.0018	0.0526
5 : 7	0.0032	0.0434
5 : 8	0.0018	0.0526
6 : 7	0.0019	0.0526
6 : 8	0.0032	0.0436
7 : 8	0.0019	0.0526

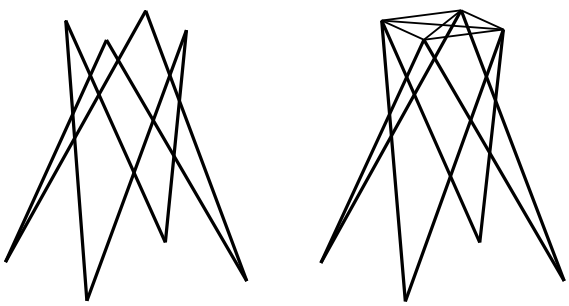

Fig. 3 Results of example 2. Left=(TTD). Right=(RTT).

Example 3.

We apply now the present algorithm to solve the problems (MV), (MC) and (MF). Consider the planar structure with 3×3 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 modulus of the material is 1.0. The initial sectional area of all the bars is 0.01.

We consider the minimum volume problem (MV) with $\overline{\gamma} = 1$ and $\underline{\lambda} = 0.05$. Our optimal design is shown in figure 4-right and the nonzero optimal volumes x^* in Table 5, with $\overline{V} = V^* = 4.731$.

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

Finally, when solving the problem of maximization the minimum eigenvalue (MF) with $\overline{V} = V^* = 4.731$ and $\overline{\gamma} = 1$, we obtain the optimal design (x^*, λ^*) with $\lambda^* = \underline{\lambda} = 0.05$. These results were also obtained in [1].

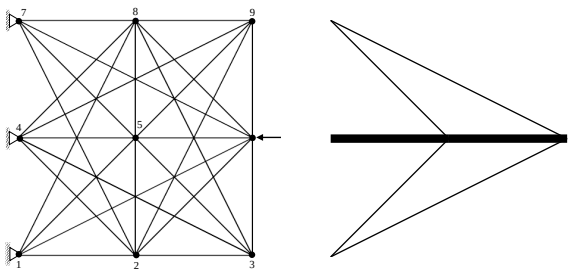

Fig. 4 Truss of example 3. Our algorithm obtains the same x^* when applied to problems (MV), (MC) and (MF).

Table 5 Results of example 3. Bar volumes (%).

	$Volume(\%)$
4 : 5	48.355
5 : 6	48.020
1 : 5	0.464
5 : 7	0.464
1 : 6	1.342
6 : 7	1.342

The two next examples consist on (MVG) problems, that require the solution of nonlinear SDP models.

Example 4.

Let be the three-dimensional truss with fixed nodes on the horizontal plane $z = 0$. The structure has 10 nodes. The initial and optimal nodal coordinates are given in Table 6 and 7 respectively. The corresponding geometries are represented in Figures 5 and 6.

Table 6 Initial coordinates of the structure of example 4.

node	*coord x*	*coord y*	*coord z*
1	−0.375	0.000	2.000
2	0.375	0.000	2.000
3	0.375	−0.375	1.000
4	0.375	0.375	1.000
5	−0.375	0.375	1.000
6	−0.375	−0.375	1.000
7	1.000	−1.000	0.000
8	1.000	1.000	0.000
9	−1.000	1.000	0.000
10	−1.000	−1.000	0.000

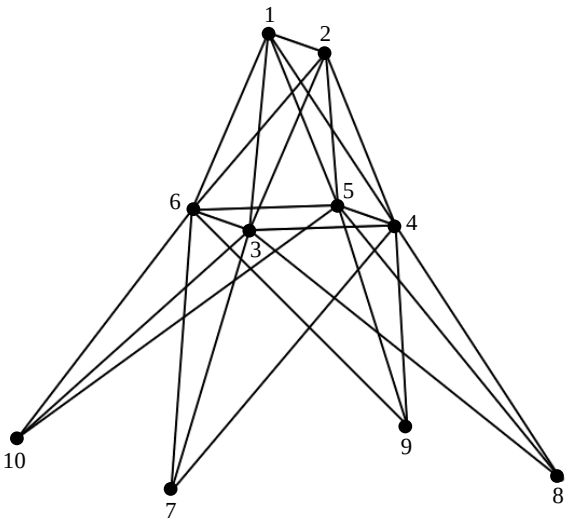

Fig. 5 Example 4 - 3D Truss. Initial design

A single load case $P = \{p\}$ is defined. The components of p acting at the nodes 1 and 2, on the plane $z = 2$ and 5 and 6, on the plane $z = 1$ are given by: $p_{1x} = 0.1$, $p_{1y} = p_{2y} = 1$, $p_{1z} = p_{2z} = -0.5$, $p_{5x} = p_{6x} = 0.05$. All bars have a sectional area of 0.1 and $\rho = 1$.
The box constraints for this example are: $0.1 \leq x_3 \leq 0.6$, $-0.6 \leq y_3 \leq -0.1$, $0.5 \leq z_3 \leq 1.5$, $0.1 \leq x_4 \leq 0.6$, $0.1 \leq y_4 \leq 0.6$, $0.5 \leq z_4 \leq 1.5$, $-0.6 \leq x_5 \leq -0.1$, $0.1 \leq y_5 \leq 0.6$, $0.5 \leq z_5 \leq 1.5$ and $-0.6 \leq x_6 \leq -0.1$, $-0.6 \leq y_6 \leq -0.1$, $0.5 \leq z_6 \leq 1.5$.

Table 7 Final coordinates of the structure of example 4.

node	*coord x*	*coord y*	*coord z*
1	−0.3750	0.0000	2.0000
2	0.3750	0.0000	2.0000
3	0.1382	−0.1222	1.3583
4	0.1383	0.1222	1.3583
5	0.1382	0.1222	1.3583
6	0.1382	−0.1222	1.3584
7	1.0000	−1.0000	0.0000
8	1.0000	1.0000	0.0000
9	−1.0000	1.0000	0.0000
10	−1.0000	−1.0000	0.0000

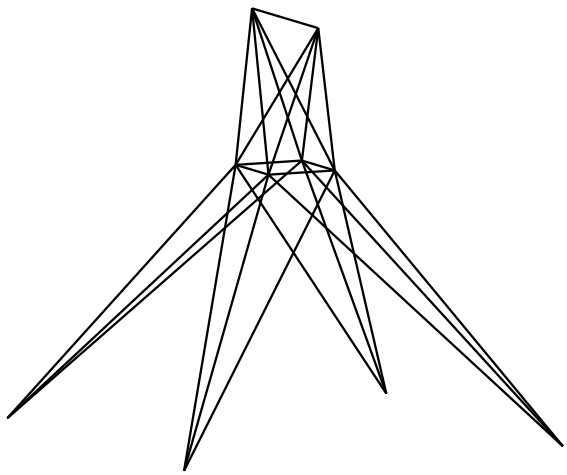

Fig. 6 Example 4. - Optimal design

Example 5

We apply now (MVG) problem to the 2D truss considered in example 1, submitted to the same primary loads. The optimal nodal coordinates are given in Table 8 and shown in Figure 7.

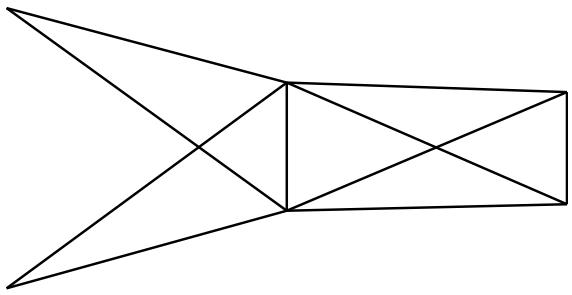

Fig. 7 Optimal design of example 5

Table 8 Final coordinates of the structure of example 5.

node	*coord x*	*coord y*
1	0.0000	0.0000
2	0.0000	1.0000
3	1.0000	0.2767
4	1.0000	0.7341
5	2.0000	0.3000
6	2.0000	0.7000

The box constraints for this example are: $0.7 \leq y_1 \leq 1.3$, $-0.3 \leq y_2 \leq 0.3$, $0.7 \leq y_3 \leq 1.3$ and $-0.3 \leq y_4 \leq 0.3$

7 Conclusions

In this paper, a new approach for nonlinear semidefinite programming is presented and supported by strong

theoretical results. In particular, global convergence to a critical point was proved. The present technique computes a descent feasible direction based on Newton-like iterations to solve KKT optimality conditions. To obtain a search direction, it is merely required the solution of two linear system with the same coefficient matrix followed by an inexact line search.

Linear SDP test problems from SDPLIB library were solved very efficiently. The numbers of iterations are in general comparable to the size of of the problem and the value of the objective function is similar to the values published in [11].

In this paper we have presented a large number of linear and nonlinear SDP problems coming from structural optimization. The numerical results exposed show the performances of the proposed algorithm in linear and nonlinear cases, in particular the number of iterations are very raisonable when we consider Rule 1 to update Λ.

The results obtained solving the structural geometric optimization of minimum volume where the nodal coordinates are the design variables shows the interesting performances of the presented algorithm in the nonlinear case.

The numerical tests were successfully performed with the same parameters.

8 Acknowledgments

The authors thank the Brazilian Research Councils CAPES, CNPq and FAPERJ, the Brazilian program "Ciência Sem Fronteiras", the French Research Councils CNRS and INRIA and the Brazilian-French Network in Mathematics, for the financial support.

References

1. Achtziger, W., Kočvara, M.: Structural topology optimization with eigenvalues. Tech. rep., University of Dortmunt (2006)
2. Achtziger, 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., Haeberly, J.P.A., Overton, M.L.: Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results. SIAM Journal on Optimization **8**, 746 768 (1998)
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. Bendsøe, 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. Hartwig, R.E.: How to partially order regular element ? Math. Japon **25**, 1–13 (1980)
18. Herskovits, J.: Developement of a numerical method for nonlinear optimization. Ph.D. thesis, Paris IX University, INRIA Rocquencort (1982)
19. Herskovits, J.: A two-stage feasible directions algorithm for nonlinear constrained optimization. Mathematical Programming **36**, 19–38 (1986)
20. Herskovits, J.: A feasible directions interior point technique for nonlinear optimization. Journal of Optimization Theory and Applications **99**, 121–146 (1998)
21. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press (1985)
22. Jarre, F.: An interior method for nonconvex semidefinite programs. Optimization and Engineering **1**, 347–372 (2000)
23. Kanno, Y., Takewaki, I.: Sequential semidefinite program for maximum robustness design of structures under load uncertainty. Journal of Optimization Theory and Applications **130**(2), 265–287 (2006)
24. Kanzow, C., Nagel, C., Kato, H., Fukushima, M.: Successive linearization methods for nonlinear semidefinite programs. Computational Optimization and Applications **31**, 251–273 (2005)
25. de Klerk, E.: Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications. Kluwer (2002)
26. Kočvara, M., Stingl, M.: Pennon : A code for convex nonlinear and semidefinite programming. Optimization Methods and Software **18**, 317–333 (2003)
27. Kočvara, M., Stingl, M.: Free material optimization for stress constraints. Structural and Multidisciplinary Optimization **33**, 323–335 (2007)

28. Lewis, A.S., Overton, M.L.: Eigenvalue optimization. Acta Numerica **5**, 149190 (1996)
29. Li, C., Sun, W.: On filter-successive linearization methods for nonlinear semidefinite programming. Science in China Series A: Mathematics **52**(11), 2341–2361 (2009)
30. Lima, E.L.: Análise Real, vol. 2, 3 edn. Coleção Matemática Universitária (2007)
31. Luenberger, D.G., Ye, Y.: Linear and Nonlinear Programming, 3 edn. Springer (2008)
32. Luo, H., Wu, H., Chen, G.: On the convergence of augmented lagrangian methods for nonlinear semidefinite programming. Journal of Global Optimization **54**(3), 599–618 (2012)
33. M. Fukuda M. Kojima, K.M., Nakata, K.: Exploiting sparsity in semidefinite programming via matrix completions i: General framework. SIAM Journal of Optimization **11**, 647– 674 (2000)
34. 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)
35. Powell, M.J.D.: The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations, in Nonlinear Programming 3. Academic Press, London (1978)
36. Shapiro, A.: First and second order analysis of nonlinear semidefinite programs. Mathematical Programming **77**, 301–320 (1994)
37. Shapiro, A.: On uniqueness of lagrange multipliers in optimization problems subject to cone constraints. SIAM Journal on Optimization **7**, 508 518 (1997)
38. Stingl, M.: On the solution of nonlinear semidefinite programs by augmented lagrangian methods. Ph.D. thesis, University of Erlangen-Nrnberg (2005)
39. Stingl, M., Kočvara, M., Leugering, G.: A sequencial convex semidefinite programming algorithm with an application to multiple-load free material optimization. SIAM Journal on Optimization **2009**, 130–155
40. 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)
41. Sun, D., Sun, J., Zhang, L.: The rate of convergence of the augmented lagrangian method for nonlinear semidefinite programming. Mathematical Programming **114**(2), 349–391 (2008)
42. Todd, M.J.: Semidefinite optimization. Acta Numerica **10**, 515–560 (2001)
43. Yamashita, H., Yabe, H.: Local and superlinear convergence of a primal-dual interior point method for nonlinear semidefinite programming. Mathematical Programming **132**(1-2), 1–30 (2012)
44. Yamashita, H., Yabe, H., Harada, K.: A primal-dual interior point method for nonlinear semidefinite programming. Math. Program. **135**(1-2), 89–121 (2012)
45. Yang, L., Yu, B.: A homotopy method for nonlinear semidefinite programming. Computational Optimization and Applications **56**(1), 81–96 (2013)
46. Zowe, J., Kocvara, M., Bendsoe, M.P.: Free material optimization via mathematical programming. Mathematical Programming **79**, 445–466 (1997)
47. 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)