J Glob Optim (2017) 68:169–187
DOI 10.1007/s10898-016-0460-2

Application of the sequential parametric convex approximation method to the design of robust trusses

Alfredo Canelas[1] · Miguel Carrasco[2] · Julio López[3]

Received: 20 February 2016 / Accepted: 28 August 2016 / Published online: 1 September 2016

Abstract We study an algorithm recently proposed, which is called sequential parametric approximation method, that finds the solution of a differentiable nonconvex optimization problem by solving a sequence of differentiable convex approximations from the original one. We show as well the global convergence of this method under weaker assumptions than those made in the literature. The optimization method is applied to the design of robust truss structures. The optimal structure of the model considered minimizes the total amount of material under mechanical equilibrium, displacements and stress constraints. Finally, Robust designs are found by considering load perturbations.

Keywords Sequential parametric convex approximation · Truss optimization · Robust design · Stress constraints

1 Introduction

In a recent paper Beck et al. [7] have proposed the sequential parametric convex approximation method (SPCA), to find the solution of a differentiable nonconvex optimization problem by

The first author was supported by the Uruguayan Councils ANII and CSIC. The second and third authors were supported by CONICYT-Chile, via FONDECYT projects 1130905 and 1160894, respectively.

✉ Alfredo Canelas
acanelas@fing.edu.uy

Miguel Carrasco
micarrasco@uandes.cl

Julio López
julio.lopez@udp.cl

1 Instituto de Estructuras y Transporte, Facultad de Ingeniería, Universidad de la República, Julio Herrera y Reissig 565, Montevideo, Uruguay

2 Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes, Mons. Álvaro del Portillo 12455, Las Condes, Santiago, Chile

3 Facultad de Ingeniería, Universidad Diego Portales, Ejército 441, Santiago, Chile

solving a sequence of differentiable convex ones. Let us consider the following optimization problem:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} \; f(x) \\ \text{s.t.} \quad g_i(x) \le 0 \quad i = 1,\dots,m, \end{cases} \tag{P}$$

where g_i, $i = 1,\dots,p$ are differentiable nonconvex functions, whilst f and g_i, $i = p+1,\dots,m$ are differentiable convex ones. Given $x_0 \in \mathbb{R}^n$, a feasible point of (P), for each $k \in \mathbb{N}$ the SPCA solves iteratively the following differentiable convex optimization problem:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} \; f(x) \\ \text{s.t.} \quad G_i(x, \psi_i(x_k)) \le 0 \quad i = 1,\dots,p, \\ \qquad\; g_i(x) \le 0 \quad i = p+1,\dots,m, \end{cases} \tag{P_k}$$

where $G_i : \mathbb{R}^n \times Y \to \mathbb{R}, i = 1,\dots,p$ are continuous functions such that $G_i(\cdot, y) : \mathbb{R}^n \to \mathbb{R}$ is a differentiable convex upper estimate of the nonconvex g_i. The set Y is called the *admissible parameters set*. The continuous functions $\psi_i : \mathbb{R}^n \to Y$, $i = 1,\dots,p$ play the role in providing a convenient value for the parameter at the actual iterate x_k; in particular, the parameter $\psi_i(x_k)$ must be such that $g_i(x_k) = G_i(x_k, \psi_i(x_k))$ ensuring that $G_i(\cdot, \psi_i(x_k))$ be an approximate of g_i in a neighborhood of x_k, and that the feasible iterate x_k of (P) be the same also for (P_k). The next iterate x_{k+1} is defined by the SPCA method as the optimal solution of (P_k).

As other methods of nonlinear programming, the SPCA replaces the original constrained optimization problem by a simpler auxiliary one, which is solved to obtain an approximate solution. For instance, in penalty and barrier methods [6,22,23] the auxiliary problem is an unconstrained optimization one that can be solved using known algorithms for unconstrained problems. In the SPCA the key feature of the auxiliary problem is convexity, which allows us the application of algorithms for nonlinear convex ones, e.g. the efficient interior-point algorithms of convex programming [9,12]. Once the approximate solution is obtained, a new auxiliary problem can be defined to obtain a more accurate solution. While in penalty and barrier methods the new auxiliary problem is obtained by updating the penalty and barrier parameters, in the SPCA this is obtained by evaluating the functions ψ_i at the actual solution. Another important feature of the SPCA is that the functions G_i, $i = 1,\dots,p$, upper estimate the functions g_i in the original problem; hence the feasible set of each auxiliary one is completely contained in the feasible set of the original problem. Therefore, as in barrier and feasible direction methods [16,22], the entire sequence $\{x_k\}_{k\in\mathbb{N}}$ generated by the SPCA method is also feasible. As mentioned by Herskovits [17], feasible approaches are appropriate for engineering design optimization, where function evaluation is in general quite expensive. Since any intermediate design can be employed, the iterations can be safely stopped when the objective reduction per each becomes small enough.

Some sequential quadratic programming (SQP) methods [16,22] are also based on the solution of convex auxiliary optimization problems which are given by a quadratic convex approximation of the Lagrangian function and linear approximations of the constraints. However, in SQP methods, the feasible set of the auxiliary problems is not necessarily contained in the original one, although there exist several SQP variants that generate a feasible sequence of solutions [31,32]. In SQP methods the auxiliary problems are easily defined by using local information (derivatives) of the original functions; hence SQP algorithms are generally applicable to any nonlinear optimization problem. However, since the auxiliary problems of the SQP represent well the original optimization one only in a neighborhood of the actual iterate, line-search or trust-region approaches must be used to define a suitable next iterate.

The SPCA is instead applicable only to problems where suitable convex upper estimates of the nonconvex constraints are known. This is maybe one of the main drawbacks of the SPCA, since there is not a general systematic procedure to obtain upper convex approximations of general functions. However, if such estimates are known for a particular problem and they represent well the original constraints in a broad region of the feasible set, then the SPCA could be a better alternative, especially if the convex auxiliary problems admit the use of known efficient algorithms.

In this paper we present a new proof of global convergence to Karush–Kuhn–Tucker (KKT) points of (P) for the SPCA method, considering weaker assumptions than those studied in [7]. In this context, global convergence means monotone convergence of the objective function and KKT conditions satisfied for each accumulation point of the sequence generated by the SPCA method. Our global convergence result is obtained under a Slater-like qualification condition instead of the linear independence constraint qualification and strict convexity of the objective function considered in [7]. We note that the qualification condition assumed in this paper is absolute necessary if interior point algorithms are used to solve the auxiliary problems (P_k). The convergence analysis is done by viewing the SPCA method as the repeated application of a closed point-to-set map and by using the classical Zangwill theorem of convergence. Comments about the application of this approach to non-differentiable problems are also given. More efficient approaches can be devised by allowing the use of an approximate solution of (P_k) at each iteration. For this purpose, we study the application of closed feasible descent algorithms to (P_k).

The second part of this paper is devoted to the study of optimal designs of robust trusses under mechanical displacements and stress constraints. To obtain a robust design we assume that in addition to set of primary external loads, which are applied only at the nodes of the truss, there exists also a set of secondary loads that are uncertain in size and direction, which can be viewed as perturbations of the main loads. The objective is to find the truss that minimizes the total amount of material or weight, i.e., the most *economical* structure, satisfying stress and displacements constraints under the main loads and any possible load perturbation, i.e., we follow the worst-case formulation of the robust design problem, which leads to a nonlinear, nonconvex, semi-infinite mathematical programming problem. We show that if the set of secondary loads takes the form of an ellipsoid, then the optimization problem can be reformulated as a nonlinear optimization one with a finite number of nonlinear, nonconvex constraints. This last problem can then be solved numerically by using the SPCA method.

The paper layout is as follows. Section 2 provides the convergence analysis of the SPCA method. Section 3 describes the proposed model for the optimal design of robust trusses and shows how to state a SPCA for this model. Section 4 presents some numerical results showing that the proposed formulation is effective to obtain a robust design. Finally, Sect. 5 presents the conclusions.

2 General algorithm

Given $x \in \mathbb{R}^n$, let (P_x) be the following differentiable convex optimization problem:

$$\begin{cases} \min\limits_{z \in \mathbb{R}^n} & f(z) \\ \text{s.t.} & G_i(z, \psi_i(x)) \leq 0\,, \quad i = 1, \ldots, p\,, \\ & g_i(z) \leq 0\,, \qquad\qquad\; i = p+1, \ldots, m. \end{cases} \tag{P_x}$$

Then, we can view the SPCA method as the repeated application of a map $A\colon X \rightrightarrows \mathbb{R}^n$, where $X := \{x \in \mathbb{R}^n \mid g_i(x) \leq 0,\ i = 1, \dots, m\}$ is the feasible set of (P), and A is given by:

$$A(x) := \text{Argmin } (P_x)\,. \tag{1}$$

Consider the following assumptions:

Assumption 1 Functions G_i and ψ_i, $i = 1, \dots, p$, satisfy:

$$g_i(x) \leq G_i(x, y) \qquad \text{for every } x \in \mathbb{R}^n\,,\ \ y \in Y\,, \tag{2}$$

$$g_i(x) = G_i(x, \psi_i(x)) \qquad \text{for every } x \in \mathbb{R}^n\,. \tag{3}$$

Assumption 2 For any feasible point x of (P) there exist a point $\bar{y}$ such that

$$G_i(\bar{y}, \psi_i(x)) < 0\,, \qquad i = 1, \dots, p\,, \tag{4}$$

$$g_i(\bar{y}) < 0\,, \qquad i = p+1, \dots, m\,. \tag{5}$$

Assumption 2 denotes that at any feasible point x of (P), the auxiliary problem (P_x) satisfies the Slater constraint qualification. Note that this assumption is necessary if we intend to solve (P_x) by using a feasible interior point method. A direct consequence of Assumption 2 is that any minimum point x^* of (P_x) is a KKT point of (P_x) [11, Proposition 3.3.9]. In addition, the following lemma holds:

Lemma 1 *Under assumptions 1 and 2, any local minimum x^* of (P) is a KKT point of (P).*

Proof First note that, by Assumption 1, x^* being feasible for (P) it is the same for (P_{x^*}). Since x^* is a local minimum of (P), then it is also a local minimum of (P_{x^*}) and therefore by convexity is a global minimum. Hence, from Assumption 2, convexity and differentiability of (P_{x^*}) we have that x^* is a KKT point of (P_{x^*}), i.e. calling $y_i = \psi_i(x^*)$, there exist $\lambda_i \geq 0$, $i = 1, \dots, m$, such that

$$\nabla f(x^*) + \sum_{i=1}^{p} \lambda_i \nabla_x G_i(x^*, y_i) + \sum_{i=p+1}^{m} \lambda_i \nabla g_i(x^*) = 0, \tag{6}$$

$$\lambda_i G_i(x^*, y_i) = 0, \quad i = 1, \dots, p, \tag{7}$$

$$\lambda_i g_i(x^*) = 0, \quad i = p+1, \dots, m. \tag{8}$$

From Assumption 1, $G_i(x^*, y_i) = g_i(x^*)$, $i = 1, \dots, p$, and x^* is a global minimum of the function $x \mapsto G_i(x, y_i) - g_i(x)$. Therefore

$$\nabla_x G_i(x^*, y_i) - \nabla g_i(x^*) = 0\,. \tag{9}$$

Then, notation G_i in (6)–(8) can be replaced by the notation g_i, hence x^* is a KKT point of (P). □

Remark 1 From the proof of the previous lemma we easily see that if x^* is not a KKT point of (P) then x^* is not a KKT of (P_{x^*}). In the case that x^* is a KKT point of (P) then, by Assumption 1, it is also a KKT point of (P_{x^*}) and therefore, by convexity, it is a global minimum of (P_{x^*}).

Remark 2 In [7], the authors considered these three elements: Assumption 1, the linear independence constraint qualification (LICQ) and strict convexity of the objective function.

Since Assumption 1 and LICQ imply Assumption 2 and we do not require strict convexity of the objective function, then, the assumptions considered here are weaker than the ones considered in [7]. To prove that Assumption 1 and LICQ imply Assumption 2, we must take any feasible regular point x of (P) and call $y_i = \psi_i(x)$. Let I be the set of indexes of the active constraints at x. For any $i \in I$, we have $g_i(x) = 0$, so that, if $i \leq p$, then by Assumption 1 (9) holds at x. Then, x is a regular point of (P_x), and since LICQ implies the Mangasarian-Fromovitz constraint qualification, there exists a direction d in the cone of interior directions of (P_x) at x [6, chapter 5]. Then, taking $\delta > 0$ small enough, $\bar{y} = x + \delta d$ is strictly interior to the feasible set of (P_x), therefore, Assumption 2 holds.

Remark 3 Equation (9) was considered an independent assumption in [7, Property A Eq. 2.2]. However, in the differentiable case (9) is a consequence of Assumption 1. The existence of G_i satisfying Assumption 1 is then very important in the theoretical analysis of (P). In the non-differentiable case a condition similar to (9), regarding the subdifferential sets of subgradients, must be independently assumed, since it cannot be obtained directly from Assumption 1. This fact restricts the practical application of the SPCA method to the non-differentiable case, see Sect. 2.1 below for more details.

Consider the following definitions:

Definition 1 A point-to-set map $A: X \rightrightarrows Y$ is said to be closed if given $\{x_k\}_{k\in\mathbb{N}} \subseteq X$ with $x_k \to \bar{x}$, and $z_k \to \bar{z}$, with $z_k \in A(x_k)$, then we get $\bar{z} \in A(\bar{x})$.

Definition 2 $F: X \to \mathbb{R}$ is said to be a descent function for a point-to-set map $A: X \rightrightarrows X$ and a set $\Gamma \in X$ if, for all $x \in X$, it satisfies: (i) $x \notin \Gamma$ and $z \in A(x)$ then $F(z) < F(x)$; (ii) $x \in \Gamma$ and $z \in A(x)$ then $F(z) \leq F(x)$.

In order to prove the global convergence of the SPCA method, we consider the following theorem:

Theorem 1 (Zangwill) *Let $A: X \rightrightarrows X$ be a closed point-to-set map, $\Gamma \subseteq X$ a given solution set and $F: X \to \mathbb{R}$ a descent function for A and Γ. Assume that a sequence $\{x_k\}_{k\in\mathbb{N}}$ is generated by A, i.e. $x_{k+1} \in A(x_k)$, and that it is contained in a compact subset of X. Then, every accumulation point of the sequence $\{x_k\}_{k\in\mathbb{N}}$ belongs to Γ.*

Proof See [6,22]. □

We note that in this theorem the terminology *global convergence* refers to the inclusion of the set of accumulation points of the sequence $\{x_k\}_{k\in\mathbb{N}}$ in the solution set. In the following, we assume that Assumptions 1 and 2 hold.

Lemma 2 *Let $x_k \to \bar{x}$, $z_k \to \bar{z}$, with $z_k \in A(x_k)$ and $\hat{z}$ be such that $G_i(\hat{z}, \psi_i(\bar{x})) < 0$ for $i = 1, \ldots, p$, and $g_i(\hat{z}) \leq 0$ for $i = p+1, \ldots, m$. Then $f(\hat{z}) \geq f(\bar{z})$.*

Proof By continuity of G_i and ψ_i we get $G_i(\hat{z}, \psi_i(x_k)) \to G_i(\hat{z}, \psi_i(\bar{x}))$, therefore, there exists $\bar{k}$ such that $G_i(\hat{z}, \Psi(x_k)) \leq 0$ for all $k \geq \bar{k}$, $i = 1, \ldots, p$. Then, $\hat{z}$ is feasible for (P_{x_k}) for all $k \geq \bar{k}$, which implies that $f(\hat{z}) \geq f(z_k)$ for all $k \geq \bar{k}$. Taking the limit as $k \to \infty$, we obtain $f(\hat{z}) \geq f(\bar{z})$. □

The main theorem of this paper is given below:

Theorem 2 *Let Γ be the set of KKT points of (P), and A the point-to-set map given by (1). Then we obtain:*

(a) f is a descent function for A and Γ.
(b) the map A is closed.

Proof Let x be a feasible point of (P). Then we have: (i) if $x \in \Gamma$, then by definition $f(z) \leq f(x)$ for all $z \in A(x)$; (ii) if $x \notin \Gamma$ then x is not a KKT point of (P) and therefore x is not a KKT point of (P_x), see Remark 1. Then x is not a minimum point of (P_x) so that $f(x) > f(z)$ for all $z \in A(x)$. From (i) and (ii) we have (a).

To prove (b) consider $\{x_k\}_{k\in\mathbb{N}} \subseteq X$ with $x_k \to \bar{x}$, and $z_k \to \bar{z}$, with $z_k \in A(x_k)$. By continuity of functions G_i, ψ_i, for $i = 1, \ldots, p$, and g_i, for $i = p+1, \ldots, m$, we get $\bar{x} \in X$. Take any feasible point z of $(P_{\bar{x}})$, i.e., a point z satisfying $G_i(z, \psi_i(\bar{x})) \leq 0$ for $i = 1, \ldots, p$, and $g_i(z) \leq 0$ for $i = p+1, \ldots, m$. Take $\bar{y}$ be the point satisfying (4)–(5) in Assumption 2 with $x \equiv \bar{x}$. The sequence defined by $z_j = z + (1/j)(\bar{y} - z)$ is such that $G_i(z_j, \psi_i(\bar{x})) < 0$ for $i = 1, \ldots, p$, $g_i(z_j) < 0$ for $i = p+1, \ldots, m$, and satisfies $z_j \to z$ as $j \to \infty$. Using the result of Lemma 2, we get $f(z_j) \geq f(\bar{z})$ for all j, and conclude that $f(z) \geq f(\bar{z})$ for any feasible z. Therefore $\bar{z} \in A(\bar{x})$.

Remark 4 Under the assumption of compactness of the feasible set X of (P) and using Zangwill's theorem, we obtain that any accumulation point of the sequence $\{x_k\}_{k\in\mathbb{N}}$ generated by SPCA is a KKT point of (P). Furthermore, as a consequence of Theorem 2 and the compactness of X, the monotonically decreasing sequence $\{f(x_k)\}_{k\in\mathbb{N}}$ converges.

2.1 The non-differentiable case

Let us assume in this case that f and g_i, for $i = 1, \ldots, m$, are possibly non-differentiable Lipschitz functions, as well as the upper convex approximation functions G_i, $i = 1, \ldots, p$. Let $\partial f(x)$ denote the Clark subdifferential of f [15]. In addition to Assumptions 1 and 2 we have to consider the following assumption regarding the approximation functions G_i:

Assumption 3 Given any feasible point x for (P), x is a KKT point of (P_x), in the sense that there exist real values $\lambda_i \geq 0$, $i = 1, \ldots, m$, such that

$$0 \in \partial f(x) + \sum_{i=1}^{p} \lambda_i \partial_x G_i(x, y_i) + \sum_{i=p+1}^{m} \lambda_i \partial g_i(x)\,, \tag{10}$$

with $y_i = \psi_i(x)$, only if x is a KKT point of the original problem (P), i.e., there exist real values $\lambda_i \geq 0$, $i = 1, \ldots, m$ satisfying

$$0 \in \partial f(x) + \sum_{i=1}^{m} \lambda_i \partial g_i(x)\,. \tag{11}$$

Note that in the non-differentiable case we cannot prove that x is a KKT point of the original problem (P) whenever x is a KKT point of (P_x), since $\partial_x G_i(x, y) \neq \partial g_i(x)$ in the general case (in fact, from Assumption 1 we can prove that $\partial g_i(x) \subset \partial_x G_i(x, y)$, but the subdifferential $\partial g_i(x)$ could not contain all the subgradients in $\partial_x G_i(x, y)$). Assumption 3 is of main importance for the success of the SPCA method. If a point x were a KKT point of (P_x) but not a KKT point of (P), the SPCA method could get stuck at a point that is not a KKT point of (P). This situation cannot be accepted, see Fig. 1.

Assumption 3 allow us to prove the non-differentiable version of Lemma 1:

Lemma 3 *If (P_x) satisfies Assumptions 1, 2 and 3, then any local minimum x^* of (P) is a KKT point of (P).*

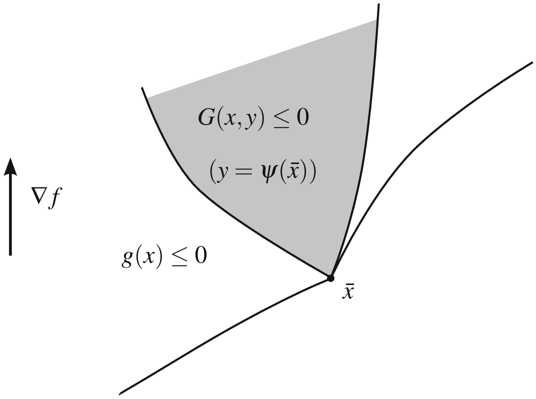


Fig. 1 Example with a single nonconvex non-differentiable constraint, where a point $\bar{x}$ is a KKT point of $(P_{\bar{x}})$ but not a KKT point of (P)

Proof By Assumption 1, x^* is feasible for (P_{x^*}). Since x^* is a local minimum of (P), it is also a local minimum of (P_{x^*}). Thanks to Assumption 2, x^* is a KKT point of (P_{x^*}) [15, Theorem 6.4.4], and by Assumption 3, x^* must be KKT point of (P).

Remark 5 Note that Lemma 2 holds in the non-differentiable case as it is, since it does not recall on the differentiability of (P). Theorem 2 also holds in the non-differentiable case if (P_x) satisfies the Assumption 3. Then, under Assumptions 1, 2 and 3, the sequence generated by the SPCA method converges globally to a KKT point of (P).

Remark 6 Note that Assumption 3 is the weakest possible condition in order that Lemma 3 hold. The disadvantage of Assumption 3 is that it is a rather difficult condition to be verified in a practical application. A more easily verifiable condition could be the assumption $\partial_x G_i(x, y_i) = \partial g_i(x)$, $i = 1, \ldots, p$, which implies Assumption 3 but has the disadvantage of being more restrictive.

2.2 Using other closed feasible descent algorithms

In most of the practical applications, Algorithm (1) cannot be implemented analytically, and an optimization algorithm must be used to obtain an approximate solution to (P_x). This procedure strictly falls outside the framework studied up to this section, since Algorithm (1) is actually replaced by other algorithm for which we have to prove independently the descent and closedness properties to ensure global convergence to a KKT point of (P).

Hence, the application of the SPCA to practical problems involves the execution of a nested iteration, where a complete iteration to solve (P_x) is performed at each iteration of the main algorithm. The inner iteration must then be performed very efficiently in order to allow the application of the SPCA. Note that global convergence to a KKT point of the original problem (P) will be obtained provided the approximate solution to (P_x) is found performing one or more iterations of a closed feasible descent algorithm. Next, we show a simple problem that we can efficiently solve by reducing the inner iteration to just one of a closed feasible descent optimization algorithm, and this simple procedure can even overcome Algorithm (1).

Let us consider here the following problem:

$$\begin{cases} \min\limits_{x \in \mathbb{R}^2} & f(x) = (x_1 - a)^2 + (x_2 - a)^2 \\ \text{s.t.} & g(x) = x_1 x_2 \leq 1\,, \\ & 0.01 \leq x_1, x_2 \leq 100\,, \end{cases} \tag{12}$$

Table 1 Performance of the algorithms

	Iter	$\bar{x}$	$\|\bar{x} - x^*\|$	$f(\bar{x}) - f(x^*)$
a = 1.5				
SPCA-Exact	24	(1.000075, 0.999925)	1.06×10^{-4}	8.40×10^{-9}
FDIPA	11	(1.000000, 1.000000)	2.19×10^{-9}	2.17×10^{-10}
SPCA-FDIPA	15	(1.000000, 1.000000)	3.85×10^{-8}	4.20×10^{-9}
a = 2				
SPCA-Exact	30	(1.199921, 0.833372)	2.60×10^{-1}	1.15×10^{-3}
FDIPA	30	(1.013915, 0.986275)	1.95×10^{-2}	3.81×10^{-8}
SPCA-FDIPA	30	(1.056752, 0.946286)	7.81×10^{-2}	2.94×10^{-5}

where a is a real parameter. The same problem with $a = 2$ was considered in [7]. Since the value $a = 2$ makes (12) to be ill-conditioned (the Hessian of the Lagrangian restricted to the tangent plane at the solution is singular), we also consider the value $a = 1.5$. In both cases the unique solution is $x^* = (1, 1)$. The SPCA version of (12) is given by:

$$G(x, \lambda) = \frac{\lambda}{2} x_1^2 + \frac{1}{2\lambda} x_2^2, \qquad \psi(x) = \frac{x_2}{x_1}. \tag{13}$$

Table 1 gives the number of main iterations required to obtain the approximate $\bar{x}$ to the exact solution x^*, when starting from the point $x_0 = (5, 0.02)$, for three different algorithms: SPCA-Exact: SPCA with analytical solution of the auxiliary problems; FDIPA: the Feasible Directions Interior Point Algorithm [17,18] applied to the original problem (12); SPCA-FDIPA: SPCA with an inexact solution corresponding to one iteration of FDIPA. The algorithms were stopped if $f(\bar{x}) - f(x^*) \leq 1.0 \times 10^{-8}$, or 30 main iterations were performed.

Note that even though SPCA-Exact finds the analytical solution of the auxiliary problem (P_k), it requires more main iterations than the other two algorithms to solve (P) with similar accuracy. FDIPA, and SPCA-FDIPA perform much better than SPCA-Exact, being FDIPA slightly better than SPCA-FDIPA. This last fact is probably due to the reinitialization of the Lagrange multipliers. Figures 2 and 3 show the iterates obtained by the three algorithms. The figures show why SPCA-Exact is less efficient: it performs shorts steps when the iterates approach the boundary of the feasible region of (P), especially when $a = 2$. FDIPA and SPCA-FDIPA do not exhibit this behavior.

The results obtained for this example suggest that for a general problem (P) and having the numerical algorithm ‘A’, the better strategy should be (i) if A can solve (P_x) but not (P), then use SPCA-A with a large tolerance to reduce the number of inner iterations; (ii) if A can directly handle the original problem (P), then use A alone.

3 Structural optimization model

This section shows an application of the SPCA to the topology design of robust trusses [1,2,10]. The SPCA is used to find the truss of minimum total weight among those that satisfy some stress and displacement constraints. Stress constrained structural optimization problems are computationally hard, and it is known that the optimum could not be the best design because of the well-known stress singularity problem [26]. Techniques to handle this

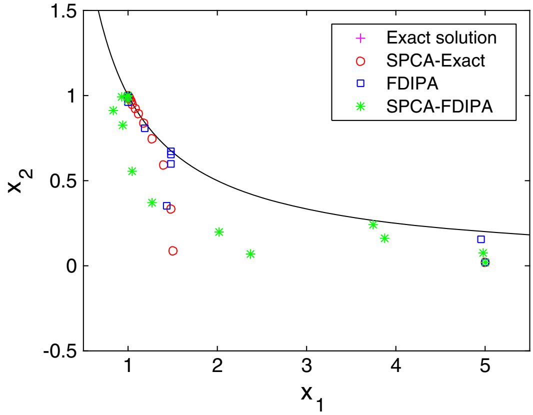


Fig. 2 Result obtained for problem (12) with $a = 1.5$

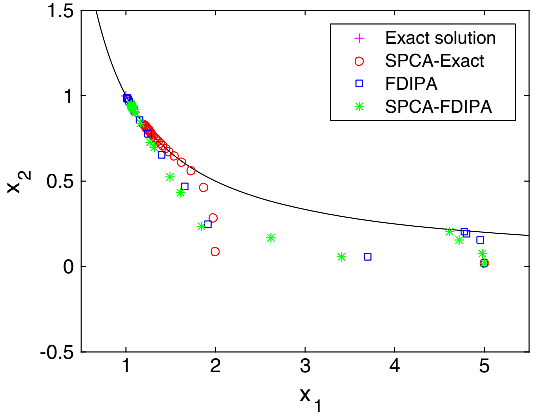


Fig. 3 Result obtained for problem (12) with $a = 2$

problem have been proposed, see references [13,14,21,24,26,28,29]. In this paper the stress singularity problem is not addressed. Therefore, the present approach should be applied on top of a good initial design, e.g. obtained by solving a classical non-robust compliance model. Other mechanical constraints such as global and local buckling constraints, natural frequency constraints, etc., are beyond the scope of this paper.

It is well known that optimal structures obtained by some models are unstable from the mechanical point of view, thus great effort has been made in order to obtain formulations having robust optimal trusses, see e.g. [3,5,8]. In the optimization model considered here, robust solutions are obtained by considering a set of small *secondary* loadings that are uncertain in size and direction, and can act over the structure concurrently with the main loadings.

Let us consider a two or a three-dimensional ground structure with L nodes, m initial potential bars, and n degrees of freedom. For the sake of simplicity, let us consider a single primary loading $f \in \mathbb{R}^n$. We propose the robust structural design model defined by the following nonconvex semi-infinite mathematical programming problem:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^m} \sum\limits_{i=1}^{m} x_i \\ \text{s.t.} \quad \left|u_j(\xi, x)\right| \le \bar{u}_j \quad \forall \xi \in \mathcal{E}\,, \quad j \in J \subseteq N, \\ \qquad\ \ |\sigma_i(\xi, x)| \le \bar{\sigma}_i \quad \forall \xi \in \mathcal{E}\,, \quad i \in I \subseteq M, \\ \qquad\ \ \varepsilon \le x \le U. \end{cases} \tag{P_w}$$

In (P_w), x is the vector of bar volumes, $u_j(x, \xi)$ and $\sigma_i(x, \xi)$ denote, respectively, the displacement corresponding to the jth degree of freedom and the stress in the ith bar, when the external force $f + \xi$ is acting over the structure. The values $\bar{u}_j$ and $\bar{\sigma}_j$ are the upper bounds on the displacements and stresses, respectively. N is the set of node indexes and M is the set of bar indexes. The lower bound ε is assumed as positive (displacements and stresses may be undefined for configurations with zero bar volumes). The vector $\xi \in \mathbb{R}^n$ represents a small perturbation that belongs to the set $\mathcal{E} \subseteq \mathbb{R}^n$ of secondary loadings. For given x and ξ, the displacement vector $u(x, \xi)$ is the unique solution to the following mechanical equilibrium equation:

$$K(x)u = f + \xi\,,$$

where $K(x)$ is the stiffness matrix given by

$$K(x) = \sum_{i=1}^{m} x_i K^i\,,$$

with $K^i = b^i(b^i)^\top$ denoting the stiffness matrix corresponding to the ith bar of unitary volume and b^i is a vector that depends on the coordinates of the bar and its material properties. As usual, to obtain a well posed optimization problem we assume that the sum $\sum_{i=1}^m K^i$ is positive definite. The values $\bar{u_j}$ and $\bar{\sigma}_i$ are the bounds for the displacements and the stresses, respectively.

Note that the displacement $u_j(x, \xi)$ and the stress $\sigma_i(x, \xi)$ are given by similar expressions:

$$u_j(x, \xi) = (e^j)^\top u(x, \xi)\,, \quad \sigma_i(x, \xi) = \sqrt{E_i}\,(b^i)^\top u(x, \xi)\,,$$

where e^j is the canonical vector, and E_i corresponds to the *Young modulus* of the ith bar, see e.g. [20, Chapter 1] and [25, Chapter 2] for details. In order to simplify the notation we denote by $C = \{1, \ldots, c\}$ with $c = |I| + |J|$ the total number of displacement and stress constraints. Then, (P_w) can be conveniently expressed as

$$\begin{cases} \min\limits_{x\in\mathbb{R}^m} \sum\limits_{i=1}^{m} x_i \\ \text{s.t.} \quad |(v^j)^\top u(x, \xi)| \le \bar{v}_j \quad \forall \xi \in \mathcal{E}\,, \; j \in C, \\ \qquad\ \ \varepsilon \le x \le U. \end{cases} \tag{P'_w}$$

To address the infinite number of constraints, we reformulate (P'_w) as the following non-convex mathematical programming problem:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^m} \sum\limits_{i=1}^{m} x_i \\ \text{s.t.} \quad \max\{(v^j)^\top u(x, \xi) \mid \xi \in \mathcal{E}\} \le +\bar{v}_j, \; j \in C, \\ \qquad\ \ \min\{(v^j)^\top u(x, \xi) \mid \xi \in \mathcal{E}\} \ge -\bar{v}_j, \; j \in C, \\ \qquad\ \ \varepsilon \le x \le U, \end{cases} \tag{P_B}$$

where we have implicitly assumed that the internal problems have optimal solutions, which is true if we consider a compact set of secondary loadings. In the following we will consider the ellipsoid $\mathcal{E} = \{Qe \mid \|e\| \leq 1\}$, where $Q \in \mathbb{R}^{n\times d}$ is a full-rank matrix and d is the dimension of $\mathcal{E}$. Each element of $\mathcal{E}$ in this paper can be viewed as a small perturbation of the main load f. This idea has been applied to model the uncertainties of external loads and also other model parameters in the field of structural optimization, see [19] and references therein.

In this case, since the displacement vector is linear with respect to e, the internal problems of (P_B) can be solved analytically. In fact, we have

$$u(x, e) = K(x)^{-1}(f + Qe)\,, \tag{14}$$

and the unique optimal solutions of the internal problems are obtained by solving the KKT conditions:

$$e^j_{max} = \frac{+\nabla_e[(v^j)^\top u(x, e)]}{\|\nabla_e[(v^j)^\top u(x, e)]\|} = \frac{+Q^\top K(x)^{-1}v^j}{\|Q^\top K(x)^{-1}v^j\|}\,, \tag{15}$$

$$e^j_{min} = \frac{-\nabla_e[(v^j)^\top u(x, e)]}{\|\nabla_e[(v^j)^\top u(x, e)]\|} = \frac{-Q^\top K(x)^{-1}v^j}{\|Q^\top K(x)^{-1}v^j\|}\,. \tag{16}$$

A direct calculation, using (14) and (15)–(16), give us the inequality constraints of (P_B) in the form:

$$f^\top K(x)^{-1}v^j + \|Q^\top K(x)^{-1}v^j\| \leq +\bar{v}_j\,,\ j \in C,$$
$$f^\top K(x)^{-1}v^j - \|Q^\top K(x)^{-1}v^j\| \geq -\bar{v}_j\,,\ j \in C,$$

which can be equivalently expressed as

$$\left|f^\top K(x)^{-1}v^j\right| + \|Q^\top K(x)^{-1}v^j\| \leq \bar{v}_j\,,\ j \in C.$$

Then, (P_B) can be formulated as the following nonconvex finite-dimensional model:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^m} \sum\limits_{i=1}^m x_i \\ \text{s.t.} \quad \left|f^\top K(x)^{-1}v^j\right| + \|Q^\top K(x)^{-1}v^j\| \leq \bar{v}_j,\ j \in C, \\ \qquad \varepsilon \leq x \leq U. \end{cases} \tag{P'_B}$$

Finally, it is convenient to introduce the additional scalar variables τ^1_j and τ^2_j, the vector variable $r^j \in \mathbb{R}^d$, and the set $D = \{1, \ldots, d\}$ to formulate (P'_B) as:

$$\begin{cases} \min\limits_{x,\tau^1_j,\tau^2_j,r^j} \sum\limits_{i=1}^m x_i \\ \qquad \|r^j\| \leq \tau^2_j\,, & j \in C, \\ \qquad \tau^1_j + \tau^2_j \leq \bar{v}_j\,, & j \in C, \\ \qquad \left|f^\top K(x)^{-1}v^j\right| \leq \tau^1_j\,, & j \in C, \\ \qquad \left|(q^\ell)^\top K(x)^{-1}v^j\right| \leq r^j_\ell\,, & j \in C,\ \ell \in D\,, \\ \qquad \varepsilon \leq x \leq U, \end{cases} \tag{P''_B}$$

where q^ℓ is the ℓth column of Q.

We note that the previous formulation considers only one main force f and one ellipsoid of perturbations. However, the model can be easily extended to consider several independent loadings and ellipsoids, with the consequent increase of the problem dimension and number of constraints.

3.1 SPCA approximation of the structural model

Note that (P''_B) have some second-order cone (SOC) constraints, some linear constraints and some nonlinear nonconvex constraints of the form $\left|q^\top K(x)^{-1}v\right| \leq \tau$. We recall that a SOC constraint has the form

$$\|Ax + b\| \leq c^\top x + d,$$

where $A \in \mathbb{R}^{m\times n},\ b \in \mathbb{R}^m,\ c \in \mathbb{R}^n$ and $d \in \mathbb{R}$, see e.g. [4].

The SPCA approximation of (P''_B) is based on the convex upper estimate function $F(x, \lambda, h)$ proposed in [7] for the function $H(x) = \left|q^\top K(x)^{-1}v\right|$:

$$F(x, \lambda, h) = \frac{\lambda}{2} q^\top K(x)^{-1} q + \frac{1}{2\lambda} v^\top K(x)^{-1} v + \frac{1}{\lambda} v^\top h + \frac{1}{2\lambda} h^\top K(x) h. \tag{17}$$

Theorem 3 *For given $\lambda > 0$ and $h \in \mathbb{R}^n$ satisfying $q^\top h = 0$, the function $F(\cdot, \lambda, h)$ defined in the open convex set $S = \{x \in \mathbb{R}^m \mid x > 0\}$ is a convex upper estimate of H. In addition, at any x such that $H(x) \neq 0$, we have $F(x, \lambda(x), h(x)) = H(x)$, where the functions $\lambda(x)$ and $h(x)$ are given by*

$$\lambda(x) = |\theta(x)|\ , \quad h(x) = K(x)^{-1}(\theta(x)q - v)\ , \quad \theta(x) = \frac{q^\top K(x)^{-1} v}{q^\top K(x)^{-1} q}\,.$$

Proof See [7].

Using the above theorem, the SPCA approach presented in Sect. 2 can be applied to (P''_B) as follows: given $x_0 \in \mathbb{R}^m$ a feasible point of (P''_B), we generate iteratively the sequence $\{x_k\}_{k\in\mathbb{N}}$ by solving the convex optimization problem (P''_{Bk}). This problem is obtained by replacing each one of the constraints of the form $\left|q^\top K(x)^{-1}v\right| \leq \tau$ in (P''_B) by:

$$\frac{\lambda(x_k)}{2} q^\top K(x)^{-1} q + \frac{1}{2\lambda(x_k)} v^\top K(x)^{-1} v + \frac{1}{\lambda(x_k)} v^\top h(x_k) + \frac{1}{2\lambda(x_k)} h(x_k)^\top K(x) h(x_k) \leq \tau. \tag{18}$$

The next iterate x_{k+1} corresponds to a solution of (P''_{Bk}).

Defining the additional scalar variables α_1 and α_2, the inequality constraint (18) can be expressed as:

$$q^\top K(x)^{-1} q \leq \alpha_1\,, \qquad v^\top K(x)^{-1} v \leq \alpha_2, \tag{19}$$

and

$$\frac{\lambda(x_k)\alpha_1}{2} + \frac{\alpha_2}{2\lambda(x_k)} + \frac{v^\top h(x_k)}{\lambda(x_k)} + \frac{h(x_k)^\top K(x) h(x_k)}{2\lambda(x_k)} \leq \tau. \tag{20}$$

Note that (20) is a linear constraint on the variables α_1, α_2 and x, while (19) can be equivalently expressed as a set of linear and SOC constraints, as shown in the following proposition.

Proposition 1 *If $\varepsilon \geq 0$, the constraints $x \geq \varepsilon$ and $q^\top K(x)^{-1} q \leq \tau$ can be equivalently expressed introducing vector variables $s \in \mathbb{R}^m$ and $\beta \in \mathbb{R}^m$ as:*

$$\sum_{i=1}^m s_i b^i = q\,, \quad \sum_{i=1}^m \beta_i \leq \tau\,, \quad (s_i)^2 \leq x_i \beta_i\,,\ i \in M\,, \quad x \geq \varepsilon\,. \tag{21}$$

Proof See [4, Sec. 2] and also [7,9]. The inequality $(s_i)^2 \le x_i\beta_i$ defines a rotated SOC constraint, which can be expressed equivalently as $\|(2s_i, x_i - \beta_i)\| \le x_i + \beta_i$.

Finally, by applying Theorem 3 and Proposition 1, after some computations we obtain the following second-order cone programming (SOCP) problem

$$\begin{cases} \min \sum_{i=1}^m x_i \\ \text{s.t. } \|r^j\| \le \tau_j^2,\ j \in C, \quad \tau_j^1 + \tau_j^2 \le \bar{v}_j,\ j \in C, \quad (s_i)^2 \le x_i\beta_i,\ i \in M, \\ \quad (p_{ij})^2 \le x_i\gamma_{ij},\ i \in M,\ j \in C, \quad (z_{i\ell})^2 \le x_i\sigma_{i\ell},\ i \in M,\ \ell \in D, \\ \quad \frac{\lambda_j(x_k)\alpha_1}{2} + \frac{\alpha_2^j}{2\lambda_j(x_k)} + \frac{(v^j)^\top h_j(x_k)}{\lambda_j(x_k)} + \frac{h_j(x_k)^\top K(x) h_j(x_k)}{2\lambda_j(x_k)} \le \tau_j^1, \\ \quad j \in C, \\ \quad \frac{\tilde{\lambda}_\ell^j(x_k)\alpha_3^\ell}{2} + \frac{\alpha_2^j}{2\tilde{\lambda}_\ell^j(x_k)} + \frac{(v^j)^\top \tilde{h}_\ell^j(x_k)}{\tilde{\lambda}_\ell^j(x_k)} + \frac{\tilde{h}_\ell^j(x_k)^\top K(x)\tilde{h}_\ell^j(x_k)}{2\tilde{\lambda}_\ell^j(x_k)} \le r_\ell^j, \\ \quad j \in C,\ \ell \in D, \\ \quad \sum_{i=1}^m s_i b^i = f, \quad \sum_{i=1}^m p_{ij} b^i = v^j,\ j \in C, \quad \sum_{i=1}^m z_{i\ell} b^i = q^\ell,\ \ell \in D, \\ \quad \sum_{i=1}^m \beta_i \le \alpha_1, \quad \sum_{i=1}^m \gamma_{ij} \le \alpha_2^j,\ j \in C, \quad \sum_{i=1}^m \sigma_{i\ell} \le \alpha_3^\ell,\ \ell \in D, \\ \quad \varepsilon \le x \le U, \end{cases} \tag{P''_{Bk}}$$

where:

$$\lambda_j(x) = |\theta_j(x)|, \quad h_j(x) = K(x)^{-1}(\theta_j(x)f - v^j),$$
$$\tilde{\lambda}_\ell^j(x) = |\tilde{\theta}_\ell^j(x)|, \quad \tilde{h}_\ell^j(x) = K(x)^{-1}(\tilde{\theta}_\ell^j(x)q^\ell - v^j),$$
$$\theta_j(x) = \frac{f^\top K(x)^{-1}v^j}{f^\top K(x)^{-1}f}, \quad \tilde{\theta}_\ell^j(x) = \frac{(q^\ell)^\top K(x)^{-1}v^j}{(q^\ell)^\top K(x)^{-1}q^\ell}.$$

4 Numerical examples

In this section we present some examples to illustrate the solution of formulation (P'_B) considering $Q = 0$, namely *non-robust* formulation, and $Q \ne 0$, what we call *robust* formulation. In all the examples considered here, the ellipsoids given by Q are set as balls of secondary loads not greater than 5% of the original forces. Additionally, once the optimized structure is obtained, we measure the maximal absolute displacement u_{max} among all nodes, and the maximal absolute stress σ_{max}, among all bars, computed by considering the worst perturbation given by (15)–(16).

In order to find an initial feasible point, we check if $x_i^0 = 1$, $i = 1, \dots, m$ is feasible, and if it is so, then we set $x_i^0 = 1/j$, with $j \in \mathbb{N}$ the largest integer such that x^0 is that way. If $x_i^0 = 1$ is infeasible then we set $x_i^0 = j$ with $j \in \mathbb{N}$ the smallest integer such that x^0 is feasible. A lower bound $\varepsilon = 1 \times 10^{-8}$ was used for all the examples given. The SPCA strategy is used to solve (P'_B), i.e., for each feasible x_k the iterate x_{k+1} is found by solving the convex auxiliary problem (P''_{Bk}). These problems are solved using *SeDuMi* [30] with default settings. The outer iteration is stopped when the reduction of the objective function is less than 10^{-3}. In summary, in order to solve the problem (P'_B) we use the following algorithm:

Algorithm 1 SPCA for solving the problem (P'_B).

Step 0 Find an initial feasible point $x_0 \in \mathbb{R}^m$ by using the criterion described above. Set $k = 0$.
Step 1 Compute a solution x_{k+1} of SOCP problem (P''_{Bk}).
Step 2 If x_{k+1} satisfies a prescribed stopping rule, then stop.
Step 3 Replace k by $k + 1$ and go to step 1.

This algorithm was implemented in MATLAB 7.13, and the numerical experiments were performed on a computer with Intel Core i5 processor (2.6 GHz) and 8 GB RAM.

In the pictures presenting the optimized solutions, Figs. 5, 6, 8, 10, 11, 13 and 14, the bars that resulted with a volume lower than 0.01% of the largest bar of the optimized structure are not depicted.

Example 1 We take this example from [8]. In the reference configuration, the left nodes are fixed, while the main loading corresponds to four forces as shown in Fig. 4. The bar stresses are limited by the bound $\bar{\sigma} = 45$ and the Young modulus of all bars is $E = 1$. The optimized structure for the non-robust formulation is depicted in Fig. 5, while Fig. 6 shows the robust solution when the force at the top-right node is perturbed by a ball of secondary loads. Table 2 displays the total amount of material of the optimized structure, the maximal displacement u_{max}, and the maximal stress σ_{max} obtained for the non and robust formulations. The table shows that the non-robust solution presents high displacements and stresses when small perturbations are applied at the top-right node. In fact, the values u_{max} and σ_{max} obtained numerically are finite just because the lower bound ε is positive.

Fig. 4 Example 1: ground structure

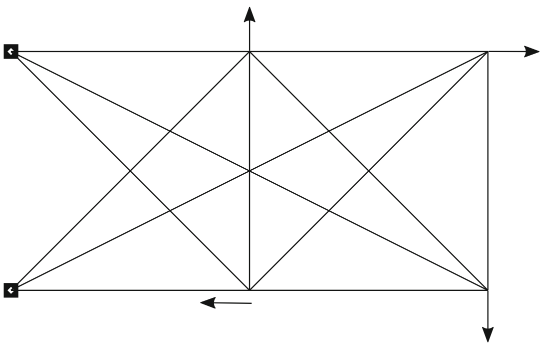
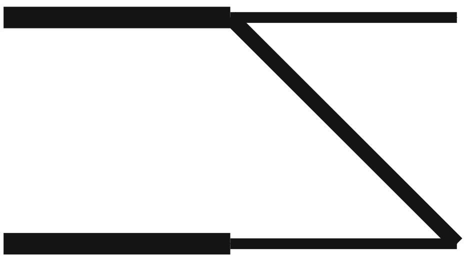


Fig. 5 Example 1: optimized structure, non-robust formulation



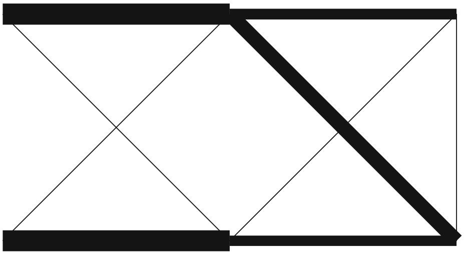


Fig. 6 Example 1: optimized structure, robust formulation

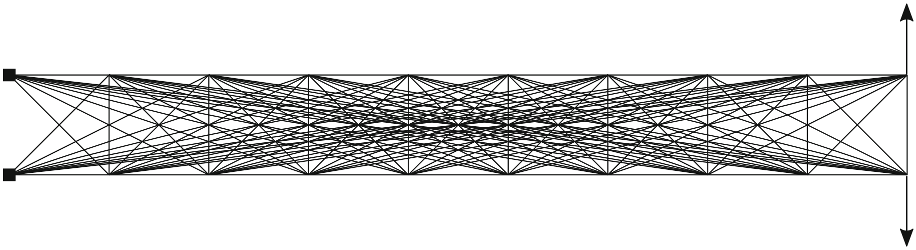


Fig. 7 Example 2: ground structure

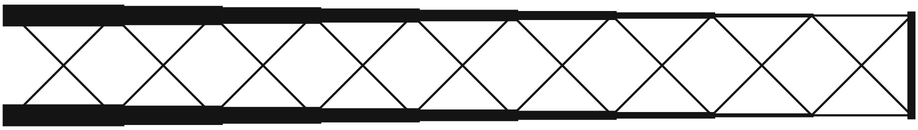


Fig. 8 Example 2: optimized structure, robust formulation

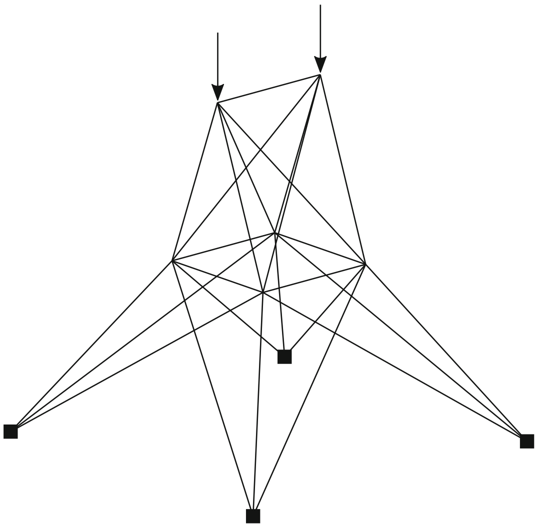


Fig. 9 Example 3: ground structure

Example 2 The second example corresponds to a cantilever truss structure, similar to Example 5.1 in [7]. The left nodes of the structure are fixed and opposite forces are applied at the free end of the cantilever as shown in Fig. 7. In this example, $E = 1$ and the displacements of the nodes at the free end are bounded by a value $\bar{u} = 0.35$. The loading acting on the

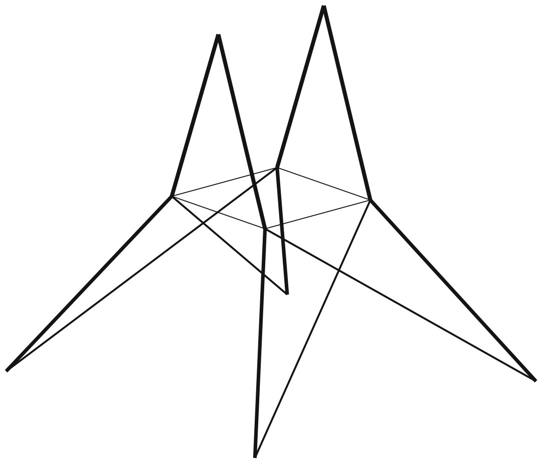


Fig. 10 Example 3: optimal structure, non-robust formulation

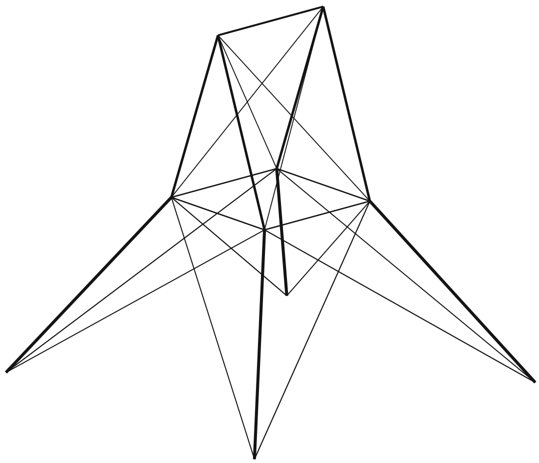


Fig. 11 Example 3: optimal structure, robust formulation

structure makes it necessary to consider a robust model so as to obtain a stable structure, since the non-robust formulation concentrates all the structural material in the bar by joining the nodes where the forces are applied. The solution obtained for the robust formulation when the main load is perturbed considering a ball of secondary loads at the bottom-right node is shown in Fig. 8.

Example 3 A three-dimensional truss with two load scenarios is considered in this example, see Fig. 9. Each scenario corresponds to only one of the vertical forces represented in the figure. The example has the same geometry as the one considered in [7] and Problem 16 in [27], but different loads are applied. The problems with several loading scenarios are more likely to provide a stable truss when considering non-robust formulations. However, certain load configurations can still provide unstable solutions, which is the case of those considered here, according to Table 2. In this example, $E = 1 \times 10^5$ all bar stresses are bounded by a value $\bar{\sigma} = 2000$, and all nodal displacements are bounded by a value $\bar{u} = 0.35$. In the robust case, the main loads are perturbed by a ball of secondary ones acting where the original forces are applied. The non and robust solutions are depicted in Figs. 10 and 11.

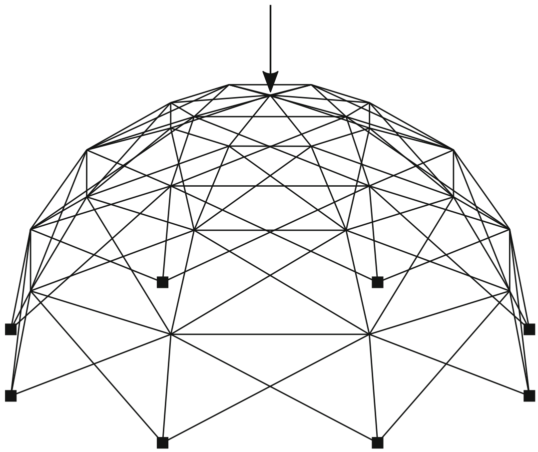


Fig. 12 Example 3: ground structure

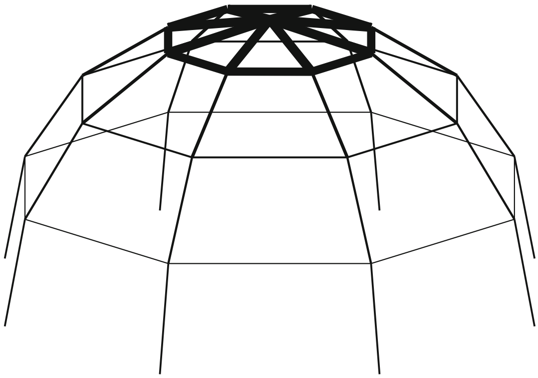


Fig. 13 Example 3: optimal structure Non-robust case

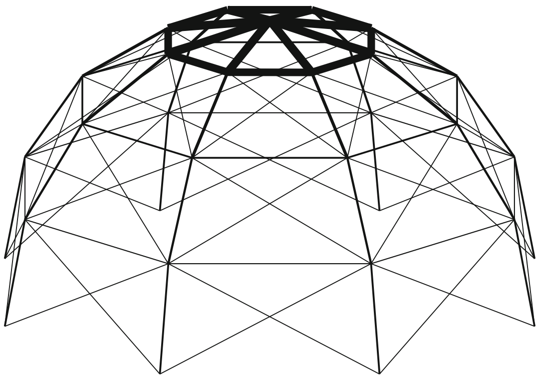


Fig. 14 Example 3: optimal structure Robust case

Example 4 The four-story dome of this example was considered in [5], see also [2]. It has a vertical load applied just on the top, see Fig. 12. In this example, the stresses are bounded by a value $\bar{\sigma} = 155$, $E = 1$, and the load on the top is perturbed by a three-dimensional ball of secondary loads. The solution obtained for the non-robust formulation is shown in Fig. 13, the robust counterpart is shown in Fig. 14. Table 2 shows that in this example the non-robust

Table 2 Results

Example	Model	Iter	CPU time (s)	w^*	u_{max}	σ_{max}
Example 1	Non-robust	40	19.3	0.711	$3.25e7$	$1.75e9$
	Robust	41	29.2	0.758	2.387	45
Example 2	Non-robust	76	19.4	1.4286	$1.79e8$	$3.02e6$
	Robust	178	50.9	1496.7	0.35	0.024
Example 3	Non-robust	76	6.8	3455	$6.20e4$	$2.75e8$
	Robust	207	35.8	5326	0.35	1077
Example 4	Non-robust	23	25.6	0.0077	418	8650
	Robust	29	28.4	0.0084	0.51	155

solution present high displacements and stresses when submitted to the worst secondary load, that are reduced considerably by using the robust formulation.

5 Conclusions

In this paper, we have shown the global convergence of the SPCA method to solve nonconvex optimization problems. The proof presented is based on Zangwill's theorem and requires weaker hypotheses than those considered previously. The extension of the theorem to the non-differentiable case was also discussed. By considering a simple example, we have shown that the strategy of using an approximate solution of the auxiliary problem can outperform the exact solution strategy. Given that the proof of convergence presented here is based on Zangwill's theorem, an approximate solution provided by a closed feasible descent algorithm can be used instead of the exact solution.

In Sect. 3, we have presented a semi-infinite nonconvex optimization model to design robust trusses. This model leads to a nonconvex mathematical problem, which was applied to find robust solutions. An SPCA was presented for this problem. The convex auxiliary problems were formulated as second-order cone constrained ones that can be solved by using available and efficient interior-point algorithms as *SeDuMi*. The numerical examples given illustrate the ability of the robust model to provide a mechanically stable structure in several situations where the non-robust formulation has an unstable solution.

References

1. Achtziger, W.: Truss topology optimization including bar properties different for tension and compression. Struct. Optim. **12**(1), 63–74 (1996)
2. Achtziger, W.: Topology optimization of discrete structures: an introduction in view of computational and nonsmooth aspects. Topology optimization in structural mechanics. CISM Courses and Lectures, vol. 374, pp. 57–100. Springer, Vienna (1997)
3. Achtziger, W., Bendsøe, M., Ben-Tal, A., Zowe, J.: Equivalent displacement based formulations for maximum strength truss topology design. Impact Comput. Sci. Eng. **4**(4), 315–345 (1992)
4. Alizadeh, F., Goldfarb, D.: Second-order cone programming. Mathematical Programming. A Publication of the Mathematical Programming Society 95(1, Ser. B), 3–51: ISMP 2000, Part 3. Atlanta, GA (2003)
5. Alvarez, F., Carrasco, M.: Minimization of the expected compliance as an alternative approach to multiload truss optimization. Struct. Multidiscip. Optim. **29**(6), 470–476 (2005)

6. Bazaraa, M., Sherali, H., Shetty, C.M.: Nonlinear Programming, 3rd edn. Wiley, Hoboken (2006)
7. Beck, A., Ben-Tal, A., Tetruashvili, L.: A sequential parametric convex approximation method with applications to nonconvex truss topology design problems. J. Glob. Optim. **47**(1), 29–51 (2010)
8. Ben-Tal, A., Nemirovski, A.: Robust truss topology design via semidefinite programming. SIAM J. Optim. **7**(4), 991–1016 (1997)
9. Ben-Tal, A., Nemirovski, A.: Lectures on modern convex optimization. MPS/SIAM Series on Optimization. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2001)
10. Bendsøe, M.: Optimization of Structural Topology, Shape, and Material. Springer, Berlin (1995)
11. Bertsekas, D.: Nonlinear Programming, 2nd edn. Athena Scientific, Belmont (1999)
12. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
13. Bruggi, M.: On an alternative approach to stress constraints relaxation in topology optimization. Struct. Multidiscip. Optim. **36**(2), 125–141 (2008)
14. Cheng, G., Guo, X.: ε-relaxed approach in structural topology optimization. Struct. Optim. **13**(4), 258–266 (1997)
15. Clarke, F.H.: Optimization and nonsmooth analysis. In: Classics in Applied Mathematics, vol. 5, 2nd edn. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1990)
16. Fletcher, R.: Practical Methods of Optimization, 2nd edn. Wiley, New York (2001)
17. Herskovits, J.: Feasible direction interior-point technique for nonlinear optimization. J. Optim. Theory Appl. **99**(1), 121–146 (1998)
18. Herskovits, J., Mappa, P., Goulart, E., Mota Soares, C.M.: Mathematical programming models and algorithms for engineering design optimization. Comput. Methods Appl. Mech. Eng. **194**(30–33), 3244–3268 (2005)
19. Kang, Z., Bai, S.: On robust design optimization of truss structures with bounded uncertainties. Struct. Multidiscip. Optim. **47**(5), 699–714 (2013)
20. Landau, L., Lifshitz, E.M.: Theory of Elasticity. Butterworth Heinemann, Oxford (1986)
21. Le, C., Norato, J., Bruns, T., Ha, C., Tortorelli, D.: Stress-based topology optimization for continua. Struct. Multidiscip. Optim. **41**(4), 605–620 (2010)
22. Luenberger, D.G.: Linear and Nonlinear Programming, 2nd edn. Kluwer Academic Publishers, Boston (2003)
23. Nocedal, J., Wright, S.: Numerical optimization. In: Springer Series in Operations Research and Financial Engineering, 2nd edn. Springer, New York (2006)
24. París, J., Navarrina, F., Colominas, I., Casteleiro, M.: Topology optimization of continuum structures with local and global stress constraints. Struct. Multidiscip. Optim. **39**(4), 419–437 (2009)
25. Popov, E., Balan, T.: Engineering Mechanics of Solids. Prentice Hall, Upper Saddle River (1998)
26. Rozvany, G.: On design-dependent constraints and singular topologies. Struct. Multidiscip. Optim. **21**(2), 164–172 (2001)
27. Stolpe, M.: Global optimization of minimum weight truss topology problems with stress, displacement, and local buckling constraints using branch-and-bound. Int. J. Numer. Methods Eng. **61**(8), 1270–1309 (2004)
28. Stolpe, M., Svanberg, K.: On the trajectories of the epsilon-relaxation approach for stress-constrained truss topology optimization. Struct. Multidiscip. Optim. **21**(2), 140–151 (2001)
29. Stolpe, M., Svanberg, K.: A note on stress-constrained truss topology optimization. Struct. Multidiscip. Optim. **25**(1), 62–64 (2003)
30. Sturm, J.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. **11**(12), 625–653 (1999)
31. Wright, S.J., Tenny, M.J.: A feasible trust-region sequential quadratic programming algorithm. SIAM J. Optim. **14**(4), 1074–1105 (2004)
32. Zhu, Z., Jian, J.: An efficient feasible SQP algorithm for inequality constrained optimization. Nonlinear Anal. Real World Appl. **10**(2), 1220–1228 (2009)