

# Robust truss topology optimization via semidefinite programming with complementarity constraints: a difference-of-convex programming approach

Yoshihiro Kanno<sup>1</sup> 

Received: 20 February 2017 / Published online: 9 June 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

**Abstract** The robust truss topology optimization against the uncertain static external load can be formulated as mixed-integer semidefinite programming. Although a global optimal solution can be computed with a branch-and-bound method, it is very time-consuming. This paper presents an alternative formulation, semidefinite programming with complementarity constraints, and proposes an efficient heuristic. The proposed method is based upon the concave–convex procedure for difference-of-convex programming. It is shown that the method can often find a practically reasonable truss design within the computational cost of solving some dozen of convex optimization subproblems.

**Keywords** Robust optimization · Design-dependent load · Complementarity constraint · Semidefinite programming · Difference-of-convex programming · Concave–convex procedure

## 1 Introduction

Many studies have been done on robust optimization of structures against uncertainty in external loads. A possibilistic (or bounded-but-unknown) model of uncertainty, assuming only the set of values that the input data can possibly take, might be useful when reliable statistical property of uncertainty, which is required for a probabilistic model of uncertainty, is unavailable or imprecise. With a possibilistic model of uncer-

---

✉ Yoshihiro Kanno  
kanno@mist.i.u-tokyo.ac.jp

<sup>1</sup> Mathematics and Informatics Center, The University of Tokyo, Hongo 7-3-1, Tokyo 113-8656, Japan

tainty, design optimization considering structural robustness against the uncertainty is treated within the framework of robust optimization [5].

Attention of this paper is focused on robust topology optimization of truss structures against uncertainty in the static nodal external load.<sup>1</sup> Namely, we attempt to find a truss design that minimizes the worst-case compliance, i.e., the maximal value of the compliance among the specified set of external loads.<sup>2</sup> The seminal work of Ben-Tal and Nemirovski [7] shows that, based on the conventional ground structure method, this optimization problem can be formulated as *semidefinite programming* (SDP).<sup>3</sup> SDP is a class of convex optimization, and can be solved efficiently with a primal-dual interior-point method [3,60]. Closely related formulations of continuum-based robust structural optimization can be found in Cherkaev and Cherkaev [12,13], Calafiore and Dabbene [11], de Gournay et al. [16], Takezawa et al. [57], Brittain et al. [10], Hashimoto and Kanno [25] and Thore [58]. Furthermore, nonlinear SDP approaches to robust structural optimization have been proposed in Guo et al. [22,24], Kanno and Takewaki [33], and Holmberg et al. [26]. This paper discusses and deals with intrinsic difficulty in robust truss topology optimization.

Suppose that uncertain external forces can possibly be applied at any nodes, and that no external force is applied at an intermediate point of a truss member. If the set of existing nodes is specified, then the robust truss optimization problem can be recast as SDP [7]. In this approach, all the nodes in the specified set remain in the obtained solution. Also, the obtained solution can possibly have some nodes other than the specified ones, but at such extra nodes no uncertain external force is considered. Thus, it is difficult to predict in advance the set of existing nodes in the robust optimal truss. In other words, the uncertainty model of external forces should be treated as a design-dependent model [32]. This design dependency can be addressed by introducing 0–1 variables to represent the set of existing members in a truss design [61]. The robust truss topology optimization problem is then formulated as *mixed-integer semidefinite programming problem* (MISDP), which can be solved globally with a branch-and-bound method [61]. Unfortunately, due to large computational cost, this MISDP approach can be applied only to small-scale problem instances [61].

Another issue that has not been considered in literature on robust truss topology optimization [7,32,61] is the treatment of parallel consecutive members in the ground structure method. Specifically, in robust truss topology optimization with uncertain external load, overlapping members in the ground structure are not redundant.<sup>4</sup> Such non-redundancy of overlapping members has also been recognized in truss topology

<sup>1</sup> A truss is an assemblage of straight bars (called members) connected by pin-joints (called nodes) that do not transfer moment. See Sect. 2 for some concrete examples.

<sup>2</sup> The compliance of a truss, formally defined by (16), is equivalent to the twice strain energy of the truss at the equilibrium state under the prescribed boundary conditions (as far as the prescribed displacements are zeros [35]). It can be regarded as a global measure of the displacements, and hence by minimizing the compliance the global stiffness of the truss is maximized.

<sup>3</sup> The ground structure method is commonly used in truss topology optimization. It prepares an initial setting, called the ground structure, consisting of many members connected by nodes. The cross-sectional areas of the members are treated as design variables, while the locations of the nodes are specified. See Sect. 2 for more account.

<sup>4</sup> With reference to concrete examples, we will thoroughly discuss this issue in Sect. 2.

optimization considering the self-weight load [8,34] and the member buckling constraints [23,41]. Consider the conventional truss topology optimization. It is often that the optimal solution has parallel consecutive members that are connected by nodes supported only in the direction of those members. A sequence of such members is sometimes called a *chain* [1]. If only the compliance is considered as the structural performance, one can remove the intermediate nodes and replace the chain with a single longer member. This procedure is called the *hinge cancellation* [1,49]. Since the hinge cancellation does not change the compliance, overlapping of members in a ground structure can be removed in advance by deleting the longer member when two members overlap. In contrast, in robust truss optimization under load uncertainties, a solution having a chain is infeasible,<sup>5</sup> while the one stabilized by hinge cancellation can be feasible. This means that, in general, a global optimal truss topology cannot be captured without incorporating overlapping members in a ground structure. However, on the other hand, presence of overlapping members in a final truss design is not allowed from a practical point of view. Therefore, a special treatment is required in a robust optimization method to prohibit the presence of overlapping members.

This paper addresses the two difficulties in robust truss optimization explained above: the design dependency of the uncertainty model of the external load and the necessity of incorporating overlapping members in a ground structure. Both can be dealt with by introducing, for each member, a 0–1 design variable indicating whether the member vanishes or exists. Therefore, the robust truss topology optimization can be formulated as MISDP; see Sect. 4.2. However, as mentioned before, this approach is applicable only to problems of small size. In contrast, in this paper we attempt to propose a heuristic that can often find a feasible solution with a reasonable objective value. Through the numerical experiments with problem instances having up to about 700 members, it is shown that the proposed method usually converges after solving only a few dozen of convex optimization subproblems, and that it finds a practically reasonable solution.

This paper is partially inspired by papers of Jara-Moroni et al. [30] and Lipp and Boyd [39]. Jara-Moroni et al. [30] present a DC (difference-of-convex) programming approach to finding a stationary point of linear programming with complementarity constraints; see also Le Thi and Pham Dinh [38] and Muu et al. [43] for DC programming approaches to complementarity constraints. A function is said to be a DC (difference-of-convex) function if it can be represented as a difference of two convex functions. A DC programming problem is a minimization problem of a DC function under some inequality constraints, where all the constraint functions are DC functions. One of well-known heuristics for finding a local optimal solution of DC programming is the concave–convex procedure<sup>6</sup> [14,18,44,50,64]. Lipp and Boyd [39] show that an extension of the concave–convex procedure can serve as an efficient heuristic for diverse nonconvex optimization problems. For more account on the DC programming and the concave–convex procedure, see Sect. 3.1 and the references therein. In this paper, we first formulate the robust truss topology optimization as *semidefinite pro-*

<sup>5</sup> A solution having a chain cannot be in equilibrium with uncertain loads applied at intermediate nodes of the chain. Therefore, the worst-case compliance of the solution is infinitely large.

<sup>6</sup> The concave–convex procedure is also known as the convex–concave procedure [39,62].

*gramming with complementarity constraints* (SDPCC). Following an idea found in Jara-Moroni et al. [30], we recast this problem as a DC programming problem. A variant of the concave–convex procedure, which is similar to the one in Lipp and Boyd [39], is then applied to this DC programming formulation. Each iteration of the proposed method consists of solving an SDP problem.

The paper is organized as follows. In Sect. 2 we explain intrinsic difficulties in robust truss topology optimization by using some illustrative examples. Section 3 provides an overview of the necessary background of the DC programming and the concave–convex procedure, and presents the general framework of the algorithm used in this paper. Section 4 briefly reviews the existing MISDP formulation for robust truss topology optimization, and extends it to the problem setting with a ground structure incorporating overlapping members. Section 5 presents a new formulation, as well as a solution method, of robust truss topology optimization. Section 6 reports the results of numerical experiments. Conclusions are drawn in Sect. 7.

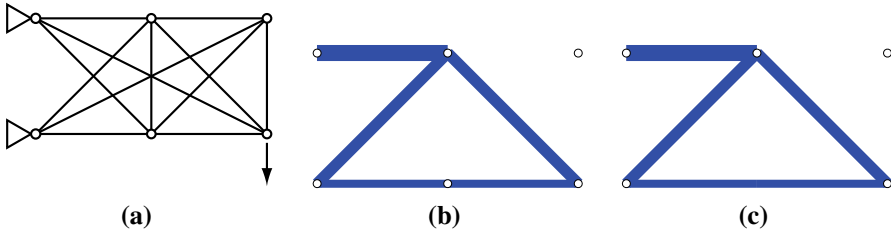
In our notation, we use  $\mathbf{x}^\top$  and  $X^\top$  to denote the transposes of vector  $\mathbf{x} \in \mathbb{R}^n$  and matrix  $X \in \mathbb{R}^{m \times n}$ , respectively. For vectors  $\mathbf{x} = (x_i) \in \mathbb{R}^n$  and  $\mathbf{y} = (y_i) \in \mathbb{R}^n$ , we write  $\mathbf{x} \geq \mathbf{y}$  if  $x_i \geq y_i$  ( $i = 1, \dots, n$ ). Particularly,  $\mathbf{x} \geq \mathbf{0}$  means  $x_i \geq 0$  ( $i = 1, \dots, n$ ). The Euclidean norm of  $\mathbf{x}$  is denoted by  $\|\mathbf{x}\| = \sqrt{\mathbf{x}^\top \mathbf{x}}$ . We use  $\mathbf{1} = (1, 1, \dots, 1)^\top$  to denote the all-ones vector. Let  $\mathcal{S}^n$  denote the set of  $n \times n$  real symmetric matrices. We write  $X \geq 0$  if  $X \in \mathcal{S}^n$  is positive semidefinite. We use  $\text{diag}(\mathbf{x})$  to denote a diagonal matrix, the vector of diagonal entries of which is  $\mathbf{x}$ . For a finite set  $T$ , we use  $|T|$  to denote the cardinality of  $T$ , i.e., the number of elements in  $T$ .

## 2 Motivation

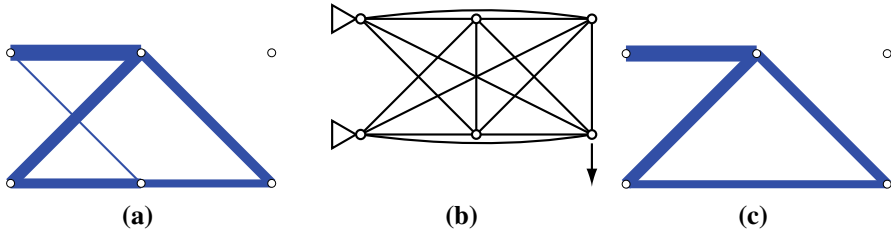
In this section, we explain intrinsic difficulties in robust truss topology optimization, which motivate us to develop the method proposed in this paper. Details of the examples in this section appear in Sect. 6.1.

Suppose that uncertain static external forces are possibly applied at all the nodes. The robust truss topology optimization is to find a truss design that minimizes the worst-case compliance, i.e., the maximum value of the compliance among possible realizations of the external load, under the upper bound constraint on the structural volume.

With reference to the examples in Figs. 1 and 2, we explain that incorporating overlapping members into a ground structure is necessary for the robust truss topology optimization. We begin with the conventional compliance minimization, without considering uncertainties. Figure 1a shows a ground structure, which consists of 12 members and has no overlapping members. The ground structure is an initial setting for truss topology optimization. The members consisting of a ground structure are called the candidate members, and their cross-sectional areas are design variables to be optimized. It is worth noting that the locations of the nodes are not treated as design variables. If the cross-sectional area of a member becomes equal to zero as a result of optimization, then the member is removed from the truss. Thus, the connectivity of members, called the topology in this research area, usually changes from the ground



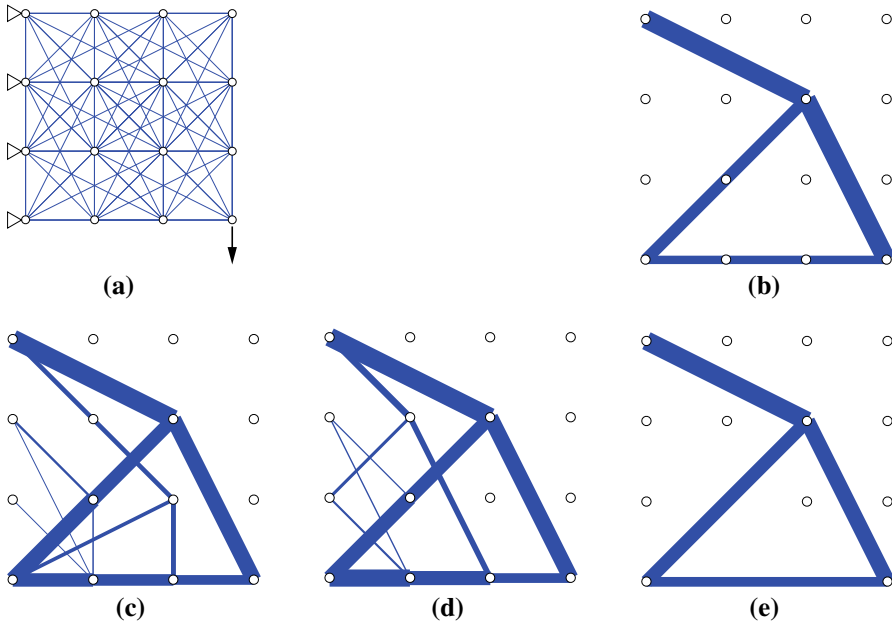
**Fig. 1** An example of the conventional compliance minimization. **a** The ground structure (with 12 members); **b** the optimal solution; and **c** the final truss design after hinge cancellation



**Fig. 2** The robust optimization corresponding to the example in Fig. 1. **a** The optimal solution obtained from the ground structure in Fig. 1a; **b** the ground structure including overlapping members (14 members in total); and **c** the global optimal solution

structure. Suppose that a vertical external load is applied at the rightmost bottom node, as shown in Fig. 1a. The optimal solution is shown in Fig. 1b, where the width of each member is proportional to its cross-sectional area. This solution has a chain consisting of two members. The hinge cancellation yields the final truss design shown in Fig. 1c. It should be clear that the objective value, i.e., the compliance, of the solution in Fig. 1c is same as the one in Fig. 1b. We next consider the robust optimization. Since uncertain external force is supposed to be applied also at the intermediate node of the chain, the solution in Fig. 1b becomes infeasible. As a result, the optimal solution has one additional member to stabilize that node, as shown in Fig. 2a. Alternatively, consider a ground structure in Fig. 2b, which consists of 14 members. The newly added two members, depicted as slightly curved lines, are in fact straight bars. Therefore, each of them overlaps with two shorter members, and is called an overlapping members. Moreover, the middle node is exactly located on a line forming an overlapping member. Therefore, we say that the overlapping member lies on the middle node. The robust optimal solution obtained from this ground structure is shown in Fig. 2c.<sup>7</sup> Namely, at the global optimal solution, the longer member is selected instead of the chain and the intermediate node of the chain is removed. Thus, overlapping members do not mean redundancy, because connection of two members via an intermediate node introduces an extra uncertain load applied to that node. Similar non-redundancy of overlapping members in a ground structure can be observed also in, e.g., the compliance minimization of trusses under the self-weight loads [8, 34].

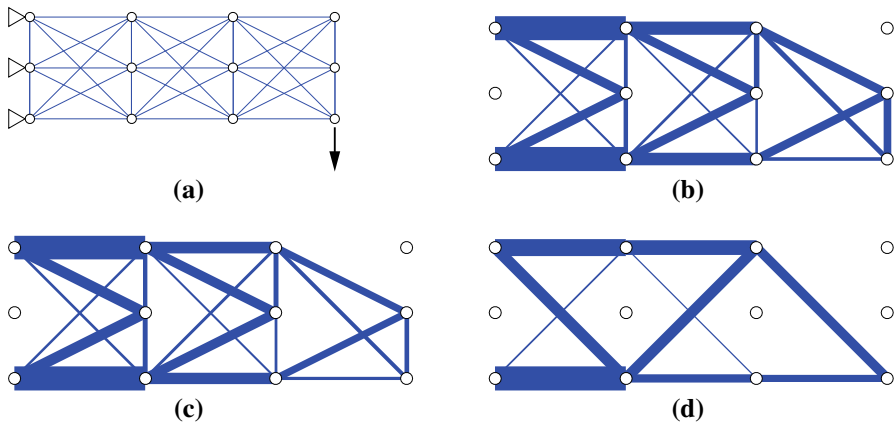
<sup>7</sup> It should be clear that the worst-case compliance for the solution in Fig. 2c is smaller than that for the solution in Fig. 2a.



**Fig. 3** Difficulties in the robust truss topology optimization. **a** The ground structure and the nominal external load; **b** the optimal solution for the nominal load; **c** a solution obtained for a design-independent uncertainty model of the external load; **d** the robust optimal solution for the design-dependent uncertainty model and the ground structure without overlapping members (the objective value is 3259.115 J); and **e** the robust optimal solution for the design-dependent uncertainty model and the ground structure with overlapping members (the objective value is 2442.708 J)

A key in this robust optimization is selecting the set of nodes which the optimal solution has. This is because the uncertainty in external loads depends on the set of existing nodes, in a manner that uncertain external forces are supposed to be applied to all the existing nodes. Moreover, a node lying on an existing member should be removed. The next example illustrates that exploring an optimal set of nodes in a heuristic manner is indeed difficult.

Consider the problem setting shown in Fig. 3a. Figure 3b shows the optimal solution of the nominal (i.e., not robust) optimization problem. A simple heuristic to predict a set of existing nodes in a robust optimal solution is to adopt the set of nodes that the nominal optimal solution has. Suppose that uncertain external forces are applied *only* to the five free nodes that the solution in Fig. 3b has. The optimal solution of this robust optimization problem is shown in Fig. 3c. This solution has two extra nodes that the solution in Fig. 3b does not have. Therefore, the solution in Fig. 3c assumes that external forces are *not* applied to these two nodes (in this sense, this solution is *not truly robust*). On the other hand, Fig. 3d shows the optimal solution of the robust topology optimization with a ground structure which does *not* include overlapping members. It is observed that one of the nodes in Fig. 3c is missing in the solution in Fig. 3d. Furthermore, Fig. 3e shows the optimal solution of the robust topology optimization with a ground structure *including* overlapping members. It is observed that the three intermediate nodes in Fig. 3b are removed and the two chains are replaced



**Fig. 4** A node which is not on a chain can also vanish as a result of the robust topology optimization. **a** The ground structure and the nominal external load; **b** the optimal solution for the nominal load; **c** the robust solution obtained for a design-independent uncertainty model of the external load (the objective value is 13934.896 J); **d** the robust optimal solution for the design-dependent uncertainty model (the objective value is 11093.750 J)

by longer members. As a result, the objective value of the solution in Fig. 3d is more than 1.33 times larger than that of the solution in Fig. 3e. Thus, it is crucial to design an optimization process so as to allow vanishment of intermediate nodes on chains.

It is also possible that a node which is *not* lying on a chain vanishes as a result of robust optimization. Consider the problem setting in Fig. 4a. The optimal solution of the nominal optimization problem is shown in Fig. 4b. If we suppose that uncertain external forces are applied to the eight free nodes in Fig. 4b, then the solution in Fig. 4c becomes optimal. Unlike the example in Fig. 3c, uncertain external forces are supposed to be applied to all the nodes that the solution in Fig. 4c has. In this sense, the solution in Fig. 4c is a local optimal solution of the robust topology optimization. In contrast, the global optimal solution of the robust topology optimization is shown in Fig. 4d.<sup>8</sup> It is observed that three nodes in Fig. 4c are missing in Fig. 4d. The objective value of the solution in Fig. 4c is more than 1.25 times larger than that of the solution in Fig. 4d. Thus, it is crucial to design an optimization algorithm that can deal with the design-dependent uncertainty model of the external load.

In Sects. 4 and 5, we propose a formulation and an algorithm for overcoming the difficulties discussed in this section.

### 3 Algorithmic framework

As preliminaries, Sect. 3.1 briefly introduces the notion of DC programming and the concave–convex procedure for solving it. In Sect. 3.2, we introduce the optimization

<sup>8</sup> In this example, overlapping longer members are *not* incorporated into the ground structure, because with overlapping members the global optimization method (YALMIP [40]) did not converge within realistic computational time.

problem that we consider in this paper, and present an extension of the concave–convex procedure.

### 3.1 Fundamentals: DC programming and concave–convex procedure

Let  $f_i, g_i : \mathbb{R}^n \rightarrow \mathbb{R}$  ( $i = 0, 1, \dots, m$ ) be convex. The optimization problem having the following form is called a DC programming problem:

$$\text{Minimize } f_0(\mathbf{x}) - g_0(\mathbf{x}) \tag{1a}$$

$$\text{subject to } f_i(\mathbf{x}) - g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m. \tag{1b}$$

For simplicity, we assume that  $g_0, g_1, \dots, g_m$  are differentiable.

The concave–convex procedure is known as a heuristic for finding a local optimal solution of problem (1). Let  $\mathbf{x}^{(k)} \in \mathbb{R}^n$  denote the incumbent value of  $\mathbf{x}$  at the  $k$ th iteration. Define  $\hat{g}_i(\cdot; \mathbf{x}^{(k)}) : \mathbb{R}^n \rightarrow \mathbb{R}$  ( $i = 0, 1, \dots, m$ ) by

$$\hat{g}_i(\mathbf{x}; \mathbf{x}^{(k)}) = g_i(\mathbf{x}^{(k)}) + \nabla g_i(\mathbf{x}^{(k)})^\top (\mathbf{x} - \mathbf{x}^{(k)}). \tag{2}$$

The concave–convex procedure updates the solution by letting  $\mathbf{x}^{(k+1)}$  be the optimal solution of the following optimization problem:

$$\text{Minimize } f_0(\mathbf{x}) - \hat{g}_0(\mathbf{x}; \mathbf{x}^{(k)}) \tag{3a}$$

$$\text{subject to } f_i(\mathbf{x}) - \hat{g}_i(\mathbf{x}; \mathbf{x}^{(k)}) \leq 0, \quad i = 1, \dots, m. \tag{3b}$$

It is worth noting that this subproblem is convex.

For the sequence,  $\{\mathbf{x}^{(k)}\}$ , generated by the concave–convex procedure, it is known that the objective value of (1), i.e.,  $\{f_0(\mathbf{x}^{(k)}) - g_0(\mathbf{x}^{(k)})\}$ , converges. However,  $\{\mathbf{x}^{(k)}\}$  does not necessarily converges to a local optimal solution; see, e.g., Lipp and Boyd [39, Sect. 1.3]. Applications of the concave–convex procedure include transductive support vector machines (transductive SVMs) [14, 18], feature selection in SVMs [44], sparse learning [19], etc.

The concave–convex procedure can be considered as a version of DCA (difference of convex algorithm) [45, 47]; see Lipp and Boyd [39] and Sriperumbudur and Lanckriet [50] for accounts of this fact. For DCA and its applications we direct the reader to Le Thi and Pham Dinh [37] and Pham Dinh and Le Thi [46]; an application of DCA in structural engineering can be found in Stavroulakis and Polyakova [52]. As shown in Sriperumbudur and Lanckriet [50], the concave–convex procedure can also be viewed as a variant of MM algorithms (majorization-minimization algorithms) [28, 36].<sup>9</sup> The MM algorithm is a generalization of the well-known EM algorithm (expectation-maximization algorithm) [15]. The MM algorithms have been frequently employed in machine learning and image processing as seen in, e.g, Hunter and Li [29], Hunter and Lange [27], Figueiredo et al. [17], Sriperumbudur et al. [51], and Sun et al. [54].

---

<sup>9</sup> In fact,  $-\hat{g}_i(\cdot; \mathbf{x}^{(k)})$  is a majorization function of  $-g_i$ .



MMA (the method of moving asymptotes) [55,56,65], which is frequently used for continuum-based topology optimization [9], also solves a sequence of convex optimization approximations of the original problem. MMA approximates the objective function and the constraint functions by convex linear fractional functions by using the function values and gradients as well as some parameters controlling the vertical asymptotes of the generated functions. Also, sequential parametric convex approximation methods with application to truss optimization can be found in Beck et al. [4] and Ben-Tal et al. [6].

### 3.2 Heuristic for convex optimization with complementarity constraints

In this paper we attempt to solve problems having the following form:

$$\text{Minimize } f(\xi, y, z) \tag{4a}$$

$$\text{subject to } (\xi, y, z) \in \Omega, \tag{4b}$$

$$y \geq \mathbf{0}, \tag{4c}$$

$$z \geq \mathbf{0}, \tag{4d}$$

$$y^\top z = 0. \tag{4e}$$

Here,  $f : \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  is convex,  $\Omega \subseteq \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^n$  is closed and convex, and the optimization variables are  $\xi \in \mathbb{R}^l$ ,  $y \in \mathbb{R}^n$ , and  $z \in \mathbb{R}^n$ . Problem (4) is convex optimization with complementarity constraints.

Following the idea in Jara-Moroni et al. [30], we can reduce problem (4) to a DC programming problem as follows. Consider a differentiable function  $\phi : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  satisfying

$$\phi(y, z) = 0 \iff y^\top z = 0,$$

$$\phi(y, z) \geq 0 \iff y \geq \mathbf{0}, z \geq \mathbf{0}.$$

The complementarity constraints, (4e), can be replaced by a penalization term as follows:

$$\text{Minimize } f(\xi, y, z) + \rho\phi(y, z) \tag{5a}$$

$$\text{subject to } (\xi, y, z) \in \Omega, \tag{5b}$$

$$y \geq \mathbf{0}, \tag{5c}$$

$$z \geq \mathbf{0}. \tag{5d}$$

Here,  $\rho > 0$  is a penalty parameter. For sufficiently large  $\rho$ , problem (5) is equivalent to problem (4). We next assume that  $\phi$  can be decomposed as

$$\phi(y, z) = \phi_+(y, z) - \phi_-(y, z),$$

where  $\phi_+$  and  $\phi_-$  are convex. Then problem (5) is reduced to the following form:

$$\text{Minimize } (f(\xi, \mathbf{y}, \mathbf{z}) + \rho\phi_+(\mathbf{y}, \mathbf{z})) - \rho\phi_-(\mathbf{y}, \mathbf{z}) \tag{6a}$$

$$\text{subject to } (\xi, \mathbf{y}, \mathbf{z}) \in \Omega, \tag{6b}$$

$$\mathbf{y} \geq \mathbf{0}, \tag{6c}$$

$$\mathbf{z} \geq \mathbf{0}. \tag{6d}$$

This is a DC programming problem, because  $f(\xi, \mathbf{y}, \mathbf{z}) + \rho\phi_+(\mathbf{y}, \mathbf{z})$  and  $\rho\phi_-(\mathbf{y}, \mathbf{z})$  are convex. There exist several different choices for  $\phi$ ,  $\phi_+$ , and  $\phi_-$  [30,38]. In this paper we adopt

$$\phi(\mathbf{y}, \mathbf{z}) = \|\mathbf{y} + \mathbf{z}\|^2 - \|\mathbf{y} - \mathbf{z}\|^2, \tag{7}$$

$$\phi_+(\mathbf{y}, \mathbf{z}) = \|\mathbf{y} + \mathbf{z}\|^2, \tag{8}$$

$$\phi_-(\mathbf{y}, \mathbf{z}) = \|\mathbf{y} - \mathbf{z}\|^2. \tag{9}$$

To solve problem (5), we apply the concave–convex procedure to problem (6) with gradually increasing the penalty parameter,  $\rho$ . For point  $(\mathbf{y}^{(k)}, \mathbf{z}^{(k)}) \in \mathbb{R}^n \times \mathbb{R}^n$ , define  $\hat{\phi}_-(\cdot; \mathbf{y}^{(k)}, \mathbf{z}^{(k)}) : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  by

$$\begin{aligned} \hat{\phi}_-(\mathbf{y}, \mathbf{z}; \mathbf{y}^{(k)}, \mathbf{z}^{(k)}) &= \phi_-(\mathbf{y}^{(k)}, \mathbf{z}^{(k)}) \\ &+ \nabla_{\mathbf{y}}\phi_-(\mathbf{y}^{(k)}, \mathbf{z}^{(k)})^\top (\mathbf{y} - \mathbf{y}^{(k)}) + \nabla_{\mathbf{z}}\phi_-(\mathbf{y}^{(k)}, \mathbf{z}^{(k)})^\top (\mathbf{z} - \mathbf{z}^{(k)}). \end{aligned} \tag{10}$$

The proposed algorithm updates the solution by letting  $(\xi^{(k+1)}, \mathbf{y}^{(k+1)}, \mathbf{z}^{(k+1)})$  be an optimal solution of the following convex optimization problem:

$$\text{Minimize } f(\xi, \mathbf{y}, \mathbf{z}) + \rho_k\phi_+(\mathbf{y}, \mathbf{z}) - \rho_k\hat{\phi}_-(\mathbf{y}, \mathbf{z}; \mathbf{y}^{(k)}, \mathbf{z}^{(k)}) \tag{11a}$$

$$\text{subject to } (\xi, \mathbf{y}, \mathbf{z}) \in \Omega, \tag{11b}$$

$$\mathbf{y} \geq \mathbf{0}, \tag{11c}$$

$$\mathbf{z} \geq \mathbf{0}. \tag{11d}$$

A reasonable stopping criterion is that the residual of the complementarity constraints is small enough, i.e.,

$$\phi(\mathbf{y}^{(k+1)}, \mathbf{z}^{(k+1)}) \leq \epsilon_1, \tag{12}$$

and the update of the incumbent solution is small enough, i.e.,

$$\|(\xi^{(k+1)}, \mathbf{y}^{(k+1)}, \mathbf{z}^{(k+1)}) - (\xi^{(k)}, \mathbf{y}^{(k)}, \mathbf{z}^{(k)})\| \leq \epsilon_2, \tag{13}$$

where  $\epsilon_1, \epsilon_2 > 0$  are thresholds. The algorithm is formally stated in Algorithm 1.<sup>10</sup>

<sup>10</sup> Choice of an initial point in the numerical experiments is explained in Sect. 6

---

**Algorithm 1** penalty concave–convex procedure for convex optimization with complementarity constraints

---

- Require:**  $\xi^{(0)} \in \mathbb{R}^l, y^{(0)} \in \mathbb{R}^n, z^{(0)} \in \mathbb{R}^n, \rho_0 > 0, \rho_{\max} > \rho_0,$  and  $\mu > 1.$   
 1:  $k \leftarrow 0.$   
 2: **repeat**  
 3:   Let  $(\xi^{(k+1)}, y^{(k+1)}, z^{(k+1)})$  be an optimal solution of problem (11).  
 4:    $\rho_{k+1} \leftarrow \min\{\mu\rho_k, \rho_{\max}\}.$   
 5:   Set  $k \leftarrow k + 1.$   
 6: **until** stopping criterion is satisfied.
- 

*Remark 1* Algorithm 1 is designed essentially based on the algorithm proposed by Lipp and Boyd [39] for solving problem (1). In their algorithm, the following sub-problem is solved to update  $\mathbf{x}^{(k)}$ :

$$\begin{aligned} &\text{Minimize} && f_0(\mathbf{x}) - \hat{g}_0(\mathbf{x}; \mathbf{x}^{(k)}) + \rho_k \sum_{i=1}^m s_i \\ &\text{subject to} && f_i(\mathbf{x}) - \hat{g}_i(\mathbf{x}; \mathbf{x}^{(k)}) \leq s_i, \quad i = 1, \dots, m, \\ &&& s_i \geq 0, \quad i = 1, \dots, m. \end{aligned}$$

Thus, penalization terms for all the constraints are added to the objective function by using the  $\ell_1$ -exact penalty function. In contrast, in Algorithm 1 only the complementarity constraints are penalized, and the other constraints of the original optimization problem are satisfied at a solution of the subproblem. □

## 4 Mixed-integer semidefinite programming formulation for robust truss topology optimization

In Sect. 4.1, we recall the existing MISDP formulation for robust truss topology optimization, without considering overlapping members in a ground structure. Section 4.2 presents treatment of overlapping members within the framework of MISDP.

### 4.1 Review of existing formulation

In this section we briefly review an MISDP formulation of the robust truss topology optimization under the load uncertainty [61]; see also Ben-Tal and Nemirovski [7].

Following the ground structure method, consider a truss consisting of candidate members connected by nodes. Let  $m$  and  $d$  denote the number of the members and the number of degrees of freedom of the nodal displacements,<sup>11</sup> respectively. We use  $x_i$  ( $i = 1, \dots, m$ ) to denote the member cross-sectional areas, which are design variables to be optimized. Throughout the paper, we assume small deformation and linear elasticity.

Let  $t_i \in \{0, 1\}$  be a variable that serves as an indicator of existence of member  $i$  such that  $t_i = 1$  means that member  $i$  exists and  $t_i = 0$  means that it vanishes. We

---

<sup>11</sup> The degrees of freedom of a truss are the possible components of the nodal displacements that define the configuration of the truss.

use  $\bar{x} > 0$  and  $\underline{x} \in [0, \bar{x}]$  to denote the specified upper and lower bounds for the cross-sectional area of an existing member, i.e.,  $x_i$  should satisfy  $x_i \in \{0\} \cup [\underline{x}, \bar{x}]$ . This constraint can be written by using  $t_i$  as

$$\underline{x}t_i \leq x_i \leq \bar{x}t_i. \tag{14}$$

We next introduce  $s_j \in \{0, 1\}$  ( $j = 1, \dots, d$ ) to represent the existence of the  $j$ th degree of freedom of the nodal displacements. A node in a ground structure is removed if and only if all the members connected to the nodes vanish. Let  $s_j = 1$  mean that the node having the  $j$ th degree of freedom exists, and  $s_j = 0$  mean that it vanishes. We use  $I(j) \subseteq \{1, \dots, m\}$  to denote the set of indices of the members connected to the node having the  $j$ th degree of freedom. Then  $s_j$  is related to  $t_1, \dots, t_m$  as follows:

$$t_i \leq s_j, \quad \forall i \in I(j). \tag{15}$$

Let  $K(\mathbf{x}) \in \mathbb{R}^{d \times d}$  denote the stiffness matrix of a truss, which is a (matrix-valued) linear function of  $\mathbf{x}$ . For a given external load, denoted  $\mathbf{p} \in \mathbb{R}^d$ , the compliance of the truss is defined by

$$\pi(\mathbf{x}; \mathbf{p}) = \sup\{2\mathbf{p}^\top \mathbf{u} - \mathbf{u}^\top K(\mathbf{x})\mathbf{u} \mid \mathbf{u} \in \mathbb{R}^d\}. \tag{16}$$

The conventional compliance minimization problem is formulated in variable  $\mathbf{x}$  as follows:

$$\text{Minimize } \pi(\mathbf{x}; \mathbf{p}) \tag{17a}$$

$$\text{subject to } \mathbf{x} \geq \mathbf{0}, \tag{17b}$$

$$\mathbf{c}^\top \mathbf{x} \leq \bar{c}. \tag{17c}$$

Here,  $c_i$  is the undeformed length of member  $i$ ,  $\mathbf{c} = (c_1, \dots, c_m)^\top$ , and  $\bar{c} > 0$  is the specified upper bound for the structural volume.

The uncertainty model of the external load is defined as follows: Let  $\tilde{\mathbf{p}} \in \mathbb{R}^d$  denote the nominal value (or the best estimate) of the external load. Define a constant matrix  $Q \in \mathbb{R}^{d \times d}$  by

$$Q = \left[ \begin{array}{c|c|c|c} \tilde{\mathbf{p}} & \alpha \mathbf{q}_1 & \cdots & \alpha \mathbf{q}_{d-1} \end{array} \right],$$

where  $\mathbf{q}_1, \dots, \mathbf{q}_{d-1} \in \mathbb{R}^d$  are the orthonormal basis vectors of the orthogonal complement of  $\tilde{\mathbf{p}}$ , and  $\alpha > 0$  is a constant representing the level (or the magnitude) of uncertainty. Then the uncertainty set of the external load, i.e., the set of all possible realizations of the external load, is defined by

$$P(s) = \{\text{diag}(s)Q\mathbf{e} \mid \|\mathbf{e}\| \leq 1\}. \tag{18}$$

For example, suppose that  $\tilde{\mathbf{p}}$  has only one nonzero entry. Then, without loss of generality we can assume  $\tilde{p}_1 \neq 0$ , and we have

$$\tilde{\mathbf{p}} = \begin{bmatrix} \tilde{p}_1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad Q = \begin{bmatrix} \tilde{p}_1 & 0 & 0 & \cdots & 0 \\ 0 & \alpha & 0 & \cdots & 0 \\ 0 & 0 & \alpha & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \alpha \end{bmatrix}. \tag{19}$$

Let  $J_f \subseteq \{1, \dots, d\}$  denote the set of indices of nonzero entries of  $\tilde{\mathbf{p}}$ , i.e.,

$$J_f = \{j \in \{1, \dots, d\} \mid \tilde{p}_j \neq 0\}.$$

The nodes to which the nominal external load,  $\tilde{\mathbf{p}}$ , is applied should not be removed in the course of optimization. Therefore, we impose the following constraint:

$$s_j = 1, \quad \forall j \in J_f. \tag{20}$$

In robust optimization, we attempt to find a truss design that minimizes the maximal compliance (i.e., the worst-case compliance) when the external load can take any value in  $P(\mathbf{s})$ . With reference to (14), (15), (17), (18), and (20), we can see that this optimization problem can be formulated as follows:

$$\text{Minimize} \quad \sup\{\pi(\mathbf{x}; \mathbf{p}) \mid \mathbf{p} \in P(\mathbf{s})\} \tag{21a}$$

$$\text{subject to} \quad s_j = 1, \quad \forall j \in J_f, \tag{21b}$$

$$t_i \leq s_j, \quad \forall i \in I(j); \quad j = 1, \dots, d, \tag{21c}$$

$$\underline{\mathbf{x}}\mathbf{t} \leq \mathbf{x} \leq \bar{\mathbf{x}}\mathbf{t}, \tag{21d}$$

$$\mathbf{c}^\top \mathbf{x} \leq \bar{c}, \tag{21e}$$

$$\mathbf{s} \in \{0, 1\}^d, \tag{21f}$$

$$\mathbf{t} \in \{0, 1\}^m. \tag{21g}$$

For  $\mathbf{x} \in \mathbb{R}^m$  ( $\mathbf{x} \geq \mathbf{0}$ ),  $\mathbf{s} \in \{0, 1\}^d$ , and  $w \in \mathbb{R}$ , define  $W(\mathbf{x}, \mathbf{s}, w) \in \mathcal{S}^{d+1}$  by

$$W(\mathbf{x}, \mathbf{s}, w) = \begin{bmatrix} wI & (\text{diag}(\mathbf{s})Q)^\top \\ \text{diag}(\mathbf{s})Q & K(\mathbf{x}) \end{bmatrix}.$$

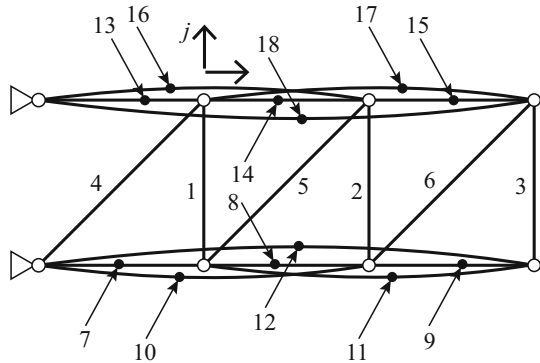
It is shown in Ben-Tal and Nemirovski [7, Lemma 2.2] that  $w \in \mathbb{R}$  satisfies

$$w \geq \sup\{\pi(\mathbf{x}; \mathbf{p}) \mid \mathbf{p} \in P(\mathbf{s})\}$$

if and only if

$$\begin{bmatrix} wI & (\text{diag}(\mathbf{s})Q)^\top \\ \text{diag}(\mathbf{s})Q & K(\mathbf{x}) \end{bmatrix} \succeq 0 \tag{22}$$

**Fig. 5** A ground structure consisting of 18 members



holds. Consequently, problem (21) is equivalently rewritten as follows:

$$\text{Minimize } w \tag{23a}$$

$$\text{subject to } W(\mathbf{x}, \mathbf{s}, w) \geq 0, \tag{23b}$$

$$s_j = 1, \quad \forall j \in J_f, \tag{23c}$$

$$t_i \leq s_j, \quad \forall i \in I(j); j = 1, \dots, d, \tag{23d}$$

$$\underline{\mathbf{x}}\mathbf{t} \leq \mathbf{x} \leq \overline{\mathbf{x}}\mathbf{t}, \tag{23e}$$

$$\mathbf{c}^\top \mathbf{x} \leq \bar{c}, \tag{23f}$$

$$\mathbf{s} \in \{0, 1\}^d, \tag{23g}$$

$$\mathbf{t} \in \{0, 1\}^m. \tag{23h}$$

Problem (23) is an MISDP problem. By relaxing the 0–1 constraints into linear inequality constraints, we obtain an SDP relaxation. Since SDP can be solved efficiently with a primal-dual interior-point method, we can find a global optimal solution of problem (23) with a branch-and-bound method [61].

### 4.2 Treatment of members lying on a line

As explained in Sect. 2, for the robust truss topology optimization it is necessary to incorporate overlapping members to a ground structure. Since the existence of overlapping members in a final truss design is not accepted, it is required to incorporate the constraints prohibiting the presence of overlapping members in a truss design. To the best of the author’s knowledge, such a consideration cannot be found in literature on robust truss topology optimization.

Recall that, for each  $j = 1, \dots, d$ ,  $s_j = 1$  means that the node having the  $j$ th degree of freedom exists, and  $s_j = 0$  means that it vanishes. Also,  $t_i = 1$  means that member  $i$  exists and  $t_i = 0$  means that it vanishes. Let  $L(j) \subseteq \{1, \dots, m\}$  denote the set of indices of the members lying on the node having the  $j$ th degree of freedom. Figure 5 shows an example of ground structure with overlapping members. It has  $m = 18$  members and  $d = 12$  degrees of freedom of the nodal displacements. In this

example, we have  $L(j) = \{16, 18\}$  and  $I(j) = \{1, 4, 13, 14, 17\}$ . If  $s_j = 1$ , then all the members in  $L(j)$  cannot exist, i.e.,  $t_i = 0$  ( $\forall i \in L(j)$ ). Also, if there exists an  $i \in L(j)$  such that  $t_i = 1$ , then the corresponding node cannot exist, i.e.,  $s_j = 0$ . These two conditions can be formulated as

$$s_j \leq 1 - t_i, \quad \forall i \in L(j). \quad (24)$$

In the following, we add (24) to problem (23).

It is worth noting that a truss design involving a chain is infeasible for the presented robust optimization problem, because it is unstable and cannot be in equilibrium with uncertain forces applied at an intermediate node of the chain. Therefore, a solution obtained by the proposed method does not involve a member that is longer than the maximum member length of the ground structure. This might be considered an advantage of the robust topology optimization, because in a conventional truss topology optimization an optimal solution may possibly have a long chain and special treatment, such as the local buckling constraints [23,41], is necessary for avoiding presence of a too long member.

## 5 Simple heuristic for robust truss topology optimization

The MISDP approach presented in Sect. 4 is only applicable to small-scale problem instances. Alternatively, in this section we present a heuristic having small computational cost. In Sect. 5.1, we reformulate our robust truss topology optimization as SDPCC. A concave–convex procedure is then applied to this formulation in Sect. 5.2.

### 5.1 Formulation as semidefinite programming with complementarity constraints

In this section, we reformulate problem (23) with constraint (24) as SDPCC, which is well suited for applying the concave–convex procedure.

We begin with constraints (23d) and (23e), which describe the relation between  $s_j$  and  $\mathbf{x}$ . Let  $r_j$  ( $j = 1, \dots, d$ ) denote the sum of the cross-sectional areas of the members that are connected to the node having the  $j$ th degree of freedom, i.e.,

$$r_j = \sum_{i \in I(j)} x_i. \quad (25)$$

Observe that  $r_j > 0$  implies  $s_j = 1$ , because at least one member connected to the corresponding node exists and, hence, the node should exist. Also,  $s_j = 0$  implies that the corresponding node vanishes, and hence  $r_j = 0$ , i.e., all the members connected to the node should vanish. These two assertions can be written as

$$(1 - s_j)r_j = 0. \quad (26)$$

Namely, constraint (23d) can be replaced with (26). For notational simplicity, in the following we write (25) as

$$r = R\mathbf{x}$$

with a constant matrix  $R \in \mathbb{R}^{d \times m}$ .

We next consider constraints (23e) and (24), which describe the relation between  $s_j$  and  $\mathbf{x}$ . Let  $v_j$  ( $j = 1, \dots, d$ ) denote the sum of the cross-sectional areas of the members that lying across the node with the  $j$ th degree of freedom, i.e.,

$$v_j = \sum_{i \in L(j)} x_i. \tag{27}$$

Observe that  $v_j > 0$  implies  $s_j = 0$ , because at least one member lying across the corresponding node exists and, thence, the node should vanish. Also,  $s_j = 1$  implies that the corresponding node exists, and hence all the members lying across the node should vanish, i.e.,  $v_j = 0$ . These two assertions can be written as

$$s_j v_j = 0. \tag{28}$$

Namely, constraint (24) can be replaced with (28). For notational simplicity, we write (27) as

$$\mathbf{v} = V\mathbf{x}$$

by using a constant matrix  $V \in \mathbb{R}^{d \times m}$ .

Finally, consider constraint (23e). For each  $i = 1, \dots, m$ , we introduce a new variable  $z_i \in \mathbb{R}$  so that  $\underline{x} - z_i$  corresponds to the lower bound for the cross-sectional area of member  $i$ . If  $x_i > 0$ , then member  $i$  exists and the lower bound should be  $\underline{x}$ , which means  $z_i = 0$ . Also, when the lower bound becomes smaller than  $\underline{x}$  (i.e.,  $z_i > 0$ ), then member  $i$  should vanish (i.e.,  $x_i = 0$ ), and hence we set  $z_i = \underline{x}$ . These relations can be written as follows:

$$\underline{x} - z_i \leq x_i \leq \bar{x}, \tag{29}$$

$$0 \leq z_i \leq \underline{x}, \tag{30}$$

$$x_i z_i = 0. \tag{31}$$

Consequently, constraints (23e) and (23h) can be replaced with (29), (30), and (31).

The upshot is that problem (23) incorporating constraint (24) is equivalently rewritten as follows:

$$\text{Minimize } w \tag{32a}$$

$$\text{subject to } W(\mathbf{x}, \mathbf{s}, w) \geq 0, \tag{32b}$$

$$s_j = 1, \quad \forall j \in J_f, \tag{32c}$$



$$\mathbf{r} = R\mathbf{x}, \tag{32d}$$

$$\mathbf{v} = V\mathbf{x}, \tag{32e}$$

$$\mathbf{0} \leq \mathbf{s} \leq \mathbf{1}, \tag{32f}$$

$$\underline{x}\mathbf{1} - \mathbf{z} \leq \mathbf{x} \leq \bar{x}\mathbf{1}, \tag{32g}$$

$$\mathbf{0} \leq \mathbf{z} \leq \underline{x}\mathbf{1}, \tag{32h}$$

$$\mathbf{c}^\top \mathbf{x} \leq \bar{c}, \tag{32i}$$

$$(1 - s_j)r_j = 0, \quad j = 1, \dots, d, \tag{32j}$$

$$s_j v_j = 0, \quad j = 1, \dots, d, \tag{32k}$$

$$x_i z_i = 0, \quad i = 1, \dots, m. \tag{32l}$$

Here,  $\mathbf{x}$ ,  $\mathbf{z}$ ,  $\mathbf{s}$ ,  $\mathbf{r}$ ,  $\mathbf{v}$ , and  $w$  are variables to be optimized. Observe that any feasible solution of problem (32) satisfies  $1 - s_j \geq 0$ ,  $r_j \geq 0$ ,  $s_j \geq 0$ ,  $v_j \geq 0$ ,  $x_i \geq 0$ , and  $z_i \geq 0$ . Therefore, constraints (32j), (32k), and (32l) are complementarity constraints. Constraint (32b) is a linear matrix inequality constraint in terms of  $\mathbf{x}$ ,  $\mathbf{s}$ , and  $w$ . Thus, problem (32) has the form of SDPCC.

*Remark 2* Since the algorithm presented in this paper consists of sequential approximation, adding some linear inequalities may possibly limit the search space and enhance the convergence. In the following, we consider linear valid inequalities, which naturally stem from the complementarity constraints and can be handled effectively in the numerical solution. Suppose that two variables,  $\alpha, \beta \in \mathbb{R}$ , are subjected to the complementarity constraint and their upper bounds are given, i.e.,

$$0 \leq \alpha \leq \bar{\alpha}, \tag{33}$$

$$0 \leq \beta \leq \bar{\beta}, \tag{34}$$

$$\alpha\beta = 0, \tag{35}$$

where  $\bar{\alpha}$  and  $\bar{\beta}$  are positive constants. It is known that the inequality

$$\bar{\beta}\alpha + \bar{\alpha}\beta \leq \bar{\alpha}\bar{\beta}$$

serves as a valid constraint for (33), (34), and (35) [42,63]. In the same manner, we can construct valid constraints for problem (32). Concerning (32j), observe that we obtain

$$\begin{aligned} 1 - s_j &\leq 1, \\ r_j &\leq \bar{x}|I(j)| \end{aligned}$$

from (32f) and (25), respectively. Therefore, the constraints

$$-\bar{x}|I(j)|s_j + r_j \leq 0, \quad j = 1, \dots, d \tag{36}$$

are valid for (32j). Similarly, inequalities

$$\bar{x}|L(j)|s_j + v_j \leq \bar{x}|L(j)|, \quad j = 1, \dots, d, \tag{37}$$

$$\underline{x}x_i + \bar{x}z_i \leq \underline{x}\bar{x}, \quad i = 1, \dots, m \tag{38}$$

are valid constraints for (32k) and (32l), respectively. In the following, we add constraints (36), (37), and (38) to problem (32).  $\square$

### 5.2 Penalty concave–convex procedure for robust truss topology optimization

Problem (32) has the form of problem (4) studied in Sect. 3.2. To see this, it is convenient to rewrite problem (32) as follows:

$$\text{Minimize } w \tag{39a}$$

$$\text{subject to } (\mathbf{x}, \mathbf{z}, \mathbf{s}, \mathbf{r}, \mathbf{v}, w) \in F, \tag{39b}$$

$$\mathbf{1} - \mathbf{s} \geq \mathbf{0}, \quad \mathbf{r} \geq \mathbf{0}, \quad (\mathbf{1} - \mathbf{s})^\top \mathbf{r} = 0, \tag{39c}$$

$$\mathbf{s} \geq \mathbf{0}, \quad \mathbf{v} \geq \mathbf{0}, \quad \mathbf{s}^\top \mathbf{v} = 0, \tag{39d}$$

$$\mathbf{x} \geq \mathbf{0}, \quad \mathbf{z} \geq \mathbf{0}, \quad \mathbf{x}^\top \mathbf{z} = 0. \tag{39e}$$

Here,  $F$  is defined by

$$\begin{aligned} F = \{(\mathbf{x}, \mathbf{z}, \mathbf{s}, \mathbf{r}, \mathbf{v}, w) \mid & W(\mathbf{x}, \mathbf{s}, w) \geq 0, \mathbf{r} = R\mathbf{x}, \mathbf{v} = V\mathbf{x}, \\ & \mathbf{0} \leq \mathbf{s} \leq \mathbf{1}, s_j = 1 \ (\forall j \in J_f), \\ & \underline{x}\mathbf{1} - \mathbf{z} \leq \mathbf{x} \leq \bar{x}\mathbf{1}, \mathbf{0} \leq \mathbf{z} \leq \underline{x}\mathbf{1}, \mathbf{c}^\top \mathbf{x} \leq \bar{c}, \\ & -\bar{x}|I(j)|s_j + r_j \leq 0 \ (j = 1, \dots, d), \\ & \bar{x}|L(j)|s_j + v_j \leq \bar{x}|L(j)| \ (j = 1, \dots, d), \\ & \underline{x}\mathbf{x} + \bar{x}\mathbf{z} \leq \underline{x}\bar{x}\mathbf{1}\}, \end{aligned}$$

which is a convex set. The complementarity constraints in (39c), (39d), and (39e) can be replaced with penalization terms added to the objective function as follows:

$$\text{Minimize } w + \rho\phi(\mathbf{1} - \mathbf{s}, \mathbf{r}) + \rho\phi(\mathbf{s}, \mathbf{v}) + \rho\phi(\mathbf{x}, \mathbf{z}) \tag{40a}$$

$$\text{subject to } (\mathbf{x}, \mathbf{z}, \mathbf{s}, \mathbf{r}, \mathbf{v}, w) \in F. \tag{40b}$$

Here,  $\rho > 0$  is a sufficiently large penalty parameter, and  $\phi$  has been defined by (7). Problem (40) has the same form as problem (5).

We are now in position to apply Algorithm 1 to problem (40). Algorithm 1 solves the subproblem in (11), which is explicitly written as follows:

$$\begin{aligned} \text{Minimize } & w + \rho_k \phi_+(\mathbf{1} - \mathbf{s}, \mathbf{r}) + \rho_k \phi_+(\mathbf{s}, \mathbf{v}) + \rho_k \phi_+(\mathbf{x}, \mathbf{z}) \\ & - \rho_k \hat{\phi}_-(\mathbf{1} - \mathbf{s}, \mathbf{r}; \mathbf{1} - \mathbf{s}^{(k)}, \mathbf{r}^{(k)}) - \rho_k \hat{\phi}_-(\mathbf{s}, \mathbf{v}; \mathbf{s}^{(k)}, \mathbf{v}^{(k)}) \\ & - \rho_k \hat{\phi}_-(\mathbf{x}, \mathbf{z}; \mathbf{x}^{(k)}, \mathbf{z}^{(k)}) \end{aligned} \tag{41a}$$

$$\text{subject to } (\mathbf{x}, \mathbf{z}, \mathbf{s}, \mathbf{r}, \mathbf{v}, w) \in F. \tag{41b}$$

Here,  $\phi_+$  and  $\hat{\phi}_-$  are defined by (8) and (10), respectively. Since the constant terms in the objective function can be neglected, problem (41) can be reduced to the following problem:

$$\begin{aligned} \text{Minimize } & w + \rho_k (\|\mathbf{x} + \mathbf{z}\|^2 + \|\mathbf{1} - \mathbf{s} + \mathbf{r}\|^2 + \|\mathbf{s} + \mathbf{v}\|^2) \\ & - 2\rho_k (\mathbf{x}^{(k)} - \mathbf{z}^{(k)})^\top \mathbf{x} - 2\rho_k (\mathbf{z}^{(k)} - \mathbf{x}^{(k)})^\top \mathbf{z} \\ & - 2\rho_k (2\mathbf{s}^{(k)} + \mathbf{r}^{(k)} - \mathbf{v}^{(k)} - \mathbf{1})^\top \mathbf{s} \\ & - 2\rho_k (\mathbf{s}^{(k)} + \mathbf{r}^{(k)} - \mathbf{1})^\top \mathbf{r} - 2\rho_k (\mathbf{v}^{(k)} - \mathbf{s}^{(k)})^\top \mathbf{v} \end{aligned} \tag{42a}$$

$$\text{subject to } (\mathbf{x}, \mathbf{z}, \mathbf{s}, \mathbf{r}, \mathbf{v}, w) \in F. \tag{42b}$$

Problem (42) is a minimization problem of a convex quadratic function under a linear matrix inequality constraint. Hence, this problem can be recast as SDP. Thus, at each iteration of Algorithm 1 we solve an SDP problem.

As mentioned in Sect. 3.2, a reasonable stopping criterion is that (12) and (13) are satisfied. In practice, however, we might use a relaxed criterion, which may save some iterations before convergence. Specifically, we terminate the algorithm when either (12) or

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| \leq \epsilon_2 \tag{43}$$

is satisfied. Then, by using the obtained solution we can fix the set of existing members and the set of existing nodes. Fixing these sets means that one variable in all the complementarity constraints in problem (32) is fixed. Therefore, the problem now becomes SDP, which is to be solved as the post process. The solutions presented in Sect. 6 are obtained in this manner.

## 6 Numerical experiments

This section reports three numerical experiments.

The proposed algorithm was implemented in MATLAB ver. 9.0. At each iteration we solved an SDP problem in (42) by using CVX, a MATLAB package for specifying and solving convex optimization problems [20,21]. SDPT3 ver. 4.0 [59] was used as a solver. Computation was carried out on a 2.2 GHz Intel Core i5 processor with 8 GB RAM. The Young modulus of the trusses in the following numerical examples is 20 GPa.

The initial point for Algorithm 1 is chosen as follows. We first solve problem (17) for the nominal external load  $\tilde{\mathbf{p}}$ , i.e., the compliance minimization without considering uncertainties,<sup>12</sup> and let  $\mathbf{x}^{(0)}$  be the obtained optimal solution. The initial values for the other variables are given by  $\mathbf{z}^{(0)} = \mathbf{0}$ ,  $\mathbf{s}^{(0)} = \mathbf{1}/2$ ,  $\mathbf{r}^{(0)} = R\mathbf{x}^{(0)}$ , and  $\mathbf{v}^{(0)} = V\mathbf{x}^{(0)}$ . The parameters of Algorithm 1 are  $\rho_0 = 10^{-2}$ ,  $\rho_{\max} = 10^6$ , and  $\mu = 1.5$ . We terminate Algorithm 1 if either (12) or (43) is satisfied, where  $\epsilon_1 = 2m \times 10^{-2} \text{ mm}^2$  and  $\epsilon_2 = 10^{-2} \text{ mm}^2$ . Then, as explained in Sect. 5.2, we fix one variable in all the complementarity constraints in problem (32), and solve the resulting SDP problem to obtain the final solution. The settings of the initial point and the parameters explained above were determined by preliminary numerical experiments. In Sect. 6.1, we consider three problem instances that could be solved with a global optimization method. Cantilever truss examples with two different loading conditions, which are frequently solved in structural optimization, are considered in Sects. 6.2 and 6.3.

### 6.1 Example (I): Comparison with global optimization

In this section, we consider the small-scale instances presented in Sect. 2. For comparison, the MISDP formulation, i.e., problem (23) with constraint (24), is solved with YALMIP [40]. YALMIP finds a global optimal solution of an MISDP problem with a branch-and-bound method, at each iteration of which an SDP problem is solved. We used YALMIP with the default setting, where SDP subproblems are solved with SeDuMi ver. 1.3 [48,53].

Consider the problem settings in Figs. 2, 3, and 4. Table 1 lists the number of members ( $m$ ), the number of degrees of freedom of displacements ( $d$ ), and the upper bound for the structural volume ( $\bar{c}$ ). In Figs. 2b and 3a, the nodes are aligned on a  $1 \text{ m} \times 1 \text{ m}$  grid. In Fig. 4a, we use a  $1 \text{ m} \times 0.5 \text{ m}$  grid. In Fig. 3a, the ground structure has all possible members connecting two nodes but are no longer than 3 m. The nominal external load,  $\tilde{\mathbf{p}}$ , is applied as shown in Figs. 2b, 3a and 4a. The uncertainty model of the external load is defined by using (19) with  $\tilde{p}_1 = 100 \text{ kN}$ ,  $\alpha = 0.75\tilde{p}_1$  for Figs. 2 and 3, and  $\alpha = 0.5\tilde{p}_1$  for Fig. 4. The lower and upper bounds for the member cross-sectional areas are  $\underline{x} = 1 \text{ mm}^2$  and  $\bar{x} = 700 \text{ mm}^2$ , respectively. In Table 1, “rob. opt.” reports the optimal value, obtained by YALMIP, of the robust optimization problem. The obtained solutions are shown in Figs. 2c, 3e and 4d. For reference, the optimal solutions of the (not robust) compliance minimization with the nominal external load,  $\tilde{\mathbf{p}}$ , are shown in Figs. 1b, 3b and 4b.<sup>13</sup> The optimal values are listed in “nom. opt.” of Table 1.

It is remarkable that, for every problem instance, the solution obtained by the proposed algorithm coincides with the global optimal solution (obtained by YALMIP).

<sup>12</sup> Problem (17) is convex. Various reformulations are known in literature; see, e.g., Achtziger et al. [2] and Jarre et al. [31]. For example, replacing  $\text{diag}(s)Q$  in (22) with  $\tilde{\mathbf{p}}$ , one can readily obtain SDP that minimizes  $w$  under constraint  $\begin{bmatrix} w & \tilde{\mathbf{p}}^T \\ \tilde{\mathbf{p}} & K(\mathbf{x}) \end{bmatrix} \succeq 0$ , (17b), and (17c). This formulation was used in the numerical experiments. It should be clear that a ground structure with overlapping members is used for generating the initial point,  $\mathbf{x}^{(0)}$ .

<sup>13</sup> Ground structures without overlapping members are used to obtain the solutions in Figs. 1b, 3b and 4b.

**Table 1** Characteristics of the problem instances in example (I)

Problem	$m$	$d$	$\bar{c}$ (mm <sup>3</sup> )	Rob. opt. (J)	Nom. opt. (J)
Figure 2	14	8	$0.4 \times 10^6$	8984.375	8000.000
Figure 3	98	24	$1.8 \times 10^6$	2442.708	2006.944
Figure 4	35	18	$1.2 \times 10^6$	11093.750	9375.000

**Table 2** Computational costs for example (I)

Problem	Proposed method		YALMIP	
	#iter.	Time (s)	#iter.	Time (s)
Figure 2	3	4.1	13	2.7
Figure 3	15	39.8	92	54.7
Figure 4	47	59.5	1141	300.3

**Fig. 6** Example (II). The problem setting for  $(N_X, N_Y) = (6, 2)$

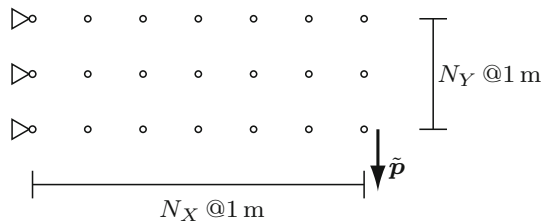


Table 2 reports the computational costs of the two methods, where “#iter.” is the number of iterations, and “time” is the required computational time. Note that the computational cost of the proposed method does not include the ones for generation of an initial point and for the post-processing. It is also worth noting that the problem size of SDP solved at each iteration of the proposed method is larger than that of YALMIP, and hence the computational time per an iteration required by the proposed method is larger than that of YALMIP. For every instance, the number of SDP problems solved by the proposed method is smaller than that of YALMIP.

In the experiments in this section, it has been observed that the proposed method converges to a global optimal solution for a small-scale problem instance. In Sects. 6.2 and 6.3, we examine large-scale instances that cannot be solved with a global optimization method within realistic computational time.

### 6.2 Example (II)

Consider the ground structures shown in Fig. 6. The nodes are aligned on a 1 m × 1 m grid, and the number of the nodes is  $(N_X + 1)(N_Y + 1)$ . The leftmost nodes are pin-supported. The candidate members are defined as follows. We first generate all possible members such that any two nodes are connected by a member. Then we

**Table 3** Characteristics of the problem instances in example (II)

$(N_X, N_Y)$	$m$	$d$	$\bar{c}$ (mm <sup>3</sup> )
(3, 7)	250	48	$4.2 \times 10^6$
(4, 6)	292	56	$4.8 \times 10^6$
(5, 5)	306	60	$5.0 \times 10^6$
(6, 4)	292	60	$4.8 \times 10^6$
(7, 3)	250	56	$4.2 \times 10^6$
(8, 2)	180	48	$3.2 \times 10^6$

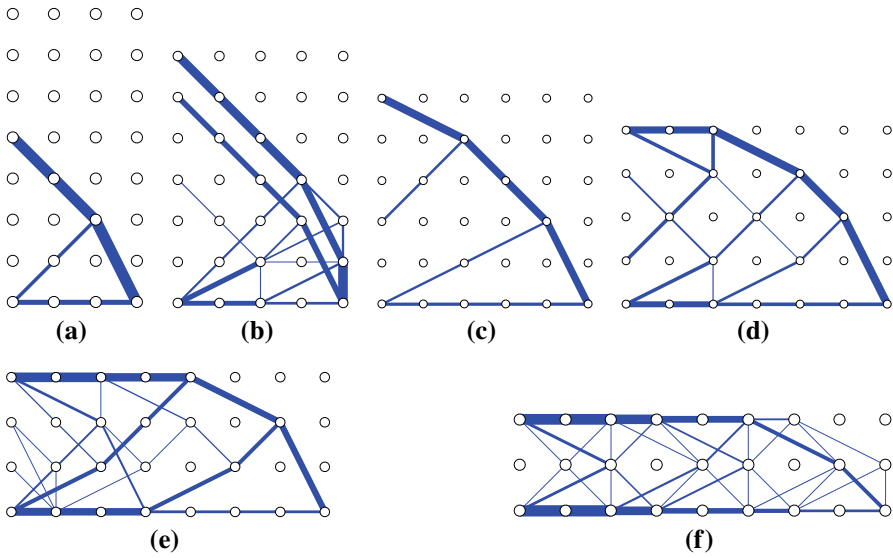
remove members that are longer than 3 m. It should be clear that the ground structure retains overlapping members.

As for the nominal external load,  $\tilde{\mathbf{p}}$ , a vertical force is applied to the bottom right-most node as shown in Fig. 6. The uncertainty model of the external load is given as explained in (19), where  $\tilde{p}_1 = 100$  kN and  $\alpha = 0.5\tilde{p}_1$ . The lower and upper bounds for the member cross-sectional areas are  $\underline{x} = 50$  mm<sup>2</sup> and  $\bar{x} = 700$  mm<sup>2</sup>. The upper bound for the structural volume is  $\bar{c} = 2N_X N_Y \times 10^5$  mm<sup>3</sup>.

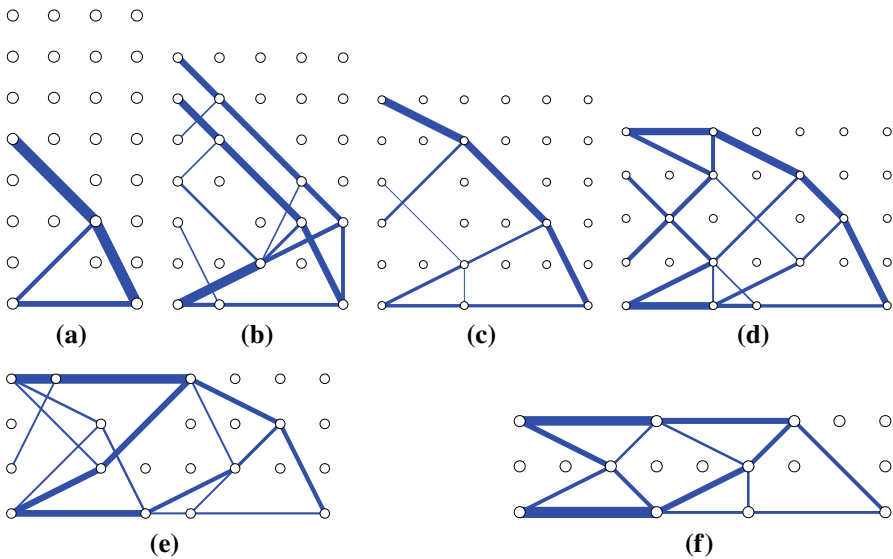
As for problem sizes, we consider six cases,  $(N_X, N_Y) = (3, 7), (4, 6), (5, 5), (6, 4), (7, 3),$  and  $(8, 2)$ . Table 3 lists the number of members, the number of degrees of freedom of the nodal displacements, and the upper bound for the structural volume. Figure 7 collects the optimal solutions of the conventional (i.e., not robust) compliance minimization in (17) for the nominal external load,  $\tilde{\mathbf{p}}$ . For the robust optimization, the solutions obtained by the proposed method are shown in Fig. 8. It is observed in Fig. 8a that the three chains in Fig. 7a are replaced with longer single members and four intermediate nodes are removed. The length of the bottom horizontal chain in Fig. 7c is 5 m. This chain is replaced with two members in Fig. 8c, because the maximum member length in the ground structure is 3 m. Similar observation can be made also in Fig. 8d, e. The nominal optimal solution in Fig. 7f has many thin members as well as many nodes. In contrast, the robust solution in Fig. 8f has simple topology, which may be considered practically preferable. In all the solutions in Fig. 8, the longest member is not longer than the maximum member length in the ground structure, as explained in Sect. 4.1.

The computational results are listed in Table 4. Here, “obj.” reports the objective value of the solution obtained by the proposed method, and  $\tilde{w}$  is the compliance of this solution for the nominal external load,  $\tilde{\mathbf{p}}$ . The optimal value of problem (17) for  $\tilde{\mathbf{p}}$  is listed as “nom. opt.” Therefore,  $\tilde{w}$  is no smaller than the value of “nom. opt.” It is observed in Table 4 that these two values are very close. Namely, in these examples, robustness can be achieved with compensation of only very small increase of the nominal compliance.

As explained in Sect. 2, one of difficulties of the robust truss topology optimization is that the uncertainty model of the external load depends on the existing nodes, i.e., on  $s$ . For comparison, we fix  $s$  to obtain a robust solution. As a simple heuristic, we construct an estimate of  $s$ , denoted  $\bar{s}$ , from the existing nodes of a solution in Fig. 7.



**Fig. 7** Example (II). The optimal solutions of the compliance minimization for the nominal external load. **a**  $(N_X, N_Y) = (3, 7)$ ; **b**  $(4, 6)$ ; **c**  $(5, 5)$ ; **d**  $(6, 4)$ ; **e**  $(7, 3)$ ; and **f**  $(8, 2)$



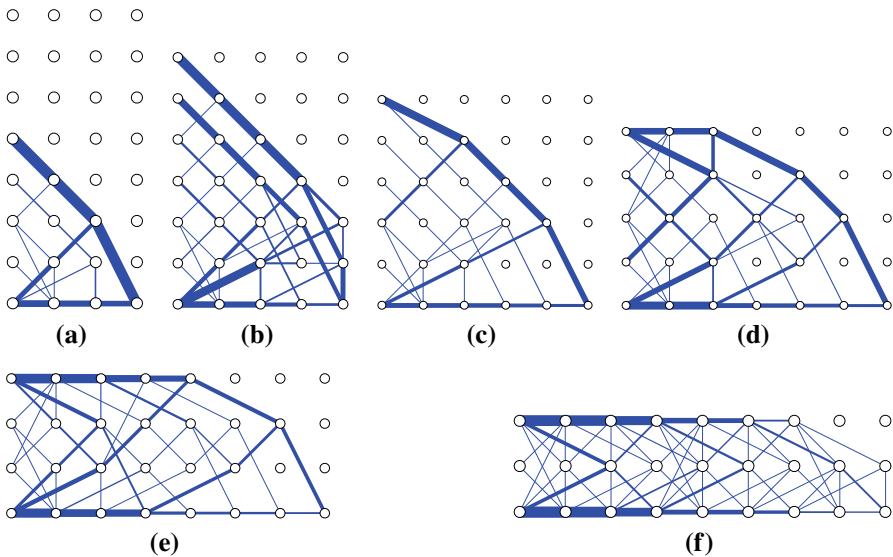
**Fig. 8** Example (II). The solutions obtained by the proposed method for the robust optimization under the load uncertainty. **a**  $(N_X, N_Y) = (3, 7)$ ; **b**  $(4, 6)$ ; **c**  $(5, 5)$ ; **d**  $(6, 4)$ ; **e**  $(7, 3)$ ; and **f**  $(8, 2)$

Then we solve the following robust optimization problem:

$$\begin{aligned}
 & \text{Minimize} && w \\
 & \text{subject to} && \sup\{\pi(\mathbf{x}; \mathbf{p}) \mid \mathbf{p} \in P(\bar{\mathbf{s}})\} \\
 & && \mathbf{x} \geq \mathbf{0}, \\
 & && \mathbf{c}^\top \mathbf{x} \leq \bar{c}.
 \end{aligned}$$

**Table 4** Computational results of example (II)

$(N_X, N_Y)$	Obj. (J)	#iter.	Time (s)	$\tilde{w}$ (J)	Nom. opt. (J)	Fixed $s$ (J)
(3, 7)	836.310	9	46.4	766.518	761.905	986.442
(4, 6)	1807.714	39	242.9	1360.425	1185.185	2534.505
(5, 5)	2382.377	35	249.6	2034.270	1929.012	3017.593
(6, 4)	5913.978	21	142.1	4427.633	4143.551	7032.673
(7, 3)	14912.232	40	193.9	10960.621	9918.356	17717.408
(8, 2)	43467.983	32	103.9	34515.627	34515.626	71121.097



**Fig. 9** The optimal solution of problem (44), where  $\bar{s}$  is defined from the solutions in Fig. 7

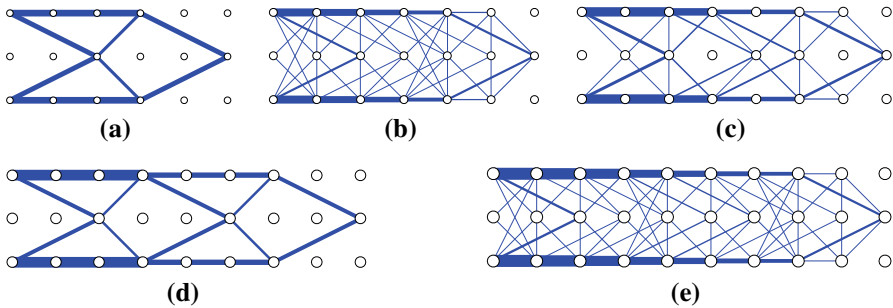
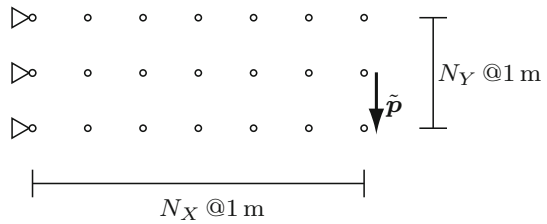
This problem can be recast as SDP. For simplicity, the lower bound constraints on the cross-sectional areas of the existing members are omitted. The obtained solutions are shown in Fig. 9. The optimal value is reported in “fixed  $s$ ” of Table 4. It should be clear that, at each solution in Fig. 9, uncertain external forces are applied only to the nodes that the corresponding solution in Fig. 7 has. Nevertheless, the objective value of a solution in Fig. 9 is much larger than that of the corresponding solution in Fig. 8. In other words, the solution obtained by the proposed method has quite high quality.

**6.3 Example (III)**

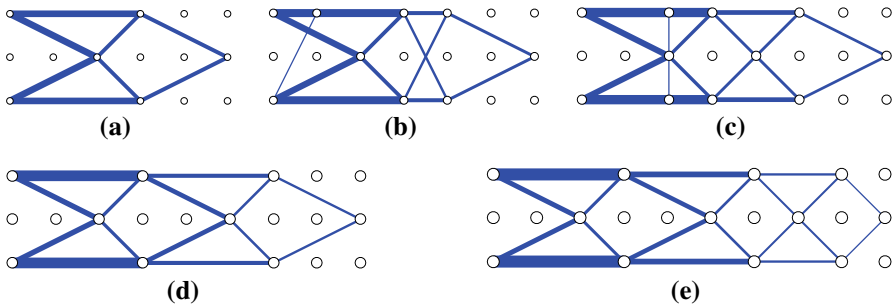
Consider the problem setting shown in Fig. 10. Ground structures are generated in the manner explained in Sect. 6.2. The maximum length of the members in a ground structure is 3 m. The uncertainty model of the external load is defined by using (19) with  $\tilde{p}_1 = 100\text{ kN}$  and  $\alpha = 0.75\tilde{p}_1$ . The lower and upper bounds for the member



**Fig. 10** Example (III). The problem setting for  $(N_X, N_Y) = (6, 2)$



**Fig. 11** Example (III)-1. The optimal solutions of the compliance minimization for the nominal external load. **a**  $(N_X, N_Y) = (5, 2)$ ; **b**  $(6, 2)$ ; **c**  $(7, 2)$ ; **d**  $(8, 2)$ ; and **e**  $(9, 2)$



**Fig. 12** Example (III)-1. The solutions obtained by the proposed method for the robust optimization under the load uncertainty. **a**  $(N_X, N_Y) = (5, 2)$ ; **b**  $(6, 2)$ ; **c**  $(7, 2)$ ; **d**  $(8, 2)$ ; and **e**  $(9, 2)$

cross-sectional areas are  $\underline{x} = 50 \text{ mm}^2$  and  $\bar{x} = 500 \text{ mm}^2$ , respectively. The upper bound for the structural volume is  $\bar{c} = 4N_X N_Y \times 10^5 \text{ mm}^3$ .

For problem instances with  $(N_X, N_Y) = (5, 2), (6, 2), \dots, (9, 2)$ , Fig. 11 shows the optimal solutions of the compliance minimization for the nominal external load. For the robust optimization, the solutions obtained by the proposed method are collected in Fig. 12. The problem sizes are listed in Table 5. The computational results are listed in Table 6. It is observed in Fig. 12a, d that the intermediate nodes on chains in Fig. 11a, d are removed as a result of robust optimization. Although the nominal optimal solutions in Fig. 11b, c, e have very complicated forms, the robust solutions in Fig. 12b, c, e are simple and practically preferable. Thus, it is often that robustness against uncertain loads and the minimal cross-sectional area constraints for the existing members yield simple truss topology.

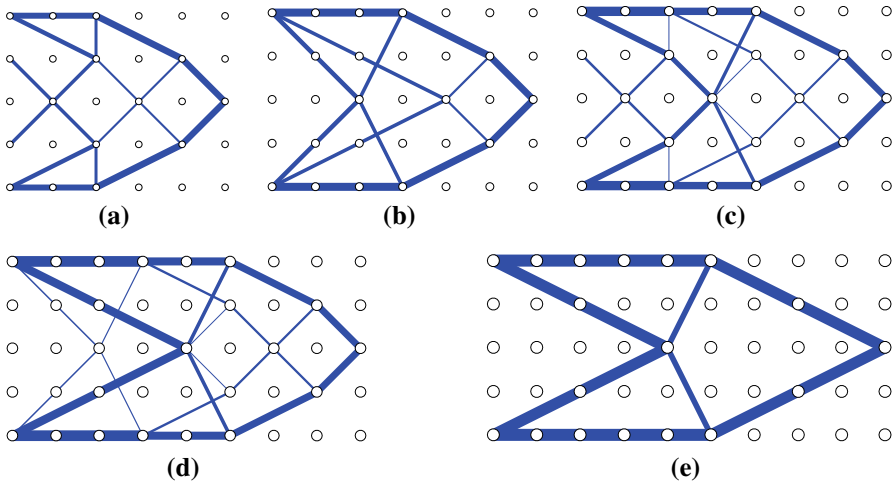
**Table 5** Characteristics of the problem instances in example (III)

$(N_X, N_Y)$	$m$	$d$	$\bar{c}$ (mm <sup>3</sup> )
(5, 2)	108	30	$4.0 \times 10^6$
(6, 2)	132	36	$4.8 \times 10^6$
(7, 2)	156	42	$5.6 \times 10^6$
(8, 2)	180	48	$6.4 \times 10^6$
(9, 2)	204	54	$7.2 \times 10^6$
(5, 4)	240	50	$8.0 \times 10^6$
(6, 4)	292	60	$9.6 \times 10^6$
(7, 4)	344	70	$11.2 \times 10^6$
(8, 4)	396	80	$12.8 \times 10^6$
(9, 4)	448	90	$14.4 \times 10^6$
(5, 6)	372	70	$12.0 \times 10^6$
(6, 6)	452	84	$14.4 \times 10^6$
(7, 6)	532	98	$16.8 \times 10^6$
(8, 6)	612	112	$19.2 \times 10^6$
(9, 6)	692	126	$21.6 \times 10^6$

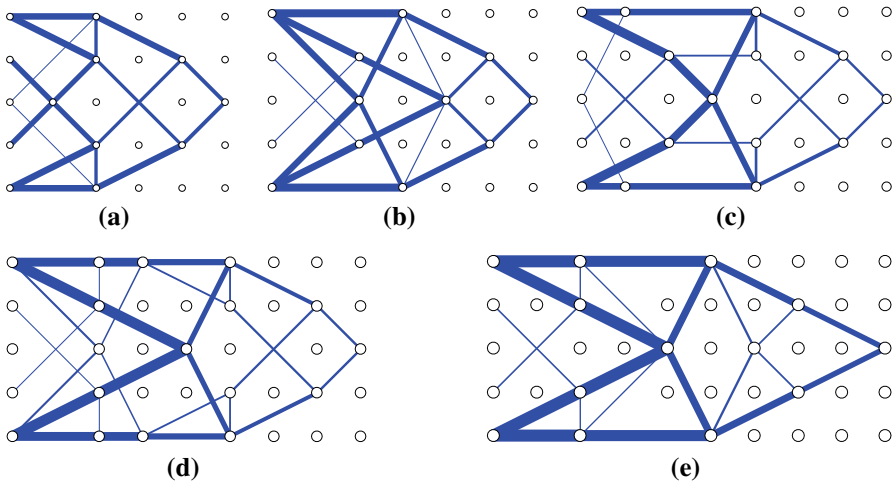
**Table 6** Computational results of example (III)

$(N_X, N_Y)$	Obj. (J)	#iter.	Time (s)	$\tilde{w}$ (J)	Nom. opt. (J)
(5, 2)	7221.094	18	25.0	5708.559	5512.500
(6, 2)	13698.325	43	70.4	9514.907	8760.417
(7, 2)	19198.058	42	87.7	13108.577	12223.214
(8, 2)	27245.117	23	64.5	17498.409	16531.250
(9, 2)	49880.911	28	100.7	23510.654	22562.500
(5, 4)	2514.685	16	62.1	1469.158	1304.012
(6, 4)	4063.725	19	113.9	2160.081	1814.815
(7, 4)	6563.146	40	320.6	3074.707	2484.871
(8, 4)	9111.073	37	440.0	3946.944	3260.031
(9, 4)	8988.477	24	286.4	5057.029	4255.319
(5, 6)	1420.620	26	312.8	828.309	575.268
(6, 6)	2168.363	19	208.0	1197.005	811.665
(7, 6)	2811.740	15	226.7	1420.604	1123.393
(8, 6)	4059.438	32	632.0	1890.087	1468.478
(9, 6)	4242.989	35	863.0	2122.988	1829.790

The solutions obtained for the instances with  $(N_X, N_Y) = (5, 4), (6, 4), \dots, (9, 4)$  are collected in Figs. 13 and 14. The robust optimal solution obtained by the proposed method has a form similar to the corresponding nominal optimal solution, but many chains in the nominal optimal solution are replaced with single members.



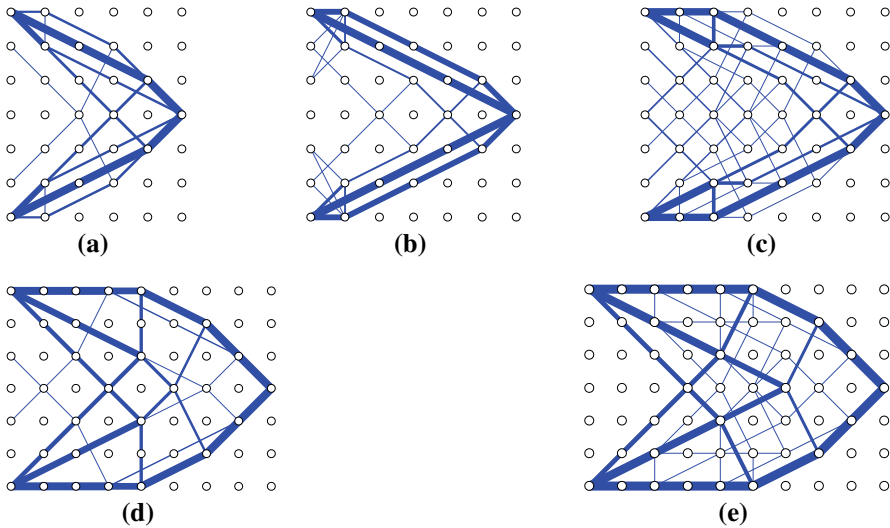
**Fig. 13** Example (III)-2. The optimal solutions of the compliance minimization for the nominal external load. **a**  $(N_X, N_Y) = (5, 4)$ ; **b**  $(6, 4)$ ; **c**  $(7, 4)$ ; **d**  $(8, 4)$ ; and **e**  $(9, 4)$



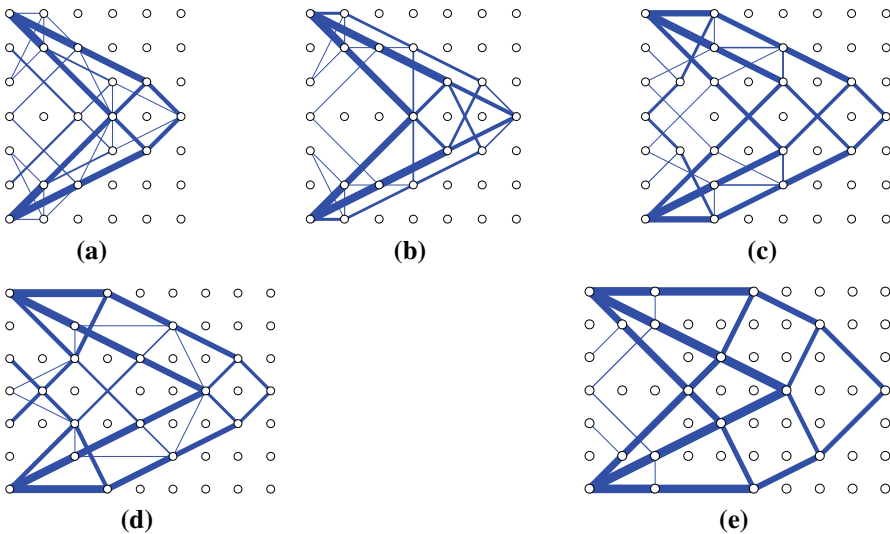
**Fig. 14** Example (III)-2. The solutions obtained by the proposed method for the robust optimization under the load uncertainty. **a**  $(5, 4)$ ; **b**  $(6, 4)$ ; **c**  $(7, 4)$ ; **d**  $(8, 4)$ ; and **e**  $(9, 4)$

The solutions obtained for the instances with  $(N_X, N_Y) = (5, 6), (6, 6), \dots, (9, 6)$  are collected in Figs. 15 and 16. It is observed that the nominal optimal solutions in Fig. 15c, e have so many thin members. In contrast, the robust solutions in Fig. 16c, e have fewer members. The layout of thick members in Fig. 16b is different from that in Fig. 15b. Also, the layout of thick members in Fig. 16d is different from that in Fig. 15d.

It is observed in Table 6 that the proposed method converged mostly within 40 iterations. The computational time for the instance with about 600 members is about 10 min. Thus, the proposed method finds a reasonable feasible solution with relatively small computational cost.



**Fig. 15** Example (III)-3. The optimal solutions of the compliance minimization for the nominal external load. **a** (5, 6); **b** (6, 6); **c** (7, 6); **d** (8, 6); and **e** (9, 6)



**Fig. 16** Example (III)-3. The solutions obtained by the proposed method for the robust optimization under the load uncertainty. **a** (5, 6); **b** (6, 6); **c** (7, 6); **d** (8, 6); and **e** (9, 6)

## 7 Conclusions

In this paper, we have presented a new formulation and algorithm for robust truss topology optimization considering uncertainty in the external load. Specifically, combinatorial aspects of the problem have been dealt with in the framework of the complementarity constraints. As for the uncertainty model of the external load, we

have supposed that uncertain external forces can be applied to all the nodes of a truss. This model depends on the set of existing nodes, and hence on the set of existing members, while the member cross-sectional areas are the design variables to be optimized. Thus, the robust optimization problem involves design-dependent constraints. Also, it has been explained that overlapping members should be incorporated to a ground structure. In the final truss design, however, presence of overlapping members is not allowed from a practical point of view. In this paper, the set of existing nodes, the selection among overlapping members, and the lower bound constraints for the cross-sectional areas of the existing members are treated by using the complementarity constraints.

In the conventional truss topology optimization, it is often that an optimal solution has a sequence of parallel consecutive members, called a chain. To stabilize a truss, a chain is replaced with a longer single member. Special consideration, such as the local buckling constraints, is needed to avoid presence of a too long member converted from a chain. In contrast, the solution obtained with the proposed method does not have a member which is longer than the maximum member length of the ground structure, because a truss design including a chain is infeasible for the presented robust optimization problem.

This paper has presented an SDPCC (semidefinite programming with complementarity constraints) formulation of a structural optimization problem. Then, its DC programming reformulation has been solved with a convex-concave procedure. This algorithm is a version of MM algorithms and EM algorithms, which are widely used in machine learning, image processing, etc. It has been shown through the numerical experiments that the proposed heuristic can converge to a high-quality solution within relatively small computational cost. The method can certainly handle complementarity constraints other than the ones presented in this paper. An example is a set of constraints that prohibits the presence of mutually crossing members in a truss design.

**Acknowledgements** This work is partially supported by JSPS KAKENHI 26420545 and 17K06633.

## References

1. Achtziger, W.: Local stability of trusses in the context of topology optimization. Part I: exact modelling. *Struct. Optim.* **17**, 235–246 (1999)
2. Achtziger, W., Bendsøe, M.P., Ben-Tal, A., Zowe, J.: Equivalent displacement based formulations for maximum strength truss topology design. *Impact Comput. Sci. Eng.* **4**, 315–345 (1992)
3. Anjos, M.F., Lasserre, J.B. (eds.): *Handbook on Semidefinite, Conic and Polynomial Optimization*. Springer, New York (2012)
4. Beck, A., Ben-Tal, A., Tretuashvili, L.: A sequential parametric convex approximation method with applications to nonconvex truss topology design problems. *J. Global Optim.* **47**, 29–51 (2010)
5. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: *Robust Optimization*. Princeton University Press, Princeton (2009)
6. Ben-Tal, A., Jarre, F., Kočvara, M., Nemirovski, A., Zowe, J.: Optimal design of trusses under a nonconvex global buckling constraint. *Optim. Eng.* **1**, 189–213 (2000)
7. Ben-Tal, A., Nemirovski, A.: Robust truss topology optimization via semidefinite programming. *SIAM J. Optim.* **7**, 991–1016 (1997)
8. Bendsøe, M.P., Ben-Tal, A., Zowe, J.: Optimization methods for truss geometry and topology design. *Struct. Optim.* **7**, 141–159 (1994)
9. Bendsøe, M.P., Sigmund, O.: *Topology Optimization*. Springer-Verlag, Berlin (2003)

10. Brittain, K., Silva, M., Tortorelli, D.A.: Minmax topology optimization. *Struct. Multidiscip. Optim.* **45**, 657–668 (2012)
11. Calafiore, G.C., Dabbene, F.: Optimization under uncertainty with applications to design of truss structures. *Struct. Multidiscip. Optim.* **35**, 189–200 (2008)
12. Cherkaev, E., Cherkaev, A.: Principal compliance and robust optimal design. *J. Elast.* **72**, 71–98 (2003)
13. Cherkaev, E., Cherkaev, A.: Minimax optimization problem of structural design. *Comput. Struct.* **86**, 1426–1435 (2008)
14. Collobert, R., Sinz, F., Weston, J., Bottou, L.: Large scale transductive SVMs. *J. Mach. Learn. Res.* **7**, 1687–1712 (2006)
15. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B (Methodological)* **39**, 1–38 (1977)
16. de Gournay, F., Allaire, G., Jouve, F.: Shape and topology optimization of the robust compliance via the level set method. *ESAIM Control Optim. Calc. Var.* **14**, 43–70 (2008)
17. Figueiredo, M.A.T., Bioucas-Dias, J.M., Nowak, R.D.: Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.* **16**, 2980–2991 (2007)
18. Fung, G., Mangasarian, O.L.: Semi-supervised support vector machines for unlabeled data classification. *Optim. Methods Softw.* **15**, 29–44 (2001)
19. Gotoh, J., Takeda, A., Tono, K.: DC formulations and algorithms for sparse optimization problems. *Math. Program.* **169**, 141–176 (2018)
20. Grant, M., Boyd, S.: Graph implementations for nonsmooth convex programs. In: Blondel, V., Boyd, S., Kimura, H. (eds.) *Recent Advances in Learning and Control (A Tribute to M. Vidyasager)*, pp. 95–110. Springer, Vidyasagar (2008)
21. Grant, M., Boyd, S.: *CVX: Matlab Software for Disciplined Convex Programming, Version 2.1* (2016 October). <http://cvxr.com/cvx>
22. Guo, X., Bai, W., Zhang, W., Gao, X.: Confidence structural robust design and optimization under stiffness and load uncertainties. *Comput. Methods Appl. Mech. Eng.* **198**, 3378–3399 (2009)
23. Guo, X., Cheng, G.D., Olhoff, N.: Optimum design of truss topology under buckling constraints. *Struct. Multidisc. Optim.* **30**, 169–180 (2005)
24. Guo, X., Du, J., Gao, X.: Confidence structural robust optimization by non-linear semidefinite programming-based single-level formulation. *Int. J. Numer. Methods Eng.* **86**, 953–974 (2011)
25. Hashimoto, D., Kanno, Y.: A semidefinite programming approach to robust truss topology optimization under uncertainty in locations of nodes. *Struct. Multidisc. Optim.* **51**, 439–461 (2015)
26. Holmberg, E., Thore, C.-J., Klarbring, A.: Worst-case topology optimization of self-weight loaded structures using semi-definite programming. *Struct. Multidisc. Optim.* **52**, 915–928 (2015)
27. Hunter, D.R., Lange, K.: Quantile regression via an MM algorithm. *J. Comput. Graphical Stat.* **9**, 60–77 (2000)
28. Hunter, D.R., Lange, K.: A tutorial on MM algorithms. *Am. Stat.* **58**, 30–37 (2004)
29. Hunter, D.R., Li, R.: Variable selection using MM algorithms. *Ann. Stat.* **33**, 1617–1642 (2005)
30. Jara-Moroni, F., Pang, J.-S., Wächter, A.: A study of the difference-of-convex approach for solving linear programs with complementarity constraints. *Math. Program.* **169**, 221–254 (2018)
31. Jarre, F., Kočvara, M., Zowe, J.: Optimal truss design by interior-point methods. *SIAM J. Optim.* **8**, 1084–1107 (1998)
32. Kanno, Y., Guo, X.: A mixed integer programming for robust truss topology optimization with stress constraints. *Int. J. Numer. Methods Eng.* **83**, 1675–1699 (2010)
33. Kanno, Y., Takewaki, I.: Sequential semidefinite program for maximum robustness design of structures under load uncertainties. *J. Optim. Theory Appl.* **130**, 265–287 (2006)
34. Kanno, Y., Yamada, H.: A note on truss topology optimization under self-weight load: mixed-integer second-order cone programming approach. *Struct. Multidisc. Optim.* **56**, 221–226 (2017)
35. Klarbring, A., Strömberg, N.: A note on the min–max formulation of stiffness optimization including non-zero prescribed displacements. *Struct. Multidisc. Optim.* **12**, 147–149 (2012)
36. Lange, K., Chi, E.C., Zhou, H.: A brief survey of modern optimization for statisticians. *Int. Stat. Rev.* **82**, 46–70 (2014)
37. Le Thi, H.A., Pham Dinh, T.: The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. *Ann. Oper. Res.* **133**, 23–46 (2005)
38. Le Thi, H.A., Pham Dinh, T.: On solving linear complementarity problems by DC programming and DCA. *Comput. Optim. Appl.* **50**, 507–524 (2011)

39. Lipp, T., Boyd, S.: Variations and extension of the convex–concave procedure. *Optim. Eng.* **17**, 263–287 (2016)
40. Löfberg, J.: YALMIP: A toolbox for modeling and optimization in MATLAB. In: 2004 IEEE International Conference on Computer Aided Control System Design, Taipei, pp. 284–289 (2004)
41. Mela, K.: Resolving issues with member buckling in truss topology optimization using a mixed variable approach. *Struct. Multidisc. Optim.* **50**, 1037–1049 (2014)
42. Mitchell, J.E., Pang, J.-S., Yu, B.: Obtaining tighter relaxations of mathematical programs with complementarity constraints. In: Terlaky, T., Curtis, F.E. (eds.) *Modeling and Optimization: Theory and Applications*, pp. 1–23. Springer, Berlin (2012)
43. Muu, L.D., Dinh, Q.T., Le Thi, H.A., Tao, P.D.: A new decomposition algorithm for globally solving mathematical programs with affine equilibrium constraints. *Acta Math. Vietnam.* **37**, 201–217 (2012)
44. Neumann, J., Schnörr, C., Steidl, G.: Combined SVM-based feature selection and classification. *Mach. Learn.* **61**, 129–150 (2005)
45. Pham Dinh, T., Le Thi, H.A.: Convex analysis approach to D. C. programming: theory, algorithms and applications. *Acta Math. Vietnam.* **22**, 289–355 (1997)
46. Pham Dinh, T., Le Thi, H.A.: Recent advances in DC programming and DCA. In: Nguyen, N.T., Le Thi, H.A. (eds.) *Transactions on computational intelligence XIII*. Springer-Verlag, Berlin (2014)
47. Pham Dinh, T., Le Thi, H.A.: A D.C. optimization algorithm for solving the trust-region subproblem. *SIAM J. Optim.* **8**, 476–505 (1998)
48. Pólik, I.: Addendum to the SeDuMi User Guide: Version 1.1.1. Technical Report. Advanced Optimization Laboratory, McMaster University, Hamilton (2005). <http://sedumi.ie.lehigh.edu/>
49. Rozvany, G.I.N.: Difficulties in truss topology optimization with stress, local buckling and system stability constraints. *Struct. Optim.* **11**, 213–217 (1996)
50. Sriperumbudur, B.K., Lanckriet, G.R.G.: On the convergence of the concave–convex procedure. *Adv. Neural Inf. Process. Syst.* **22**, 1759–1767 (2009)
51. Sriperumbudur, B.K., Torres, D.A., Lanckriet, G.R.G.: A majorization-minimization approach to the sparse generalized eigenvalue problem. *Mach. Learn.* **85**, 3–39 (2011)
52. Stavroulakis, G.E., Polyakova, L.N.: Nonsmooth and nonconvex structural analysis algorithms based on difference convex optimization techniques. *Struct. Optim.* **12**, 167–176 (1996)
53. Sturm, J.F.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optim. Methods Softw.* **11**(12), 625–653 (1999)
54. Sun, Y., Babu, P., Palomar, D.P.: Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Trans. Signal Process.* **65**, 794–816 (2017)
55. Svanberg, K.: The method of moving asymptotes—a new method for structural optimization. *Int. J. Numer. Methods Eng.* **24**, 359–373 (1987)
56. Svanberg, K.: A class of globally convergent optimization method based on conservative convex separable approximations. *SIAM J. Optim.* **12**, 555–573 (2002)
57. Takezawa, A., Nii, S., Kitamura, M., Kogiso, N.: Topology optimization for worst load conditions based on the eigenvalue analysis of an aggregated linear system. *Comput. Methods Appl. Mech. Eng.* **200**, 2268–2281 (2011)
58. Thore, C.-J.: Multiplicity of the maximum eigenvalue in structural optimization problems. *Struct. Multidisc. Optim.* **53**, 961–965 (2016)
59. Tütüncü, R.H., Toh, K.C., Todd, M.J.: Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Program.* **B95**, 189–217 (2003)
60. Wolkowicz, H., Saigal, R., Vandenberghe, L. (eds.): *Handbook on Semidefinite Programming: Theory, Algorithms and Applications*. Kluwer Academic Publishers, Boston (2000)
61. Yonekura, K., Kanno, Y.: Global optimization of robust truss topology via mixed integer semidefinite programming. *Optim. Eng.* **11**, 355–379 (2010)
62. You, S., Cheny, L., Liu, Y.: Convex–concave procedure for weighted sum-rate maximization in a MIMO interference network. In: 2014 IEEE Global Communications Conference, Austin, pp. 4060–4065 (2014)
63. Yu, B., Mitchell, J.E., Pang, J.-S.: Solving linear programs with complementarity constraints using branch-and-cut. Technical Report, Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY (2016 October)
64. Yuille, A., Rangarajan, A.: The concave–convex procedure. *Neural Comput* **15**, 915–936 (2003)
65. Zillober, C.: Global convergence of a nonlinear programming method using convex approximations. *Numer. Algorithms* **27**, 265–289 (2001)