

# Constrained incremental bundle method with partial inexact oracle for nonsmooth convex semi-infinite programming problems

Li-Ping Pang<sup>1</sup> · Jian Lv<sup>1</sup> · Jin-He Wang<sup>2</sup>

Received: 14 January 2015 / Published online: 20 November 2015  
© Springer Science+Business Media New York 2015

**Abstract** Semi-infinite problem (SIPs) are widely used in many control systems for solving complex control problem, such as polymerase chain reaction control system or other real time control system. In this paper, we present a bundle method for solving the nonsmooth convex SIPs, with the aim of working on the basis of “improvement function”, “inexact oracle” and “incomplete knowledge” of the constraints. The proposed algorithm, whenever a new stabilized center is refreshed, requires an evaluation within some accuracy for the value of constraints. Beyond that, by using the incremental technique, it does not require all information about the constraints, but only one component function value and one subgradient needed to be estimated to update the bundle information and generate the search direction. Thus the computational cost is significantly reduced. Global convergence of this method is established based on some mild assumptions. Numerical experiments show that the algorithm is efficient for solving nonsmooth convex SIPs.

**Keywords** Constrained optimization · Nonsmooth optimization · Semi-infinite programming · Bundle method

---

✉ Li-Ping Pang  
m15041171591@163.com; lppang@dlut.edu.cn

Jian Lv  
lvjian328@163.com

<sup>1</sup> School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China

<sup>2</sup> School of Computer Engineering, Qingdao Technological University, Qingdao 266033, China

### 1 Introduction

Semi-infinite problem (SIPs) can be used in polymerase chain reaction (PCR) control system for finding optimized complex control strategy. In this paper, we proposed an incremental bundle method for general nonsmooth convex SIPs. One of basic motivation is to solve a large class of distributionally robust optimization problems, which can be rewritten as nonsmooth convex SIPs by an uncountable infinite dimensional set of probability distributions. In fact, the robust stochastic optimization for minimizing an expectation function is an important variety of decision problems with uncertainty data. An abstract form for such stochastic optimization problems is

$$\min_{x \in X} E_W[f(x, \xi)],$$

where the expectation is taken concerning  $W$  which is supposed to be obtained. Unfortunately, in practical applications, such a distribution is not known precisely and has to be estimated from data. Just as in [4, 32], after defining a set  $\mathcal{D}$  of possible probability distributions for the uncertain parameters is known; the objective function can be rewritten corresponding to the “worst” case expected cost over the choice of a distribution in  $\mathcal{D}$ . Therefore, the general distributionally robust optimization problem can be reformulated as follows:

$$\min_{x \in X} \max_{W \in \mathcal{D}} E_W[F(x)],$$

where  $F$  is a disutility function or random cost we attempt to minimize in expectation. An equivalent expression is

$$\min_{x \in X} \max_{W \in \mathcal{D}} \int_{\xi \in \Pi} f(x, \xi) W(d\xi),$$

where the support set  $\Pi$  is closed and bounded, and  $f(x, \xi)$  is a convex weight function in  $x$  that depends on parameter  $\xi$ . If the set  $X$  is a bounded set, the optimal objective function value can be lied in an interval  $[\alpha_0, \alpha_1]$ . Thus, the problem can be written as a nonsmooth convex SIP:

$$\begin{aligned} &\min \alpha \\ &s.t. \int_{\Pi} f(x, \xi) W(d\xi) - \alpha \leq 0, \quad W \in \mathcal{D} \\ &\quad \alpha \in [\alpha_0, \alpha_1], \quad x \in X. \end{aligned} \tag{1.1}$$

In this paper, we consider the following general nonsmooth convex SIP:

$$\begin{aligned} &\min_x f(x) \\ &s.t. c(x, y) \leq 0, \quad \forall y \in Y, \\ &\quad x \in X, \end{aligned} \tag{1.2}$$

where the functions  $f: X \rightarrow \mathbf{R}$ ,  $c: X \times Y \rightarrow \mathbf{R}$  satisfy the following conditions:

- $X$  is a subset of  $\mathbf{R}^n$  and  $Y$  is a nonempty compact subset of  $\mathbf{R}^m$ .
- The objective function  $f$  is a convex function, in general nondifferentiable.
- $c(\cdot, y)$  is convex, in general nondifferentiable for all  $y \in Y$ ,  $c(x, \cdot)$  is upper semi-continuous on  $Y$  for all  $x \in X$ .

More broadly, such SIPs apply to optimal control, and diverse fields of engineering such as multiobjective optimization, resource allocation in decentralized systems, control systems design, and filter design in signal processing, decision making under competition, see the early literature [10, 13, 14, 30, 39, 54, 60].

The main difficulty for solving SIPs lies in that there are infinitely many constraints. The principal effort of existing methods is to reduce the infinite set  $Y$  to a finite one. The main proposed methods can be roughly divided into the following several types. Based on the KKT systems of SIPs, some semismooth Newton methods and smoothing Newton methods were presented in [17, 27, 29, 36, 41, 42, 56]. In every iteration only a system of linear equations needs to be solved, but they may not ensure the feasibility of the original, or the accumulation point is not necessarily an stationary point of the SIPs. The discretization methods, which based on the ideas of choosing a finite grid of  $Y$ , solve finite mathematical programs with  $Y$  replaced by the finite grid and updates the grid (see, e.g. [51, 53, 59]). The discretization methods are apt to implement, but the cost of per iteration increases rapidly as the cardinality of the finite grid of  $Y$  grows. The third common approach is the reduction-based method, which is to reduce the problem locally to a finite-dimensional nonlinear programming problem (see, e.g. [2, 3, 52, 58]). Unfortunately, the reduction-based method needs some highly strong assumptions including the conditions leading to the infiniteness of the set of active constraints. Another important method is the so-called exchange method (see, e.g. [6, 16, 26, 30, 55, 57, 60]). The new exchange method proposed in [60], does not require to solve a maximum problem over the index set at each iteration.

All methods above just apply to the smooth SIPs, where  $f(\cdot)$  and  $c(\cdot, \cdot)$  all are smooth. The methods for solving nonsmooth SIPs are rare, and just only some excellent theoretical research. If the set  $Y$  is finite, necessary conditions of Karush–Kuhn–Tucker (KKT) condition for optimality can be established under various constraint qualifications. For example, Abadie, Zangwill, Guignard and Mangasarian–Fromovitz constraint qualification, are denoted by ACQ, ZCQ, GCQ and MFCQ. Puente et al. in [40], introduced the locally Farkas–Minkowski (LFM) property for linear SIPs, and its role as a constraint qualification was emphasized in many later papers. By using Clarke subdifferential, [19–21, 38] presented a nonsmooth analogue of the ACQ, ZCQ, GCQ and LFM property for locally Lipschitz SIPs. [28] introduced some new notions of constraint qualifications about the epigraphs of the conjugates of these functions for convex nonsmooth SIPs. [61] provided Lagrange multiplier rules for nonsmooth SIPs. [33] gave the subdifferential studies and applications of the supremum of uniformly Lipschitzian functions over arbitrary index sets with no topology. Furthermore, necessary and sufficient optimality conditions of KKT type for these problems are further researched in recent years.

In this paper, we proposed a bundle method for solving the nonsmooth convex SIP (1.2). By defining the following “max” function as

$$h(x) := \max \left\{ c(x, y) : y \in Y \right\},$$

the (1.2) can be equivalently written as the following nonsmooth nonlinear problem:

$$\begin{aligned} \min_x & f(x) \\ \text{s.t.} & h(x) \leq 0, \quad x \in X. \end{aligned} \quad (1.3)$$

Various versions of the bundle method (see, e.g., [1, 12, 31, 35, 37, 47–49]) are recognized as one of the most stable and effective method for nonsmooth problems. However, in most of applications, it may be impossible to calculate  $h(\cdot)$  accurately, but only controllable accuracy can be obtained [5, 45]. In fact, it may even be difficult to find a point  $x \in X$  satisfying  $h(x) \leq 0$ . Therefore, the bundle methodology cannot be used to (1.3) immediately.

Before continuing with our discussions, we introduce the improvement function associates with the problem (1.3):

$$H_y(x) = \max \left\{ f(x) - f(y), h(x) \right\}, \quad \forall x \in \mathbf{R}^n, \quad (1.4)$$

which is one of the most effective tools to handle constraints in this context, and plays a significant role in this paper. The constrained optimization problem can be rewritten as an unconstrained one. In fact, the point  $x^*$  is a solution of (1.3) if and only if  $x^*$  solves the unconstrained problem of minimizing  $H_{x^*}(x)$  (see, e.g. [23, 46]). As the iteration progresses, the main motivations for our method include the improvement function, incomplete knowledge of the “max” function  $h(\cdot)$  and the inexact oracle at some points. Based on these ideas, we introduce the improvement function to deal with the constraints, and propose an infeasible bundle method in this paper. Moreover, we introduce the so-called “incremental” conception, which stems from [22] and previously been applied for various problems (see, e.g. [5, 8, 9, 24, 34]). Now we expand the idea of “incremental” conception to the nonsmooth convex SIPs. At each iteration, we only need to compute one component function value  $c(x, y)$  and one respectively subgradient to update the bundles. Whenever a new stabilized center  $\hat{x}$  is updated, we also need to evaluate  $\max_{y \in Y} c(\hat{x}, y)$  within a given fixed accuracy. Fortunately, the process can be properly tackled by some excellent method (see, e.g. [7, 11, 25]).

This paper is organized as follows. In Sect. 2, we study the improvement function and introduce the corresponding bundle information. In Sect. 3, we propose a bundle method for solving problems (1.2). In Sect. 4, we establish the convergence property of our bundle method. In Sect. 5, we give some numerical results to see the performance of our bundle method.

## 2 Preliminaries

### 2.1 Concepts

Now we recall concepts and results of variational analysis that will be used in this paper. We shall make use of  $\partial f(\bar{x})$  and  $\partial_x c(\bar{x}, y)$  to denote the subdifferential of  $f(\cdot)$

and  $c(\cdot, y)$  at the point  $\bar{x}$  as [44, Proposition 8.12], i.e.,

$$\begin{aligned} \partial f(\bar{x}) &:= \left\{ v \mid f(x) \geq f(\bar{x}) + \langle v, x - \bar{x} \rangle \right\}, \\ \partial_x c(\bar{x}, y) &:= \left\{ v \mid c(x, y) \geq c(\bar{x}, y) + \langle v, x - \bar{x} \rangle \right\}. \end{aligned}$$

Approximate subgradients in the Convex Analysis refer to the  $\varepsilon$ -subdifferential is defined as

$$\partial_\varepsilon f(\bar{x}) := \left\{ v \mid f(x) \geq f(\bar{x}) + \langle v, x - \bar{x} \rangle - \varepsilon \right\},$$

whose central property is that  $0 \in \partial_\varepsilon f(\bar{x})$  implies  $\varepsilon$ -minimality of  $\bar{x}$  for problem  $\min f(x)$ , i.e.,  $f(\bar{x}) \leq f(x) + \varepsilon$ . The directional derivative, defined as [44] with respect to  $x$  in the direction  $d \in \mathbf{R}^n$ , has an alternative representation

$$f'(x, d) := \max_{g \in \partial f(x)} g^T d \quad \text{and} \quad c'_y(x, d) := \max_{g \in \partial_x c(x, y)} g^T d \quad \text{for } y \in Y.$$

According to [15, Theorem 4.4.2] and [19, Remark 3.6], it is known that  $Y$  is a nonempty compact subset,  $c(\cdot, y)$  is convex on  $X$  for any  $y \in Y$ , and  $c(x, \cdot)$  is upper semicontinuous on  $Y$  for each  $x \in X$ , then the subdifferential of  $h(\cdot)$  can be represented as:

$$\partial h(x) = \text{conv} \left\{ \bigcup \partial_x c(x, y) : y \in A(x) \right\},$$

where the active index-set  $A(x) := \{y \in Y : h(x) = c(x, y)\}$ . The convexity of each  $c(\cdot, y)$ , for any  $y \in Y$  actually implies that  $h(\cdot)$  is jointly upper semi-continuous on  $X \times Y$ . By the Carathéodory's Theorem [43, Theorem 17.1], a point  $x$  of  $\mathbf{R}^n$  lies in the convex hull of a set  $P$ , there is a subset  $P'$  of  $P$  consisting of  $n + 1$  or fewer points such that  $x$  lies in the convex hull of  $P'$ . Hence, for every  $g \in \partial h(x)$ , there exist the corresponding  $\mu_i^g$  and  $y_i^g \in A(x), i = 1, \dots, n + 1$  such that

$$g = \sum_{i=1}^{n+1} \mu_i^g g_{y_i^g}, \quad \text{where } \mu_i^g \geq 0, \sum_{i=1}^{n+1} \mu_i^g = 1, g_{y_i^g} \in \partial_x c(x, y_i^g), y_i^g \in A(x). \tag{2.1}$$

Along iterations, the method keeps an stabilized center, denoted by  $\hat{x}^k$ , at the  $k$ th iteration. Then the  $k$ th improvement function is given as follows

$$H_{\hat{x}^k}(x) = \max \left\{ f(x) - f(\hat{x}^k), h(x) \right\}. \tag{2.2}$$

We introduce the following extended MFCQ (see [18,50]) for this nonsmooth convex SIP.

**Assumption 1 (EMFCQ)** The extended MFCQ holds at the optimum point  $x^*$ . That is, there exists some feasible direction  $d$  of  $\mathcal{F} := \{x \in X : c(x, y) \leq 0, \text{ for all } y \in Y\}$  at  $x^*$ , satisfies

$$c'_y(x^*, d) < 0, \text{ for all } y \in Y_{act}(x^*), \tag{2.3}$$

where  $Y_{act}(x^*) = \{y \in Y \mid c(x^*, y) = 0\}$ .

We now give the following necessary conditions for optimality.

**Lemma 2.1** *Suppose that Assumption 1 is satisfied. Then the following statements hold:*

(i) *If  $x^*$  is a solution to the problem (1.2), then*

$$\min\{H_{x^*}(z) : z \in X\} = H_{x^*}(x^*) = 0. \tag{2.4}$$

(ii) *If the relation (2.4) holds, then  $0 \in \partial H_{x^*}(x^*) + N_X(x^*)$ , and the following conditions hold:*

$$\begin{aligned} 0 &\in \partial f(x^*) + \mu \partial h(x^*) + N_X(x^*), \\ \mu &\geq 0, \mu h(x^*) = 0, h(x^*) \leq 0, \end{aligned} \tag{2.5}$$

where  $N_X(x^*)$  denotes the normal cone of  $X$  at  $x^*$ .

(iii) *If the relations (2.5) holds, then there exist multipliers  $\lambda_i^*$  and  $y_i^* \in Y_{act}(x^*)$ ,  $i = 1, \dots, n + 1$ , satisfying*

$$\begin{aligned} 0 &\in \partial f(x^*) + \sum_{i=1}^{n+1} \lambda_i^* \partial_x c(x^*, y_i^*) + N_X(x^*), \\ \lambda_i^* &\geq 0, \lambda_i^* c(x^*, y_i^*) = 0, \quad i = 1, \dots, n + 1. \end{aligned} \tag{2.6}$$

*Proof* Suppose  $x^*$  is a solution to the problem (1.2), i.e.,

$$f(z) \geq f(x^*), \text{ for all } z \in \mathcal{F},$$

which implies that

$$\begin{aligned} H_{x^*}(z) &= \max_{z \in \mathbf{R}^n} \{f(z) - f(x^*), h(z)\} \\ &\geq \max_{z \in \mathcal{F}} \{f(z) - f(x^*), h(z)\} \\ &\geq 0 = H_{x^*}(x^*), \end{aligned}$$

where the first inequality has used that  $\mathcal{F} \subseteq \mathbf{R}^n$ , and the last equality is from  $h(x^*) \leq 0$ . From here, the first item (i) follows.

From [23, Lemma 2.15], we obtain that  $0 \in \partial H_{x^*}(x^*) + N_X(x^*)$ , and  $x^*$  is a FJ point of the problem (1.3). Then there exist some  $v_0, v \geq 0$ , such that

$$\begin{aligned} 0 &\in v_0 \partial f(x^*) + v \partial h(x^*) + N_X(x^*), \\ v_0 + v &= 1, v h(x^*) = 0, h(x^*) \leq 0. \end{aligned} \tag{2.7}$$

Suppose that the EMFCQ holds, and  $v_0$  is zero in (2.7), we can obtain that  $v = 1$ . Thus there exists a subgradient  $\hat{g} \in \partial h(x^*)$  such that  $-\hat{g} \in N_X(x^*)$ . Furthermore, by using the complementary condition  $v h(x^*) = 0$ , we conclude that  $h(x^*) = 0$ . Then the definitions of  $A(x^*)$  and  $Y_{act}(x^*)$  yield  $A(x^*) = Y_{act}(x^*)$ .

Suppose  $d$  is a feasible direction of  $\mathcal{F}$ , from  $-\hat{g} \in N_X(x^*)$  we have  $\langle \hat{g}, d \rangle \geq 0$ . By noting the relation (2.1) (Carathéodory's Theorem), there exist  $\hat{y}_i \in A(x^*)$  and  $\hat{\mu}_i \geq 0, i = 1, \dots, n + 1$  such that

$$\hat{g} = \sum_{i=1}^{n+1} \hat{\mu}_i g_{\hat{y}_i}, \text{ where } g_{\hat{y}_i} \in \partial_x c(x^*, \hat{y}_i), \text{ for all } i = 1, \dots, n + 1.$$

Besides, from the definition of the directional derivative, we can obtain

$$\langle g_{\hat{y}_i}, d \rangle \leq \max_{g \in \partial_x c(x, \hat{y}_i)} g^T d = c'_{\hat{y}_i}(x^*, d) < 0, \text{ where } \hat{y}_i \in A(x^*).$$

The second inequality the last inequality is from (2.3) and  $A(x^*) = Y_{act}(x^*)$ . Thus, it holds that

$$\begin{aligned} 0 \leq \langle \hat{g}, d \rangle &= \sum_{i=1}^{n+1} \hat{\mu}_i \langle g_{\hat{y}_i}, d \rangle \\ &\leq \sum_{i=1}^{n+1} \hat{\mu}_i c'_{\hat{y}_i}(x^*, d) < 0. \end{aligned}$$

Then that is an apparent contradiction. Thus  $v_0$  is strictly positive, and conditions (2.5) hold by setting  $\mu = \frac{v}{v_0}$ . From here, the result item (ii) follows readily.

From the relations (2.5), then there exist a subgradient  $g^* \in \partial h(x^*)$  such that

$$-\mu g^* \in \partial f(x^*) + N_X(x^*). \tag{2.8}$$

By combining with (2.1), there are  $\mu_i^* \geq 0, y_i^* \in A(x^*)$  (for all  $i = 1, \dots, n + 1$ ) such that

$$g^* = \sum_{i=1}^{n+1} \mu_i^* g_{y_i^*},$$

where  $g_{y_i^*} \in \partial_x c(x, y_i^*)$  for all  $i = 1, \dots, n + 1$ . Now we can show that if  $i_1 \neq i_2$ , then  $y_{i_1}^* \neq y_{i_2}^*$ , i.e., the different  $g_{y_i^*}$  contains in different  $\partial_x c(x, y_i^*)$ . Let  $I := \{j_1, \dots, j_{n-1}\} = \{1, 2, \dots, n + 1\} / \{i_1, i_2\}$ . By contradiction, we suppose  $i_1 \neq i_2$  and  $y_{i_1}^* = y_{i_2}^*$ , then

$$\begin{aligned} g^* &= \sum_{i \in I} \mu_i^* g_{y_i^*} + \mu_{i_1}^* g_{y_{i_1}^*} + \mu_{i_2}^* g_{y_{i_2}^*} \\ &= \sum_{i \in I} \mu_i^* g_{y_i^*} + \left( \mu_{i_1}^* + \mu_{i_2}^* \right) \left( \frac{\mu_{i_1}^*}{\mu_{i_1}^* + \mu_{i_2}^*} g_{y_{i_1}^*} + \frac{\mu_{i_2}^*}{\mu_{i_1}^* + \mu_{i_2}^*} g_{y_{i_2}^*} \right). \end{aligned}$$

Since  $\partial_x c(x, y_{i_1}^*)$  is a convex set and  $g_{y_{i_1}^*}, g_{y_{i_2}^*} \in \partial_x c(x, y_{i_1}^*)$ , thus  $\frac{\mu_{i_1}^*}{\mu_{i_1}^* + \mu_{i_2}^*} g_{y_{i_1}^*} + \frac{\mu_{i_2}^*}{\mu_{i_1}^* + \mu_{i_2}^*} g_{y_{i_2}^*} \in \partial_x c(x, y_{i_1}^*)$ . As a result, by setting  $\hat{\lambda}_n = \mu_{i_1}^* + \mu_{i_2}^*$ ,  $\hat{\lambda}_{n+1} = 0$  and  $\hat{\lambda}_i = \mu_{j_i}^*$  (for all  $i = 1, \dots, n - 1$ ), we can obtain that the different  $g_{y_i^*}$  contains in different  $\partial_x c(x, y_i^*)$ . Thus it holds that

$$\mu g^* = \mu \sum_{i=1}^{n+1} \hat{\lambda}_i g_{y_i^*} \in \mu \sum_{i=1}^{n+1} \hat{\lambda}_i \partial_x c(x, y_i^*),$$

which combines with (2.8), we can obtain that

$$0 \in \partial f(x^*) + \mu \sum_{i=1}^{n+1} \hat{\lambda}_i \partial_x c(x, y_i^*) + N_X(x^*).$$

By setting  $\lambda_i^* = \mu \times \hat{\lambda}_i$ , the conditions (2.6) hold readily. In fact, (2.5) can be written as (2.6) equivalently. □

By noting Lemma 2.1, we can obtain that (1.3) and its KKT condition are equivalent. In addition, if the Slater constraint qualification holds, i.e., there exists a point  $\bar{x} \in X$  such that  $c(\bar{x}, y) < 0$  for any  $y \in Y$ . Then the above three statements in Lemma 2.1 are equivalent.

### 2.2 Pretreatment for inner subproblem

In this subsection, we consider the inner subproblem, i.e.,

$$\max_{y \in Y} c(\hat{x}, y). \tag{2.9}$$

We assume that the function value  $h(\hat{x})$  can only be evaluated within a given fixed accuracy  $\omega$ . In fact, the assumption for fixed accuracy is realistic in many applications. The controllable accuracy may contain the following two cases: For each constant  $\omega > 0$ , an  $\omega$ -optimal solution of (2.9) can be found; on the other hand, this may be possible only for some fixed (maybe unknown)  $\omega < \infty$ .

If  $c(\hat{x}, \cdot)$  is concave, then (2.9) can be rewritten as a constrained convex minimization programming, hence any inexact algorithms for constrained convex problems can be used; see, e.g., [25]. The method in [25] is just an inexact bundle method for minimizing a convex function over a closed convex set, which asymptotically evaluates the optimal value for (2.9) with accuracy  $\omega$  and obtains  $\omega$ -optimal solutions  $y(\hat{x})$ . Even if  $c(\hat{x}, \cdot)$  is not concave, just upper semicontinuous on  $Y$ , there are also some fine algorithms for estimating the problem (2.9). The method in [7] extended the classical cutting plane algorithm to tackle the unconstrained nonconvex nonsmooth optimizations, and whose parameters can be input for obtaining a solution with any given accuracy. Recently, W. Hare [11] proposed a proximal bundle method with inexact



oracle for minimizing a nonconvex function over a closed compact set. For a class of nonconvex nonsmooth functions, an approximate critical point can be obtained, in the condition of inexact oracles.

Suppose that  $\hat{x}$  is a current stabilized center. Let  $y(\hat{x}) \in Y$  denote a  $\omega$ -optimal solution to the maximum problem (2.9), that is

$$c(\hat{x}, y(\hat{x})) \geq h(\hat{x}) - \omega.$$

The iterative rules in [7, 11, 25] can generate more and more accurate solutions to (2.9), until a  $\omega$ -optimal solution.

### 2.3 Bundle information

Bundle method are one of the most effective approaches for solving nonsmooth optimization problems. In this subsection, we introduce some essential elements for our bundle method. Suppose that  $J_l^f$  and  $J_l^c$  are the index sets for objective function and constrained functions at the iteration  $l$  respectively. The oracle outputs are collected along iterations to form the Bundle information.

$$\begin{aligned} \mathcal{B}_l^f &:= \left\{ (x^j, f(x^j), g_f^j, \beta_j) : j \in J_l^f \right\} \quad \text{for } J_l^f \subset \{1, \dots, l\}; \\ \mathcal{B}_l^c &:= \left\{ (x^j, c(x^j, y^j), g_{c_j}^j, v_j) : j \in J_l^c \right\} \quad \text{for } J_l^c \subset \{1, \dots, l\}, \end{aligned} \tag{2.10}$$

where  $y^j$  is a point in  $Y$ ,  $g_{c_j}^j \in \partial_x c(x^j, y^j)$ ,  $g_f^j \in \partial f(x^j)$  and

$$\begin{aligned} \beta_j &:= f(x^j) + \langle g_f^j, \hat{x}^k - x^j \rangle, \quad j \in J_l^f, \quad \hat{\beta}_k := f(\hat{x}^k), \\ v_j &:= c(x^j, y^j) + \langle g_{c_j}^j, \hat{x}^k - x^j \rangle, \quad j \in J_l^c, \quad \hat{v}_k := c(\hat{x}^k, y^j). \end{aligned} \tag{2.11}$$

Having the information from (2.10), the  $k$ th improvement function is modelled by a convex function

$$\begin{aligned} \mathcal{M}_l(x) &= \max\{\check{f}_i(x) - f(\hat{x}^k), \check{c}_i(x)\}, \\ \text{where } \begin{cases} \check{f}_i(x) = \max_{j \in J_l^f} \left\{ f(x^j) + \langle g_f^j, x - x^j \rangle \right\}, \\ \check{c}_i(x) = \max_{j \in J_l^c} \left\{ c(x^j, y^j) + \langle g_{c_j}^j, x - x^j \rangle \right\}. \end{cases} \end{aligned} \tag{2.12}$$

An equivalent expression, better suited for implementation is

$$\check{f}_i(\hat{x}^k + d) = \max_{j \in J_l^f} \left\{ \beta_j + \langle g_f^j, d \rangle \right\}, \quad \check{c}_i(\hat{x}^k + d) = \max_{j \in J_l^c} \left\{ v_j + \langle g_{c_j}^j, d \rangle \right\}, \tag{2.13}$$

where  $d = x - \hat{x}^k$ . Given a positive proximal parameter  $\mu_l$ , each solution  $d^l$  is generated by solving the following quadratic program

$$\min_{d \in \mathcal{D}_k} \left\{ \mathcal{M}_l(\hat{x}^k + d) + \frac{\mu_l}{2} \|d\|^2 \right\}, \tag{2.14}$$

where  $\mathcal{D}_k := X - \hat{x}^k$ . With (2.14) as a QP with an extra scalar variable  $r$  as follows

$$\begin{aligned} \min_{(d,r) \in \mathcal{D}_k \times \mathbf{R}} \quad & r + \frac{\mu_l}{2} \|d\|^2, \\ \text{QP}(\mathcal{B}_l) \quad & \text{s.t.} \quad \beta_j - f(\hat{x}^k) + \langle g_f^j, d \rangle \leq r, \quad j \in J_l^f, \\ & v_j + \langle g_{c_j}^j, d \rangle \leq r, \quad j \in J_l^c. \end{aligned} \tag{2.15}$$

Let  $(d^l, r_l)$  be the optimal solution to  $\text{QP}(\mathcal{B}_l)$ , we can obtain that  $r_l = \mathcal{M}_l(\hat{x}^k + d^l)$ , and the new iterative point  $x^{l+1} := \hat{x}^k + d^l$ . Furthermore, it is characterized by the following conditions:

$$d^l = -\frac{1}{\mu_l}(G^l + \alpha^l), \quad r_l = m_l + \langle G^l, d^l \rangle, \tag{2.16}$$

where

$$\begin{aligned} m_l &:= \sum_{j \in J_l^f} \lambda_j^l (\beta_j - f(\hat{x}^k)) + \sum_{j \in J_l^c} \gamma_j^l v_j, \\ \sum_{j \in J_l^f} \lambda_j^l + \sum_{j \in J_l^c} \gamma_j^l &= 1, \quad \lambda_j^l \geq 0, \quad j \in J_l^f, \quad \gamma_j^l \geq 0, \quad j \in J_l^c, \\ G^l &:= \sum_{j \in J_l^f} \lambda_j^l g_f^j + \sum_{j \in J_l^c} \gamma_j^l g_{c_j}^j \in \partial \mathcal{M}_l(x^{l+1}), \quad \alpha^l \in N_X(x^{l+1}). \end{aligned} \tag{2.17}$$

The next lemma discusses the boundedness of the optimal value of the quadratic program  $\text{QP}(\mathcal{B})$ .

**Lemma 2.2** *Let  $w_l$  denote the optimal value of  $\text{QP}(\mathcal{B}_l)$ , i.e.,*

$$w_l := r_l + \frac{\mu_l}{2} \|d^l\|^2.$$

*Then*

$$w_l \leq m_l \leq H_{\hat{x}^k}(\hat{x}^k). \tag{2.18}$$

*Proof* By the definitions of  $r_l, d^l$  in (2.16), for all  $l$ , it holds that

$$\begin{aligned} r_l &= m_l + \langle G^l, d^l \rangle \\ &= m_l - \mu_l \|d^l\|^2 - \langle \alpha^l, x^{l+1} - \hat{x}^k \rangle \\ &\leq m_l - \mu_l \|d^l\|^2, \end{aligned}$$

where the last inequality has used the fact  $\alpha^l \in N_X(x^{l+1})$ . Therefore, it is obvious that

$$w_l \leq m_l - \frac{\mu_l}{2} \|d^l\|^2 < m_l.$$

Besides, by noting (2.11) and the convexity of  $f$  and  $c$ , then

$$\beta_j - f(\hat{x}^k) = f(x^j) + \langle g_f^j, \hat{x}^k - x^j \rangle - f(\hat{x}^k) \leq 0, \quad \forall j \in J_l^f,$$

and

$$v_j = c(x^j, y^j) + \langle g_{c_j}^j, \hat{x}^k - x^j \rangle \leq c(\hat{x}^k, y^j) \leq h(\hat{x}^k), \quad \forall j \in J_l^c.$$

Hence, we conclude that the inequality  $m_l \leq H_{\hat{x}^k}(\hat{x}^k)$  holds. From here the result has been proved.

The following lemma can be used to establish an approximate optimality condition.

**Lemma 2.3** *The vector  $G^l$  satisfies the following relation*

$$H_{\hat{x}^k}(x) \geq H_{\hat{x}^k}(\hat{x}^k) + \langle G^l, x - \hat{x}^k \rangle - \epsilon_l, \tag{2.19}$$

where  $\epsilon_l := H_{\hat{x}^k}(\hat{x}^k) - m_l \geq 0$ . That is  $G^l \in \partial_{\epsilon_l} H_{\hat{x}^k}(\hat{x}^k)$ .

*Proof* By (2.11) and the convexity of  $f$ , it follows that for all  $j \in J_l^f$

$$\begin{aligned} f(x) &\geq f(x^j) + \langle g_f^j, x - x^j \rangle \\ &= f(x^j) + \langle g_f^j, \hat{x}^k - x^j \rangle + \langle g_f^j, x - \hat{x}^k \rangle \\ &= f(\hat{x}^k) + \langle g_f^j, x - \hat{x}^k \rangle - f(\hat{x}^k) + \beta_j. \end{aligned}$$

Similarly we can obtain that

$$\begin{aligned} h(x) &\geq c(x, y^j) \geq c(x^j, y^j) + \langle g_{c_j}^j, x - x^j \rangle \\ &= c(x^j, y^j) + \langle g_{c_j}^j, \hat{x}^k - x^j \rangle + \langle g_{c_j}^j, x - \hat{x}^k \rangle \\ &= v_j + \langle g_{c_j}^j, x - \hat{x}^k \rangle, \end{aligned}$$

for all  $j \in J_l^c$ . By combining with above two inequalities, and using the definition of  $H_{\hat{x}^k}(\cdot)$ , we have

$$\begin{aligned} H_{\hat{x}^k}(x) &= \max\{f(x) - f(\hat{x}^k), h(x)\} \\ &\geq \sum_{j \in J_l^f} \lambda_j^l (f(x) - f(\hat{x}^k)) + \sum_{j \in J_l^c} \gamma_j^l h(x) \\ &\geq \langle G^l, x - \hat{x}^k \rangle + \sum_{j \in J_l^f} \lambda_j^l (\beta_j - f(\hat{x}^k)) + \sum_{j \in J_l^c} \gamma_j^l v_j \\ &\geq H_{\hat{x}^k}(\hat{x}^k) + \langle G^l, x - \hat{x}^k \rangle - (H_{\hat{x}^k}(\hat{x}^k) - m_l). \end{aligned} \tag{2.20}$$

Furthermore, by noting the fact that  $\beta_j - f(\hat{x}^k) \leq 0, v_j \leq c(\hat{x}^k, y^j) \leq h(\hat{x}^k)$ , as well as the definition of  $m_l$ , we can obtain that

$$\epsilon_l = H_{\hat{x}^k}(\hat{x}^k) - m_l \geq 0. \tag{2.21}$$

From here the thesis can be readily proved.

*Remark 1* Lemma 2.1 and Lemma 2.3 play a critical role in establishing an approximate optimality condition. Since  $G^l \in \partial_{\epsilon_l} H_{\hat{x}^k}(\hat{x}^k)$ , we can obtain that

$$G^l + \alpha^l \in \partial_{\epsilon_l} (H_{\hat{x}^k}(\hat{x}^k) + i_X(\hat{x}^k)),$$

where  $i_X(\cdot)$  stands for the indicator function of the set  $X$ . Therefore, if  $\|G^l + \alpha^l\|$  and  $\epsilon_l$  are small enough, the current stabilized center  $\hat{x}^k$  can be accepted as an approximate optimal solution.

### 3 A constrained incremental bundle method

Now we are ready to introduce an incremental bundle method for solving the SIP problem (1.2). Setting the following global parameters for the execution of our method:

- the optimality tolerance  $\epsilon > 0$  and the subgradient approximation measure  $\eta > 0$ ;
- the increase parameter  $\sigma > 1$ .

Three positive parameters subject to possible modifications are also needed, that is the descent parameter  $\delta$ , the proximity parameter  $\mu$  and the inner precision parameter  $\omega$ . In addition, the initial parameters are set, respectively, as  $\delta := \delta_0, \mu := \mu_0 > 1$  and  $\omega := \omega_0$  with  $\omega_0 < \epsilon$ . For convergence analysis, the following condition should be satisfied

$$\eta^2 > \frac{2\mu_0\delta_0}{\iota - 0.5}, \quad \omega_0 = \frac{\eta^2}{4\mu_0}, \quad \iota \in (0.5, 1),$$

which imply that

$$\eta^2 > \frac{2\mu_0(\delta_0 + \omega_0)}{\iota}. \tag{3.1}$$

#### Algorithm 1

##### Step 0 (Input and Initialization).

Select an initial starting point  $x^0$ , a point  $y^0 \in Y, \iota \in (0, 1)$ , the optimality tolerance  $\epsilon > 0$ , the subgradient approximation measure  $\eta > 0$ , and the increase parameter  $\sigma > 1$ . Call oracle to compute  $f(x^0), c(x^0, y^0)$ , as well as  $g_f^0 \in \partial f(x^0), g_{c_0}^0 \in \partial_x c(x^0, y^0)$ .

Initialize the iteration counter  $k = l = 0$ , choose the bundle index set  $L_0^f := \{0\}, L_0^c := \{0\}$ , the first stabilized center  $\hat{x}^0 = x^0$ , the starting prox-parameter  $\mu_0$  as well as  $\delta_0, \omega_0$ .

The main iteration

**Step 1 (Model Generation and QP Subproblem)**

Having the current stabilized center  $\hat{x}^k$ , the bundle  $\mathcal{B}_l$ , the prox-parameter  $\mu_l$ . Solve  $QP(\mathcal{B}_l)$  (2.15), or the corresponding dual problem  $DP(\mathcal{B}_l)$ , and obtain both the optimal primal and dual solutions  $(d^l, r_l)$  and  $\lambda^l, \gamma^l$ . Let  $x^{l+1} := \hat{x}^k + d^l$ . Select a point  $y^{l+1} \in Y$ , and call oracle to compute  $f(x^{l+1}), c(x^{l+1}, y^{l+1})$ , as well as  $g_f^{l+1}, g_{c_l}^{l+1}$ . If

$$\|G^l + \alpha^l\| > \eta, \tag{3.2}$$

then go to Step 3.

**Step 2 (Approximate optimality test)**

Estimate the  $c(\hat{x}^k, y(\hat{x}^k))$ , and let  $c^+(\hat{x}^k, y(\hat{x}^k)) := \max\{0, c(\hat{x}^k, y(\hat{x}^k))\}$ . If

$$c^+(\hat{x}^k, y(\hat{x}^k)) \leq m_l + \varepsilon - \omega, \tag{3.3}$$

where

$$m_l := \sum_{j \in J_f^l} \lambda_j^l (\beta_j - f(\hat{x}^k)) + \sum_{j \in J_c^l} \gamma_j^l v_j,$$

then an approximate optimal solution has been found, then stops.

Else set  $\mu_{l+1} := \sigma \mu_l, \delta_{l+1} := \delta_l / \sigma, \omega_{l+1} := \eta^2 / 4\mu_{l+1}$ , and perform a bundle restoration procedure, that is delete one or more elements from the bundle, while keeping at least the element referred to the stability center  $\hat{x}^k$ , i.e.,

$$J_f^l := \{1\}, J_c^l := \{1\}, \mathcal{B}_f^l := \{\hat{x}^k, f(\hat{x}^k), \hat{g}_f^k, \hat{\beta}_k\}, \mathcal{B}_c^l := \{\hat{x}^k, c(\hat{x}^k, y(\hat{x}^k)), \hat{g}_{c_k}^k, \hat{v}_k\},$$

where  $\hat{\beta}_k = f(\hat{x}^k), \hat{g}_f^k \in \partial f(\hat{x}^k), \hat{v}_k = c(\hat{x}^k, y(\hat{x}^k)), \hat{g}_{c_k}^k \in \partial_x c(\hat{x}^k, y(\hat{x}^k))$ . Set  $l := l + 1$  and return to Step 1.

**Step 3. (Noise Attenuation Test)**

As soon as

$$f(x^{l+1}) - f(\hat{x}^k) > \omega_l + (1 - \iota)m_l + \delta, \tag{3.4}$$

update the bundle

$$\mathcal{B}_{l+1}^f := \mathcal{B}_l^f \cup \left\{ x^{l+1}, f(x^{l+1}), g_f^{l+1}, \beta_{l+1} \right\}, \mathcal{B}_{l+1}^c := \mathcal{B}_l^c. \tag{3.5}$$

Next update the bundle index set  $l := l + 1$ , and return to Step 1.

Otherwise, estimate a point  $\bar{y} := y(x^{l+1}) \in Y$ , and launch the oracle to compute  $c(x^{l+1}, \bar{y}), g_{c_{\bar{y}}}^{l+1} \in \partial_x c(x^{l+1}, \bar{y})$ . If it fulfills another cut property

$$c(x^{l+1}, \bar{y}) > \omega + \delta, \tag{3.6}$$

then we set  $y^{l+1} = \bar{y}$ ,  $g_{c_{l+1}}^{l+1} = g_{c_{\bar{y}}}^{l+1}$ , and  $v_{l+1} := c(x^{l+1}, \bar{y}) + \langle g_{c_{\bar{y}}}^{l+1}, \hat{x}^k - x^{l+1} \rangle$ , and update the bundle as follows

$$\mathcal{B}_{l+1}^c := \mathcal{B}_l^c \cup \left\{ x^{l+1}, c(x^{l+1}, y^{l+1}), g_{c_{l+1}}^{l+1}, v_{l+1} \right\}, \quad \mathcal{B}_{l+1}^f := \mathcal{B}_l^f. \quad (3.7)$$

Next update the bundle index set  $l := l + 1$ , and return to Step 1. Otherwise, set  $\hat{x}^{k+1} = x^{l+1}$ , and exit the main iteration and go to Step 4.

**Step 4 (Bundle update)**

Update the index  $k := k + 1$  and the bundle with respect to the new stabilized center, and return to Step 1.

We now add some another comments of Algorithm 1. Step 2 aims to test the approximate optimality condition introduced in Lemma 2.3. If our algorithm stops at Step 2, namely, the termination of Algorithm 1, takes place when

$$\|G^l + \alpha^l\| \leq \eta, \quad c^+(\hat{x}^k, y(\hat{x}^k)) \leq m_l + \varepsilon - \omega.$$

By noting the fact that  $h(\hat{x}^k) \leq c(\hat{x}^k, y(\hat{x}^k)) + \omega$  and combining with (2.2), we have

$$\begin{aligned} H_{\hat{x}^k}(\hat{x}^k) &\leq c^+(\hat{x}^k, y(\hat{x}^k)) + \omega \\ &\leq m_l + \varepsilon. \end{aligned}$$

Using (2.21), it holds that

$$\epsilon_l = H_{\hat{x}^k}(\hat{x}^k) - m_l \leq \varepsilon.$$

In short, Algorithm 1 stops if and only if

$$\|G^l + \alpha^l\| \leq \eta \quad \text{and} \quad \epsilon_l \leq \varepsilon,$$

which implies, by combining with Lemma 2.1 and Lemma 2.3, that the current stabilized center  $\hat{x}^k$  satisfies KKT conditions (2.5) and (2.6) approximatively. Otherwise, the proximal parameter  $\mu$  will increase, and parameters  $\delta$  and  $\omega$  decrease properly. In the meantime, bundle information will be reset, or rather the elimination of all the elements but the stabilized center, whose related information will be fixed after the initial bundle reset. Accordingly, along with the increase of  $\mu$ , the step length  $d^l$  will become less and less, that means the trial points will be closer to the stabilized center.

It is worth noting that in Step 3, the reduction has been achieved, whenever the decrease property and the condition (3.2) are satisfied. Therefore, whenever Algorithm 1 exits from the main iteration, a new stabilized center has been obtained, and this correlative information is needed to update before a new execution of the main iteration is entered.

### 4 Convergence results

Now it is well known that each new trial point should provide either a better point of the current estimate of the minimum of  $f(\cdot)$ , or at least a better point of the approximation properties of the model  $\mathcal{M}_l(\cdot)$ . The following Lemmas 4.1, 4.2 will give two simple properties named the decrease property and the cut property, respectively.

**Lemma 4.1** *Let  $\delta$  be any positive scalar and  $\iota \in (0, 1)$ . If*

$$\begin{aligned} f(x^{l+1}) - f(\hat{x}^k) &\leq \iota r_l + (1 - \iota)m_l + \delta, \\ c(x^{l+1}, y(x^{l+1})) &\leq \omega + \delta \end{aligned} \tag{4.1}$$

hold, then  $x^{l+1}$  is the serious point, and the following decrease property holds:

$$f(x^{l+1}) \leq f(\hat{x}^k) - \mu_l \iota \|x^{l+1} - \hat{x}^k\|^2 + 2(\omega + \delta). \tag{4.2}$$

*Proof* If the first relation in (4.1) holds, we can obtain that

$$\begin{aligned} f(x^{l+1}) - f(\hat{x}^k) &\leq \iota r_l + (1 - \iota)m_l + \delta \\ &= \sum_{j \in J_l^f} \lambda_j^l (\beta_j - f(\hat{x}^k)) + \sum_{j \in J_l^c} \gamma_j^l v_j + \iota \langle G^l, d^l \rangle + \delta, \end{aligned} \tag{4.3}$$

where the last equality has used the definitions in (2.16) and (2.17). Observe that the serious point  $\hat{x}^k$  satisfies (4.1), we can obtain that

$$c(\hat{x}^k, y(\hat{x}^k)) \leq \omega + \delta.$$

In view of (2.11), and the convexity of  $f(\cdot)$  and  $c(\cdot, y)$ , it holds that

$$\beta_j - f(\hat{x}^k) \leq 0, \text{ for all } j \in J_l^f,$$

and

$$v_j \leq c(\hat{x}^k, y^j) \leq h(\hat{x}^k) \leq 2\omega + \delta, \text{ for all } j \in J_l^c,$$

where the second inequality has used the fact that  $h(\hat{x}^k) \leq c(\hat{x}^k, y(\hat{x}^k)) + \omega$ . By noting  $\sum_{j \in J_l^f} \lambda_j^l + \sum_{j \in J_l^c} \gamma_j^l = 1$ , and combining with the relation (4.3), we have

$$f(x^{l+1}) \leq f(\hat{x}^k) + \iota \langle G^l, x^{l+1} - \hat{x}^k \rangle + 2\omega + 2\delta.$$

Following (2.16), we can observe that

$$\langle G^l, x^{l+1} - \hat{x}^k \rangle = -\mu_l \|x^{l+1} - \hat{x}^k\|^2 - \langle \alpha^l, x^{l+1} - \hat{x}^k \rangle \leq -\mu_l \|x^{l+1} - \hat{x}^k\|^2, \tag{4.4}$$

where the last inequality has used the fact that  $\alpha^l \in N_X(x^{l+1})$ . As a result,

$$f(x^{l+1}) \leq f(\hat{x}^k) - \mu_l \iota \|x^{l+1} - \hat{x}^k\|^2 + 2\omega + 2\delta.$$

From here the result has been proved.

**Lemma 4.2** *Let  $\delta$  be any positive scalar and  $\iota \in (0, 1)$ .*

(i) *If*

$$f(x^{l+1}) - f(\hat{x}^k) > \iota r_l + (1 - \iota)m_l + \delta$$

*holds, then  $(d^l, r_l)$  is not feasible for problem  $QP(\mathcal{B}_{l+1})$  by resetting the bundle, such as*

$$\mathcal{B}_{l+1}^f := \mathcal{B}_l^f \cup \{x^{l+1}, f(x^{l+1}), g_f^{l+1}, \beta_{l+1}\}, \quad \mathcal{B}_{l+1}^c := \mathcal{B}_l^c. \tag{4.5}$$

(ii) *If the point  $y^{l+1} \in Y$  satisfying*

$$c(x^{l+1}, y^{l+1}) > \omega + \delta,$$

*then  $(d^l, r_l)$  is not feasible for problem  $QP(\mathcal{B}_{l+1})$  by resetting the bundle, such as*

$$\mathcal{B}_{l+1}^c := \mathcal{B}_l^c \cup \{x^{l+1}, c(x^{l+1}, y^{l+1}), g_{c_{l+1}}^{l+1}, v_{l+1}\}, \quad \mathcal{B}_{l+1}^f := \mathcal{B}_l^f. \tag{4.6}$$

*Proof* If the first condition holds, i.e.,  $f(x^{l+1}) - f(\hat{x}^k) > \iota r_l + (1 - \iota)m_l + \delta$ . By using (2.11) and (2.16), we can obtain that

$$\begin{aligned} & \beta_{l+1} - f(\hat{x}^k) + \langle g_f^{l+1}, d^l \rangle \\ &= f(x^{l+1}) - f(\hat{x}^k) \\ &> \iota r_l + (1 - \iota)m_l + \delta \\ &= r_l + \mu_l(1 - \iota)\|d^l\|^2 - (1 - \iota)\langle \alpha^l, \hat{x}^k - x^{l+1} \rangle + \delta \\ &\geq r_l + \delta, \end{aligned}$$

where the last inequality follows from  $\alpha^l \in N_X(x^{l+1})$  and  $\iota \in (0, 1)$ . Thus  $(d^l, r_l)$  is not feasible for problem  $QP(\mathcal{B}_{l+1})$ .

Since  $f$  and  $c(\cdot, y)$  for any  $y \in Y$  are convex, by noting (2.11), we have

$$\beta_j - f(\hat{x}^k) + \langle g_f^j, 0 \rangle \leq 0, \quad \text{for all } j \in J_l^f,$$

and

$$\begin{aligned} & v_j + \langle g_{c_j}^j, 0 \rangle \leq c(\hat{x}^k, y^j) \\ &\leq h(\hat{x}^k) \leq c(\hat{x}^k, y(\hat{x}^k)) + \omega \\ &\leq 2\omega + \delta, \end{aligned}$$



for all  $j \in J_l^c$ , where the last inequality has used  $c(\hat{x}^k, y(\hat{x}^k)) \leq \omega + \delta$ . Thus,  $(0, 2\omega + \delta)$  is a feasible point of the corresponding quadratic program. Using (2.16) and noting  $(d^l, r_l)$  is the solution of the corresponding quadratic program, then the optimal values  $w_l$  is smaller than  $2\omega + \delta$ , i.e.,

$$r_l + \frac{1}{2\mu_l} \|G^l + \alpha^l\|^2 = r_l + \frac{\mu_l}{2} \|d^l\|^2 \leq 2\omega + \delta,$$

which, taking into account  $\|G^l + \alpha^l\| > \eta$  and  $\omega = \eta^2/4\mu_l$ , implies that

$$r_l \leq 2\omega + \delta - \frac{1}{2\mu_l} \|G^l + \alpha^l\|^2 = \delta. \tag{4.7}$$

Moreover, if the condition (ii) holds, by using the above inequality, we can obtain

$$\begin{aligned} & v_{l+1} + \langle g_{c_{l+1}}^{l+1}, d^l \rangle \\ &= c(x^{l+1}, y^{l+1}) \\ &> \omega + \delta \\ &\geq r_l + \omega, \end{aligned}$$

where the first inequality follows from (3.6), and the last one has used the (4.7). It follows that  $(d^l, r_l)$  is not feasible for  $\text{QP}(\mathcal{B}_{l+1})$ . Thus the thesis easily follows.

We now give a preliminary result for the global convergence of Algorithm 1.

**Lemma 4.3** *Suppose that  $w_l$  and  $w_{l+1}$  are, respectively, the optimal values of  $\text{QP}(\mathcal{B}_l)$  and  $\text{QP}(\mathcal{B}_{l+1})$ . Then*

$$w_{l+1} \geq w_l + \frac{\mu_l}{2} \|d^{l+1} - d^l\|^2. \tag{4.8}$$

*Proof* Suppose that  $(\bar{d}, \bar{r})$  is any feasible point for  $\text{QP}(\mathcal{B}_l)$ , then it fulfils the following inequalities

$$\begin{aligned} \beta_j - f(\hat{x}^k) + \langle g_f^j, \bar{d} \rangle &\leq \bar{r}, \quad j \in J_l^f, \\ v_j + \langle g_{c_j}^j, \bar{d} \rangle &\leq \bar{r}, \quad j \in J_l^c. \end{aligned}$$

By combining with (2.17), we obtain that

$$m_l + \langle G^l, \bar{d} \rangle \leq \bar{r}.$$

Observe that  $(d^{l+1}, r_{l+1})$  is a feasible point for  $\text{QP}(\mathcal{B}_l)$ , then one gets

$$m_l + \langle G^l, d^{l+1} \rangle \leq r_{l+1}.$$

Since  $(d^l, r_l)$  is the optimal solution of  $QP(\mathcal{B}_l)$ , it follows from (2.16) that

$$m_l + \langle G^l, d^l \rangle = r_l.$$

By the above two relations, thus

$$\langle G^l, d^{l+1} - d^l \rangle \leq r_{l+1} - r_l. \tag{4.9}$$

Observe further that

$$\begin{aligned} w_{l+1} &= r_{l+1} + \frac{\mu_{l+1}}{2} \|d^{l+1}\|^2 \\ &\geq r_{l+1} - r_l + r_l + \frac{\mu_l}{2} \|d^{l+1} - d^l + d^l\|^2 \\ &\geq \langle G^l, d^{l+1} - d^l \rangle + w_l + \frac{\mu_l}{2} \|d^{l+1} - d^l\|^2 + \mu_l \langle d^{l+1} - d^l, d^l \rangle, \end{aligned} \tag{4.10}$$

where the last inequality has used (4.9) and the definition of  $w_l$ . By noting the fact that  $d^l = -\frac{1}{\mu_l}(G^l + \alpha^l)$  and  $\alpha^l \in N_X(x^{l+1})$ , it obtains that

$$\begin{aligned} &\mu_l \langle d^{l+1} - d^l, d^l \rangle \\ &= -\langle G^l, d^{l+1} - d^l \rangle - \langle \alpha^l, x^{l+2} - x^{l+1} \rangle \\ &\geq -\langle G^l, d^{l+1} - d^l \rangle, \end{aligned}$$

which implies, together with (4.10), that (4.8) holds.

**Lemma 4.4** *Suppose that  $(d^l, r_l)$  is the optimal solution to  $QP(\mathcal{B}_l)$ . Let  $j(k) \in J_l^f$  denote an iteration index yielding a serious step:  $\hat{x}^k = x^{j(k)} := \hat{x}^{k-1} + d^{j(k)-1}$ . The following properties hold:*

- (1)  $r_l \leq H_{\hat{x}^k}(\hat{x}^k)$ ;
- (2)  $r_l \leq H_{\hat{x}^k}(x^{l+1})$ ;
- (3)  $d^l \in V_{R_{l(k)}}(0)$  (a neighbourhood of 0 with radius  $R_{l(k)}$ ), where

$$R_{l(k)} = \begin{cases} \frac{2\|g_f^{j(k)}\|}{\mu_l}, & \text{if } m_l \leq 0; \\ \frac{\|G^l\| + \sqrt{\|G^l\|^2 + 2\mu_l H_{\hat{x}^k}(\hat{x}^k)}}{\mu_l}, & \text{if } m_l > 0. \end{cases} \tag{4.11}$$

*Proof* (1) Since  $f$  is convex, by (2.11) and (2.15), one gets

$$\begin{aligned} &\beta_l - f(\hat{x}^k) + \langle g_f^l, 0 \rangle \\ &= -(f(\hat{x}^k) - f(x^l) - \langle g_f^l, \hat{x}^k - x^l \rangle) \\ &\leq 0. \end{aligned}$$

On the other hand, by noting the convexity of  $c(\cdot, y^l)$ , we can obtain that

$$\begin{aligned} & v_l + \langle g_{c_l}^l, 0 \rangle \\ &= c(x^l, y^l) + \langle g_{c_l}^l, \hat{x}^k - x^l \rangle \\ &\leq c(\hat{x}^k, y^l). \end{aligned}$$

Recalling the definition of  $H_{\hat{x}^k}(\cdot)$ , we can obtain that  $(0, H_{\hat{x}^k}(\hat{x}^k))$  is a feasible point of  $\text{QP}(\mathcal{B}_l)$ . Thus,

$$r_l \leq r_l + \frac{\mu_l}{2} \|d^l\|^2 \leq H_{\hat{x}^k}(\hat{x}^k).$$

- (2) It follows by observing that  $r_l = \mathcal{M}_l(\hat{x}^k + d^l) \leq H_{\hat{x}^k}(\hat{x}^k + d^l)$ .
- (3) Suppose now that  $m_l \leq 0$ , it follows from (2.16) that

$$r_l \leq \langle G^l, d^l \rangle = -\mu_l \|d^l\|^2 - \langle \alpha^l, x^{l+1} - \hat{x}^k \rangle < 0.$$

Thus  $(0, 0)$  is the feasible point for  $\text{QP}(\mathcal{B}_l)$ .  
By noting the relation (2.11), it holds that

$$\begin{aligned} \beta_{j(k)} &:= f(x^{j(k)}) + \langle g_f^{j(k)}, \hat{x}^k - x^{j(k)} \rangle \\ &= f(\hat{x}^k) = \hat{\beta}_k. \end{aligned}$$

Notice that  $(0, 0)$  is the feasible point for  $\text{QP}(\mathcal{B}_l)$ , then the optimal value  $w_l = r_l + \frac{\mu_l}{2} \|d^l\|^2$  is smaller than 0. Thus we can obtain that

$$\begin{aligned} 0 &\geq r_l + \frac{\mu_l}{2} \|d^l\|^2 \\ &\geq \beta_{j(k)} - f(\hat{x}^k) + \langle g_f^{j(k)}, d^l \rangle + \frac{\mu_l}{2} \|d^l\|^2 \\ &= \langle g_f^{j(k)}, d \rangle + \frac{\mu_l}{2} \|d^l\|^2 \\ &\geq \frac{\mu_l}{2} \|d^l\|^2 - \|g_f^{j(k)}\| \|d^l\|, \end{aligned}$$

where the second inequality has used the fact that  $r_l \geq \beta_{j(k)} - f(\hat{x}^k) + \langle g_f^{j(k)}, d^l \rangle$ . Thus, we can obtain that

$$\|d^l\| \leq \frac{2\|g_f^{j(k)}\|}{\mu_l}.$$

On the other hand, if  $m_l > 0$  holds, then by noting the fact that  $(0, H_k(\hat{x}^k))$  is feasible point for  $\text{QP}(\mathcal{B}_l)$ , we obtain that the optimal value  $w_l = r_l + \frac{\mu_l}{2} \|d^l\|^2$  is smaller than  $H_k(\hat{x}^k)$ , i.e.,

$$H_{\hat{x}^k}(\hat{x}^k) \geq r_l + \frac{\mu_l}{2} \|d^l\|^2.$$

Furthermore, by noting (2.16), we obtain that

$$\begin{aligned} 0 &\geq \frac{\mu_l}{2} \|d^l\|^2 + \langle G^l, d^l \rangle + m_l - H_{\hat{x}^k}(\hat{x}^k) \\ &\geq \frac{\mu_l}{2} \|d^l\|^2 - \|G^l\| \|d^l\| - H_{\hat{x}^k}(\hat{x}^k). \end{aligned}$$

Thus

$$\|d^l\| \leq \frac{\|G^l\| + \sqrt{\|G^l\|^2 + 2\mu_l H_{\hat{x}^k}(\hat{x}^k)}}{\mu_l},$$

From here all the items have been proven.

To prove global convergence of Algorithm 1, it is necessary to obtain the finite termination of the “main iteration”. The finiteness of the “main iteration” corresponds to prove that one of the following cases can occur:

- either there are finitely many passages through Step 2, or
- the “main iteration” passes finitely many times through Step 3.

In what follows, we suppose that  $f(\cdot)$  and  $c(\cdot, y)$  for any  $y \in Y$  are boundedness from below on subset  $X$ , which is a standing assumption in nonsmooth optimization. Moreover, let  $L$  denote the Lipschitz constant of  $f(\cdot)$  and  $c(\cdot, y)$  for any  $y \in Y$  on subset  $X$ , then we can obtain that  $\|g_f^l\| \leq L, \|g_{c_y}^l\| \leq L$  for all  $y \in Y$ , and then  $\|G^l\| \leq L$ .

In the following two lemmas, we can obtain that the “main iteration” is well defined.

**Lemma 4.5** *There are finitely many passages through Step 2. That is only finite loops between Step 1 and Step 2 are executed.*

*Proof* For a contradiction, we assume there are infinitely many passages through Step 2. For convenience, let  $\hat{x}^k$  and  $t$  denote, respectively, the current fixed stabilized center and the index of the  $t$ th passage through Step 2. Furthermore, let  $(\tilde{d}^t, \tilde{r}_t), (\tilde{\lambda}^t, \tilde{\gamma}^t), \tilde{R}_t, \tilde{\mu}_t$  and  $\tilde{\omega}_t$  denote, respectively, the primal solution of the current quadratic program, the matching multipliers, the upper bound of  $\tilde{d}^t$ , the prox-parameter and the inner precision parameter.

Infinitely many passages through Step 2 imply the following conditions:

$$\|\tilde{G}^t + \tilde{\alpha}^t\| \leq \eta,$$

and

$$c^+(\hat{x}^k, y(\hat{x}^k)) > \tilde{m}_t + \varepsilon - \tilde{\omega}_t \tag{4.12}$$

are simultaneously satisfied infinitely many times. Observe that the bundle will be always restored, which ensures the elements

$$\begin{aligned} \mathcal{B}_f^l &:= \{\hat{x}^k, f(\hat{x}^k), \hat{g}_f^k, \hat{\beta}_k\}, & \text{where } \hat{\beta}_k &= f(\hat{x}^k), \\ \mathcal{B}_c^l &:= \{\hat{x}^k, c(\hat{x}^k, y(\hat{x}^k)), \hat{g}_{c_k}^k, \hat{v}_k\}, & \text{where } \hat{v}_k &= c(\hat{x}^k, y(\hat{x}^k)) \end{aligned}$$

are always preserved in the bundle. By noting (4.11) in Lemma 4.4, we get that

$$\|\tilde{d}^t\| \leq \tilde{R}_t \leq \max \left\{ \frac{2L}{\tilde{\mu}_t}, \frac{2L + \sqrt{2\tilde{\mu}_t} H_{\hat{x}^k}(\hat{x}^k)}{\tilde{\mu}_t} \right\}.$$

By noting  $\tilde{\mu}_t \rightarrow \infty$ , we can obtain that  $\|\tilde{d}^t\|$  tends to 0 as  $t \rightarrow \infty$ . Since  $(\tilde{d}^t, \tilde{r}_t)$  is the solution of  $QP(\mathcal{B})$ , then it is of course a feasible point of the problem, namely,

$$\tilde{r}_t \geq \hat{\beta}_k - f(\hat{x}^k) + \langle \hat{g}_f^k, \tilde{d}^t \rangle \geq -L\|\tilde{d}^t\|,$$

and

$$\tilde{r}_t \geq \hat{\nu}_k + \langle \hat{g}_{c_k}^k, \tilde{d}^t \rangle \geq c(\hat{x}^k, y(\hat{x}^k)) - L\|\tilde{d}^t\|,$$

which imply that

$$\tilde{r}_t \geq c^+(\hat{x}^k, y(\hat{x}^k)) - L\|\tilde{d}^t\|.$$

Besides, since  $f$  and  $c(\cdot, y)$  for any  $y \in Y$  are convex, by noting (2.11), we have

$$\beta_j - f(\hat{x}^k) + \langle g_f^j, 0 \rangle \leq 0, \text{ for all } j \in \tilde{J}_t^f,$$

and

$$\nu_j + \langle g_{c_j}^j, 0 \rangle \leq c(\hat{x}^k, y^j) \leq h(\hat{x}^k) \leq c(\hat{x}^k, y(\hat{x}^k)) + \tilde{\omega}_t, \text{ for all } j \in \tilde{J}_t^c,$$

which imply that  $(0, c^+(\hat{x}^k, y(\hat{x}^k)) + \tilde{\omega}_t)$  is the feasible point of the matching quadratic program. Observe that  $(\tilde{d}^t, \tilde{r}_t)$  is the solution of the matching quadratic program, then the corresponding optimal values  $\tilde{w}_t$  is smaller than  $c^+(\hat{x}^k, y(\hat{x}^k)) + \tilde{\omega}_t$ , i.e.,

$$\tilde{w}_t := \tilde{r}_t + \frac{\tilde{\mu}_t}{2} \|\tilde{d}^t\|^2 \leq c^+(\hat{x}^k, y(\hat{x}^k)) + \tilde{\omega}_t.$$

Hence we have that

$$c^+(\hat{x}^k, y(\hat{x}^k)) + \tilde{\omega}_t \geq \tilde{r}_t \geq c^+(\hat{x}^k, y(\hat{x}^k)) - L\|\tilde{d}^t\|.$$

Now, by noting the fact that  $\lim_{t \rightarrow \infty} \tilde{\omega}_t = 0$  and  $\lim_{t \rightarrow \infty} \|\tilde{d}^t\| = 0$ , we have

$$\lim_{t \rightarrow \infty} \tilde{r}_t = c^+(\hat{x}^k, y(\hat{x}^k)).$$

Moreover, by using (2.16), we obtain that  $\tilde{m}_t \rightarrow c^+(\hat{x}^k, y(\hat{x}^k))$  as  $t \rightarrow \infty$ , which is a contradiction with (4.12). From here the result has been proved.

**Lemma 4.6** *The “main iteration” is impossible to loop infinitely many times between Step 1 and Step 3.*

*Proof* Suppose first, for a contradiction, there are infinitely many passages through Step 3. Note the quadratic program  $QP(\mathcal{B})$  will be solved infinitely many times, with the cut property (3.4) or (3.6) holds. For convenience, let  $\hat{x} = \hat{x}^k$ ,  $\tilde{\mu}$  and  $\tilde{\omega}$  denote, respectively, the current fixed stabilized center, the current fixed prox-parameter and the fixed inner precision parameter. Let  $t$ ,  $(\tilde{d}_t, \tilde{r}_t)$  and  $(\tilde{\lambda}^t, \tilde{\gamma}^t)$  stand for, respectively, the index of the  $t$ th passage through Step 3, the primal solution and the matching multipliers of the current quadratic program.

Then the matching optimal values can be denoted as

$$\tilde{w}_t := \tilde{r}_t + \frac{\tilde{\mu}}{2} \|\tilde{d}^t\|^2.$$

According to Lemma 2.2 and Lemma 4.3, we obtain that  $\{\tilde{w}_t\}$  is monotonously non-decreasing and bounded, thus convergent. The latter implies, again by Lemma 4.3, that

$$\|\tilde{d}^{t+1} - \tilde{d}^t\|^2 \rightarrow 0 \text{ as } t \rightarrow \infty,$$

which combines with the boundedness of  $\{\|\tilde{d}^t\|\}$ , it holds that  $\{\tilde{d}^t\}$  is convergent. As a consequence, by noting that  $\tilde{r}_t = \frac{\tilde{\mu}}{2} \|\tilde{d}^t\|^2 - \tilde{w}_t$ , then the sequence  $\{\tilde{r}_t\}$  is also convergent.

Infinitely many passages through Step 3 imply the following conditions hold infinitely many times

$$\|\tilde{G}^t + \tilde{\alpha}^t\| > \eta,$$

and either

$$f(\tilde{x}^{t+1}) - f(\hat{x}) > \iota \tilde{r}_t + (1 - \iota) \tilde{m}_t + \delta, \tag{4.13}$$

or

$$c(\tilde{x}^{t+1}, y(\tilde{x}^{t+1})) > \tilde{\omega} + \delta. \tag{4.14}$$

If the inequality (4.13) holds infinitely many times. Observe that  $(\tilde{d}^{t+1}, \tilde{r}_{t+1})$  is the solution of  $QP(\mathcal{B}_{t+1})$ , thus it is of course the feasible point of the current quadratic program, namely,

$$\begin{aligned} \tilde{r}_{t+1} &\geq \tilde{\beta}_{t+1} - f(\hat{x}) + \langle \tilde{g}_f^{t+1}, \tilde{d}^{t+1} \rangle \\ &= f(\tilde{x}^{t+1}) - f(\hat{x}) + \langle \tilde{g}_f^{t+1}, \tilde{d}^{t+1} - \tilde{d}^t \rangle \\ &> \iota \tilde{r}_t + (1 - \iota) \tilde{m}_t + \delta + \langle \tilde{g}_f^{t+1}, \tilde{d}^{t+1} - \tilde{d}^t \rangle, \end{aligned}$$

where the last inequality follows from (4.13). Thus, we can obtain that

$$\begin{aligned}
 & \delta + \langle \tilde{g}_f^{t+1}, \tilde{d}^{t+1} - \tilde{d}^t \rangle \\
 & \leq (\tilde{r}_{t+1} - \tilde{r}_t) + (1 - \iota)(\tilde{r}_t - \tilde{m}_t) \\
 & = (\tilde{r}_{t+1} - \tilde{r}_t) + (1 - \iota)\langle \tilde{G}^t, \tilde{d}^t \rangle \\
 & \leq \tilde{r}_{t+1} - \tilde{r}_t,
 \end{aligned} \tag{4.15}$$

where the last inequality has used the relation (4.4), and the equality follows from (2.16). Passing to the limit and noting the boundedness of  $\tilde{g}_f^{t+1}$ , it holds that  $\delta \leq 0$ , which is a contradiction.

On the other hand, if (4.14) holds infinitely many times, by noting  $(\tilde{d}^{t+1}, \tilde{r}_{t+1})$  is the solution of the matching quadratic program, we can obtain that

$$\begin{aligned}
 \tilde{r}_{t+1} & \geq \tilde{v}_{t+1} + \langle \tilde{g}_{c_{t+1}}^{t+1}, \tilde{d}^{t+1} \rangle \\
 & = c(\tilde{x}^{t+1}, y(\tilde{x}^{t+1})) + \langle \tilde{g}_{c_{t+1}}^{t+1}, \tilde{d}^{t+1} - \tilde{d}^t \rangle \\
 & \geq \tilde{\omega} + \delta - L\|\tilde{d}^{t+1} - \tilde{d}^t\| \\
 & \geq \tilde{\omega} + \tilde{r}_t - L\|\tilde{d}^{t+1} - \tilde{d}^t\|,
 \end{aligned}$$

where the last inequality has used the fact  $\delta \geq \tilde{r}_t$  in (4.7). Thus it holds that

$$\tilde{r}_{t+1} - \tilde{r}_t \geq \tilde{\omega} - L\|\tilde{d}^{t+1} - \tilde{d}^t\|.$$

Passing to the limit as  $t \rightarrow \infty$ , we obtain that  $\tilde{\omega} \leq 0$ , which is a contradiction. □

Lastly, we can prove global convergence of Algorithm 1.

**Theorem 4.1** *Consider solving the nonsmooth convex SIPs (1.2) with Algorithm 1. For any  $\varepsilon > 0$  and  $\eta > 0$ , Algorithm 1 stops after finitely many executions of the “main iteration” at an approximate optimum point  $\tilde{x}$ , satisfying the following approximate optimality condition:*

$$\|\tilde{G} + \tilde{\alpha}\| \leq \eta \text{ and } \tilde{\varepsilon} := H_{\tilde{x}}(\tilde{x}) - \tilde{m} \leq \varepsilon,$$

where  $\tilde{m}$  has been defined in (2.17).

*Proof* For a contradiction, we assume there are an infinite number of “main iteration” executions, which mean that the stabilized center is updated infinitely. Let  $k, (\hat{d}^k, \hat{r}_k)$  and  $(\hat{\lambda}^k, \hat{\gamma}^k)$  denote the index of the  $k$ th “main iteration”, the primal solution of the current QP and the matching multipliers, and  $\hat{J}_k^f, \hat{J}_k^c$  stand for the matching index sets.

We observe that the stabilized center  $\hat{x}^k$  should be updated only if

$$\|\hat{G}^k + \hat{\alpha}^k\| > \eta$$

holds. Let  $p_k$  denote number of bundle resets carried out during the  $k$ th “main iteration”. It follows from Lemma 4.5 that  $p_k$  is always is bounded. Moreover, by Lemma 4.1, we can obtain that

$$\begin{aligned}
 f(\hat{x}^{k+1}) - f(\hat{x}^k) &\leq -\mu_k \iota \|\hat{d}^k\|^2 + 2(\delta + \omega) \\
 &= -\frac{\iota}{\mu_k} \|\hat{G}^k + \hat{\alpha}^k\|^2 + 2(\delta + \omega) \\
 &\leq -\frac{\eta \eta^2}{\mu_k} + 2(\delta + \omega) \\
 &= \left(-\frac{\eta \eta^2}{\mu_0} + 2\delta_0 + 2\omega_0\right) \frac{1}{(\sigma)^{pk}},
 \end{aligned}$$

where the first equality has used (2.16) and the last equality is from Step 2 in Algorithm 1. Therefore, by summing up the above relation, we can obtain that

$$f(\hat{x}^{k+1}) - f(\hat{x}^0) \leq a \sum_{j=0}^k \frac{1}{(\sigma)^{pj}},$$

where  $a := -\frac{\eta \eta^2}{\mu_0} + 2\delta_0 + 2\omega_0$ , which is negative follows from (3.1). By noting the boundness of  $p_j$ , then  $\frac{1}{(\sigma)^{pj}}$  is uniformly bounded away from zero. As a result, passing to the limit, we can obtain

$$\lim_{k \rightarrow \infty} f(\hat{x}^{k+1}) - f(\hat{x}^0) \leq a \lim_{k \rightarrow \infty} \sum_{j=0}^k \frac{1}{(\sigma)^{pj}} \leq -\infty,$$

which is a contradiction, since  $f$  is bounded from below.

Therefore, Algorithm 1 will eventually stop at Step 2, and output an approximate point  $\tilde{x}$ , which satisfies the following conditions:

$$\|\tilde{G} + \tilde{\alpha}\| \leq \eta, \tag{4.16}$$

and

$$H_{\tilde{x}}(\tilde{x}) - \omega \leq c^+(\tilde{x}, y(\tilde{x})) \leq \tilde{m} + \varepsilon - \omega,$$

implies that  $H_{\tilde{x}}(\tilde{x}) - \tilde{m} \leq \varepsilon$ . By Lemmas 2.1 and 2.3, it holds that

$$\tilde{G} + \tilde{\alpha} \in \partial_\varepsilon H_{\tilde{x}}(\tilde{x}) + N_X(\tilde{x}),$$

which, combining with (4.16), implies that  $\tilde{x}$  satisfies the approximate optimality conditions.

### 5 Computational results

We complete Algorithm 1 in MATLAB and the numerical experiments were done by using a PC with 1.80GHz CPU. Quadratic programming solver is QuadProg.m, which is available in the Optimization Toolbox. In our experiments, we choose the values for all the parameters as follows:

$$\varepsilon = 10^{-3}, \eta = 10^{-3}, \delta_0 = 10^{-6}, \iota = 0.56, \sigma = 10, \mu_0 = 10.$$



Now we try to handle the maximum problem (2.9), i.e.,

$$\max_{y \in Y} c(\bar{x}, y),$$

where  $\bar{x}$  is the current iterative point. There are a few excellent methods for solving both convex and nonconvex optimization, and the solution with any accuracy can be returned [7, 11]. In particular, [11] proposed an inexact nonconvex bundle if the  $Y$  is a compact set and simple enough. If the objective function  $c(\bar{x}, \cdot)$  is concave, then (2.9) is a constrained concave maximization program. Therefore, any standard algorithms for minimizing constrained convex problem are suitable for our algorithm.

### 5.1 Computational results for nonsmooth convex semi-infinite problems

In this subsection, we tested 8 nonsmooth convex semi-infinite examples, namely, Examples 1-8.

*Example 1*

$$\begin{aligned} \min f(x) &:= 1.21 \exp(|x_1|) + \exp(|x_2|), \\ \text{s.t. } c(x, y) &:= y - \exp(x_1 + x_2) \leq 0, \\ &y \in [-10, 1]. \end{aligned}$$

*Example 2*

$$\begin{aligned} \min f(x) &:= \frac{1}{3}|x_1| + \frac{1}{2}|x_1| + |x_2|, \\ \text{s.t. } c(x, y) &:= (1 - x_1^2 y^2)^2 - x_1 y^2 - |x_2| + x_2 \leq 0, \\ &y \in [-1, 1]. \end{aligned}$$

*Example 3*

$$\begin{aligned} \min f(x) &:= |x_1 - 2x_2 + 5x_2^2 - x_2^3 - 13| + |x_1 - 14x_2 + x_2^2 + x_2^3 - 29|, \\ \text{s.t. } c(x, y) &:= |x_1| + 2x_2 y^2 + \exp(x_1 + x_2) - \exp(y) \leq 0, \\ &y \in [0, 1]. \end{aligned}$$

*Example 4*

$$\begin{aligned} \min f(x) &:= |x_1 - 2| + |x_2|, \\ \text{s.t. } c(x, y) &:= |x_1| \cos(y_1) - x_2 \sin(y_2) - 4 \leq 0, \\ &y \in [0, 2\pi] \times [0, 2\pi]. \end{aligned}$$

*Example 5*

$$\begin{aligned} \min f(x) &:= |x_1| + |x_2| + |x_3|, \\ \text{s.t. } c(x, y) &:= x_1 + x_2 \exp(x_3 y) - \exp(2x_1 y) + \sin(4y) \leq 0, \\ &y \in [0, 1]. \end{aligned}$$

**Example 6**

$$\begin{aligned} \min f(x) &:= |x_1| + |x_2| + |x_3|, \\ \text{s.t. } c(x, y) &:= x_1 + x_2 \exp(x_3 y_1) - \exp(2y_2) + \sin(4y_1) \leq 0, \\ &y \in [0, 1] \times [0, 1]. \end{aligned}$$

**Example 7**

$$\begin{aligned} \min f(x) &:= \frac{1}{2}(|x_1| + |x_2| + |x_3| + |x_4|), \\ \text{s.t. } c(x, y) &:= \sin(y_1 y_2) - x_1 - x_2 y_1 - x_3 y_2 - x_4 y_1 y_2 \leq 0, \\ &y \in [0, 1] \times [0, 1]. \end{aligned}$$

**Example 8**

$$\begin{aligned} \min f(x) &:= \frac{1}{2}(|x_1| + |x_2| + |x_3| + |x_4| + |x_5| + |x_6|), \\ \text{s.t. } c(x, y) &:= \exp(y_1^2 + y_2^2) - x_1 - x_2 y_1 - x_3 y_2 - x_4 y_1^2 - x_5 y_1 y_2 - x_6 y_2^2 \leq 0, \\ &y \in [0, 1] \times [0, 1]. \end{aligned}$$

The numerical results are listed in Tables 1 and 2, and notations are used as follows:

$$\begin{aligned} \text{stop}_{\eta_k} &:= \|\hat{G}^k + \hat{\alpha}^k\|, \\ \text{stop}_{\varepsilon_k} &:= c^+(\hat{x}^k, y(\hat{x}^k)) - \hat{m}_k \\ (x_0, y_0) &\text{---the initial point,} \\ f^* &\text{---the final objective value,} \\ h^* &\text{---the final constrained value,} \\ N_i &\text{---the number of null iterations,} \\ k &\text{---the number of serious iterations,} \\ \text{Time}_{\max} &\text{---the CPU time for solving (2.9)(sec.),} \\ \text{Time} &\text{---the CPU time(sec.).} \end{aligned}$$

The test results are summarized in Tables 1 and 2, and show that Algorithm 1 performs well for Examples 1–8. From Table 1, we can note the value of  $\text{stop}_{\eta_k}$  and  $\text{stop}_{\varepsilon_k}$  drop to zero fast in the last three iterations, which means that Algorithm 1 is convergent quickly. From Table 2, the value of  $h^*$  shows that Algorithm 1 can ensure the feasibility for all the test examples. Also, we can note the CPU time for estimating (2.9) ( $\text{Time}_{\max}$ ) is extremely short, which shows the inexact oracle idea is reasonable. Above all, the  $\text{Time}$  shows that our method is efficient for solving nonsmooth SIP problem (1.2).

## 5.2 Comparison of Algorithm 1 with classic algorithms

In this subsection, we first test performance on the following five classical convex SIPs (Examples 9–13). We compare the performance of Algorithm 1 with that of the central cutting plane algorithm (CCPA) as detailed in [26] and the SIP solver *fseminf* in MATLAB toolbox.

**Table 1** The last three iterates generated by Algorithm 1 for Examples 1–8

Example	$k$	$\text{Stop}_{\eta_k}$	$\text{Stop}_{\varepsilon_k}$	Example	$k$	$\text{Stop}_{\eta}$	$\text{Stop}_{\varepsilon}$
1	9	0.0592	0.0353	2	8	0.0742	0.0316
	10	0.0037	0.0048		9	0.0037	0.0093
	11	4.0137e-4	6.1780e-4		10	3.4673e-4	8.4378e-4
3	8	0.0129	0.0458	4	11	0.0085	0.0467
	9	0.0073	0.0104		12	0.0073	0.0062
	10	6.1751e-4	6.8305e-4		13	5.6038e-4	3.9324 e-4
5	9	0.1032	0.0739	6	11	0.0347	0.0073
	10	0.0367	0.0084		12	0.0071	0.0020
	11	8.1764e-4	5.7030e-4		13	7.3020e-4	8.5014e-4
7	10	0.0629	0.0538	8	14	0.0873	0.0130
	11	0.0572	0.0143		15	0.0119	0.0036
	12	7.7490e-4	7.3080e-4		16	6.9007e-4	7.5431e-4

**Table 2** Test results by Algorithm 1 for Examples 1–8

Example	$x_0/y_0$	$k \text{Ni}$	$\text{Stop}_{\eta^*} \text{stop}_{\varepsilon^*}$	$f^*$	$h^*$	$\text{Time}_{\max}$	Time
1	(1,1)/1	11 21	4.01e-4   6.10e-4	1.24	0	0.08	1.07
2	(-1,-1)/1	10 19	3.46e-4   8.43e-4	2.34	5.34e-7	0.09	1.61
3	(-1,1)/1	10 21	6.17e-4   6.83e-4	37.24	0	0.10	1.04
4	(-1,-1)/(1,0)	13 20	5.60e-4   3.93 e-4	0	0	0.11	1.62
5	(1,1,1)/1	11 18	8.17e-4   5.70e-4	0	0	0.10	1.04
6	(-1,-1,-1)/(0, 1)	13 17	7.30e-4   8.50e-4	0	0	0.13	1.47
7	(1, 2, 1, 2)/(1, 0)	12 20	7.74e-4   7.30e-4	0.04	3.37e-8	0.12	1.60
8	(-2,-2,-2,-2,-2,-2)/(1, 1)	16 19	6.90e-4 7.54e-4	1.48	0	0.11	1.47

*Example 9* Consider the following convex SIP problem, which satisfies the Slater CQ and the feasible set is bounded.

$$\begin{aligned} \min f(x) &= (x_1 - 2)^2 + (x_2 - 0.2)^2, \\ \text{s.t. } c(x, y) &= (5\sin(\pi\sqrt{y})/(1 + y^2))x_1^2 - x_2 \leq 0, \\ x_1 &\in [0, 1], \quad x_2 \in [0, 1] \text{ and } Y = [0, 1]. \end{aligned}$$

The problem is stemmed from Tichatschke and Nebeling [54], and was also tested by Kortanek and No [26]. Obviously, it satisfies the Slater CQ, and its objective function is strictly convex

*Example 10* This problem is convex SIP problem, and the objective function is strictly convex.

$$\begin{aligned} \min \quad & f(x) = x_1^2 + x_2^2, \\ \text{s.t.} \quad & c(x, y) = \cos(y)x_1 + \sin(y)x_2 - (1 + \cos(y) + \sin(y)) \leq 0, \\ & Y = [\pi, \frac{3}{2}\pi], \end{aligned}$$

which is discussed by Goberna and López [10], and Zhang and Wu [60]. Although the feasible set is not bounded, the objective function is level bounded on the feasible set.

*Example 11* Consider the following convex SIP problem, and has been tested by Kortanek and No [26].

$$\begin{aligned} \min \quad & f(x) = x_1^2 + x_2^2 + x_3^3, \\ \text{s.t.} \quad & c(x, y) = x_1 + x_2 \exp(x_3 y) + \exp(2y) - 2 \sin(4y) \leq 0, \\ & x_1 \in [-2, 2], x_2 \in [-2, 2], x_3 \in [-2, 2], Y = [0, 1], \end{aligned}$$

where the objective function is strictly convex, and the problem satisfies the Slater CQ.

*Example 12* The problem is a convex SIP problem with multidimensional index set, and has been tested by Kortanek and No [26].

$$\begin{aligned} \min \quad & f(x) = x_1^2 + x_2^2, \\ \text{s.t.} \quad & c(x, y) = (x_1^2 + x_2^2 - 4)y_1 + ((x_1 - 2)^2 + (x_2 - 2)^2 - 4)y_2 \leq 0, \\ & x_1 \in [0, 2], x_2 \in [0, 2], Y = [0, 1] \times [0, 1]. \end{aligned}$$

Observe that the problem satisfies the Slater CQ and has strictly convex objective function, the feasible set is bounded.

*Example 13* Consider the following quadratically constrained convex quadratic SIP problem, which stems from linear-phase FIR digital filter design with weighted peak-constrained least-square error. The convex SIP problem has been tested by Zhang and Wu [60].

$$\begin{aligned} \min \quad & f(x) = x^T H x - 2c^T x, \\ \text{s.t.} \quad & c_1(x, y) = |\varphi(y)^T x - 1| - 0.05 \leq 0, \text{ for } y \in Y_1 = [0, 0.05], \\ & c_2(x, y) = |\varphi(y)^T x| - 0.01 \leq 0, \text{ for } y \in Y_2 = [0.1, 0.5], \end{aligned}$$

where

$$\begin{aligned} \varphi(y) &= (2 \cos(2\pi y(n - 1)), 2 \cos(2\pi y(n - 2)), \dots, 2 \cos(2\pi y), 1)^T, \quad n = 18. \\ H &= \int_0^{0.05} \varphi(y)\varphi(y)^T dy + 1000 \int_{0.1}^{0.5} \varphi(y)\varphi(y)^T dy, \quad c = \int_0^{0.05} \varphi(y)dy \end{aligned}$$

Algorithm 1 will obtain the solution of this problem after 21 iterations, where

$$\begin{aligned} x^* &= (0.0053, 0.0033, 0.00067, -0.0034, -0.0083, -0.01343, \\ &\quad -0.0211, -0.0223, -0.0219, -0.0149, -0.0007, 0.0199, \\ &\quad 0.0447, 0.0737, 0.1002, 0.1232, 0.1382, 0.1436)^T. \end{aligned}$$

**Table 3** Comparison of results for Examples 9–13

Example	Algorithm	$x^*$	$f^*$	CPU (s)	Iter
9	Algorithm 1	(0.2928, 0.2929)	3.2212	0.26	8
	CCPA	(0.2929, 0.2929)	3.2212	1.69	141
	<i>fseminf</i>	(0.2929, 0.2929)	3.2212	2.09	6
10	Algorithm 1	(0.2052, 0.2000)	0.0821	0.15	4
	CCPA	(0.2052, 0.2000)	0.0821	7.32	32
	<i>fseminf</i>	(0.3012, 0.3101)	0.1868	1.98	53
11	Algorithm 1	(−0.2133, −1.3610, −1.8531)	5.3318	0.73	6
	CCPA	(−0.2133, −1.3615, −1.8535)	5.3346	2.71	27
	<i>fseminf</i>	(−0.2133, −1.3615, −1.8535)	5.3346	3.17	46
12	Algorithm 1	(0.5858, 0.5858)	0.6863	0.10	4
	CCPA	(0.5858, 0.5858)	0.6863	0.30	9
	<i>fseminf</i>	(0.5858, 0.5858)	0.6863	17.01	107
13	Algorithm 1		−0.1618	67.02	21
	CCPA	–	–	–	–
	<i>fseminf</i>	–	–	–	–

Comparative computational results are summarized in Table 3.

It is worth noting that our algorithm is much more effective than CCPA and SIP solver *fseminf*. Specifically, the SIP solver *fseminf* cannot obtain the optimal solution when it is used to solve Example 10. Besides, perhaps because of the dimension of Example 13 is too large, both of the CCPA and *fseminf* need too much time for solving Example 13, thus they are obliged to stop running. The solution of Example 13 has been reported in itself.

*Example 14* Consider the following unconstrained SIP [32]:

$$\min_{x \in [-1, 1]^n} \max_{t \in [0, 1]} \sum_{i=1}^n (ix_i - i/n - \sin(2\pi t + i))^2,$$

which is a classic SIPs.

We compared the performance of Algorithm 1 on Example 14 with the new exchange method (EXM) in [60] and the SIP solver *fseminf* in Matlab toolbox. By setting  $n = 2, 5, 10, 15, 20, 25, 30$ , the numerical results are summarized in Table 4.

Table 4 shows, the *fseminf* is acceptable for lower dimension problems, but it takes too much time for  $10 \leq n \leq 20$ , Besides, we have to give up running *fseminf* for  $n \geq 25$ , since it spends too much time to stop, or did not converge. The performance of the EXM is excellent for Example 14, but it needs much more time than Algorithm 1 for most problems.

**Table 4** Comparison of Algorithm 1, the new EXM and *fseminf* for Example 14

<i>n</i>	Algorithm 1						EXM		fseminf	
	Stop <sub>η</sub>	Stop <sub>ε</sub>	<i>k</i>	<i>N<sub>i</sub></i>	Time <sub>max</sub>	Time	<i>k</i>	Time	<i>k</i>	Time
2	1.4760e-4	6.0483e-4	9	11	0.68	1.01	11	1.13	8	1.07
5	3.4742e-4	5.2431e-4	13	17	2.73	4.24	17	6.47	14	8.47
10	2.4330e-4	4.4192e-4	19	33	3.66	33.49	39	28.26	97	583.32
15	7.8522e-4	7.1243e-4	44	48	4.27	36.39	63	46.17	287	895.45
20	6.2390e-4	7.4094e-4	66	38	4.40	45.46	92	59.52	393	1457.58
25	3.3173e-4	6.0724e-4	95	60	6.09	69.42	132	88.26	–	–
30	9.1076e-5	4.2451e-4	123	72	10.93	94.43	174	113.51	–	–

**5.3 Computational results for moment robust optimization**

To test the efficiency of Algorithm 1 for solving the moment robust optimization, we compute the following robust stochastic constraints optimization problem as [32].

*Example 15*

$$\begin{aligned} \min f(x) &:= (x_1 - 2)^2 + (x_2 - 0.2)^2, \\ \text{s.t. } E_W \left[ \frac{5 \sin \pi \sqrt{\xi}}{1 + \xi^2} x_1^2 - x_2 \right] &\leq 0, \quad W \in \mathcal{D}, \\ x_1 &\in [-1, 1], \quad x_2 \in [0, 0.2], \end{aligned}$$

where  $\mathcal{D} := \left\{ W : E_W[\xi^i] = \frac{1}{i+1}, i = 0, 1, \dots, m \right\}$ .

In our numerical experiment, we set the initial point  $x^0 = (1.3, 0.7)$ , the increase parameter  $\sigma = 2.5$ , the initial parameters  $\mu_0 = 4.5, \delta_0 = 10^{-6}, \iota = 0.08$ , the optimality tolerance  $\varepsilon = 10^{-3}$ .

By setting different moment constraints  $m = 1, 2, 3, 4, 5, 6, \infty$ , Example 15 gives various classical robust optimization model of the problem. It should be noted that the solutions for solving Example 15 for increasing values of  $t$  correspond to fewer and fewer conservative solutions. The objective value increases along with the increasing values of  $t$ . Fortunately, the inner subproblem (2.9) can be solved effectively, which implies the inexact oracle idea is reasonable. All in all, the numerical results show that Algorithm 1 can solve the robust stochastic constraints optimization problem effectively.

The numerical results are listed in the following Table 5.

**Table 5** The numerical results gave by Algorithm 1 for Example 15 with different moment constraints

$t$	$\text{Stop}_\eta$	$\text{Stop}_\varepsilon$	$x_1$	$x_2$	$f^*$	$k$	$Ni$	$\text{Time}_{max}$	Time
1	2.1475e-4	7.7081e-4	0.2407	0.2	3.0951	13	20	0.23	1.04
2	6.1034e-4	9.8001e-5	0.2493	0.2	3.0760	17	26	0.37	1.21
3	3.1179e-4	5.0781e-4	0.2672	0.2	3.0026	19	26	0.34	1.27
4	8.1930e-4	7.7314e-4	0.2679	0.2	2.9964	24	29	1.21	2.73
5	7.9013e-5	5.3107e-5	0.2709	0.2	2.9898	22	34	0.85	2.39
6	6.1347e-4	9.3174e-5	0.2713	0.2	2.9884	22	37	1.05	2.77
$\infty$	7.1743e-4	6.1814e-5	0.2721	0.2	2.9856	31	40	1.17	3.07

**Acknowledgments** Partially supported by the Natural Science Foundation of China, Grant 11171049 and 31271077.

## References

- Ackooij, W.V., Sagastizbal, C.: Constrained bundle methods for upper inexact oracles with application to joint chance constrained energy problems. *SIAM J. Optim.* **24**, 733–765 (2014)
- Bhattacharjee, B., Lemonidis, P., Green Jr, W.H., Barton, P.I.: Global solution of semi-infinite programs. *Math. Program.* **103**, 283–307 (2005)
- Cánovas, M.J., Hantoute, A., Láopez, M.A., Parra, J.: Stability of indices in the KKT conditions and metric regularity in convex semi-infinite optimization. *J. Optim. Theory Appl.* **139**, 485–500 (2008)
- Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Oper. Res.* **58**, 595–612 (2010)
- Emiel, G., Sagastizábal, C.: Incremental-like bundle methods with application to energy planning. *Comput. Optim. Appl.* **46**, 305–332 (2010)
- Fang, S.C., Lin, C.J., Wu, S.-Y.: Solving quadratic semi-infinite programming problems by using relaxed cutting-plane scheme. *J. Comput. Appl. Math.* **129**, 89–104 (2001)
- Fuduli, A., Gaudioso, M., Giallombardo, G.: Minimizing nonconvex nonsmooth functions via cutting planes and proximity control. *SIAM J. Optim.* **14**, 743–756 (2005)
- Fuduli, A., Gaudioso, M., Giallombardo, G., Miglionico, G.: A partially inexact bundle method for convex semi-infinite minmax problems. *Commun Nonlinear Sci Numer Simulat* **21**, 172–180 (2014)
- Gaudioso, M., Giallombardo, G., Miglionico, G.: An incremental method for solving convex finite Min-Max problems. *Math. Oper. Res.* **31**, 173–187 (2006)
- Goberna, M.A., Láopez, M.A.: *Linear Semi-infinite Optimization*. Wiley, New York (1998)
- Hare, W., Sagastizábal, C., Solodov, M.: A proximal bundle method for nonsmooth nonconvex functions with inexact information. *Comput. Optim. Appl.* doi:10.1007/s10589-015-9762-4
- Helmberg, C., Rendl, F.: A spectral bundle method for semi-definite programming. *SIAM J. Optim.* **10**, 673–696 (2000)
- Hettich, R.: An implementation of a discretization method for semi-infinite programming. *Math. Program.* **34**, 354–361 (1986)
- Hettich, R., Kortanek, K.O.: *Semi-infinite programming: theory, methods and applications*. *SIAM Rev.* **35**, 380–429 (1993)
- Hiriart-Urruty, J.B., Lemarechal, C.: *Convex analysis and minimization algorithms*. Springer, Berlin, Heidelberg (1991)
- Hu, H.: A one-phase algorithm for semi-infinite linear programming. *Math. Program.* **46**, 85–103 (1990)
- Jin, P., Ling, C., Shen, H.F.: A smoothing Levenberg-Marquardt algorithm for semi-infinite programming. *Comput. Optim. Appl.* **60**, 675–695 (2015)
- Jongen, HTh, Rückmann, J.-J., Stein, O.: Generalized semi-infinite optimization: a first order optimality condition and examples. *Math. Program.* **83**, 145–158 (1998)

19. Kanzi, N.: Necessary optimality conditions for nonsmooth semi-infinite programming problems. *J. Glob. Optim.* **49**, 713–725 (2011)
20. Kanzi, N., Nobakhtian, S.: Optimality conditions for nonsmooth semi-infinite programming. *Optimization* **53**, 717–727 (2008)
21. Kanzi, N., Nobakhtian, S.: Nonsmooth semi-infinite programming problems with mixed constraints. *J. Math. Anal. Appl.* **351**, 170–181 (2008)
22. Kibardin, V.M.: Decomposition into functions in the minimization problem. *Avtomat. i Telemekh.*, no. 9 (1979), 66–79 (in Russian). *Automat. Remote Control* **40**, 1311–1323 (1980). (in English)
23. Kiwiel, K.C.: *Methods of Descent for Nondifferentiable Optimization*. Lecture Notes in Mathematics. Springer, Berlin (1985)
24. Kiwiel, K.C.: Convergence of approximate and incremental subgradient methods for convex optimization. *SIAM J. Optim.* **14**, 807–840 (2003)
25. Kiwiel, K.C.: A proximal bundle method with approximate subgradient linearizations. *SIAM J. Optim.* **16**, 1007–1023 (2006)
26. Kortanek, K.O., No, H.: A central cutting plane algorithm for convex semi-infinite programming problems. *SIAM J. Optim.* **3**, 901–918 (1993)
27. Li, D.H., Qi, L., Tam, J., Wu, S.-Y.: A smoothing Newton method for semi-infinite programming. *J. Glob. Optim.* **30**, 169–194 (2004)
28. Li, C., Ng, K.F., Pong, T.K.: Constraint qualifications for convex inequality systems with applications in constrained optimization. *SIAM J. Optim.* **19**, 163–187 (2008)
29. Ling, C., Ni, Q., Qi, L.Q., Wu, S.-Y.: A new smoothing Newton-type algorithm for semi-infinite programming. *J. Glob. Optim.* **47**, 133–159 (2010)
30. López, M.A., Still, G.: Semi-infinite programming. *Eur. J. Oper. Res.* **180**, 491–518 (2007)
31. Lv, J., Pang, L.P., Wang, J.H.: Special backtracking proximal bundle method for nonconvex maximum eigenvalue optimization. *Appl. Math. Comput.* **265**, 635–651 (2015)
32. Mehrotra, S., Papp, D.: A cutting surface algorithm for semi-infinite convex programming with an application to moment robust optimization. *SIAM J. Optim.* **24**, 1670–1697 (2014)
33. Mordukhovich, Boris S., Nghia, T.T.A.: Subdifferentials of nonconvex supremum functions and their applications to semi-infinite and infinite programs with Lipschitzian data. *SIAM J. Optim.* **23**, 406–431 (2013)
34. Nedić, A., Bertsekas, D.P.: Incremental subgradient methods for nondifferentiable optimization. *SIAM J. Optim.* **12**, 109–138 (2001)
35. Nguyen, T.T.V., Strodio, J.J., Nguyen, V.H.: A bundle method for solving equilibrium problems. *Math. Program.* **116**, 529–552 (2009)
36. Ni, Q., Ling, C., Qi, L., Teo, K.L.: A truncated projected Newton-type algorithm for large-scale semi-infinite programming. *SIAM J. Optim.* **16**, 1137–1154 (2006)
37. Oliveira, W., Sagastizábal, C., Scheimberg, S.: Inexact bundle methods for two-stage stochastic programming. *SIAM J. Optim.* **21**, 517–544 (2011)
38. Pang, L. P., Wang, M. Z., Xia, Z. Q.: First order necessary optimality conditions for a class of nonsmooth generalized semiinfinite optimization problems. *Comput. Math. Appl.* **56**, 1457–1464 (2008)
39. Polak, E.: On the use of consistent approximations in the solution of semi-infinite optimization and optimal control problems. *Math. Program.* **62**, 385–414 (1993)
40. Puente, R., VeraDe Serio, V.N.: Locally farkas minkowski linear inequality systems. *Top* **7**, 103–121 (1999)
41. Qi, L., Wu, S.Y., Zhou, G.: Semismooth Newton methods for solving semi-infinite programming problems. *J. Global Optim.* **27**, 215–232 (2003)
42. Qi, L., Ling, C., Tong, X.J., Zhou, G.: A smoothing projected Newton-type algorithm for semi-infinite programming. *Comput. Optim. Appl.* **42**, 1–30 (2009)
43. Rockafellar, R.T.: *Convex Analysis*. Princeton University Press, Princeton (1970)
44. Rockafellar, R.T., Wets, J.J.-B.: *Variational Analysis*. Springer, Berlin (1998)
45. Sagastizábal, C.: Divide to conquer: decomposition methods for energy optimization. *Math. Program.* **134**, 187–222 (2012)
46. Sagastizábal, C., Solodov, M.V.: An infeasible bundle method for nonsmooth convex constrained optimization without a penalty function or a filter. *SIAM J. Optim.* **16**, 146–169 (2005)
47. Salmon, G., Strodio, J.-J., Nguyen, V.H.: A bundle method for solving variational inequalities. *SIAM J. Optim.* **14**, 869–893 (2004)



48. Shen, J., Pang, L. P.: A proximal analytic center cutting plane algorithm for solving variational inequality problems. *J. Appl. Math.* (2012). doi:[10.1155/2012/503242](https://doi.org/10.1155/2012/503242)
49. Solodov, M.V.: A bundle method for a class of bilevel nonsmooth convex minimization problems. *SIAM J. Optim.* **18**, 242–259 (2007)
50. Stein, O.: On constraint qualifications in nonsmooth optimization. *J. Optim. Theory Appl.* **121**, 647–671 (2004)
51. Still, G.: Discretization in semi-infinite programming: the rate of convergence. *Math. Program.* **91**, 53–69 (2001)
52. Tanaka, Y., Fukushima, M., Ibaraki, T.: A globally convergent SQP method for semi-infinite nonlinear optimization. *J. Comput. Appl. Math.* **23**, 141–153 (1988)
53. Teo, K.L., Yang, X.Q., Jennings, L.S.: Computational discretization algorithms for functional inequality constrained optimization. *Ann. Oper. Res.* **28**, 215–234 (2000)
54. Tichatschke, R., Nebeling, V.: A cutting plane method for quadratic semi-infinite programming Problems. *Optimization* **19**, 803–817 (1988)
55. Wu, S.-Y., Fang, S.C., Lin, C.J.: Relaxed cutting plane method for solving linear semi-infinite programming problems. *J. Optim. Theory Appl.* **99**, 759–779 (1998)
56. Wu, S.-Y., Li, D.H., Qi, L., Zhou, G.: An iterative method for solving KKT system of the semi-infinite programming. *Optim. Methods Softw.* **20**, 629–643 (2005)
57. Wu, S.-Y., Fang, S.C.: Solving convex programs with infinitely many linear constraints by a relaxed cutting plane method. *Comput. Math. Appl.* **38**, 23–33 (1999)
58. Xu, M.W., Wu, S.-Y., Ye, J.J.: Solving semi-infinite programs by smoothing projected gradient method. *Comput. Optim. Appl.* **59**, 591–616 (2014)
59. Xu, Q.J., Jian, J.B.: A nonlinear norm-relaxed method for finely discretized semi-infinite optimization problems. *Nonlinear Dyn.* **73**, 85–92 (2013)
60. Zhang, L.P., Wu, S.-Y., López, M.A.: A new exchange method for convex semi-infinite programming. *SIAM J. Optim.* **20**, 2959–2977 (2010)
61. Zheng, X.Y., Yang, X.Q.: Lagrange multipliers in nonsmooth semi-infinite optimization problems. *Math. Oper. Res.* **32**, 168–181 (2007)