

# Exact Penalization, Level Function Method and Modified Cutting-Plane Method for Stochastic Programs with Second Order Stochastic Dominance Constraints

Hailin Sun<sup>1</sup>

Department of Mathematics, Harbin Institute of Technology, Harbin, 150001, China  
(mathhlsun@gmail.com)

Huifu Xu

School of Mathematics, University of Southampton, Southampton, SO17 1BJ, UK  
(h.xu@soton.ac.uk)

Rudabeh Meskarian

School of Mathematics, University of Southampton, Southampton, SO17 1BJ, UK  
(rm3g08@soton.ac.uk)

Yong Wang

Department of Mathematics, Harbin Institute of Technology, Harbin, 150001, China  
(mathwy@hit.edu.cn)

February 20, 2012

## Abstract

Level function methods and cutting plane methods have been recently proposed to solve stochastic programs with stochastic second order dominance (SSD) constraints. A level function method requires an exact penalization setup because it can only be applied to the objective function, not the constraints. Slater constraint qualification (SCQ) is often needed for deriving exact penalization. In this paper, we show that although the original problem usually does satisfy the SCQ but, through some reformulation of the constraints, the constraint qualification can be satisfied under some moderate conditions. Exact penalization schemes based on  $L_1$ -norm and  $L_\infty$ -norm are subsequently derived through Robinson's error bound on convex system and Clarke's exact penalty function theorem. Moreover, we propose a modified cutting plane method which constructs a cutting plane through the maximum of the reformulated constraint functions. In comparison with the existing cutting methods, it is numerically more efficient because only a single cutting plane is constructed and added at each iteration. We have carried out a number of numerical experiments and the results show that our methods display better performances particularly in the case when the underlying functions are nonlinear w.r.t. decision variables.

**Key words.** Slater constraint qualification, exact penalization, modified cutting-plane method, level function method

## 1 Introduction

Stochastic dominance is a fundamental concept in decision theory and economics. A random outcome  $a(\omega)$  is said to dominate another random outcome  $b(\omega)$  in the second order, written as

---

<sup>1</sup>The work of this author is carried out while he is visiting the second author in the School of Mathematics, University of Southampton sponsored by China Scholarship Council.

$a(\omega) \succeq_2 b(\omega)$ , if  $\mathbb{E}[v(a(\omega))] \geq \mathbb{E}[v(b(\omega))]$  for every concave nondecreasing function  $v(\cdot)$ , for which the expected values are finite, see monograph [14] for the recent discussions of the concept.

In their pioneering work [2, 4], Dentcheva and Ruszczyński introduced a stochastic programming model with second order stochastic dominance constraints :

$$\begin{aligned} \min_x \quad & \mathbb{E}[F(x, \xi(\omega))] \\ \text{s.t.} \quad & G(x, \xi(\omega)) \succeq_2 Y(\xi(\omega)), \\ & x \in \mathcal{X}, \end{aligned} \tag{1.1}$$

where  $\mathcal{X}$  is a closed convex subset of  $\mathbb{R}^n$ ,  $\xi : \Omega \rightarrow \Xi$  is a vector of random variables defined on probability space  $(\Omega, \mathcal{F}, P)$  with support set  $\Xi \subset \mathbb{R}^q$ ,  $F : \mathbb{R}^n \times \Xi \rightarrow \mathbb{R}$  is convex continuous function w.r.t.  $x$ ,  $G : \mathbb{R}^n \times \Xi \rightarrow \mathbb{R}$  is concave continuous function w.r.t.  $x$  and  $\mathbb{E}[\cdot]$  denotes the expected value with respect to the distribution of  $\xi(\omega)$ . Here we make a blanket assumption that the expected value of the random function is well defined.

A simple economic interpretation of the model can be given as follows. Let  $G(x, \xi(\omega))$  be a profit function which depends on decision vector  $x$  and a random variable  $\xi(\omega)$ , let  $F = -G$  and  $Y(\xi(\omega))$  be a benchmark profit. Then (1.1) can be viewed as an expected profit maximization problem subject to the constraint that the profit dominates the benchmark profit in second order.

Let  $F_1(X; \eta)$  denote cumulative distribution function of random variable  $X$ , that is,

$$F_1(X; \eta) := P(X \leq \eta),$$

and

$$F_2(G(x, \xi(\omega)); \eta) := \int_{-\infty}^{\eta} F_1(G(x, \xi(\omega)); t) dt.$$

$G(x, \xi(\omega))$  is said to dominate  $Y(\xi(\omega))$  in first order, denoted by  $G(x, \xi(\omega)) \succeq_1 Y(\xi(\omega))$ , if for all  $\eta \in \mathbb{R}$ ,

$$F_1(G(x, \xi(\omega)); \eta) \leq F_1(Y(\xi(\omega)); \eta).$$

$G(x, \xi(\omega))$  is said to dominate  $Y(\xi(\omega))$  in second order, denoted by  $G(x, \xi(\omega)) \succeq_2 Y(\xi(\omega))$ , if

$$F_2(G(x, \xi(\omega)); \eta) \leq F_2(Y(\xi(\omega)); \eta), \forall \eta \in \mathbb{R}. \tag{1.2}$$

It is easy to observe that first order stochastic dominance implies second order stochastic dominance. It is well known that second order dominance constraint in (1.1) can be reformulated as

$$\mathbb{E}[(\eta - G(x, \xi(\omega)))_+] \leq \mathbb{E}[(\eta - Y(\xi(\omega)))_+], \forall \eta \in \mathbb{R}, \tag{1.3}$$

where  $(x)_+ = \max(0, x)$ ; see [5]. Ogryczak and Ruszczyński [15] investigated the relationship between stochastic second order dominance and mean-risk models. In a more recent development [3], the second order dominance is shown to be equivalent to the constraint of conditional value at risk through Fenchel conjugate duality. Using the reformulation of the second order dominance constraints, Dentcheva and Ruszczyński [5] reformulated (1.1) as:

$$\begin{aligned} \min_x \quad & \mathbb{E}[F(x, \xi(\omega))] \\ \text{s.t.} \quad & \mathbb{E}[(\eta - G(x, \xi(\omega)))_+] \leq \mathbb{E}[(\eta - Y(\xi(\omega)))_+], \forall \eta \in \mathbb{R}, \\ & x \in \mathcal{X}. \end{aligned} \tag{1.4}$$

To ease notation, we will use  $\xi$  to denote the random vector  $\xi(\omega)$  and a deterministic vector, depending on the context. It is well known that (1.4) does not satisfy the well known Slater constraint qualification, a condition that is often needed for deriving first order optimality conditions of the problem and developing a numerically stable method for solving the problem. Subsequently, a so-called *relaxed* form of (1.4) is proposed:

$$\begin{aligned} \min_x \quad & \mathbb{E}[F(x, \xi)] \\ \text{s.t.} \quad & \mathbb{E}[(\eta - G(x, \xi))_+] \leq \mathbb{E}[(\eta - Y(\xi))_+], \quad \forall \eta \in [a, b], \\ & x \in \mathcal{X}, \end{aligned} \tag{1.5}$$

where  $[a, b]$  is a closed interval in  $\mathbb{R}$ . Over the past few years, Dentcheva and Ruszczyński have developed comprehensive theory of optimality and duality for (1.5), see [2, 3, 4, 5].

Unfortunately, problem (1.5) is difficult to solve since it is a stochastic semi-infinite nonsmooth programming problem. In the case when  $G(x, \xi)$  and  $F(x, \xi)$  are linear w.r.t.  $x$ , Dentcheva and Ruszczyński [3] reformulated problem (1.5) as a linear programming (LP) problem by introducing new variables which represent positive parts in each constraint of problem (1.5). The reformulation effectively tackles the nonsmoothness in the second order constraints and the approach can easily be used to the case when  $G$  and  $F$  are nonlinear. This reformulation, however, introduces many new variables particularly when the random variable  $\xi$  has many distributional scenarios. Apparently this does not have significant impact on numerical implementation as the existing solvers for LP are very powerful (can deal with millions of variables). However, the impact will be much more significant when  $F$  and  $G$  are nonlinear and this is indeed one of the key issues this paper is to address.

Rudolf and Ruszczyński [19] and Fábíán et al [6] proposed cutting-plane methods for solving a stochastic program with second order stochastic dominance constraints. A crucial element of the method in [6] is based on an observation that when  $F$  and  $G$  are linear w.r.t.  $x$  and probability space  $\Omega$  is finite, the constraint function in the second order dominance constraint is the convex envelope of a finitely many linear functions. Subsequently, an iterative scheme which exploits the fundamental idea of classical cutting-plane method is proposed where at each iterate “cutting-plane” constraints are constructed and added. This also effectively tackles the nonsmoothness issue caused by the plus function. While the method displays strong numerical performance, it relies heavily on discreteness of the probability space as well as the linearity of  $F$  and  $G$ .

In this paper we consider problem (1.4) with a focus on the case when  $\xi$  has a discrete distribution, that is,

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \sum_{i=1}^N p_i (\eta - G(x, \xi^i))_+ \leq \sum_{i=1}^N p_i (\eta - Y(\xi^i))_+, \quad \eta \in \mathbb{R}, \\ & x \in \mathcal{X}. \end{aligned} \tag{1.6}$$

Here the random variable  $\xi$  has a finite distribution, that is,  $P(\xi = \xi^i) = p_i$ , for  $i = 1, \dots, N$ . When  $p_i = \frac{1}{N}$ , problem (1.6) can be viewed as a sample average approximation of problem (1.4). We investigate the Slater constraint qualification of the problem and its reformulation, exact penalization schemes and numerical methods. Specifically, we make the following contributions.

- We develop penalization schemes for problem (1.6). We do so by exploiting Clarke’s exact penalty function theorem [1, Proposition 2.4.3] and Robinson’s error bound [16]. The latter requires SCQ. Unfortunately, problem (1.6) or its reformulation (2.10) does not satisfy the constraint qualification (see discussions by Dentcheva and Ruszczyński at pages 558-559 in [2]). Here we propose an *alternative way* to reformulate the constraints of problem (2.10). We then demonstrate that the newly reformulated problem (see (2.12)) satisfies the SCQ under some moderate conditions (see Theorem 2.1). Two exact penalization schemes based on  $L_1$ -norm and  $L_\infty$ -norm are subsequently derived and are shown that they are exact penalization of problem (2.10) although the latter does not satisfy SCQ (see Theorems 3.1 and 3.2). Note that Liu and Xu [12] proposed an exact penalization scheme with  $L_\infty$ -norm for the relaxed problem (1.5). A crucial condition is the SCQ of problem (1.5) which relies on the *relaxation* because the original problem (1.4) may not satisfy the SCQ. Our penalization schemes in this paper differ from Liu and Xu’s in that they are proposed for the original problem rather than for the relaxed problem, which means that they are established without the SCQ of the original problem. This makes the penalization scheme more appealing given the fact that the original problem usually does not satisfy the SCQ particularly in the case when  $\xi$  satisfies discontinuous distribution. Based on the exact penalization formulations, we apply a well known level function method in nonsmooth optimization [11, 20] to the penalized problems. An obvious advantage of this approach is that we can effectively deal with excessive number of constraints, nonsmoothness in the constraints and nonlinearity of the underlying functions.
- We also propose a modified cutting-plane method to solve the problem. The cutting-plane method differs from those in the literature [19] in that it applies to the maximum of the constraint functions rather than each constraint function. This saves considerable computation time because at each iteration, our cutting-plane method requires to add a couple of linear constraints whereas the cutting-plane method in [19] requires to add  $N$  constraints ( $N$  is the cardinality of the support set  $\Xi$ ). The approach also differs from that in [7, 8] because our modified cutting-plane method uses the cutting-plane representation proposed in [9]. The idea of applying the cutting-plane method to the maximum of the constraint functions is similar to the idea in algorithm proposed by Fábíán, Mitra and Roman, see the algorithm at page 48 in [6]. Note that Fábíán, Mitra and Roman’s algorithm is applied to linear models while Algorithm 4.1 is applicable to nonlinear case. Therefore we may regard our algorithm as an extension of theirs.
- We have carried out extensive numerical tests on our proposed methods in comparison with the cutting plane method in [6]. The numerical results show that our proposed methods are more efficient. Specifically, we have discovered that level function method based on exact penalization scheme with  $L_\infty$ -norm is most efficient in terms of computation time; the modified cutting-plane method (Algorithm 4.1) performs also efficiently.

The rest of the paper is organized as follows. In section 2, we discuss the SCQ of problem (1.6) and its reformulation. In section 3, we propose two exact penalization schemes for problem (2.10) and apply a level function method to solve them. In section 4, a modified cutting-plane method has been proposed for solving the problem and finally in section 5, we report some numerical test results.

Throughout this paper, we use the following notation.  $x^T y$  denotes the scalar product of two vectors  $x$  and  $y$ ,  $\|\cdot\|$ ,  $\|\cdot\|_1$  and  $\|\cdot\|_\infty$  denote the Euclidean norm,  $L_1$ -norm and  $L_\infty$ -norm of a vector and a compact set of vectors respectively. We also use  $\|\cdot\|$  to denote the infinity norm of a

continuous function space and its induced norm of a linear operator.  $d(x, \mathcal{D}) := \inf_{x' \in \mathcal{D}} \|x - x'\|$ ,  $d_1(x, \mathcal{D}) := \inf_{x' \in \mathcal{D}} \|x - x'\|_1$  and  $d_\infty(x, \mathcal{D}) := \inf_{x' \in \mathcal{D}} \|x - x'\|_\infty$  denote the distance from point  $x$  to set  $\mathcal{D}$  in Euclidean norm,  $L_1$ -norm and  $L_\infty$ -norm respectively. For a real valued-function  $h(x)$ , we use  $\nabla h(x)$  to denote the gradient of  $h$  at  $x$ .

## 2 Slater constraint qualification

In the literature of stochastic programs with second order dominance constraints, SCQ has been used as a key condition for deriving optimality conditions and exact penalization, see for instances [2, 12].

Recall that problem (1.6) is said to satisfies the SCQ if there exists  $x_0 \in \mathcal{X}$  such that

$$\sum_{i=1}^N p_i (\eta - G(x_0, \xi^i))_+ - \sum_{i=1}^N p_i (\eta - Y(\xi^i))_+ < 0, \quad \eta \in \mathbb{R}. \quad (2.7)$$

Unfortunately, this kind of constraint qualification is not satisfied. To see this, let

$$Y(\Xi) := \{Y(\xi^i) : i = 1, \dots, N\}$$

and

$$\underline{y} := \min\{Y(\xi^1), \dots, Y(\xi^N)\}. \quad (2.8)$$

For any  $\eta \leq \underline{y}$ , it is easy to verify that  $\mathbb{E}[(\eta - Y(\xi))_+] = 0$ . For those  $\eta$ , the feasible constraint of problem (1.5) reduces to

$$\mathbb{E}[(\eta - G(x, \xi))_+] - \mathbb{E}[(\eta - Y(\xi))_+] = 0$$

because the term at the left hand side of the equation is non-negative. This means that there does not exist a feasible point  $x_0 \in \mathcal{X}$  such that (2.7) holds.

Dentcheva and Ruszczyński [2] observed this issue and tackled it by considering a relaxed problem (1.5) which effectively restricts  $\eta$  to take value from a specified  $[a, b]$ . In other words, the feasible region of the original problem (1.6) is enlarged. In this context, their relaxation scheme can be written as follows:

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \sum_{i=1}^N p_i (\eta - G(x, \xi^i))_+ \leq \sum_{i=1}^N p_i (\eta - Y(\xi^i))_+, \quad \eta \in [a, b], \\ & x \in \mathcal{X}. \end{aligned} \quad (2.9)$$

Under some circumstance, it is possible to choose a proper value  $a$  such that problem (2.9) satisfies the SCQ. For instance, if there exists a point  $x_0 \in \mathcal{X}$  such that

$$G(x_0, \xi) \succeq_1 Y(\xi)$$

and for every  $\xi \in \Xi$ ,  $G(x_0, \xi) > \underline{y}$ , then  $x_0$  is a feasible point of problem (2.9) and

$$\int_{-\infty}^{\eta} F_1(G(x_0, \xi); t) dt < \int_{-\infty}^{\eta} F_1(Y(\xi); t) dt$$

for all  $\eta > \underline{y}$ . In such a case, it is easy to verify that the SCQ holds for any  $a > \underline{y}$ . Note that problem (2.9) is a relaxed problem of (1.6) which depends on  $[a, b]$  and when  $[a, b]$  contains  $Y(\Xi)$ , the SCQ fails.

In this section, we propose an *alternative way* to address the issue of SCQ of problem (1.6) without relaxation. To this end, let us use [2, Proposition 3.2] to reformulate problem (1.6) as follows:

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ \leq \gamma_j, \quad j = 1, \dots, N, \\ & x \in \mathcal{X}, \end{aligned} \tag{2.10}$$

where  $Y_j := Y(\xi^j)$ ,  $\gamma_j := \sum_{i=1}^N p_i (Y_j - Y(\xi^i))_+$ . Like the original problem (1.6), the reformulated problem (2.10) does not satisfy SCQ.

We consider the power set of  $\{1, \dots, N\}$ , that is, a collection of all subsets of  $\{1, \dots, N\}$  including empty set and itself. For the simplicity of notation, let  $\mathcal{N}$  denote the power set excluding the empty set and for  $j = 1, \dots, N$ ,

$$\psi_j(x) := \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i (Y_j - G(x, \xi^i)) - \gamma_j. \tag{2.11}$$

Consider problem

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \psi_j(x) \leq 0, \text{ for } j = 1, \dots, N, \\ & x \in \mathcal{X}. \end{aligned} \tag{2.12}$$

In what follows, we will show that problem (2.12) is equivalent to problem (2.10) but, under some circumstance, the former satisfies the SCQ.

**Lemma 2.1** For  $j = 1, \dots, N$ , let

$$\varphi_j(x) := \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i (Y_j - G(x, \xi^i)).$$

Then

$$\sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ = \max\{\varphi_j(x), 0\}. \tag{2.13}$$

**Proof.** We prove the claim by going through two cases: 1.  $\varphi_j(x) \leq 0$ ; 2.  $\varphi_j(x) > 0$ .

Case 1. Since  $\varphi_j(x) \leq 0$ , then  $\max\{\varphi_j(x), 0\} = 0$  and  $Y_j - G(x, \xi^i) \leq 0$  for  $j \in \{1, \dots, N\}$ . The latter implies

$$\sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ = 0$$

and hence (2.13).

Case 2. Since  $\varphi_j(x) > 0$ , then there exists a nonempty subset  $\mathcal{J} \subseteq \{1, \dots, N\}$  such that

$$\varphi_j(x) = \sum_{i \in \mathcal{J}} p_i(Y_j - G(x, \xi^i)) > 0.$$

It suffices to show that

$$\sum_{i \in \mathcal{J}} p_i(Y_j - G(x, \xi^i)) = \sum_{i=1}^N p_i(Y_j - G(x, \xi^i))_+$$

or equivalently  $\mathcal{J}$  consists of every index  $i$  with

$$Y_j - G(x, \xi^i) > 0.$$

Indeed, if  $\mathcal{J}$  does not include such an index, then adding it to  $\mathcal{J}$  would increase the quantity  $\sum_{i \in \mathcal{J}} p_i(Y_j - G(x, \xi^i))$  and this contradicts the fact that  $\varphi_j(x)$  is the maximum. Likewise,  $\mathcal{J}$  does not consist of an index  $i$  with

$$Y_j - G(x, \xi^i) < 0,$$

because, otherwise, removing the index will also increase the quantity  $\sum_{i \in \mathcal{J}} p_i(Y_j - G(x, \xi^i))$ . This completes the proof.  $\blacksquare$

We are now ready to state the main result in this section.

**Theorem 2.1** *Let  $G(x, \xi)$  and  $Y(\xi)$  be defined as in problem (1.6) and  $\psi_j$  be defined as in (2.11). Then*

(i)  $G(x, \xi) \succeq_2 Y(\xi)$  if and only if

$$\psi_j(x) \leq 0, \quad \forall j = 1, \dots, N; \quad (2.14)$$

(ii) problems (2.10) and (2.12) are equivalent;

(iii) if there exists a feasible point  $x_0$  such that  $G(x_0, \xi) \succeq_1 Y(\xi)$  and  $G(x_0, \xi) > \underline{y}$  for all  $\xi \in \Xi$ , then the system of inequalities (2.14) satisfies the SCQ.

**Proof.** Part (i). By [2, Proposition 3.2],  $G(x, \xi) \succeq_2 Y(\xi)$  if and only if

$$\sum_{i=1}^N p_i(Y_j - G(x, \xi^i))_+ \leq \gamma_j, \quad j = 1, \dots, N, \quad (2.15)$$

or equivalently for  $j = 1, \dots, N$ ,

$$\max \left\{ \sum_{i=1}^N p_i(Y_j - G(x, \xi^i))_+ - \gamma_j, 0 \right\} = 0.$$

By (2.13)

$$\max \left\{ \sum_{i=1}^N p_i(Y_j - G(x, \xi^i))_+ - \gamma_j, 0 \right\} = \max \{ \max \{ \varphi_j(x), 0 \} - \gamma_j, 0 \}.$$

Note that for any number  $a \in \mathbb{R}$  and  $r > 0$ , it is easy to verify that

$$\max\{\max\{a, 0\} - r, 0\} = \max\{a - r, 0\}. \quad (2.16)$$

Using (2.16), we have that

$$\max\{\max\{\varphi_j(x), 0\} - \gamma_j, 0\} = \max\{\varphi_j(x) - \gamma_j, 0\} = \max\{\psi_j(x), 0\}.$$

The last equality is due to the definition of  $\psi_j$ . The discussion above demonstrates that (2.15) is equivalent to (2.14) and hence the conclusion.

Part (ii) follows straightforwardly from Part (i) in that the feasible set of the two problems coincide, i.e.,

$$\left\{ x \in \mathcal{X} : \sum_{i=1}^N p_i(Y_j - G(x, \xi^i))_+ - \gamma_j \leq 0 \right\} = \{x \in \mathcal{X} : \psi_j(x) \leq 0\}.$$

Part (iii). Let  $\gamma_{\underline{y}} := \sum_{i=1}^N p_i(\underline{y} - Y(\xi^i))_+$ , where  $\underline{y}$  is defined in (2.8). By the definition of  $\underline{y}$ , the right hand side equals to 0. Therefore  $\gamma_{\underline{y}} = 0$ . Likewise, the assumption  $G(x_0, \xi) > \underline{y}$  for  $\xi \in \Xi$  implies

$$\sum_{i \in \mathcal{J}} p_i(\underline{y} - G(x_0, \xi^i)) < 0$$

for every nonempty index set  $\mathcal{J} \subseteq \{1, \dots, N\}$ . This shows

$$\max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i(\underline{y} - G(x_0, \xi^i)) - \gamma_{\underline{y}} < 0. \quad (2.17)$$

For the convenience of notation and discussion, let us use  $Y_1, \dots, Y_N$  denote the  $N$  elements in set  $Y(\Xi)$ , where

$$Y_1 \leq Y_2 \leq \dots \leq Y_N.$$

By the definition of  $\psi_j(x)$  (see (2.14), inequality (2.17) means that

$$\psi_1(x_0) = \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i(Y_1 - G(x_0, \xi^i)) - \sum_{i=1}^N p_i(Y_1 - Y(\xi^i))_+ < 0.$$

In what follows, we show

$$\psi_j(x_0) < 0$$

for  $j = 2, \dots, N$ . By definition

$$\begin{aligned} \psi_j(x_0) &= \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i(Y_j - G(x_0, \xi^i)) - \sum_{i=1}^N p_i(Y_j - Y(\xi^i))_+ \\ &\leq \max \left\{ \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} p_i(Y_j - G(x_0, \xi^i)), 0 \right\} - \sum_{i=1}^N p_i(Y_j - Y(\xi^i))_+ \\ &= \sum_{i=1}^N p_i((Y_j - G(x_0, \xi^i))_+ - (Y_j - Y(\xi^i))_+) \\ &= \int_{-\infty}^{Y_j} (F_1(G(x_0, \xi), t) - F_1(Y(\xi), t)) dt. \end{aligned} \quad (2.18)$$

The second last equality follows from Lemma 2.1 and the last equality is due to the equivalent representation of second order dominance between (1.2) and (1.3).

Assume without loss of generality that  $Y_2 > Y_1$  (otherwise  $\psi_2(x_0) = \psi_1(x_0) < 0$ ). Let  $\bar{\eta} \in (Y_1, \min\{\min_{\xi \in \Xi} G(x_0, \xi^i), Y_2\})$ . Note that by assumption  $Y_1 < \min\{\min_{\xi \in \Xi} G(x_0, \xi^i), Y_2\}$ ,  $\bar{\eta}$  exists. Then

$$\begin{aligned} \int_{-\infty}^{Y_j} F_1(G(x, \xi), t) - F_1(Y(\xi), t) dt &= \int_{-\infty}^{\bar{\eta}} F_1(G(x, \xi), t) - F_1(Y(\xi), t) dt \\ &+ \int_{\bar{\eta}}^{Y_j} F_1(G(x, \xi), t) - F_1(Y(\xi), t) dt. \end{aligned}$$

Note that

$$\int_{-\infty}^{\bar{\eta}} F_1(G(x, \xi), t) - F_1(Y(\xi), t) dt = 0 - p_1(\bar{\eta} - Y_1) < 0$$

where  $p_1$  is the probability that  $Y(\xi)$  takes value  $Y_1$ . On the other hand,  $G(x_0, \xi) \succeq_1 Y(\xi)$  implies

$$\int_{\bar{\eta}}^{Y_j} (F_1(G(x_0, \xi), t) - F_1(Y(\xi), t)) dt \leq 0.$$

This shows

$$\int_{-\infty}^{Y_j} (F_1(G(x_0, \xi), t) - F_1(Y(\xi), t)) dt < 0, \text{ for } j = 2, \dots, N. \quad (2.19)$$

The conclusion follows by combining (2.17), (2.18) and (2.19).  $\blacksquare$

Theorem 2.1 says that although problems (1.6) and (2.10) do not satisfy SCQ, the reformulated problem (2.12) may do under some circumstance. The fundamental reason behind this is to do with the plus function  $(\cdot)_+$ . Consider a single variate function  $a(x) = x$ . It is easy to see that the single inequality  $a(x) \leq 0$  satisfies SCQ but  $(a(x))_+ \leq 0$  does not although the two inequalities represent the same set  $(-\infty, 0]$ . Clearly, the constraint qualification is closely related to the function which represents the feasible set. In problem (2.12), we give an alternative presentation of the feasible constraints of (1.6) and (2.10) without the plus function (which could potentially destroy the SCQ).

### 3 Exact penalization schemes and level function method

Problem (2.12) is an ordinary nonlinear programming problem with finite number of constraints. This means that we can apply any existing NLP code to solve it. However, from numerical point of view, problem (2.12) is difficult to solve because every constraint  $\psi_j(x)$  is a maximum function of  $2^N - 1$  functions. That means problem (2.12) contains  $(2^N - 1)N$  constraints which depends on  $N$ , the cardinality of support set  $\Xi$ , and this may make the problem difficult to solve by well-known NLP methods such as the active set method even when  $N$  is not very large.

This motivates us to consider an exact penalty function method which is well known in nonlinear programming. Liu and Xu [12] proposed an  $L_\infty$ -norm based penalization scheme for the relaxed problem (1.5). In this context, their penalization scheme can be written as follows:

$$\min_x \sum_{i=1}^N p_i F(x, \xi^i) + \rho \max_{\eta \in [a, b]} \left( \sum_{i=1}^N p_i ((\eta - G(x, \xi^i))_+ - (\eta - Y(\xi^i))_+) \right)_+. \quad (3.20)$$

Justification of the penalty scheme (the equivalence of problem (1.5) and (3.20)) requires SCQ but the constraint qualification is not satisfied when  $Y(\xi) \subset [a, b]$ .

In this section, we apply the penalty function method to problem (2.12). There are essentially two ways to apply the penalty function method in this paper. One is to apply an exact penalty function method with  $L_\infty$ -norm to problem (2.12). The other is to use an exact penalty function method with  $L_1$ -norm. In this section, we consider both of them.

To this end, we need the following technical result.

**Lemma 3.1** *Let  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  be a continuous function and  $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$  be a continuous vector-valued function whose components are convex. Let  $X \subseteq \mathbb{R}^n$  be a compact and convex set. Consider the following constrained minimization problem*

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & x \in X. \end{aligned} \tag{3.21}$$

(i) *If  $g(x)$  satisfies SCQ, that is, there exists a point  $x_0$  and a real number  $\delta > 0$  such that*

$$\delta \mathcal{B} \subset g(x_0) + K,$$

*and the feasible set, denoted by  $S$ , is bounded, then*

$$d(x, S) \leq \delta^{-1} D d_1(0, g(x) + K)$$

*and*

$$d(x, S) \leq \delta^{-1} D d_\infty(0, g(x) + K),$$

*where  $\mathcal{B}$  denotes the closed unit ball in  $\mathbb{R}^m$  and  $K := [0, +\infty)^m$ , and  $D$  denotes the diameter of  $S$ .*

(ii) *If  $\phi(x)$  is Lipschitz continuous on  $X$  with modulus  $\kappa$ , then for*

$$\rho > \kappa \delta^{-1} D,$$

*the set of optimal solutions of (3.21) coincides with the set of optimal solutions of problem*

$$\begin{aligned} \min \quad & f(x) + \rho \|(g(x))_+\|_1 \\ \text{s.t.} \quad & x \in X, \end{aligned} \tag{3.22}$$

*and that of*

$$\begin{aligned} \min \quad & f(x) + \rho \|(g(x))_+\|_\infty \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{3.23}$$

**Proof.** Part (i) follows from Robinson's error bound for convex systems [16] and Part (ii) follows from Part (i) and Clarke's exact penalty function theorem [1, Proposition 2.4.3]. ■

### 3.1 Exact penalization with $L_1$ -norm

A popular exact penalty scheme in optimization is based on  $L_1$ -norm. Here we consider the penalization scheme for (2.12):

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) + \rho_1 \sum_{j=1}^N (\psi_j(x))_+ \\ \text{s.t.} \quad & x \in \mathcal{X}, \end{aligned} \quad (3.24)$$

and for (2.10):

$$\begin{aligned} \min_x \quad \vartheta(x) := \quad & \sum_{i=1}^N p_i F(x, \xi^i) + \rho_1 \sum_{j=1}^N \left( \sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ - \gamma_j \right)_+ \\ \text{s.t.} \quad & x \in \mathcal{X}. \end{aligned} \quad (3.25)$$

In what follows, we show that the two penalty schemes are equivalent, and estimate the penalty parameter. As we discussed in the preceding section, (2.10) does not satisfy the SCQ, but (2.12) does under some moderate conditions. A key point we want to make here is that exact penalization function scheme (3.25) is justified despite (2.10) does not satisfy the SCQ.

We need the following assumption on the underlying random functions of problem (1.6).

**Assumption 3.1**  $F(x, \xi^i)$ ,  $G(x, \xi^i)$  are continuously differentiable w.r.t.  $x$  in an open neighborhood of  $\mathcal{X}$ , for  $i = 1, \dots, N$ . Moreover, they are globally Lipschitz over  $\mathcal{X}$ , that is, there exists  $\kappa(\xi) < +\infty$  such that

$$\max(\|\nabla_x F(x, \xi^i)\|, \|\nabla_x G(x, \xi^i)\|) \leq \kappa(\xi^i)$$

for  $i = 1, \dots, N$ .

**Theorem 3.1** Assume: (a) problem (2.12) satisfies SCQ, (b) Assumption 3.1 holds; (c) the feasible set of problem (2.12) is bounded. Then

- (i) problems (3.24) and (3.25) are equivalent;
- (ii) there exist positive constants  $\bar{\delta}$  and  $\bar{D}$  such that when

$$\rho_1 > \sum_{i=1}^N p_i \kappa(\xi^i) \bar{\delta}^{-1} \bar{D},$$

the set of optimal solutions of (2.12) coincides with that of (3.24), and the set of optimal solutions of (2.10) coincides with that of (3.25).

**Proof.** Part (i). Through Lemma 2.1 and (2.16), it is easy to verify that

$$\sum_{j=1}^N (\psi_j(x))_+ = \sum_{j=1}^N (\varphi_j(x) - \gamma_j)_+ = \sum_{j=1}^N \left( \sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ - \gamma_j \right)_+, \quad (3.26)$$

where  $\varphi_j(\cdot)$  is defined in Lemma 2.1. The conclusion follows from (3.26).

Part (ii). Let

$$\Phi(x) := \begin{pmatrix} \psi_1(x) \\ \vdots \\ \psi_N(x) \end{pmatrix}$$

and  $\mathcal{Q}$  be the feasible set of problem (2.12). Since  $\mathcal{Q}$  is bounded,  $\sum_{i=1}^N p_i F(x, \xi^i)$  is Lipschitz continuous with modulus  $\sum_{i=1}^N p_i \kappa(\xi^i)$ , problem (2.12) is a convex programming problem and satisfies SCQ, by Lemma 3.1, there exists real numbers  $\bar{\delta} > 0$  and  $\bar{D} > 0$  such that when

$$\rho_1 > \sum_{i=1}^N p_i \kappa(\xi^i) \bar{\delta}^{-1} \bar{D},$$

the set of optimal solutions of problem (2.12) coincides with that of (3.24). Moreover, since problem (2.10) and (2.12) are equivalent, while problem (3.24) and (3.25) are equivalent, the set of optimal solutions of problem (2.10) coincides with that of (3.25). ■

Theorem 3.1 shows that the exact penalization (with  $L_1$ -norm) of problem (2.10) can be derived although it does not satisfy SCQ. This is achieved through problem (2.12). Since the reformulation of (2.10) depends on the distribution of random variable  $\xi$ , it is unclear whether Theorem 3.1 can be generalized to the case when  $\xi$  satisfies continuous distribution.

### 3.2 Exact penalization with $L_\infty$ -norm

Another popular penalty scheme in optimization is based on  $L_\infty$ -norm. Here we consider the penalization scheme for (2.12)

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) + \bar{\rho} \max_{j \in \{1, \dots, N\}} (\psi_j(x))_+ \\ \text{s.t.} \quad & x \in \mathcal{X}, \end{aligned} \quad (3.27)$$

and for (2.10)

$$\begin{aligned} \min_x \quad & \hat{\vartheta}(x) := \sum_{i=1}^N p_i F(x, \xi^i) + \bar{\rho} \max_{j \in \{1, \dots, N\}} \left( \sum_{i=1}^N p_i (Y_j - G(x, \xi^i))_+ - \gamma_j \right)_+ \\ \text{s.t.} \quad & x \in \mathcal{X}. \end{aligned} \quad (3.28)$$

Similar to the discussions in the preceding subsection, we need to show that the two penalty schemes are equivalent and give an estimate of the penalty parameter  $\bar{\rho}$ .

**Theorem 3.2** *Assume: (a) problem (2.12) satisfies SCQ, (b) Assumption 3.1 holds; (c) the feasible set of problem (2.12) is bounded. Then*

(i) *problems (3.27) and (3.28) are equivalent;*

(ii) *there exist positive constants  $\hat{\delta}$  and  $\hat{D}$  such that when*

$$\bar{\rho} > \sum_{i=1}^N p_i \kappa(\xi^i) \hat{\delta}^{-1} \hat{D},$$

the set of optimal solutions of (2.12) coincides with that of (3.27), and the set of optimal solutions of (2.10) coincides with that of (3.28).

**Proof.** The proof is similar to Theorem 3.1, we omit details here. ■

Analogous to the comments following Theorem 3.1, we note that a main contribution of Theorem 3.2 is to show exact penalization scheme with  $L_\infty$ -norm can be established for problem (2.10) despite it does not satisfy SCQ. The observation makes the exact penalization schemes more appealing because they can be applied to a fairly class of problems.

Note that our exact penalty schemes are established through Clarke's penalty function theorem [1, Proposition 2.4.3] and Robinson's error bound [16] for convex systems. The latter requires SCQ as a key condition. It is unclear if the exact penalization schemes can be derived through other avenues. For instance, Dentcheva and Ruszczyński [2] observed that first order optimality conditions of (2.10) may be established without SCQ. It might be interesting to explore whether this can be exploited to derive the error bound and exact penalization. We leave this for our future research.

### 3.3 Level function methods

Level function method is popular numerical approach for solving deterministic nonsmooth optimization problems. It is proposed by Lemaréchal et al [11] for solving nonsmooth convex optimization problems and extended by Xu [20] for solving quasiconvex optimization problems. Meskarian et al [13] recently applied a level function method to (3.20) where the distribution of  $\xi$  is discrete. In this subsection, we apply the level function method in [20] to problems (3.25) and (3.28).

Let  $v : \mathbb{R}^n \rightarrow \mathbb{R}$  be a locally Lipschitz continuous function. Recall that the *Clarke generalized derivative* of  $v$  at a point  $x$  in direction  $d$  is defined as

$$v^\circ(x, d) := \limsup_{y \rightarrow x, t \downarrow 0} \frac{v(y + td) - v(y)}{t}.$$

The *Clarke generalized gradient* (also known as Clarke subdifferential) is defined as

$$\partial v(x) := \{\zeta : \zeta^T d \leq v^\circ(x, d)\}.$$

See [1, Chapter 2] for the details of the concepts. In the case when  $v$  is convex, the Clarke subdifferential coincides with the usual convex subdifferential [17].

Let  $\vartheta(x)$  and  $\hat{\vartheta}(x)$  be defined as in (3.25) and (3.28) respectively. In the algorithm to be stated below, we need to calculate an element of the subdifferential of  $\vartheta(x)$  and  $\hat{\vartheta}(x)$ .

**Algorithm 3.1** (*Level function method for penalized problem (3.25) (or (3.28))*)

*Step 1.* Let  $\epsilon > 0$  be a constant and select a constant  $\tau \in (0, 1)$  and a starting point  $x_0 \in \mathcal{X}$ ; set  $k := 0$ .

*Step 2.* Calculate  $\zeta_k \in \partial_x \vartheta(x_k)$  (for problem (3.28),  $\zeta_k \in \partial_x \hat{\vartheta}(x_k)$ ). Set

$$\sigma_{x_k}(x) = \zeta_k^T (x - x_k) / \|\zeta_k\|$$

and

$$\sigma_k(x) = \max\{\sigma_{k-1}(x), \sigma_{x_k}(x)\}$$

where  $\sigma_{-1}(x) := -\infty$ . Let

$$x_k = \operatorname{argmin}\{\vartheta(x_j) : j \in 0, \dots, k\}$$

and

$$x_{k+1} \in \pi(x_k, Q_k),$$

where

$$Q_k = \{x \in \mathcal{X} : \sigma_k(x) \leq -\tau\Delta(k)\}, \quad \Delta(k) = -\min_{x \in \mathcal{X}} \sigma_k(x),$$

and  $\pi(x, Q_k)$  is the Euclidean projection of the point  $x$  on a set  $Q$ .

Step 3. If  $\Delta(k) \leq \epsilon$ , stop; otherwise, set  $k := k + 1$ ; go to step 2.

**Theorem 3.3** [20, Theorem 3.3] Let  $\{x_k\}$  be generated by Algorithm 3.1. Assume that  $\vartheta : \mathbb{R}^n \rightarrow \mathbb{R}$  is a continuous function and that the sequence of level functions  $\{\sigma_{x_k}(x)\}$  is uniformly Lipschitz with constant  $M$ . Then,

$$\Delta(k) \leq \epsilon, \text{ for } k > M^2 \Upsilon^2 \epsilon^{-2} \tau^{-2} (1 - \tau^2)^{-1},$$

where  $\Upsilon$  represents the diameter of  $X$ ,  $\epsilon$  and  $\tau$  are given in Algorithm 3.1.

## 4 A modified cutting plane method

Rudolf and Ruszczyński [19] and Fábíán et al [6] proposed cutting plane methods to solve a stochastic program with second order stochastic dominance constraints when the underlying random variable has finite distribution. The method is an extension of the cutting-plane method developed by Haneveld and Vlerk in [9] for integrated chance constraints (ICC). Here we revisit the cutting-plane method [19, 6] by considering a modification of the procedure where a cut is constructed. Our modified cutting-plane method is aimed to solve problem (2.12).

In [6, 19], the authors reformulated problem (2.10) as follows:

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \max_{\mathcal{J} \subseteq \{1, \dots, N\}} \sum_{i \in \mathcal{J}} p_i (Y_j - G(x, \xi^i)) - \gamma_j \leq 0, \forall j = 1, \dots, N, \\ & x \in \mathcal{X}. \end{aligned} \tag{4.29}$$

They proposed a cutting plane method for solving problem (4.29). Specifically, at iteration  $t$ , they considered a collection of subsets (events)

$$\{\mathcal{J}_{j,t} \subseteq \{1, \dots, N\} : j = 1, \dots, N\},$$

which depend on the  $t$ -th iterate, denoted by  $x_t$ , and solve subproblem

$$\begin{aligned} \min_x \quad & \sum_{i=1}^N p_i F(x, \xi^i) \\ \text{s.t.} \quad & \sum_{i \in \mathcal{J}_{j,t}} p_i (Y_j - G(x, \xi^i)) - \gamma_j \leq 0, \text{ for } j = 1, \dots, N \text{ and } l = 1, \dots, t, \\ & x \in \mathcal{X}. \end{aligned} \tag{4.30}$$

Specifically, in [6], constraints

$$\sum_{i \in \mathcal{J}_{j,t}} p_i(Y_j - G(x, \xi^i)) - \gamma_j \leq 0, \text{ for } j = 1, \dots, N \quad (4.31)$$

are added at iteration  $t$  (each of which corresponds to a cut) to the  $N \times (t - 1)$  constraints

$$\sum_{i \in \mathcal{J}_{j,t}} p_i(Y_j - G(x, \xi^i)) - \gamma_j \leq 0, \text{ for } j = 1, \dots, N \text{ and } l = 1, \dots, t - 1,$$

inherited from the previous iterations. Here  $\mathcal{J}_{j,t}$  is the index set such that

$$\sum_{i \in \mathcal{J}_{j,t}} p_i(Y_j - G(x_t, \xi^i)) = \max_{\mathcal{J} \subseteq \{1, \dots, N\}} \sum_{i \in \mathcal{J}} p_i(Y_j - G(x_t, \xi^i)).$$

It is observed that such  $\mathcal{J}_{j,t}$  can be identified as follows:

$$\mathcal{J}_{j,t} = \{i : Y_j - G(x_t, \xi^i) > 0, \text{ for } i = 1, \dots, N\},$$

see a comment at page 45 in [6].<sup>2</sup> Since the algorithm above is the modified Haneveld and Vlerk's cutting-plane method [9] based on the comment by Fábíán, Mitra and Roman in [6]. Let us call the cutting-plane method as *HVFM R's cutting-plane method*.

In [19],  $\mathcal{J}_{j,t} = \mathcal{J}_{\hat{j},t}$  for  $j = 1, \dots, N$ , where  $\hat{j}$  is any  $j \in \{1, \dots, N\}$  such that

$$\sum_{i=1}^N p_i(Y_j - G(x_t, \xi^i))_+ - \gamma_j > 0.$$

This differs from the previous approach because here the index set  $\mathcal{J}_{j,t}$  is constructed for the  $\hat{j}$ -th constraint which is violated at  $x_t$ , rather than for every  $j$ -th constraint.

The focus of [6, 19] is on the case when  $F$  and  $G$  are linear functions of  $x$  and subsequently subproblem (4.30) is a linear programming problem. In the case when  $G$  is nonlinear w.r.t.  $x$ , (4.30) is a nonlinear program and therefore the approach is not a cutting plane method in the classical sense. Rudolf and Ruszczyński [19] call it *cut generation method*, see details in [6, 19].

In what follows, we reformulate problem (2.12) as follows:

$$\begin{aligned} \min_{x,y} \quad & y \\ \text{s.t.} \quad & \psi(x) := \max_{j \in \{1, \dots, N\}} \psi_j(x) \leq 0, \\ & \sum_{i=1}^N p_i F(x, \xi^i) - y \leq 0, \\ & x \in \mathcal{X}, y \in \mathcal{Y}, \end{aligned} \quad (4.32)$$

where  $\mathcal{Y}$  is a closed convex compact set such that

$$\left\{ \sum_{i=1}^N p_i F(x, \xi^i) : x \in \mathcal{X} \right\} \subset \mathcal{Y}.$$

Existence of  $\mathcal{Y}$  is due to the fact that  $F(x, \xi^i)$ ,  $i = 1, \dots, N$ , is a continuous function and  $\mathcal{X}$  is a compact set. Note that  $G(x, \xi)$  is concave and  $F(x, \xi)$  is convex w.r.t.  $x$ , which implies that  $\psi(x)$  is convex w.r.t.  $x$  and  $\sum_{i=1}^N p_i F(x, \xi^i) - y$  is convex w.r.t.  $(x, y)$ . Moreover problem (4.32) is equivalent to problem (2.12). We apply the classical cutting-plane method [10] to both  $\psi(x)$  and  $\sum_{i=1}^N p_i F(x, \xi^i) - y$ . Specifically, we propose the following algorithm.

<sup>2</sup>Note that Fábíán, Mitra and Roman's [6] did not propose a detailed algorithm, instead, they observed that the cutting plane method due to Haneveld and Vlerk [9] can be easily applied to (4.29).

**Algorithm 4.1 (A modified cutting plane method)** Define the current problem  $CP_t$  at iteration  $t$  as

$$\begin{aligned} \min_{x,y} \quad & y \\ \text{s.t.} \quad & x \in \mathcal{X}, y \in \mathcal{Y} \\ & (x, y) \in S_t := \{(x, y) \in \mathcal{X} \times \mathcal{Y} : (a_l^{j_l^*})^T x \leq b_l^{j_l^*}, d_l^T x + e_l y \leq k_l, \quad l = 1, \dots, t\}, \end{aligned} \tag{4.33}$$

set  $t := 0$ ,  $S_0 := \mathcal{X} \times \mathcal{Y}$ . For each  $t$ , carry out the following.

Step 1. Solve the LP problem (4.33) and let  $(x_t, y_t)$  denote the optimal solution. If problem (4.33) is infeasible, Stop: the original problem is infeasible.

Step 2. Find  $j_t^*$  such that

$$j_t^* = \operatorname{argmax}\{\psi_j(x_t), j = 1, \dots, N\}$$

and construct an index set

$$\mathcal{J}_t := \{i : Y_{j_t^*} - G(x_t, \xi^i) > 0\}.$$

Step 3. If

$$\sum_{i=1}^N p_i (Y_{j_t^*} - G(x_t, \xi^i))_+ - \gamma_{j_t^*} \leq 0$$

and

$$\sum_{i=1}^N p_i F(x_t, \xi^i) - y_t \leq 0,$$

stop:  $(x_t, y_t)$  is the optimal solution. Otherwise, construct feasible cuts

$$(a_{t+1}^{j_t^*})^T x \leq b_{t+1}^{j_t^*}$$

and

$$d_{t+1}^T x + e_{t+1} y \leq k_{t+1},$$

where

$$\begin{aligned} a_{t+1}^{j_t^*} &:= - \sum_{i \in \mathcal{J}_t} p_i \nabla_x G(x_t, \xi^i), \\ b_{t+1}^{j_t^*} &:= \sum_{i \in \mathcal{J}_t} p_i (-\nabla_x G(x_t, \xi^i))^T x_t + G(x_t, \xi^i) - Y_{j_t^*} + \gamma_{j_t^*}, \\ d_{t+1} &:= \sum_{i=1}^N p_i \nabla_x F(x_t, \xi^i), \\ e_{t+1} &:= -1, \\ k_{t+1} &:= \sum_{i=1}^N p_i (\nabla_x F(x_t, \xi^i))^T x_t - F(x_t, \xi^i). \end{aligned}$$

and set

$$S_{t+1} := S_t \cap \left\{ (x, y) \in \mathcal{X} \times \mathcal{Y} : (a_{t+1}^{j_t^*})^T x \leq b_{t+1}^{j_t^*}, d_{t+1}^T x + e_{t+1} y \leq k_{t+1} \right\}.$$

Proceed with iteration  $t + 1$ .

**Remark 4.1** We make a few comments on Algorithm 4.1.

(i) The main difference between Algorithm 4.1 and the cutting plane methods in [19] is in the way how feasible cuts are constructed. In [19],  $N$  constraints/cuts are added at iteration  $t$ , see (4.31). These constraints/cuts are not necessarily the extreme support (tangent plane) of  $\psi(x)$  at  $x_t$ . In Algorithm 4.1, we exclude all those non-support constraints because we believe a cut at the extreme support (to  $\psi(x)$  at  $x_t$ ) is most effective and a single linear cut is adequate to ensure the convergence as we will demonstrate in Theorem 4.1; all other non-support constraints/cuts may potentially reduce numerical efficiency. Our approach is similar to the algorithm proposed by Fábíán, Mitra and Roman, see the algorithm at page 48 in [6]. Note that Fábíán, Mitra and Roman's algorithm is applied to linear models while Algorithm 4.1 is applicable to nonlinear case. Therefore we may regard the latter as an extension of the former.

(ii) In order to apply Algorithm 4.1 to problem (4.32), we need to identify index  $j_t^*$  where

$$j_t^* = \operatorname{argmax}\{\psi_j(x_t) : j = 1, \dots, N\}$$

at iteration  $t$ . There are two ways to do this. One is to calculate the value of  $\psi_j(x_t)$  for  $j \in \{1, \dots, N\}$  and identify the index corresponding to the maximum. The other is to solve the following DC-program

$$\max_{\eta \in [a, b]} \psi(x_t, \eta) := \sum_{i=1}^N p_i [(\eta - G(x_t, \xi^i))_+ - (\eta - Y(\xi^i))_+], \quad (4.34)$$

where  $[a, b] \supseteq \{y_1, \dots, y_N\}$ . We prefer the latter because when  $N$  is large, the former requires  $N$  function evaluations of  $\psi_j(x_t)$  which is numerically expensive, whereas the latter only requires to solve a single program albeit it is not convex.

- (iii) At Step 2, if there exists more than one index  $j_t^*$  such that the maximum is achieved, then we just pick up anyone of them. In such a case, the graph of function  $\psi(x)$  has a kink at  $x_t^*$ . Our algorithm requires to construct a support plane to one of the active piece (note that  $\psi(x)$  is piecewise smooth) and such a support plane is also a support to  $\psi(x)$  at  $x_t$ .
- (iv) When  $F$  is linear w.r.t.  $x$ , there is no need to introduce additional variable  $y$  because the objective is linear.
- (iv) Note that although problem (2.10) does not satisfy the SCQ, under some moderate conditions, problem (2.12) may satisfy it (Theorem 2.1). Moreover, since each subproblem (4.33) is a relaxation of problem (2.12), it also satisfies the SCQ.

The following theorem states convergence of the algorithm.

**Theorem 4.1** *Let  $\{(x_t, y_t)\}$  be a sequence generated by Algorithm 4.1. Let*

$$S := \{(x, y) \in \mathcal{X} \times \mathcal{Y} : \psi(x) \leq 0, \mathbb{E}[F(x, \xi)] - y \leq 0\} \subset \mathcal{X} \times \mathcal{Y},$$

where  $\psi(x)$  is defined in problem (4.32). Assume: (a)  $F(x, \xi)$  is continuously differentiable and convex and  $G(x, \xi)$  is continuously differentiable and concave w.r.t.  $x$  for almost every  $\xi$ ; (b)  $\mathcal{X} \times \mathcal{Y} \in \mathbb{R}^n$  is a compact set; (c) there exists a positive constant  $L$  such that the Lipschitz modulus of  $\mathbb{E}[F(x, \xi)]$  and  $\psi(x)$  are bounded by  $L$  on  $\mathcal{X} \times \mathcal{Y}$ ; (d)  $S$  is nonempty. Then  $\{(x_t, y_t)\}$  contains a subsequence which converges to a point  $(x^*, y^*) \in S$ , where  $(x^*, y^*)$  is the optimal solution and  $y^*$  is the optimal value of (4.32).

**Proof.** The proof is similar to [10, Theorem]. Note that in every iteration  $t > 0$ ,  $a_{t+1}^{j_i^*} \in \partial_x \psi(x_t)$ ,  $d_{t+1} = \nabla \mathbb{E}[F(x_t, \xi)]$  and  $e_{t+1} = \nabla_y (\mathbb{E}[F(x_t, \xi)] - y_t) = -1$ . Then  $(a_{t+1}^{j_i^*})^T x - b_{t+1}^{j_i^*}$  and  $d_{t+1}^T x + e_{t+1} y - k_{t+1}$  are the extreme support to the graphs of  $\psi(x)$  and  $\mathbb{E}[F(x, \xi)] - y$  at point  $(x_t, y_t)$  respectively. By condition (a),  $\psi(x)$  and  $\mathbb{E}[F(x, \xi)] - y$  are convex and continuous functions w.r.t.  $(x, y)$ . Thus, if  $(x, y) \in S$  and  $\max(\psi(x), \mathbb{E}[F(x, \xi)] - y) \leq 0$ , then

$$\max((a_{t+1}^{j_i^*})^T x - b_{t+1}^{j_i^*}, d_{t+1}^T x + e_{t+1} y - k_{t+1}) \leq 0.$$

On the other hand, for  $(x_t, y_t) \notin S$ ,

$$\max\{(a_{t+1}^{j_i^*})^T x_t - b_{t+1}^{j_i^*}, d_{t+1}^T x_t + e_{t+1} y_t - k_{t+1}\} = \max\{\psi(x_t), \mathbb{E}[F(x_t, \xi)] - y_t\} > 0.$$

Thus, when  $(x_t, y_t) \notin S$ , the set  $S$  and the point  $(x_t, y_t)$  lie on opposite sides of the cutting angle  $\max\{(a_{t+1}^{j_i^*})^T x - b_{t+1}^{j_i^*}, d_{t+1}^T x + e_{t+1} y - k_{t+1}\} = 0$ .

Note that from the definition of  $S_t$  and  $(x_t, y_t)$ , we know that  $S \subset S_t \subset S_{t-1}$ ,  $(x_t, y_t)$  minimizes  $y$  in  $S_t$  and  $y_{t-1} \leq y_t$ . In the case when  $(x_t, y_t) \in S$ , it is easy to verify that  $(x_t, y_t)$  is the optimal solution of problem (4.32).

In the rest of proof, we focus on the case when  $(x_t, y_t) \notin S$  for all  $t$ . Since  $\mathcal{X} \times \mathcal{Y}$  is a compact set, the sequence  $\{(x_t, y_t)\}$  contains a subsequence which converges to  $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ . We want to show that  $(x^*, y^*) \in S$ , which means that  $x^*$  is an optimal solution and  $y^*$  is the corresponding optimal value. Observe that  $(x_t, y_t)$  minimizes  $y$  in  $S_t$ , that is, it satisfies the inequalities:

$$(a_{l+1}^{j_i^*})^T x - b_{l+1}^{j_i^*} \leq 0, \quad (4.35)$$

and

$$d_{l+1}^T x + e_{l+1} y - k_{l+1} \leq 0, \quad (4.36)$$

for  $l = 0, \dots, t-1$  and by condition (c),  $\max\{\|(a_{l+1}^{j_i^*})\|, \|d_{l+1}\|\} \leq L$  for all  $l$ . For the simplicity of notation, let  $\{x_l, y_l\}$  denote the subsequence. We claim that  $\{\max(\psi(x_t), \mathbb{E}[F(x_t, \xi)] - y_t)\}$  converges to 0. Note that since

$$\begin{aligned} b_{l+1}^{j_i^*} &= \sum_{i \in \mathcal{J}_l} p_i (-\nabla_x G(x_l, \xi^i))^T x_l + G(x_l, \xi^i) - Y_{j_i^*} + \gamma_{j_i^*} \\ &= (a_{l+1}^{j_i^*})^T x_l - \psi_{j_i^*}(x_l) \\ &= (a_{l+1}^{j_i^*})^T x_l - \psi(x_l), \end{aligned}$$

then (4.35) implies

$$\psi(x_l) + (a_{l+1}^{j_i^*})^T (x - x_l) \leq 0.$$

Likewise, by the definition of  $e_{l+1}$ ,  $k_{l+1}$ , we have from (4.36) that

$$\mathbb{E}[F(x_l, \xi)] + d_{l+1}^T (x - x_l) - y \leq 0.$$

Assume for the sake of a contradiction that the desired convergence does not occur. Then there exists an  $r > 0$  independent of  $t$  such that

$$\begin{aligned} r &\leq \max(\psi(x_l), \mathbb{E}[F(x_l, \xi)] - y_l) \\ &\leq \max((a_{l+1}^{j_i^*})^T (x_l - x_t), d_{l+1}^T (x_l - x_t) - (y_l - y_t)) \\ &\leq (L+1) \|(x_l, y_l) - (x_t, y_t)\|, \end{aligned}$$

for all  $0 \leq l \leq t-1$ , which means  $\{(x_t, y_t)\}$  does not converge, a contradiction! This shows  $\{\max(\psi(x_t), \mathbb{E}[F(x_t, \xi)] - y_t)\}$  converges to 0 and hence  $(x^*, y^*) \in S$  is the optimal solution and  $y^*$  is the optimal value of (4.32).  $\blacksquare$

## 5 Numerical tests

In this section, we apply the proposed Algorithms 3.1 and 4.1 to an academic problem and a practical portfolio selection problem, and compare them with *HVFMRS cutting-plane method*<sup>3</sup>.

We have carried out a number of numerical tests on the two problems in MATLAB 7.10 installed on a Viglen PC with Windows XP operating system and 2.96 GB of RAM. We use Matlab optimization solver “fmincon” to solve the nonlinear subproblems constructed from Algorithm 3.1 and *HVFMRS cutting-plane method*. The linear subproblem constructed from Algorithm 4.1 is solved by Matlab linear optimization solver “linprog”. For Algorithm 3.1, we use the stopping criteria parameters  $\epsilon = 10^{-4}$  and  $\tau = 0.5$ . The Algorithm 4.1 and the cutting-plane method proposed in [6] terminates when the solution of any subproblem become a feasible point of the original problem, see the Step 3. of Algorithm 4.1 for details.

**Example 5.1** Consider second order dominance constrained optimization problem (2.10) with  $F(x, \xi) = -x\xi$ ,  $G(x, \xi) = x\xi - \frac{1}{2}x^2$ ,  $Y(\xi) = G(1, \xi)$ ,  $\mathcal{X} = [0, 50]$ ,  $\xi$  is a random variable with finite distribution

$$P(\xi = 2 + \frac{i-1}{N}) = \frac{1}{N}$$

for  $i = 1, \dots, N$  and  $N = 101$ . The second order dominance constrained optimization problem can be specifically presented as:

$$\begin{aligned} \min_x \quad & -\frac{1}{101} \sum_{i=1}^{101} x(2 + \frac{i-1}{101}), \\ \text{s.t.} \quad & \frac{1}{101} \sum_{i=1}^{101} (\eta - x(2 + \frac{i-1}{101}) + \frac{1}{2}x^2)_+ \leq \frac{1}{101} \sum_{i=1}^{101} (\eta - (2 + \frac{i-1}{101}) + \frac{1}{2})_+, \forall \eta \in \mathbb{R}, \\ & x \in \mathcal{X}. \end{aligned} \tag{5.37}$$

It is difficult to work out the feasible set precisely. Here we only need to find out the optimal solution of problem (5.37). For  $x \in [1, 3]$ ,

$$P(G(x, \xi) \leq \eta) \leq P(Y(\xi) \leq \eta), \text{ for all } \eta \in \mathbb{R},$$

which means

$$G(x, \xi) \succeq_1 G(1, \xi)$$

and hence

$$G(x, \xi) \succeq_2 G(1, \xi).$$

For  $x > 3$ ,

$$\sum_{i=1}^N p_i (1.5 - G(x, \xi^i))_+ > \sum_{i=1}^N p_i (1.5 - Y(\xi^i))_+$$

which implies

$$G(x, \xi) \not\succeq_2 G(1, \xi).$$

This shows that any point in the interval  $[1, 3]$  is a feasible point of problem (5.37) whereas any point  $x$  with  $x > 3$  is infeasible. It is easy to see that  $x^* = 3$  is the optimal solution (with corresponding optimal value  $-7.5$ ) because the objective function is linear w.r.t.  $x$ .

<sup>3</sup>The method has been described at the beginning of Section 4 of the paper.

We apply the  $L_1$ -norm based penalty scheme (3.25) and  $L_\infty$ -norm based penalty scheme (3.28) to problem (5.37) respectively and obtain the corresponding penalty problems

$$\begin{aligned} \min_x \quad & -\frac{1}{101} \sum_{i=1}^{101} x(2 + \frac{i-1}{101}) \\ & + \rho \sum_{j=1}^{101} \left( \frac{1}{101} \sum_{i=1}^{101} ((Y_j - x(2 + \frac{i-1}{101}) + \frac{1}{2}x^2)_+ - (Y_j - (2 + \frac{i-1}{101}) + \frac{1}{2}))_+ \right) \\ \text{s.t.} \quad & x \in \mathcal{X}, \end{aligned} \tag{5.38}$$

and

$$\begin{aligned} \min_x \quad & -\frac{1}{101} \sum_{i=1}^{101} x(2 + \frac{i-1}{101}) \\ & + \rho \max_{j \in \{1, \dots, N\}} \left( \frac{1}{101} \sum_{i=1}^{101} ((Y_j - x(2 + \frac{i-1}{101}) + \frac{1}{2}x^2)_+ - (Y_j - (2 + \frac{i-1}{101}) + \frac{1}{2}))_+ \right) \\ \text{s.t.} \quad & x \in \mathcal{X}. \end{aligned} \tag{5.39}$$

where  $Y_j := (2 + \frac{j-1}{101}) - \frac{1}{2}$ .

In what follows, we discuss the SCQ of problem (5.37) and estimate the penalty parameter  $\rho$ .

Consider formulation (2.12) for problem (5.37). Let  $\underline{y} := \min_{i=1}^N Y(\xi^i)$ . It is easy to show that  $\underline{y} = Y(2)$ . For  $x_0 = 2$ , it is a feasible point. Moreover,

$$G(x_0, \xi) \succeq_1 Y(\xi),$$

and

$$G(x_0, \xi^i) > \underline{y}$$

for all  $i = 1, \dots, N$ . This verifies the conditions of Theorem 2.1. Hence formulation (2.12) of problem (5.37) satisfies the SCQ.

Next we estimate the penalty parameter  $\rho$  through Theorem 3.1. We need to work out  $\kappa, \delta$  and  $D$  defined in Lemma 3.1. Observe first that the objective function of problem (5.37) is Lipschitz continuous function with modulus  $\kappa = 3$ . Let

$$\delta := - \max_{j \in \{1, \dots, N\}} \left( \frac{1}{N} \max_{\mathcal{J} \in \mathcal{N}} \sum_{i \in \mathcal{J}} (Y_j - G(x_0, \xi^i)) - \frac{1}{N} \sum_{i=1}^N (Y_j - Y(\xi^i))_+ \right).$$

It is easy to calculate that  $\delta = 0.005$ . On the other hand, it is easy to verify that the feasible set of problem (5.37) is contained in  $[0, 3]$ . Let  $D = 3$ . Let

$$\rho > \kappa \delta^{-1} D = 1800.$$

By Theorem 3.1 problem (5.37) and (5.38) are equivalent in terms of optimal solutions. Likewise, by Theorem 3.2, the problem (5.37) is equivalent to (5.39).

We have carried out numerical tests on four algorithms for problem (5.37): Algorithm 3.1 based on exact penalization with  $L_1$ -norm (Algorithm 3.1 ( $L_1$ ) for short), Algorithm 3.1 based on exact penalization with  $L_\infty$ -norm (Algorithm 3.1 ( $L_\infty$ ) for short), Algorithm 4.1 and the *HVFMR'S cutting-plane method*.

The numerical results are displayed in Table 1. A few words about notation. *Opt.Sol* denotes the numerical optimal solution and *Opt.Val* denotes the corresponding optimal value. To check the efficiency of the algorithms, we have recorded the CPU time (in minutes) for each of the algorithms. Note that “fmincon” requires an initial point. We set the initial point  $x_0 = 1$  for all subproblems in Algorithm 3.1 and *HVFMR’s cutting-plane method*. Algorithm 4.1 does not require a starting point as the subproblem there is a linear program.

	<i>CPU Time</i>	<i>Opt.Sol</i>	<i>Opt.Val</i>
Algorithm 3.1 ( $L_1$ )	0.0374	3.000	-7.4850
Algorithm 3.1 ( $L_\infty$ )	0.0148	3.000	-7.4850
HVFMR’s cutting-plane method	0.0463	3.003	-7.5099
Algorithm 4.1	0.0196	3.000	-7.4850

Table 1: Numerical results for (5.37)

The results show that all four algorithms perform efficiently albeit Algorithm 3.1 ( $L_1$ ) and *HVFMR’s cutting-plane method* takes slightly more CPU time. It seems that the problem size is a bit too small to tell the difference of the algorithms. Therefore we propose to test them with a portfolio optimization problem of larger size (with 93 decision variables).

**Example 5.2** Consider a short selling allowed portfolio optimization problem with nonlinear transaction costs. Let  $r_j : \Omega \rightarrow \mathbb{R}$ , denote the random return rate of asset  $j$  for  $j = 1, \dots, n$ , and  $R := (r_1, \dots, r_n)$ . We assume that  $\mathbb{E}[|r_j|] < \infty$ . Denoting by  $x_j$  the fraction of the initial capital invested in asset  $j$ , we can easily derive the total return rate:

$$R(\omega)^T x = r_1(\omega)x_1 + r_2(\omega)x_2 + \dots + r_n(\omega)x_n,$$

and the total return with transaction costs:

$$G(x, R) := R(\omega)^T x - 0.01 \times (|x_1| + |x_2| + \dots + |x_n|), \quad (5.40)$$

where the number 0.01 is the unit transaction cost.

Our aim is to investigate the optimal investment policy for a fixed capital in the  $n$  assets with some desirable characteristics of the portfolio. The main difficulty in formulating a meaningful portfolio optimization problem is the definition of the preference structure among feasible portfolios. There are several ways to incorporate risk-aversion into the model to address practical needs, see an excellent review by Roman and Mitra [18]. The stochastic dominance approach [3] which introduces a comparison to a benchmark return rate into the optimization problem, is a way which receives increasing attention over the past few years.

Let  $G(x, R)$  be defined as in (5.40), associated with the decision vector  $x \in \mathcal{X} \subset \mathbb{R}^n$ , where  $\mathcal{X}$  is a compact and convex set. In the literature of portfolio optimization,  $x$  is known as a portfolio and  $\mathcal{X}$  is the set of feasible portfolios.

Let  $\bar{x}$  denote the benchmark portfolio. Our aim is to find an optimal portfolio  $x$  such that  $G(x, R)$  dominates  $G(\bar{x}, R)$  in second order and  $\mathbb{E}[G(x, R)]$  is maximized. We formulate the decision making problem as a stochastic minimization problem with second order dominance constraints, namely (1.4), with  $F(x, \xi) = -G(x, R)$ ,  $G(x, \xi) = G(x, R)$  and  $Y(\xi) = Y(R)$ . The main issue here is that we don’t know the distribution of random vector  $R(\omega)$ . To get around

the difficulty, we use historical data (sample) and consider the following approximation problem:

$$\begin{aligned}
\min_x \quad & -\frac{1}{N} \sum_{i=1}^N (x^T R^i - 0.01 \|x\|_1) \\
\text{s.t.} \quad & \frac{1}{N} \sum_{i=1}^N ((Y_j - x^T R^i + 0.01 \|x\|_1)_+ - (Y_j - \bar{x}^T R^i + 0.01 \|\bar{x}\|_1)_+) \leq 0, \quad j = 1, \dots, N, \\
& x \in \mathcal{X},
\end{aligned} \tag{5.41}$$

where  $\|x\|_1 = |x_1| + \dots + |x_n|$ ,  $\{R^1, \dots, R^N\}$  is a set of historical data and

$$Y_j = \bar{x}^T R^j - 0.01 \|\bar{x}\|_1.$$

Problem (5.41) is also known as a sample average approximation of problem (1.4). Taking into account the short selling policy, we set an upper bound 2 and a lower bound  $-1$  for each  $x_i$ , consequently

$$\mathcal{X} = \{x : \sum_{i=1}^n x_i = 1, x_i \in [-1, 2], \text{ for } i = 1, \dots, n\}.$$

Note that the problem (5.41) is different from the portfolio optimization problem investigated in [6, 21] in that the underlying function  $G(x, R)$  is nonlinear w.r.t.  $x$ .

We have carried out numerical tests on the four algorithms for problem (5.41): Algorithm 3.1 ( $L_1$ ), Algorithm 3.1 ( $L_\infty$ ), Algorithm 4.1 and *HVFM*R's *cutting-plane method*. Since it is difficult to estimate the penalty parameter for the two penalization schemes to be applied to problem (5.41), we set an estimate of  $\rho := 1800$  for both of the penalization schemes in the tests.

We have collected 300 daily returns of 93 FTSE100 assets prior to March 2011. The first 200 of them are used to construct the portfolio strategy and set up a backtest in order to investigate the performance of the portfolio in-sample. The remaining 100 are used to carry out an out-of-sample test to examine the efficiency of the portfolio out-of-sample.

Table 2 displays the test results of the four algorithms in terms of CPU time and optimal return based on the first 200 data. The four algorithms generate almost identical optimal portfolio and return. The numerical efficiency, however, seems different in terms of CPU time: *HVFM*R's *cutting-plane method* apparently requires more computational time. This is because the subproblem of *HVFM*R's *cutting-plane method* is a nonlinear program in this example and the Matlab optimization solver "fmincon" cannot solve them efficiently particularly when the problem size is large.

	<i>CPU Time</i>	<i>Opt.Return</i>
Algorithm 3.1 ( $L_1$ )	1.0312	0.0868
Algorithm 3.1 ( $L_\infty$ )	0.9065	0.0868
<i>HVFM</i> R's <i>cutting-plane method</i>	35.2830	0.0868
Algorithm 4.1	1.0308	0.0868

Table 2: Numerical results for (5.41).

In order to investigate the performance of the selected portfolio, we have carried out a backtest and an out-of-sample test for each of the algorithms. Figures 1 and 2 display the return based on optimal portfolios obtained through Algorithm 3.1 ( $L_1$ ) and Algorithm 3.1 ( $L_\infty$ ) respectively, in comparison with the benchmark return (FTSE100 Index). The performance of

the portfolio strategies generated by the *HVFMR's cutting-plane method* and Algorithm 4.1 are displayed in Figures 3 and 4.

Let us look at Figure 1. The curve which looks “horizontal” represents the daily return of the portfolio based on FTSE100 index. The solid curve and the dashed curve with significant fluctuations represent the portfolio returns based on optimal portfolio obtained by Algorithm 3.1 ( $L_1$ ) and Algorithm 3.1 ( $L_\infty$ ) respectively. Observe first that the two curves are mostly located above the “horizontal” curve. This means that the selected portfolios outperform the benchmark one with high returns. Second, fluctuation/variation of the solid curve and the dashed curve is largely due to the short selling policy in the model which generates a higher return in most scenarios. In case when short selling is not allowed, the optimal portfolio return still outperforms the benchmark one but with less variations. We leave this out in the figure.

Figure 2 depicts the out-of-sample test results of returns based on portfolio strategies generated by Algorithm 3.1 ( $L_1$ ) and Algorithm 3.1 ( $L_\infty$ ) in comparison with FTSE100 index. It is easy to observe that for the rest 100 observations, the performance of selected portfolio returns is better than that of the benchmark portfolio. Similar patterns can be observed from Figures 3 - 4 which depict the backtest and out-of-sample test results of HVFMR's cutting-plane method and Algorithm 4.1 respectively.

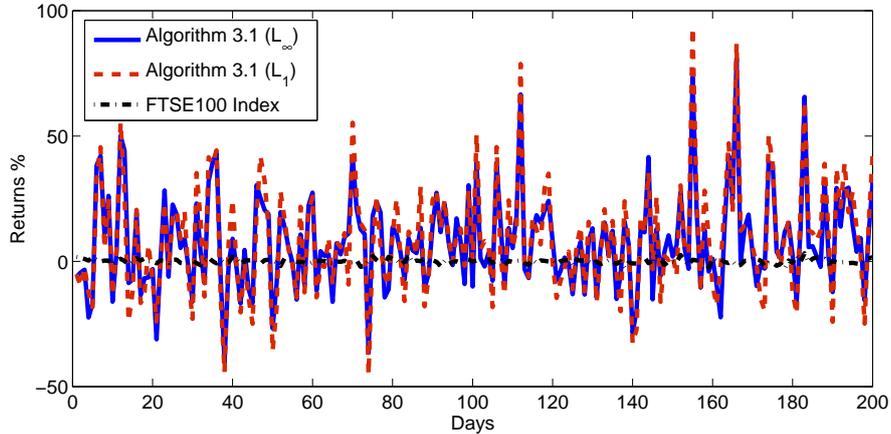


Figure 1: Backtest of the selected portfolios using Algorithm 3.1 ( $L_1$ ) and Algorithm 3.1 ( $L_\infty$ ) in comparison with FTSE100 index.

Table 3 displays the returns and risks based on the selected portfolios and the benchmark one both in-sample and out-of-sample. Here we use the Value-at-Risk (VaR) as the risk measure, which is one of the most commonly used measures of risk in finance. It is defined as

$$\text{VaR}_\alpha(-G(x, R)) := \min_{\eta \in \mathbb{R}} \{\eta : \text{Prob}\{-G(x, R) \leq \eta\} \geq \alpha\},$$

where  $\alpha \in (0, 1)$  and  $-G(x, R)$  is the loss function. In this context, the formulation above means that with the probability less than  $1 - \alpha$ , the loss  $-G(x, R)$  will be greater than  $\text{VaR}_\alpha(-G(x, R))$  or equivalently, the return  $G(x, R)$  will be less than  $-\text{VaR}_\alpha(-G(x, R))$ . For a fixed  $\alpha$ , a smaller  $\text{VaR}_\alpha(-G(x, R))$  means smaller risk. Three values of  $\alpha$  are commonly considered: 0.90, 0.95, 0.99. In our analysis, we consider  $\alpha = 0.95$ . From Table 3, we can see that the optimally selected portfolios generate higher returns with lower risks in comparison with those of the benchmark portfolio both in-sample and out-of-sample.

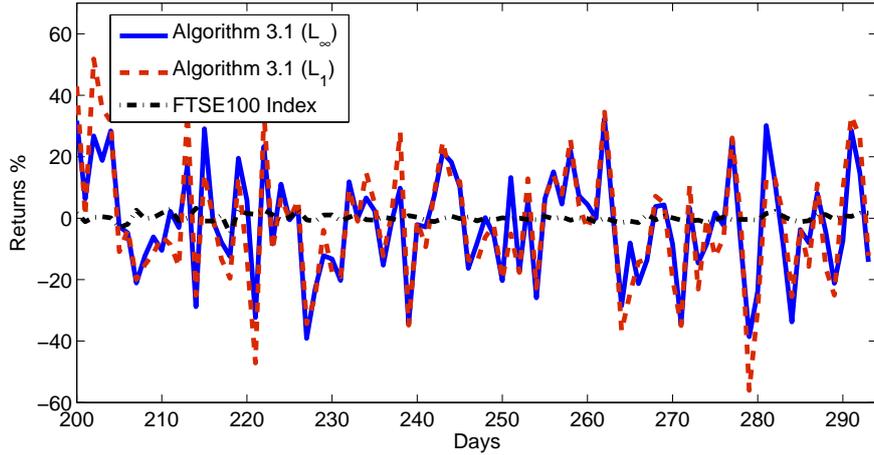


Figure 2: Out-of-sample test of the selected portfolios using Algorithm 3.1 ( $L_1$ ) and Algorithm 3.1 ( $L_\infty$ ) in comparison with FTSE100 index.

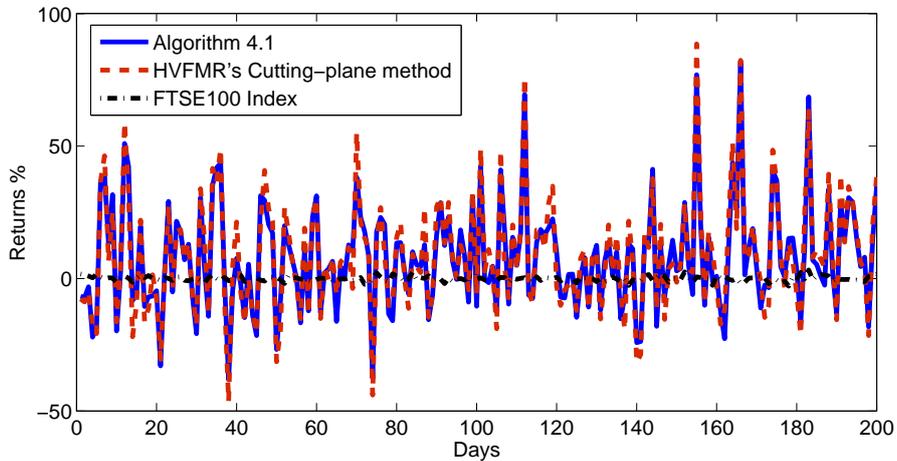


Figure 3: Backtest of the selected portfolios using HVFMR'S cutting-plane method and Algorithm 4.1 in comparison with FTSE100 index.

Finally, we have carried out some sensitivity analysis of the four algorithms with respect to the change of problem size and number of scenarios. Figure 5 depicts CPU time of the four algorithms as the number of assets increase from 10 to 2500. It shows that Algorithm 3.1 ( $L_\infty$ ) and Algorithm 4.1 require considerably less CUP times as the number of assets increases. The underlying reason that Algorithm 3.1 ( $L_\infty$ ) outperforms Algorithm 3.1 ( $L_1$ ) is that the former requires to calculate a subgradient of a single nonsmooth function while the latter requires to calculate a subgradient of the sum of  $N$  nonsmooth functions which takes more times as problem size increases. *HVFMR's cutting-plane method* is proposed to solve linear problems and in this nonlinear setting, it performs reasonably well with respect to large number of assets considered. Figure 6 displays similar phenomena. As the size of scenarios increases, *HVFMR's cutting-plane method* have more nonlinear constraints while Algorithm 3.1 ( $L_1$ ) takes more time to calculate a subgradient. There seems to be no significant impact on the other two algorithms.

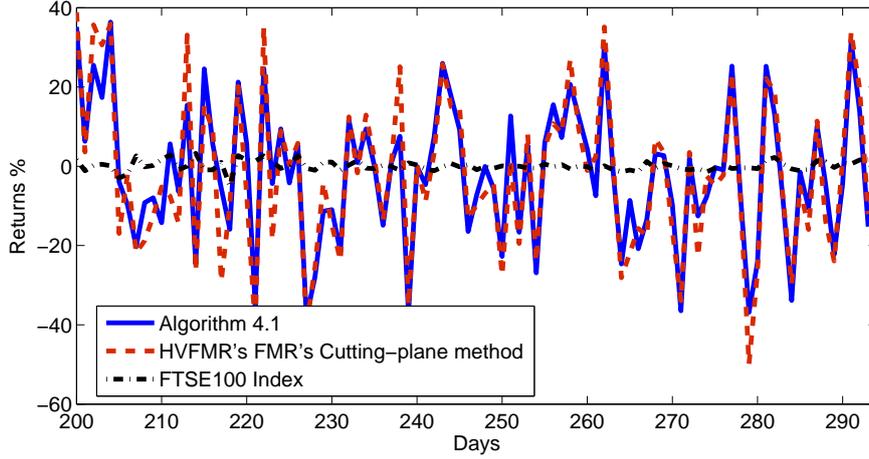


Figure 4: Out-of-sample test of the selected portfolios using HVFMR'S cutting-plane method and Algorithm 4.1 in comparison with FTSE100 index.

Data	Portfolio	Return	VaR
In-sample	Selected portfolio	0.0868	-0.0752
	Benchmark portfolio	0.00031	0.0008
Out-of-sample	Selected portfolio	0.0208	-0.0107
	Benchmark portfolio	0.0085	0.0015

Table 3: Comparison of the selected portfolio to the benchmark portfolio.

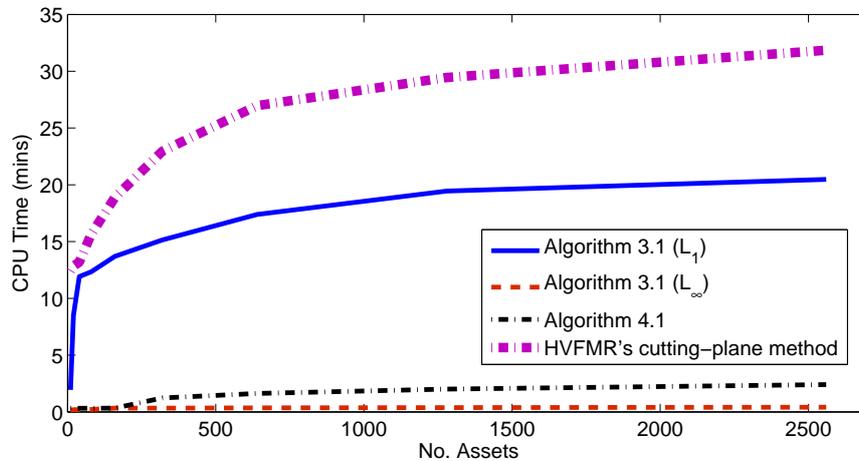


Figure 5: computational time versus the number of assets for a fixed number of observations.

**Acknowledgements.** The authors would like to thank Professor CasBa Fábíán for his comments on an earlier version of the paper which helped us clarify the cutting plane algorithms in [6].

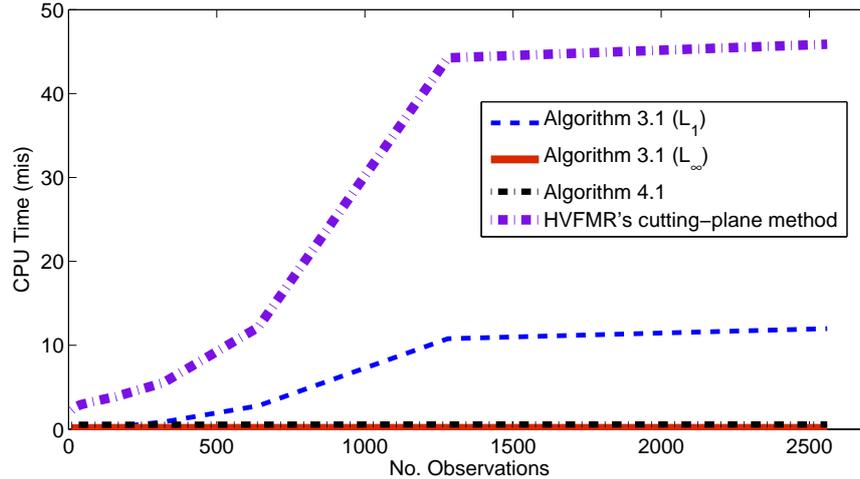


Figure 6: computational time versus the number of observations for a fixed number of assets.

## References

- [1] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.
- [2] D. Dentcheva and A. Ruszczyński, Optimization with stochastic dominance constraints, *SIAM Journal on Optimization*, Vol. 14, pp. 548-566, 2003.
- [3] D. Dentcheva and A. Ruszczyński, Portfolio optimization with stochastic dominance constraints, *Journal of Banking and Finance*, Vol. 30, pp. 433-451, 2006.
- [4] D. Dentcheva and A. Ruszczyński, Semi-infinite probabilistic constraints: optimality and convexification, *Optimization*, Vol. 53, pp. 583-601, 2004.
- [5] D. Dentcheva and A. Ruszczyński, Optimality and duality theory for stochastic optimization with nonlinear dominance constraints, *Mathematical Programming*, Vol. 99, pp. 329-350, 2004.
- [6] C. Fábián, G. Mitra and D. Roman, Processing second-order stochastic dominance models using cutting-plane representations. To appear in *Mathematical Programming*, 2010.
- [7] T. Homem-de-Mello and S. Mehrotra, A cutting surface method for uncertain linear programs with polyhedral stochastic dominance constraints, *SIAM Journal on Optimization*, Vol. 20, pp. 1250-1273, 2009.
- [8] J. Hu, T. Homen-De-Mello and S. Mehrotra, Sample average approximation of stochastic dominance constrained programs. To appear in *Mathematical Programming*, 2010.
- [9] W. Klein Haneveld and M. van der Vlerk, Integrated chance constraints: reduced forms and an algorithm, *Computational Management Science*, Vol. 3, pp. 245-269, 2006.
- [10] J. E. Kelley, The cutting-plane method for solving convex programs, *SIAM Journal on Applied Mathematics*, Vol. 8, pp. 703-712, 1960.
- [11] C. Lemarechal, A. Nemirovskii and Y. Nesterov, New variants of bundle methods, *Mathematical Programming*, Vol. 69, pp. 111-147, 1995.

- [12] Y. Liu and H. Xu, *Stability and sensitivity analysis of stochastic programs with second order dominance constraints*, Preprint, School of Mathematics, University of Southampton, June 2010.
- [13] R. Meskarian, H. Xu and J. Fliege, Numerical methods for stochastic programs with second order dominance constraints with applications to portfolio optimization, *European Journal of Operations Research*, Vol. 216, pp. 376-385, 2011.
- [14] A. Müller and M. Scarsini, Eds., *Stochastic Orders and Decision under Risk*, Institute of Mathematical Statistics, Hayward, CA, 1991.
- [15] W. Ogryczak and A. Ruszczyński, From stochastic dominance to mean-risk models: Semideviations as risk measures, *European Journal of Operational Research*, Vol. 116, pp. 33-50, 1999.
- [16] S. M. Robinson, An application of error bounds for convex programming in a linear space, *SIAM Journal on Control and Optimization*, Vol. 13, pp. 271-273, 1975.
- [17] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, 1982.
- [18] D. Roman and G. Mitra, Portfolio selection models: A review and new directions, *Wilmott Journal*, Vol. 1, pp. 69-85, 2009.
- [19] G. Rudolf and A. Ruszczyński, Optimization problems with second order stochastic dominance constraints: duality, compact formulations, and cut generation methods, *SIAM Journal on Optimization*, Vol. 19, pp. 1326-1343, 2008.
- [20] H. Xu, Level function method for quasiconvex programming, *Journal of Optimization Theory and Applications*, Vol. 108, pp. 407-437, 2001.
- [21] H. Xu and D. Zhang, Smooth sample average approximation of stationary points in non-smooth stochastic optimization and applications, *Mathematical Programming*, Vol. 119, pp. 371-401, 2009.