European Journal of Operational Research 216 (2012) 376–385

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Stochastics and Statistics

Numerical methods for stochastic programs with second order dominance constraints with applications to portfolio optimization

Rudabeh Meskarian *, Huifu Xu, Jörg Fliege

CORMSIS, School of Mathematics, University of Southampton, Southampton, SO17 1BJ, UK

ARTICLE INFO

Article history:
Received 8 December 2010
Accepted 22 July 2011
Available online 2 August 2011

Keywords:
Stochastic programming
Portfolio optimization
Penalty methods
Second order dominance
Stochastic approximation

ABSTRACT

Inspired by the successful applications of the stochastic optimization with second order stochastic dominance (SSD) model in portfolio optimization, we study new numerical methods for a general SSD model where the underlying functions are not necessarily linear. Specifically, we penalize the SSD constraints to the objective under Slater's constraint qualification and then apply the well known stochastic approximation (SA) method and the level function method to solve the penalized problem. Both methods are iterative: the former requires to calculate an approximate subgradient of the objective function of the penalized problem at each iterate while the latter requires to calculate a subgradient. Under some moderate conditions, we show that w.p.1 the sequence of approximated solutions generated by the SA method converges to an optimal solution of the true problem. As for the level function method, the convergence is deterministic and in some cases we are able to estimate the number of iterations for a given precision. Both methods are applied to portfolio optimization problem where the return functions are not necessarily linear and some numerical test results are reported.

1. Introduction

The notion of stochastic dominance as a constraint for optimization problems was introduced by Dentcheva and Ruszczýnski [4]. The concept of stochastic dominance is fundamental when comparing two random variables, it allows one to define preference among random variables. This concept has been playing an important role in portfolio optimization. Let $g(x,\xi)$ be a concave function, with decision vector x and random variable ξ. It is said that $g(x,\xi)$ stochastically dominates $g(y,\xi)$ in the first order, denoted by $g(x,\xi) \succeq_{(1)} g(y,\xi)$, if

$$F(g(x,\xi);\eta) \leqslant F(g(y,\xi);\eta), \quad \forall \eta \in \mathbb{R}, \tag{1.1}$$

where $F(g(x,\xi);\eta)$ and $F(g(y,\xi);\eta)$ are the cumulative distribution functions of $g(x,\xi)$ and $g(y,\xi)$, respectively. Similarly, $g(x,\xi)$ stochastically dominates $g(y,\xi)$ in the second order, denoted by $g(x,\xi) \succeq_{(2)} g(y,\xi)$, if

$$\int_{-\infty}^{\eta} F(g(x,\xi);\alpha)d\alpha \leqslant \int_{-\infty}^{\eta} F(g(y,\xi);\alpha)d\alpha, \quad \forall \eta \in \mathbb{R}. \tag{1.2}$$

The concept of stochastic dominance is widely used in economics and finance because of its relation to models of risk-averse preferences [15].

In this paper, we consider the following optimization problem with second order stochastic dominance constraints:

$$\begin{array}{ll} \max\limits_{x} & \mathbb{E}[f(x,\xi)], \\ \text{s.t.} & g(x,\xi) \succeq_{(2)} g(y,\xi), \\ & x \in X, \end{array} \tag{1.3}$$

where $f : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$, $g : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$ are concave continuous functions both in x and ξ, $x \in X$ is a decision vector with X being a nonempty convex subset of $\mathbb{R}^n$, $y \in X$ is a predefined vector, and $\xi : \Omega \to \Xi \subset \mathbb{R}^k$ is a random vector defined on probability space $(\Omega, \mathcal{F}, P)$ with support Ξ, $\mathbb{E}[\cdot]$ denotes the expected value w.r.t. the probability distribution of ξ.

Dentcheva and Ruszczyński analyzed several aspects of the stochastic dominance model including optimality and duality [5,6], as well as numerical methods [4]. Roman et al. [29] proposed a multi-objective portfolio selection model with second order stochastic dominance constraints, and Fábián [14] developed an efficient method to solve this model based on a cutting plane scheme. In a more recent development, Dentcheva and Ruszczyński [7] introduced concept of positive linear multivariate stochastic dominance and obtained necessary conditions of optimality for non-convex problems. Furthermore, Homem-de-Mello and Mehrota [20] proposed a sample average cutting-surface algorithm for optimization problems with multidimensional polyhedral linear second-order stochastic dominance constraints.

* Corresponding author. Tel.: +44 2380592660.
E-mail address: rm3g08@soton.ac.uk (R. Meskarian).

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.07.044

It is well known [25,33] that the second order dominance constraints in (1.3) can be reformulated as

$$\mathbb{E}\big[(\eta - g(x,\xi))_+\big] \leqslant \mathbb{E}[(\eta - g(y,\xi))_+], \quad \forall \eta \in \mathbb{R}, \tag{1.4}$$

where $(\eta - g(x,\xi))_+ = \max(\eta - g(x,\xi), 0)$. Consequently, problem (1.3) can be formulated as a stochastic semi-infinite programming problem:

$$\begin{aligned} \min_x \quad & -\mathbb{E}[f(x,\xi)], \\ \text{s.t.} \quad & G(x,\eta) := \mathbb{E}[(\eta - g(x,\xi))_+] - \mathbb{E}[(\eta - g(y,\xi))_+] \leqslant 0, \quad \forall \eta \in \mathbb{R}, \\ & x \in X. \end{aligned} \tag{1.5}$$

To overcome serious technical difficulties associated with the dominance constraint, a so-called *relaxed form* of the program is proposed:

$$\begin{aligned} \min_x \quad & -\mathbb{E}[f(x,\xi)], \\ \text{s.t.} \quad & G(x,\eta) \leqslant 0, \quad \forall \eta \in [a,b], \\ & x \in X, \end{aligned} \tag{1.6}$$

where $[a,b]$ is a closed interval in $\mathbb{R}$. Dentcheva and Ruszczyński [4] showed that, if ξ has uniformly bounded distribution, problem (1.6) is equivalent to problem (1.5) for some appropriate interval $[a,b]$. However, under general conditions, (1.6) is a relaxation of (1.5) in the sense that (1.6) has a larger set of feasible solutions and subsequently its optimal value gives a lower bound for the problem (1.5). Furthermore, the relaxed problem (1.6) is more likely to satisfy the Slater condition which is closely related to numerical stability.

This paper is concerned with numerical methods for solving (1.6). Specifically, we apply two well known methods: the stochastic approximation (SA) method and the level function methods to solve (1.6). The stochastic approximation (SA) method can be traced back to the pioneering work of Robbins and Monro [27]. Since then the SA algorithm has become widely used in stochastic optimization (see, [1,9–12,17,13], and reference therein). In this paper, we focus on a stochastic quasi-gradient method (SQG) which generalizes the SA method. The SQG method is a stochastic algorithmic procedure for solving general constrained optimization problems with non-differentiable, non-convex functions. Poljak [26] proposed techniques for investigating the local convergence of stochastic optimization processes and proved some results concerning differentiable optimization. A formal investigation of the asymptotic rate of convergence of SQG procedures was also carried out by Poljak [26].

The SQG method uses a quasi-subgradient of the objective function at each iteration. However, it might be helpful to use the subgradient information at the previous iterate. This motivates us to resort to the level function method from non-smooth optimization proposed by Lemarechal et al. [18] and extended by Xu [34]. The basic idea of the method is to use a subgradient to construct a linear level function which characterizes the level set of the objective function at the iterate and then treat the minimizer of the maximum of the linear functions as the next iterate.

As far as we are concerned, the main contribution of this paper can be summarized as follows:

- We exploit a recently developed exact penalization scheme for stochastic programming models with SSD constraints and apply the stochastic approximation method and the level function method to solve the penalized problem. The SA method requires to calculate only one approximate subgradient per iteration and can be applied to the case when the underlying functions are highly nonlinear and/or non-smooth, and the distribution of the random variable may be unknown. The level function method requires to calculate a subgradient instead of an approximate subgradient of the objective function at each iterate and therefore it applies to the problem with known distribution of the random variable or the sample average approximated problems. The corresponding numerical scheme provides an alternative approach to the existing cutting plane methods for SSD models [14,20]. A clear advantage of the method is that we can estimate the number of iterations needed for a specified precision.
- We apply the penalization scheme and the numerical methods to some portfolio problems where the underlying return functions are not necessarily linear and present some test results. Moreover, we use real world test data to set up both backtest and out-of-sample test for investigating the performance of the portfolio based on SSD model in comparison with the Markowitz model.

The rest of the paper is organized as follows. In Section 2, we discuss preliminaries needed throughout the paper. In Section 3 we discuss the stochastic quasi-gradient algorithm and analyze the convergence of optimal solutions. In Section 4, we discuss the level method and its complexity. Finally in Section 5, we apply the proposed methods to portfolio optimization problems and report some numerical test results.

2. Preliminaries and exact penalization

In this section, we consider an exact penalization of problem (1.5). We start by discussing preliminaries needed.

Throughout this paper, we use the following notation. Let $x^T y$ denotes the scalar products of two vectors x and y, and let $\|\cdot\|$ denotes the Euclidean norm.

For a real valued smooth function $h(x)$, we use $\nabla h(x)$ to denote the gradient of h at x. Let "conv" denotes the convex hull of a set.

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

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

The function v is said to be *Clarke regular* at x if the usual one sided directional derivative, denoted by $v'(x,d)$, exists for all $d \in \mathbb{R}^n$ and $v^o(x,d) = v'(x,d)$. The *Clarke generalized gradient* (also known as Clarke subdifferential) is defined as

$$\partial_x v(x) := \{\zeta : \zeta^T d \leqslant v^o(x,d)\},$$

see [3, Chapter 2].

Proposition 2.1. *Let $G(x,\eta)$ be defined as in (1.5). Assume that $g(x,\xi)$ is continuous w.r.t x and ξ is Lipschitz continuous w.r.t. x with integrably bounded Lipschitz modulus $\kappa(\xi)$. Let $T = [a,b]$*

$$P(x,\eta) := \max(G(x,\eta), 0) \tag{2.1}$$

and

$$\vartheta(x) := \max_{\eta \in T} P(x,\eta). \tag{2.2}$$

For any fixed $x \in X$, let $T^*(x)$ denote the set of $\bar{\eta} \in T$ such that $P(x,\bar{\eta}) = \max_{\eta \in T} P(x,\eta)$. Then

$$\partial_x P(x,\eta) = \begin{cases} \{0\}, & \text{if } G(x,\eta) < 0, \\ \operatorname{conv}\{0, \partial_x G(x,\eta)\}, & \text{if } G(x,\eta) = 0, \\ \partial_x G(x,\eta), & \text{if } G(x,\eta) > 0. \end{cases} \tag{2.3}$$

Moreover, $\vartheta(x)$ is Lipschitz continuous with modulus $\mathbb{E}[\kappa(\xi)]$ and

$$\partial_x \vartheta(x) = \operatorname{conv}\left\{ \bigcup_{\eta \in T^*(x)} \partial_x P(x,\eta) \right\}. \tag{2.4}$$

Proof. Since $g(x,\xi)$ is concave, then $G(x,\eta)$ is convex in x and hence it is Clarke regular, see [3, Proposition 2.3.6]. By [3, Proposition 2.3.12],

$$\partial_x [G(x,\eta)]_+ = \begin{cases} \{0\}, & \text{if } G(x,\eta) < 0, \\ \operatorname{conv}\{0, \partial_x G(x,\eta)\}, & \text{if } G(x,\eta) = 0, \\ \partial_x G(x,\eta), & \text{if } G(x,\eta) > 0. \end{cases} \tag{2.5}$$

The verification of Lipschitzness of $\vartheta(x)$ is straightforward. Applying the Levin–Valadier theorem (see [30, Section 2, Theorem 51]) to $\vartheta(x)$, we obtain (2.4). □

The focus of this paper is on numerical methods for solving the relaxed SSD problem (1.6). There are three issues to deal with: (a) the expectation of random functions in both the objective and constraints, (b) the infinite number of constraints, (c) the non-smoothness resulting from the max functions.

Our first step is to use an exact penalty function to move the infinite number of constraints to the objective, The penalty function method is well known [16] and has recently been used by Liu and Xu [22] for (1.6). Specifically, they considered the following problem:

$$\begin{aligned} \min_x \quad & \varphi(x,\rho) = -\mathbb{E}[f(x,\xi)] + \rho\vartheta(x), \\ \text{s.t.} \quad & x \in X, \end{aligned} \tag{2.6}$$

where $\rho > 0$ is a penalty parameter. Liu and Xu established the equivalence between problem (1.5) and penalized problem (2.6) in the sense of optimal solutions under some moderate conditions. Penalty methods for stochastic programs have also been discussed by Branda [2] and Dupačová et al. [8].

Theorem 2.1. [22, Theorem 2.3]. *Assume that problem (1.6) satisfies the Slater condition, that is, there exists a positive number δ and a point $\bar{x} \in X$ such that*

$$\max_{\eta \in T} G(\bar{x},\eta) < -\delta. \tag{2.7}$$

Assume also that X is a compact set, and $f(\cdot,\xi)$ and $g(\cdot,\xi)$ are locally Lipschitz continuous w.r.t. x and their Lipschitz modulus are bounded by an integrable function $\kappa(\xi) > 0$. Then there exists a positive constant $\hat{\rho}$ such that for any $\rho > \hat{\rho}$, the set of optimal solutions of the problems (1.5) and (2.6) coincide.

The proof is provided in the Appendix A.

Remark 2.1. Under the conditions of Theorem 2.1, problem (2.6) is a convex minimization problem with the objective function $\varphi(x,\rho)$ being Lipschitz continuous. The optimality condition of the problem can be written as

$$0 \in -\mathbb{E}[\nabla f(x,\xi)] + \rho\partial_x\vartheta(x) + 0\mathcal{N}_X(x), \tag{2.8}$$

where $\mathcal{N}_X(x)$ denotes the normal cone to X at point x in the sense of convex analysis [28]. Let $P_X(x) = \arg\min_{y\in X}\|x-y\|$ denote the orthogonal projection of x on X. Then the optimality condition (2.8) can be stated as follows: there exists $w \in \partial_x\vartheta(x)$ such that

$$P_X(x + \mathbb{E}[\nabla f(x,\xi)] - \rho w) = x. \tag{2.9}$$

We will use this in the next section.

In the rest of this paper, we apply the stochastic approximation method and the level function methods to solve (2.6).

3. Stochastic approximation method

In this section we discuss the stochastic quasi-gradient (SQG) method for solving the penalized problem (2.6). One of the main reasons that we apply this method is that the objective function of (2.6) is non-smooth.

Let $x_k \in X$ be an approximate solution of (2.6). The SQG method calculates a quasi-gradient, denoted by ζ_k, of $\varphi(x,\rho)$ at x_k such that

$$\mathbb{E}[\zeta_k/\{x_0,\ldots,x_k\}] \in -\mathbb{E}[\nabla f(x_k,\xi)] + \rho\partial_x\vartheta(x_k) + v_k, \tag{3.1}$$

where v_k is a controlled error, and by Proposition 2.1

$$\partial_x\vartheta(x_k) = \operatorname{conv}\left\{ \bigcup_{\eta\in T^*(x_k)} \partial_x P(x_k,\eta) \right\}, \tag{3.2}$$

where $T^*(x_k)$ is the set of solutions to (2.2) for $x = x_k$. In order to calculate an element of $\partial_x\vartheta(x_k)$, we need to find an $\eta \in +T^*(x_k)$. This amounts to solving optimization problem (2.2) w.r.t. η. Note that $P(x,\eta) := \max(G(x,\eta),0)$, and $G(x,\eta) = \mathbb{E}[(\eta - g(x,\xi))_+] - \mathbb{E}[(\eta - g(y,\xi))_+]$. Obviously for a fixed x, $G(x,\eta)$ is the difference of two convex functions in η, which means $P(x,\eta)$ is not a convex function in η. Homem-de-Mello and Mehrota [20] tackled this type of challenge with a branch and cut method: reformulating the problem as a DC-programming and then solving it with branch and cut algorithm. Here, we propose to approximate this subgradient through sampling. Let $\xi^1,\ldots,\xi^N$ be a sampling of ξ, and w_{k_i} be a subgradient of $\vartheta(x)$ at x_k. Then we may choose

$$\zeta_k = \frac{1}{N}\sum_{i=1}^{N}(-\nabla f(x_k,\xi^i) + \rho w_{k_i}).$$

Let us explain how to calculate the w_{k_i}. By Dentcheva and Ruszczyński [4, Proposition 3.2], we reformulate the constraints

$$\mathbb{E}[(\eta - g(x,\xi))_+] \leqslant \mathbb{E}[(\eta - g(y,\xi))_+], \quad \forall \eta \in [a,b],$$

as

$$\mathbb{E}[(\eta_i - g(x,\xi))_+] \leqslant \mathbb{E}[(\eta_i - g(y,\xi))_+], \quad i = 1,\ldots,N,$$

where $\eta_i = g(x,\xi^i)$, $i = 1,\ldots,N$. Assume that $g(x,\xi)$ is bounded. Then we may choose the interval $[a,b]$ such that $g(x,\xi) \in [a,b]$ for all $x \in X, \xi \in \Xi$, which means $\eta_i \in [a,b]$, $i = 1,\ldots,N$. Consequently we can reformulate problem (2.2) as follows:

$$\max_{\eta_i} \frac{1}{N}\sum_{i=1}^{N}(\eta_i - g(x_k,\xi^i))_+ - \frac{1}{N}\sum_{i=1}^{N}(\eta_i - g(y,\xi^i))_+. \tag{3.3}$$

Based on the discussions above, we present a stochastic quasi-subgradient algorithm for solving problem (2.6).

Algorithm 3.1.

Step 1. Set a sequence of stepsizes $\{\lambda_k\}$ satisfying

$$\sum_{k=0}^{\infty}\lambda_k^2 < \infty, \quad \sum_{k=0}^{\infty}\lambda_k = \infty, \quad \lambda_k \geqslant 0. \tag{3.4}$$

Choose an initial vector $x_0 \in X$, set $k = 0$.

Step 2. At x_k calculate an approximated subgradient of $\varphi(x,\rho)$, denoted by ζ_k, that is

$$\mathbb{E}[\zeta_k/x_0,\ldots,x_k] \in -\mathbb{E}[\nabla f(x_k,\xi)] + \rho\partial_x\vartheta(x_k) + v_k, \tag{3.5}$$

where $\partial_x\vartheta(x)$ is defined as in Proposition 2.1 and v_k is a controlled error satisfying

$$\sum_{k=0}^{\infty} \mathbb{E}[\lambda_k \|v_k\| + \lambda_k^2 \|\zeta_k\|^2] < \infty. \tag{3.6}$$

Step 3. Set

$$x_{k+1} := P_X(x_k - \lambda_k \zeta_k), \tag{3.7}$$

where $P_X(x)$ is the orthogonal projection of x on X.

Step 4. If $x_{k+1} = x_k$ and $v_k = 0$, stop. Otherwise, set $k := k + 1$, go to *Step 2.*

Let us make a comment on the stopping rule. In the case when $x_{k+1} = P_X(x_k - \lambda_k \zeta_k) = x_k$, we have

$$-\lambda \zeta_k \in \mathcal{N}_X(x_k)$$

and hence

$$-\zeta_k \in \mathcal{N}_X(x_k). \tag{3.8}$$

Since $v_k = 0$, then

$$0 \in -\mathbb{E}[\nabla f(x_k, \xi)] + \rho \partial_x \vartheta(x_k) + \mathcal{N}_X(x_k), \tag{3.9}$$

which, by Remark 2.1, implies that x_k is an optimal solution of (2.6).

Let us now consider the case that $x_k = x_{k_0}$ for $k \geqslant k_0$ but $v_k \neq 0$. By (3.8),

$$0 \in -\mathbb{E}[\nabla f(x_k, \xi)] + \rho \partial_x \vartheta(x_k) + v_k + \mathcal{N}_X(x_k).$$

Under (3.6), $v_k \to 0$ as $k \to \infty$. By taking a limit on the equation above, we have

$$0 \in -\mathbb{E}[\nabla f(x_{k_0}, \xi)] + \rho \partial_x \vartheta(x_{k_0}) + \mathcal{N}_X(x_{k_0})$$

which implies that x_{k_0} satisfies the first order optimality condition and hence x_{k_0} is an optimal solution.

In what follows, we study the convergence of the general case.

Definition 3.1. Let $\|\cdot\|$ denotes the Euclidean norm. A random process $\{x_k\}$ valued in $\mathbb{R}^n$ and adapted to the filtration $\mathcal{F}_k$ is called a random quasi-Feyer sequence w.r.t. a set $S \subseteq \mathbb{R}^n$, if $\mathbb{E}[\|x_0\|] < \infty$, and for any $s \in S$,

$$\mathbb{E}[\|x_{k+1} - s\| / \mathcal{F}_k] \leqslant \|x_k - s\| + \sigma_k$$

and

$$\sum_{k=0}^{\infty} \mathbb{E}[\sigma_k] < \infty, \quad \sigma_k \geqslant 0,$$

where σ_k is an error.

Lemma 3.1 [10, p. 98]. *Let $\{x_k\}$ be a stochastic quasi-Feyer sequence w.r.t. Z. Then the following assertions hold.*

(i) *The sequence $\{\|z - x_k\|^2\}$ converges w.p.1 for any $z \in Z$, and $\mathbb{E}[\|z - x_k\|^2] < C < \infty$ for some constant C.*
(ii) *The set of accumulation points of $\{x_k\}$ is not empty. Suppose that an accumulation point of $\{x_k\}$ belongs to Z. Then $\{x_k\}$ has only one limiting point.*

We are now ready to present our main results.

Theorem 3.1. *Let $\{x_k\}$ be generated by the Algorithm 3.1 and let X^* denote the set of optimal solutions of (2.6). Assume: (a) $f(x, \xi)$ and $g(x, \xi)$ are concave for almost every ξ and continuous w.r.t. both x and ξ, (b) X is a convex compact set, (c) there exists a constant $C > 0$ such that $\mathbb{E}[\|\zeta_k\|^2 / \mathcal{F}_k] \leqslant C$, ζ_k satisfy (3.5), $\{\lambda_k\}$, and v_k satisfy conditions (3.4) and (3.6) w.p.1. Then there is a subsequence $\{x_{k_i}\}$ such that $\{x_{k_i}\} \to x^*$ and $\varphi(x_{k_i}, \rho) \to \varphi(x^*, \rho)$, where $x^* \in X^*$.*

The proof is rather standard and we provide it in the Appendix A for completeness. Let us make a comment on condition (c) of the theorem. If the sequence of the approximate solutions $\{x_k\}$ is contained in a compact set w.p.1 and the controlled error v_k is bounded, then it follows from (3.1) that $\mathbb{E}[\nabla f(x_k, \xi)]$ is bounded. Moreover it follows from Proposition 2.1 that $\vartheta(x)$ is globally Lipschitz continuous with modulus $\mathbb{E}[\kappa(\xi)]$ which implies that $\partial_x \vartheta(x_k)$ is also bounded by $\mathbb{E}[\kappa(\xi)]$.

Before concluding this section, we make a few general comments on stochastic quasi-subgradient method. The stochastic quasi-subgradient method [9–11] has been developed to solve stochastic problems with complicated functions. The main advantage of this method is that, at each iteration, the search direction is a stochastic subgradient of the objective function. Another advantage of stochastic approximation methods is that it allows working directly with the samples of random variables, rather than the full distributions. However, this advantage comes at a cost. One difficulty is the choice of the stepsize. In general, choosing the stepsize requires some experimentation, and there are no hard or fast rules for making the choice.

4. Level function methods

In this section, we consider another approach for solving (2.6). The stochastic approximation method discussed in the previous section uses only one subgradient of the objective function at the current iterate in order to move to the next iterate. From a computational point of view it might be helpful to use some of previous iterates as well as subgradients at the points. This motivates us to apply the so-called level function method to solve (2.6). The fundamental idea of the method is to use a subgradient of the objective function at each iteration to construct a linear function and treat the minimizer of the maximum of the linear function as the next iterate. The method was first proposed by Lemarechal et al. [18] and extended by Xu [34].

Let us start with some basic definition of the method.

Let $\alpha \in \mathbb{R}$ be a scalar and $\varphi(x, \rho)$ be a general continuous function. We use

$$T_\varphi(\alpha) = \{x \in X : \varphi(x, \rho) < \alpha\},$$

to denote the strict lower level set of φ. We discuss the case where the distribution of random variable ξ is known and a subgradient could be calculated based on the available scenarios. This will aid us in the calculation of a subgradient of the objective function at each iteration.

Definition 4.1. Let $\varphi(x, \rho)$ be continuous function and $x \in X$, where X is a nonempty convex subset of $\mathbb{R}^n$. A function $\sigma : \mathbb{R}^n \to \mathbb{R}$ is called a level function of φ at x if it satisfies the following conditions:

(a) $\sigma(x) = 0$,
(b) σ is a continuous convex function,
(c) $T_\varphi(\varphi(x,\rho)) \subset T_\sigma(0)$.

From the definition, we can see that the minimizers of φ are contained in $T_\varphi(\varphi(x, \rho))$ and x is optimal if and only if $T_\varphi(\varphi(x, \rho)) = \emptyset$.

In what follows, we apply the level function method to (2.6). Let $f(x, \xi)$ and $g(x, \xi)$ be continuous convex and concave functions, respectively. Let $\zeta_k \in \partial_x \varphi(x_k, \rho)$, then

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

is a level function of $\varphi(x, \rho)$ at x_k.

Algorithm 4.1.

Step 1. Let $\epsilon > 0$ be a constant, select a starting point $x_0 \in X$; set $k := 0$.

Step 2. Calculate $\zeta_k \in \partial_x \varphi(x_k, \rho)$.
Define the functions $\sigma_{x_k}(x)$ and $\sigma_k(x)$ by

$$\sigma_{x_k}(x) = \zeta_k^T(x - x_k)/\|\zeta_k\|,$$
$$\sigma_k(x) = \max\{\sigma_{k-1}(x), \sigma_{x_k}(x)\},$$

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

$$x_{k+1} \in \arg\min_{x \in X} \sigma_k(x)$$

and

$$\Delta(k) = -\sigma_k(x_{k+1}).$$

Step 3. If $\Delta(k) \leqslant \epsilon$, stop. Otherwise, set $k := k + 1$, go to *Step 2.*

It is important to note that here we need to calculate a subgradient of $\varphi(x, \rho)$ at each iterate. This is more demanding than the stochastic approximation method where only a quasi-subgradient is calculated at each iterate. However, in some practical instances, the random variable may have a finite distribution, in that case $\varphi(x, \rho)$ can be written as a sum of a finite number of deterministic functions. Calculating a subgradient of such a function might be numerically possible. In the case when we are not able to obtain a closed form of the expected value of the underlying functions, we may use the sample average approximation method [31] to approximate $\varphi(x, \rho)$ and reduce it to a finite sum.

Theorem 4.1. *Let $\varphi(x, \rho)$ be defined as in (2.6) and let the assumptions of Theorem 2.1 hold. Then $\lim_{k\to\infty}\Delta(k) = 0$ and there exists a subsequence of $\{x_k\}$ converging to a global minimizer of φ over X.*

Proof. Under the assumptions, each of the level functions $\{\sigma_{x_k}(x)\}$ generated by Algorithm 4.1 is Lipschitz on X. The conclusion follows from [34, Theorem 3.2]. □

The Algorithm 4.1 takes a minimizer of $\sigma_k(x)$ as the next iterate, the main drawback is that it is not possible to predict the maximum number of iterations required to reduce $\Delta(k)$ to a prescribed precision. To overcome this problem, Xu [34] modified the Algorithm 4.1 by updating an iterate using projection of the current point to a level set of $\sigma_k(x)$. This projection idea belongs to Lemarechal et al. [18], who applied it to convex programming.

Algorithm 4.2.

Step 1. Let $\epsilon > 0$ be a constant, and select a constant $\lambda \in (0, 1)$ and a starting point $x_0 \in X$; set $k = 0$.

Step 2. Calculate a level function $\sigma_{x_k}(x)$ of φ at x, and set

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

where $\sigma_{-1} = -\infty$. Let

$$\hat{x_k} = \arg\min\{\varphi(x_j, \rho) : j \in 0, \ldots, k\}$$

and

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

where

$$Q_k := \{x \in X : \sigma_k(x) \leqslant -\lambda\Delta(k)\}, \quad \Delta(k) = -\min_{x \in X} \sigma_k(x)$$

and P_{Q_k} is the Euclidean projection of the point x on a set Q_k.

Step 3. If $\Delta(k) \leqslant \epsilon$, stop. Otherwise, set $k := k + 1$, go to *Step 2.*

Note that, when $\lambda = 1$, Q_k becomes the set of minimizers of σ_k over X. Consequently, Algorithm 4.2 becomes identical to Algorithm 4.1. The following convergence results follow directly from [34, Theorem 3.3].

Theorem 4.2. *Let $\{x_k\}$ be generated by Algorithm 4.2. Assume the conditions of Theorem 4.1. Then*

$$\Delta(k) \leqslant \epsilon, \quad \text{for} \quad k > M^2 d^2 \epsilon^{-2} \lambda^{-2} (1 - \lambda^2)^{-1},$$

where ϵ is specified as in Algorithm 4.2, M is the Lipschitz modulus of φ over X, and d is the diameter of X defined as

$$d = \sup_{x,y}\{\|x - y\|, x, y \in X\}.$$

5. Numerical results

We have carried out a number of numerical tests on the proposed algorithms by using nonlinear programming solver **fmincon** built in MATLAB 7.10 installed on a Viglen PC with Windows XP operating system and 2.96 GB of RAM. In this section, we report the test results.

We consider primarily two portfolio optimization problems to examine the SSD model (1.5) and efficiency of our proposed numerical methods, that is, the penalization approach (2.6) and algorithms discussed in Sections 3 and 4.

Suppose that we have a fixed capital to be invested in n assets. Let R_i, $i = 1, \ldots, n$, denotes the return of asset i. In practice, the return is often uncertain and we use a random variable ξ to describe the uncertainty. Specifically, we write R_i as $R_i(\xi)$ and in doing so we are assuming that all n assets have identical random factor.

To simplify the discussion, we normalize the capital to 1 and use x_i, $i = 1, \ldots, n$, to denote the fraction of capital to be invested in asset i. The portfolio return can then be formulated as:

$$g(x, \xi) := R_1(\xi)x_1 + R_2(\xi)x_2 + \cdots + R_n(\xi)x_n. \tag{5.1}$$

We apply the SSD model (1.5) to optimize our investment strategy. To ease the citation, we repeat the model:

$$\begin{aligned} \min_x \quad & \mathbb{E}[f(x, \xi)], \\ \text{s.t.} \quad & g(x, \xi) \succeq_{(2)} g(y, \xi), \\ & x \in X, \end{aligned} \tag{5.2}$$

where g is defined by (5.1). We need to specify $f(x, \xi)$ and X. We will start with the simplest case of $f(x, \xi) = -g(x, \xi)$ and $X := \{x : \sum_{i=1}^n x_i = 1, x_i \geqslant 0\}$ and then consider a variation which allows f to include a quadratic term and x_i to take a negative value in order to address some practical need where investment in a particular asset is not too small and/or the short selling occurs. We will come to the details of the variations later on. Here y denotes a benchmark investment with $y_i = \frac{1}{n}$, for $i = 1, \ldots, n$.

To examine the appropriateness of the SSD model, we calculate the Conditional Value at Risk (CVaR) for random variables $g(x^*, \xi)$ and $g(y, \xi)$ where x^* is an approximate optimal solution obtained from solving (5.1). Recall that, by definition for a specified probability level α, the Value at Risk (VaR) of a portfolio is the lowest amount C such that, with probability α, the profit does not fall below C. The CVaR_α is the conditional expectation of profit below C. In our context,

$$\text{CVaR}_\alpha(g(x^*, \xi)) = \sup_{\mathrm{C}} \left\{ \mathrm{C} - \frac{1}{\alpha}\mathbb{E}[(\mathrm{C} - g(x^*, \xi))_+] \right\}, \tag{5.3}$$

where $\alpha \in (0, 1)$ is a prespecified constant.

Dentcheva and Ruszczyński [5] showed that there is a fundamental relationship between the concept of CVaR and the second

order stochastic dominance constraint. Specifically they showed that

$$g(x,\xi) \succeq_{(2)} g(y,\xi),$$

if and only if

$$\mathrm{CVaR}_\alpha(g(x,\xi)) \geqslant \mathrm{CVaR}_\alpha(g(y,\xi)), \quad \forall \alpha \in (0,1],$$

which means that as the return of a portfolio increases the CVaR of that portfolio also increases. Three values of α are commonly considered: 0.90, 0.95, 0.99. However, in our analysis we focus on the case of $\alpha = 0.95$.

5.1. Numerical performance

Example 5.1. We consider a history of percentage returns, for $m = 6$ and $m = 10$ time periods, for a group of $n = 2$ and $n = 5$ assets in Tables 5.1 and 5.2, respectively.

Our aim is to find an optimal investment strategy for a fixed capital in the n assets which maximizes the expected profit subject to certain risk averse measures. Particularly we consider the following model:

$$\begin{aligned} \min_x \quad & \mathbb{E}[f(x,\xi)] = -\mathbb{E}[g(x,\xi)], \\ \text{s.t.} \quad & g(x,\xi) \succeq_{(2)} g(y,\xi), \\ & x \in X. \end{aligned} \tag{5.4}$$

For the purpose of this example we set the upper and lower bound on the fraction of capital invested in each asset to 0.6 and 0, respectively. We do this to ensure diversification. Minimizing this function can be regarded as an attempt to get as close as possible to meeting requirements on both return and risk.

We apply the exact penalization as discussed in Section 2 to Examples 5.1 and set the penalty parameter $\rho = 1000$. We solve the reformulated problem with Algorithms 3.1, 4.1, and 4.2. For Algorithm 3.1, we use the step size:

$$\lambda_k = \frac{1}{k}$$

and the stopping rule:

$$\|x_{k+1} - x_k\| \leqslant \delta_x \|x_{k+1}\|, \quad \|\varphi(x_{k+1},\rho) - \varphi(x_k,\rho)\| \leqslant \delta_\varphi \|\varphi(x_{k+1},\rho)\|,$$

where $\delta_x = 0.001$ and $\delta_\varphi = 0.001$ are specified precisions.

For Algorithms 4.1 and 4.2 we use $\epsilon = 0.001$ and $\lambda = 0.5$.

Consider Example 5.1, we apply the Algorithms 3.1, 4.1 and 4.2 to solve this problem. The optimal fractions of the invested capital from the starting point $x_0 = (1,0)^T$ are shown in Table 5.3.

Table 5.1
Monthly rates of return on two assets.

	Returns % for period					
	January	February	March	April	May	June
Asset 1	1.2	1.3	1.4	1.5	1.1	1.2
Asset 2	1.3	1.0	0.8	0.9	1.4	1.3

Table 5.2
Rates of return on five assets over ten periods.

	Returns % for period									
	1	2	3	4	5	6	7	8	9	10
Asset 1	1.2	1.3	1.4	1.5	1.1	1.2	1.1	1.0	1.0	1.1
Asset 2	1.3	1.0	0.8	0.9	1.4	1.3	1.2	1.1	1.2	1.1
Asset 3	0.9	1.1	1.0	1.1	1.1	1.3	1.2	1.1	1.0	1.1
Asset 4	1.1	1.1	1.2	1.3	1.2	1.2	1.1	1.0	1.1	1.2
Asset 5	0.8	0.75	0.65	0.75	0.8	0.9	1.0	1.1	1.1	1.2

Table 5.3
Example 5.1 using data in Table 5.1. Time is in minutes, the expected return of the benchmark portfolio $\mathbb{E}[g(y,\xi)] = 1.200$ and its CVaR = 0.897.

Algorithm	Iter.	Time	x	$\mathbb{E}[g(x,\xi)]$	CVaR
3.1	6	0.845	(0.600, 0.400)	1.217	0.9824
4.1	6	0.631	(0.599, 0.401)	1.218	0.9824
4.2	6	0.628	(0.599, 0.401)	1.218	0.9824

In order to investigate the accuracy of the solution, we calculated the norm of the subgradient at this solution. The norm of subgradient at $x = (0.6, 0.4)$ is equal to 0.0068 which confirms that the solution is close to optimal.

In Table 5.3 and the rest of the tables "Iter" refers to the number of iterations, "Alg" is the short form for algorithm, and "S-sell" refers to short selling.

The results are obtained after 6 iterations by Algorithms 4.1 and 4.2 and are equivalent to the results obtained by Algorithm 3.1. Additionally, as it is expected both the return and its CVaR of the selected portfolio are higher than the benchmark return and CVaR.

Further, we use the data of 5 assets over 10 periods in Table 5.2 and examine Example 5.1. Table 5.4 shows the results of this example.

As it can be seen, all three algorithms result in a similar optimal portfolio. Also the selected portfolio dominates the benchmark portfolio in a sense that the CVaR of the selected portfolio is greater than that of the benchmark portfolio. However, the number of iterations is different and as a result the computation times differ. Algorithm 4.2 performed better than the other algorithms as it converged to the optimal portfolio with the least number of iterations.

Note that in Example 5.1, both f and g are linear. In what follows, we consider nonlinear portfolio optimization problems where either f or g or both are nonlinear. This is to demonstrate that the proposed algorithms can cope with both linear and nonlinear portfolio optimization problems.

Example 5.2. In Example 5.1 we considered an optimization problem where any fraction of capital between 0 and 0.6 was acceptable. However, due to transaction cost, investors do not like to invest very small amount of their capital in different assets. We now reformulate Example 5.1 into a slightly more complicated problem in which we do not want to invest very small amounts in an asset. We consider the following performance function:

$$f(x,\xi) = -g(x,\xi) - \sum_{i=1}^{n} x_i^2 \tag{5.5}$$

and incorporate (5.5) into the optimization problem (5.4). In the section we will consider two cases:

- Short-selling is allowed and upper and lower bounds on the fraction of capital invested in each asset are set to 2 and −1.
- Short-selling is prohibited and the bounds are set to 0.6 and 0 to ensure diversification.

Table 5.4
Example 5.1 using data in Table 5.2. Time is in minutes, the expected return of the benchmark portfolio $\mathbb{E}[g(y,\xi)] = 1.093$ and its CVaR = 0.895.

Alg.	Iter.	Time	x	$\mathbb{E}[g(x,\xi)]$	CVaR
3.1	115	5	(0.325, 0.231, 0.177, 0.266, 0)	1.147	1.004
4.1	7	0.8931	(0.322, 0.231, 0.177, 0.266, 0)	1.148	1.004
4.2	5	0.6355	(0.325, 0.231, 0.177, 0.266, 0)	1.148	1.004

Table 5.5
Example 5.2 using data in Table 5.2. Time is in minutes, the expected return of the benchmark portfolio $\mathbb{E}[g(y,\xi)] = 1.200$ and its CVaR = 0.897.

Alg.	Problem	Iter.	Time	x	$\mathbb{E}[g(x,\xi)]$	CVaR
3.1	S-sell	684	16	$(0.127, 0.495, 0.550, 0.380, -0.553)$	1.230	1.045
	No S-sell	226	9	$(0.600, 0, 0, 0.400, 0)$	1.1740	1.007
4.1	S-sell	6	0.899	$(0.400, 0.514, 0.314, 0.390, -0.500)$	1.260	1.063
	No S-sell	6	0.768	$(0.600, 0, 0, 0.400, 0)$	1.1740	1.007
4.2	S-sell	4	0.649	$(0.39, 0.527, 0.287, 0.3, -0.5)$	1.260	1.063
	No S-sell	4	0.577	$(0.600, 0, 0, 0.400, 0)$	1.1740	1.007

Table 5.6
Example 5.2 using FTSE100 historical return. Time is in minutes, "No. assets" represent the number of assets in the optimal portfolio. The expected return of the benchmark portfolio $\mathbb{E}[g(y,\xi)] = -0.051$ and its CVaR = 0.023.

Alg.	Problem	Iter.	Time	No. assets	$\mathbb{E}[g(x,\xi)]$	CVaR
3.1	S-sell	792	28.542	32	0.089	0.079
	No S-sell	685	25.43	27	0.036	0.025
4.1	S-sell	16	1.274	40	0.094	0.082
	No S-sell	13	0.973	26	0.037	0.026
4.2	S-sell	12	0.640	40	0.094	0.082
	No S-sell	10	0.561	26	0.037	0.026

The invested fractions which solve this problem using the discussed algorithms and the data in Table 5.1 are the same as the result obtained in Example 5.1. However, the results of the problem using data in Table 5.2 are shown in Table 5.5.

Furthermore, we collect 300 daily historical returns of 95 FTSE100 assets prior to March 2011 and a cash account paying 5% interest. We used the first 200 daily return to construct the portfolio strategy based on the performance function (5.5). We also used the FTSE100 Index as the benchmark. The results are presented in Table 5.6.

It can be seen that when short-selling is allowed the optimal portfolio has higher return compared to the case where short-selling is prohibited, however this higher return is associated with a higher risk. A rational risk-averse investor is expected to discourage short-selling as the excess return is not worth the extra risk. Additionally, the financial authorities in many countries including the UK and the USA restrict many financial institutions such as pension funds from the practice of short-selling.

Note that both of the level function algorithms converge to very similar portfolios. The portfolio returns are higher than the benchmark portfolio return and as it was expected the CVaR of the selected portfolios are higher than the CVaR of benchmark portfolio. Although the fraction of the capital invested in each asset differ from the results from SA method, but the optimal portfolios return and risk are very close. The number of iterations in the level function method are much lower compared to the stochastic approximation method, consequently the optimization time is lower for the level function method. Further it could be seen that the projected level function algorithm converges to the optimal solution with fewer number of iterations compared to the scaled level function algorithm. This makes Algorithm 4.2 more attractive than Algorithm 4.1.

In next section, we demonstrate the advantage of taking stochastic dominance constraints into account using real world data for a portfolio optimization problem followed with a bakctest and a out-of-sample analysis.

5.2. Portfolio performance

In this section we focus on optimization problem (5.4) where $g(y,\xi)$ is set to be equal to FTSE100 Index and draw some conclusions.

Table 5.7
Result of the problem using FTSE100 data. Time is in minutes, No. assets represents the number of assets in the optimal portfolio. The expected return of the benchmark portfolio $\mathbb{E}[g(y,\xi)] = -0.051$ and its CVaR = 0.023.

Algorithm	Iter.	Time	No. assets	$\mathbb{E}[g(x,\xi)]$	CVaR
3.1	735	25.64	31	0.036	0.027
4.1	9	1.717	29	0.037	0.027
4.2	7	1.215	29	0.037	0.027

We use the FTSE100 data collected and a cash account paying 5% interest. We used the first 200 daily return to construct the portfolio strategy and the further 100 daily return for an out-of-sample test. In practice there are many strict regulations imposed by authorities on short selling and as a result many financial institutions prohibit any short selling activity. Consequently, in this example we only consider the case where short selling is not allowed and set the upper and lower bounds on portion of capital invested at 0.6 and 0, in order to ensure diversification.

We solve the above optimization problem using the FTSE100 data and compare the proposed algorithms. The results are presented in Table 5.7.

As it is explained in Section 4, Algorithm 4.1 takes a minimizer of $\sigma_k(x)$ as the next iterate, the main drawback is that it is not possible to predict the maximum number of iterations required to reduce $\Delta(k)$ to a prescribed precision. To overcome this problem, the projected level function method was introduced. In the remainder of this section we concentrate on the investigating the efficiency of selected portfolio by Algorithms 3.1 and 4.2.

To illustrate the benefit of using stochastic dominance constraints, we set up a backtest which is a key component of effective trading-system development in finance. It is accomplished by reconstructing, with historical data, trades that would have occurred in the past using rules defined by a given strategy. Furthermore, we set up an out-of-sample test to evaluate the performance of the selected portfolio over the remaining 100 samples. For the backtest the model finds the optimal portfolio weights from 200 historical market data, then the portfolio strategy is applied to the same data and daily portfolio return is calculated for each day (Fig. 5.1). In the out-of-sample test, the same portfolio strategy is applied to the remaining data of 100 days and the portfolio return is again calculated for each day (Fig. 5.2). In both tests the

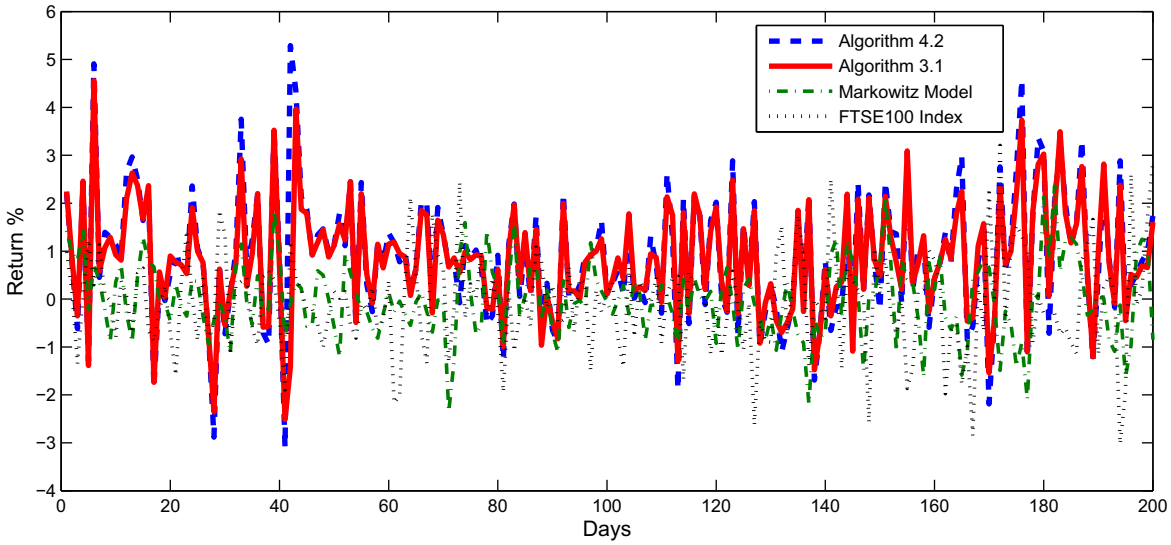

Fig. 5.1. Backtesting of the selected portfolios and comparison with FTSE100 Index and Markowitz model.

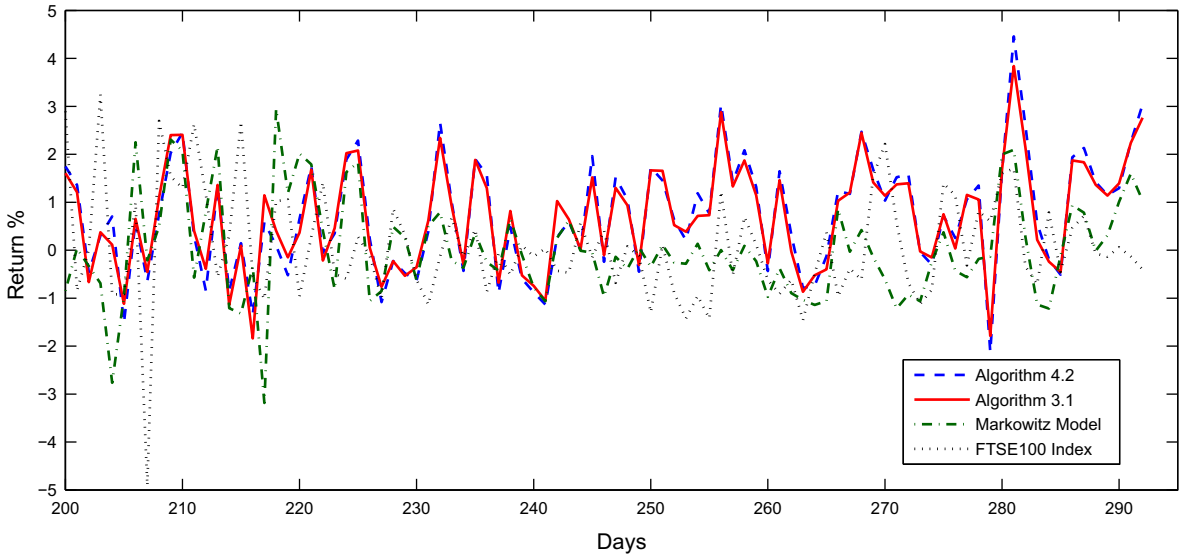

Fig. 5.2. Out-of-sample test for the selected portfolios and comparison with FTSE100 Index and Markowitz model.

portfolio performance is compared with FTSE100 Index and an investment strategy generated by a Markowitz model as described below:

$$\begin{aligned} \max_{x\in X} \quad & \mathbb{E}[g(x,\xi)] - \lambda \mathbb{E}[R(x,\xi)], \\ \text{s.t.} \quad & \mathbb{E}[g(x,\xi)] \geqslant R_b, \\ & \sum_{i=1}^{n} x_i = 1, \quad x \geqslant 0,\ x \in X, \end{aligned} \tag{5.6}$$

where $\lambda = 1$ is a fixed nonnegative number, $\mathbb{E}[R(x,\xi)]$ is the portfolio variance, R_b is the benchmark return set equal to the FTSE100 Index, and $\mathbb{E}[g(x,\xi)]$ is the return defined as in (5.1). The Markowitz model (5.6) assumes that portfolio can be characterized by their mean return and variance.

It can be seen that the portfolios constructed by the SSD model (5.4) and solved by proposed Algorithms 4.2 and 3.1 performs better than the Markowitz model (5.6) and a FTSE100 Index. Note that the two algorithms generate similar results, but their numerical efficiency differ significantly in terms of CPU time and the number of iterations. Fig. 5.3 shows the CPU time of different numbers of assets for Algorithms 3.1 and 4.2.

5.3. Conclusions

Our preliminary numerical tests show that Algorithm 4.2 (projected level function) is numerically more efficient than Algorithm 3.1 (Stochastic approximation). However, Algorithm 3.1 has a unique advantage; that is at each iteration only one approximated subgradient of the objective function is calculated.

Furthermore, the portfolio optimization problem with SSD constraints performs better than the Markowitz model and it also outperforms the benchmark both in-sample and out-of-sample in sense of portfolio return, which was shown by the results from the backtest and out-of-sample test (Figs. 5.1 and 5.2).

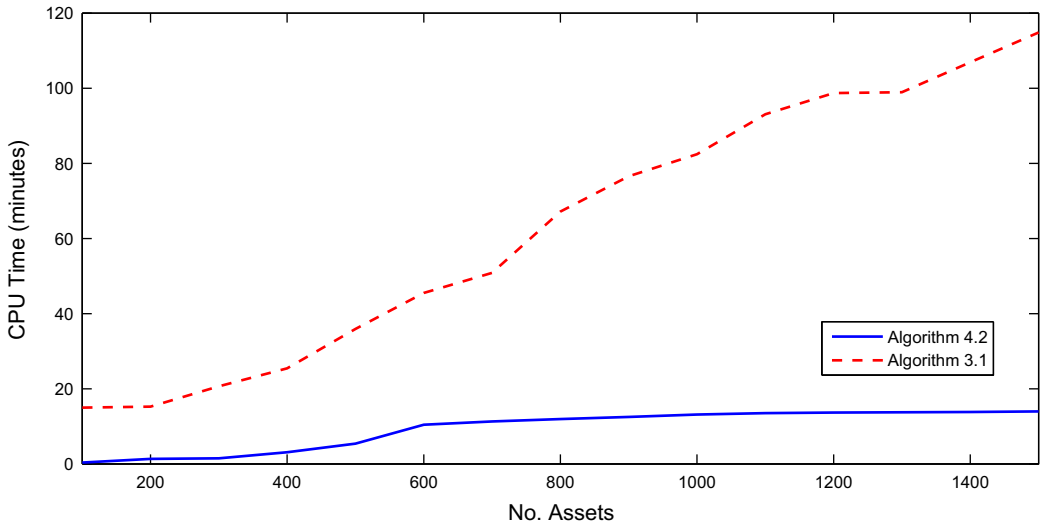

Fig. 5.3. CPU Time in minutes versus the number of instruments.

Appendix A

Proof of Theorem 2.1. Let $d(x,D) := \inf_{x' \in D}\|x - x'\|$ denotes the distance from a point x to a set D. Under the Slater condition, it follows by Lin and Xu [22, Lemma 2.5] that there exists a constant $\beta > 0$ such that

$$d(x,\mathcal{F}) \leqslant \beta \|\mathbb{E}[G(x,\eta)]_+\|_\infty, \quad \forall x \in X, \tag{6.7}$$

where $\mathcal{F}$ denotes the feasible set of problem (2.6). Let C denote the Lipschitz modulus of $\mathbb{E}[f(x,\xi)]$. By Clarke [3, Proposition 2.4.3] for any $\rho > \beta C$, the two optimal solutions of problem (1.5) and (2.6) coincide. Note that under the Slater condition assumption, we can set an $C = \mathbb{E}[\kappa(\xi)]$. This shows the existence of a positive constant $\bar{\rho} := \beta C$. The proof is complete. □

Proof of Theorem 3.1. Let $\{x_k\}$ be generated by (3.7) and $x^* \in X^*$. By definition

$$\begin{aligned}\|x^* - x_{k+1}\|^2 &= \|x^* - P_X(x_k - \lambda_k\zeta_k)\|^2 \leqslant \|x^* - x_k + \lambda_k\zeta_k\|^2 \\ &= \|x^* - x_k\|^2 + 2\lambda_k\zeta_k^T(x^* - x_k) + \lambda_k^2\|\zeta_k\|^2. \end{aligned}\tag{6.8}$$

Let $\mathcal{F}_k = \{x_1, \ldots, x_k\}$. Taking the conditional expectation on both sides of the above inequality w.r.t. $\mathcal{F}_k$, we have

$$\begin{aligned}\mathbb{E}[\|x^* - x_{k+1}\|^2/\mathcal{F}_k] \leqslant \mathbb{E}[\|x^* - x_k\|^2] + 2\lambda_k\mathbb{E}[\zeta_k/\mathcal{F}_k]^T(x^* - x_k) \\ + \lambda_k^2\mathbb{E}[\|\zeta_k\|^2/\mathcal{F}_k].\end{aligned}$$

Observe from (3.5) that $\mathbb{E}[\zeta_k/\mathcal{F}_k] - v_k \in \partial_x\varphi(x_k,\rho)$. By the convexity of $\varphi(x,\rho)$, we have

$$\mathbb{E}[\zeta_k/\mathcal{F}_k]^T(x^* - x_k) - v_k^T(x^* - x_k) \leqslant \varphi(x^*,\rho) - \varphi(x_k,\rho). \tag{6.9}$$

Using conditions (b) and (c) and taking into account that $\varphi(x^*,\rho) - \varphi(x_k,\rho) \leqslant 0$, we obtain from the above two inequalities that

$$\mathbb{E}[\|x^* - x_{k+1}\|^2/\mathcal{F}_k] \leqslant \|x^* - x_k\|^2 + \widetilde{C}\left(\lambda_k\|v_k\| + \lambda_k^2\|\zeta_k\|^2\right),$$

where $\widetilde{C}$ is a constant.

In view of (3.6) and Definition 3.1, it is clear that $\{x_k\}$ is a stochastic quasi-Feyer sequence w.r.t. X^*. Consequently, the sequence $\|x_k - x^*\|^2 \to 0$ w.p.1. Furthermore, the set of accumulation points of $\{x_k\}$ is not empty. Consequently, if we show that one of the accumulation points belongs to X^*, then from condition (c) it follows that $\{x_k\}$ converges w.p.1. to a point in X^* [10].

Referring back to (6.8) and taking expectations, we have

$$\begin{aligned}\mathbb{E}[\|x^* - x_{k+1}\|^2] \leqslant \mathbb{E}[\|x^* - x_1\|^2] + 2\sum_{i=1}^k \lambda_i\mathbb{E}[\zeta_i/\mathcal{F}_i]^T(x^* - x_i) \\ + \sum_{i=1}^k \mathbb{E}[\lambda_i^2\|\zeta_i\|^2/\mathcal{F}_i],\end{aligned}$$

through (6.9), this yields

$$\begin{aligned}\mathbb{E}[\|x^* - x_{k+1}\|^2] &\leqslant \mathbb{E}[\|x^* - x_1\|^2] + 2\sum_{i=1}^k \lambda_i\mathbb{E}[\varphi(x^*,\rho) - \varphi(x_i,\rho) \\ &\quad + \|v_i\|\|x_i - x^*\|] + \sum_{i=1}^k \mathbb{E}[\lambda_i^2\|\zeta_i\|^2/\mathcal{F}_i], \\ &\leqslant \mathbb{E}[\|x^* - x_1\|^2] + 2\sum_{i=1}^k \lambda_i\mathbb{E}[\varphi(x^*,\rho) - \varphi(x_i,\rho)] \\ &\quad + \widetilde{C}\sum_{i=1}^k \mathbb{E}\left[\lambda_i\|v_i\| + \lambda_i^2\|\zeta_i/\mathcal{F}_i\|^2\right].\end{aligned}$$

This and condition (3.6) imply

$$\sum_{i=1}^\infty \lambda_i\mathbb{E}[\varphi(x_i,\rho) - \varphi(x^*,\rho)] < \infty.$$

Since

$$\sum_{i=1}^\infty \lambda_i = \infty \quad \text{and} \quad \varphi(x_i,\rho) - \varphi(x^*,\rho) \geqslant 0,$$

then there exists a subsequence x_{k_i} such that

$$\varphi(x_{k_i},\rho) - \varphi(x^*,\rho) \to 0, \quad \text{w.p.1.}$$

This shows there exists a subsequence such that $\|x_{k_i} - x^*\| \to 0$ w.p.1 and this completes the proof. □

References

[1] B.M. Baker, J. Sheasby, Accelerating the convergence of subgradient optimization, European Journal of Operational Research 117 (1999) 136–144.
[2] M. Branda, Reformulation of General Chance Constrained Problems Using the Penalty Functions, vol. 2, Stochastic Programming E-print Series, 2010.
[3] F.H. Clarke, Optimization and Non-smooth Analysis, Wiley, New York, 1983.
[4] D. Dentcheva, A. Ruszczyński, Optimization with stochastic dominance constraints, SIAM Journal of Optimization 13 (2003) 548–566.
[5] D. Dentcheva, A. Ruszczyński, Portfolio optimization with stochastic dominance constraints, Journal of Banking and Finance 30 (2007) 433–451.
[6] D. Dentcheva, A. Ruszczyński, Composite semi-infinite optimization, Control and Cybernetics 36 (2007) 634–646.
[7] D. Dentcheva, A. Ruszczyński, Optimization with multivariate stochastic dominance constraints, Mathematical programming Series B 117 (2009) 111–127.
[8] J. Dupačová, A. Gaivoronski, Z. Kos, T. Szantai, Stochastic programming in water management: A case study and a comparison of solution techniques, European Journal of Operational Research 52 (1991) 28–44.
[9] Y. Ermoliev, Random optimization and stochastic programming, Lecture Notes in Mathematics 112 (1970) 104–115.
[10] Y. Ermoliev, Stochastic Programming Methods, Nauka, Moscow, 1976.
[11] Y. Ermoliev, Stochastic quasi-gradient methods and their application to system optimization, Stochastics 9 (1983) 1–36.
[12] Y. Ermoliev, R.J.B. Wets, Numerical Techniques for Stochastic Optimization, Springer-Verlag, Berlin, 1988.
[13] Y. Ermoliev, V.I. Norkin, On nonsmooth and discontinuous problems of stochastic systems optimization, European Journal of Operational Research 101 (1997) 230–244.
[14] C.I. Fábián, G. Mitra, D. Roman, Processing second order stochastic dominance models using cutting-plane representations, Mathematical Programming Series A (2009) 1–25.
[15] P.C. Fishburn, Decision and Value Theory, John Wiley & Sons, New York, 1964.
[16] S. Gustafson, Investigating semi-infinite programs using penalty function and lagrangian methods, Australian Mathematical Society, Series B 28 (1986) 158–169.
[17] T. Larsson, M. Patriksson, A. Strömberg, Conditional subgradient optimization – Theory and application, European Journal of Operational Research 88 (1996) 382–403.
[18] C. Lemarechal, A. Nemirovskii, Y. Nestrov, New variants of bundle methods, Mathematical Programming 69 (1995) 111–147.
[20] T. Homem-de-Mello, S. Mehrota, A cutting surface method for uncertain linear programs with polyhedral stochastic dominance constraints, SIAM Journal of Optimization 20 (2009) 1250–1273.
[22] Y. Lin, H. Xu, Stability and sensitivity analysis of stochastic programs with second order dominance constraints, Stochastic Programming E-Print Series. <http://edoc.hu-berlin.de/docviews/abstract.php?id=36988>, 2010.
[25] W. Ogryczak, A. Ruszczyński, From stochastic dominance to mean-risk models: semideviations as risk measures, European Journal of Operational Research 116 (1999) 33–50.
[26] B.T. Poljak, Nonlinear programming methods in the presence of noise, Mathematical programming 14 (1978) 87–97.
[27] H. Robbins, S. Monro, A stochastic approximation method, Annual Mathematical Statistics 22 (1951) 400–407.
[28] R.T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, NJ, 1970.
[29] D. Roman, K. Darby-Dowman, G. Mitra, Portfolio construction based on stochastic dominance and target return distributions, Mathematical Programming 108 (2000) 541–569.
[30] A. Ruszczyński, A. Shapiro, Stochastic Programming, Handbook in Operations Research and Management Science, Elsevier, 2003.
[31] A. Shapiro, Monte Carlo sampling methods, in: Stochastic Programming, Handbook in Operations Research and Management Science, Stochastic Programming, vol. 10, 2003, pp. 353–425.
[33] G.A. Whitmore, M.C. Findlay, Stochastic Dominance: An Approach to Decison-Making Under Risk, D.C. Health, Lexington, MA, 1978.
[34] H. Xu, Level function method for quasiconvex programming, Journal of Optimization Theory and Applications 108 (2001) 407–437.