An iterative method for solving a bi-objective constrained portfolio optimization problem

Madani Bezoui, Mustapha Moulaï, Ahcène Bounceur, Reinhardt Euler

Noname manuscript No.
(will be inserted by the editor)

An iterative method for solving a bi-objective constrained portfolio optimization problem

Madani Bezoui · Mustapha Moulaï · Ahcène Bounceur · Reinhardt Euler

Received: date / Accepted: date

Abstract In this work, we consider the problem of portfolio optimization under cardinality and quantity constraints. We use the standard model of mean-variance in its bi-objective form which is presented here as a bi-objective quadratic programming problem under cardinality and quantity constraints. This problem is NP-hard, which is why the majority of methods proposed in the literature use metaheuristics for its resolution. In this paper, we propose an iterative method for solving constrained portfolio optimization problems. Experiments are performed with major market indices, such as the Hang Seng, DAX, FTSE, S&P 100, Nikkei, S&P 500 and Nasdaq using real-world datasets involving up to 2196 assets. Comparisons with two exact methods and a metaheuristic are performed. These results show that the new method allows to find efficient portfolio fronts in reasonable time.

Keywords Cardinality and Quantity constraints · Cardinality portfolio selection · Bi-objective programming · Mixed integer programming · Steepest descent method · Pascoletti-Serafini method

Mathematics Subject Classification (2000) 90C29 · 90C90 · 90C11 · 90B50 · 91B28

M. Bezoui
University M'hamed Bougara of Boumerdes, 35000.
LaROMad, Faculty of Mathematics, (USTHB),
BP 32, Bab Ezzouar 16111, Algiers, Algeria.
E-mail: mbezoui@univ-boumerdes.dz

M. Moulaï
LaROMad, Faculty of Mathematics, (USTHB),
BP 32, Bab Ezzouar 16111, Algiers, Algeria.

A. Bounceur
Lab-STICC - UMR CNRS 6285, UBO, France.

R. Euler
Lab-STICC - UMR CNRS 6285, UBO, France.

1 Introduction

Portfolio optimization is an important area of finance, and the most significant advance in this field during the last century is due to the Nobel price laureate Harry Markowitz. In his seminal work [34], which outlines the basics of modern portfolio theory (MPT), Markowitz has modeled the portfolio selection problem as a problem of mean-variance optimization with two criteria: maximize return (estimated by the mean), and minimize portfolio risk (measured by variance). These criteria must be optimized simultaneously, that is why in this work we use a bi-objective model.

Markowitz' mean-variance theory (MV) provides a classic solution to the portfolio selection problem. The risk of a portfolio can be reduced by combining assets with imperfectly correlated returns [21]. After the work of Markowitz [34], several studies tried to improve and extend the standard MV model in three directions: simplification of the type and the quantity of input data [6,18], implementation of other risk measures [2,41] and integration of real-world constraints. Our work deals with this latter direction and will be detailed in the next paragraph.

However, to have a portfolio in real-world conditions, we must consider many constraints, such as segmentation [15], cardinality [11,19,27,32,38,42, 46], quantity [32,33,35], pre-assignment [22,32], round lot [31], logical and budget [45], etc. In this work, we consider the cardinality and the quantity constraints.

The original Markowitz problem is NP-hard [7,37,44]. It becomes even more difficult by adding cardinality constraints since it can be formulated as a mixed integer quadratic optimization problem which can be solved by exact methods only for small instances. That is why most of the suggested solution methodologies in the literature that tackle discrete features in portfolio selection use approximative algorithms, like for example: Genetic algorithms [3, 11,46], Ant Colony [26] and Particle Swarm [13]. A comprehensive review of heuristic methods for solving portfolio selection problems can be found in [1, 13,29,47].

Several methods are based on scalarization of the multiobjective optimization problem. These methods consist in transforming the initial problem with several objective functions into a singleobjective optimization problem, which requires adding parameters and constraints. The well-known scalarization methods are the weighted sum method [23], the ε-constraint method [36], the Pascoletti-Serafini method [40], etc. In this work, we propose a new scalarization technique for a portfolio optimization problem in its bi-objective form, which consists of supporting only the linear function (return) in the optimization process and adding a constraint to support the other function. Unlike the ε-constraint method, in this technique, we do not find the second function in the constraints explicitly, but we use its direction of improvement.

In the literature, only a few number of exact methods is dealing with the problem considered in this paper. Borchers and Mitchel [8] propose an improved branch-and-bound algorithm for solving a mixed integer 0-1 pro-

gramming problem with convex objective functions and constraints. Bienstock presents a computational experience with a branch-and-cut algorithm to solve a quadratic programming problem [7], in [28], Lee and Mitchell introduce an interior-point algorithm within a parallel branch-and-bound framework for solving mixed integer nonlinear programs. A convergent Lagrangian and a contour-domain cut method are proposed by Li et al. [30] for solving constrained portfolio selection problems by exploiting some prominent features of the mean-variance formulation and the portfolio model under consideration. Frangioni and Gentile proposed an improved branch-and-cut method based on the perspective cut technique [20].

In [44], Shaw et al. developed a Lagrangian relaxation method dedicated to solve cardinality-constrained quadratic programming problems. In [48], Vielma et al. propose a branch-and-bound procedure based on a lifted polyhedral relaxation of conic quadratic constraints. In [5], Bertsimas and Shioda introduced a branch-and-bound procedure in which the continuous relaxation of subproblems is solved using Lemke's pivoting technique. In [24], Gulpinar et al. proposed an exact solution method based on the difference of convex functions to solve cardinality constrained portfolio optimization problems.

To our knowledge, all existing exact methods are gourmand in memory space, and they are not useful for solving large-scale problem instances. According to [16], no efficient algorithmic solution to this issue has been proposed. In this work, we propose an iterative method, which can find the frontier of efficient solutions for large-scale constrained portfolio problems in a reasonable time.

After this introduction, we present the model considered in this work. In Section 3, we give some definitions and a theoretical basis for the proposed method. In Section 4, we present the method and in Section 5, its validation by theoretical and numerical results. Section 6 contains our conclusions and proposes future work.

2 Model formulation

This paper is focused on a model that more accurately reflects the real setting, and which is, of course, harder to solve than the standard Markowitz model. We include the cardinality restriction, which limits a portfolio to hold a designated number of assets:

$$\begin{cases} \min x^T \sigma x & \text{minimize risk} \\ \max r^T x & \text{maximize return} \\ \text{s.t. } e^T x = 1 & \text{budget constraint} \\ \quad 0 \leq x_i \leq 1 & \forall i \in \{1, .., n\}, \\ \quad y_i = \begin{cases} 1 & \text{if } x_i > 0 \\ 0 & \text{if } x_i = 0 \end{cases} & \forall i \in \{1, .., n\}, \\ \quad e^T y = K & \text{cardinality contraint,} \end{cases} \tag{1}$$

where

x_i is the decision variable representing the proportion held of asset i,
n is the number of available assets,
σ called "risk of portfolio", is a symmetric $n \times n$-matrix, each component σ_{ij} representing the covariance between assets i and j,
r , called expected "return of portfolio", is an n-vector, each component r_i of which represents the expected return of asset i,
e is an n-vector of 1's.
K is the number of selected assets in a portfolio.

Let a and b be two n-vectors of, respectively, floor and ceiling limits to invest in each asset i, if it is selected. Then we can formulate our problem as follows:

$$\begin{cases} \min x^T \sigma x \\ \max r^T x \\ \text{s.t. } e^T x = 1 \\ \quad a_i y_i \leq x_i \leq b_i y_i \ \forall i \in \{1,..,n\}, \text{ quantity constraint} \\ \quad e^T y = K \\ \quad y_i \in \{0,1\} \qquad \forall i \in \{1,..,n\}, \end{cases} \tag{2}$$

where $0 < a_i < b_i < 1, \forall i \in \{1,..,n\}$.

Problem (2) is called the *cardinality and quantity constrained portfolio optimization problem*, which is a bi-objective mixed integer quadratic programming problem, shown in [7] to be NP-hard.

3 Preliminaries

3.1 Multiobjective optimization

Multiobjective optimization is the search for compromise solutions to simultaneously optimize more than one objective function. It is clear that this problem is mathematically harder to solve than singleobjective optimization problems, but it is important to express the real problems as faithful as possible. In this work, we consider the multiobjective portfolio selection model. Before we start to present our method, we give some definitions and notations.

Let us consider the following multiobjective optimization problem:

$$\begin{cases} \text{"max"} \ F(x) = (f_1(x), f_2(x), .., f_m(x)) \\ \quad s.t. \quad x \in X \end{cases} \tag{3}$$

where:

$X \subset \mathbb{R}^n$ is the set of feasible points, such that $X \neq \emptyset$,
$f_j : X \longmapsto \mathbb{R}$, $j = \overline{1,m}$, are m objective functions.

For any two vectors u and $v \in \mathbb{R}^m$ we say that:

u *dominates* v, denoted as "$u \succeq v$", if and only if $u_j \geq v_j, \forall j \in \{1,2,..,m\}$ such that $\exists j^0 \in \{1,2,..,m\}, u_{j^0} > v_{j^0}$,

u strictly dominates v, denoted as "$u \succ v$", if and only if $u_j > v_j, \forall j \in \{1, 2, .., m\}$.

The concept of efficiency is very important in multiobjective optimization. The most preferred solution with respect to different objectives is said *"Pareto optimal"* or "efficient". A vector $x \in X$ is *efficient*, if there exists no vector $y \in X$, such that $F(y) \succeq F(x)$. A vector $x \in X$ is *weakly efficient*, if there exists no vector $y \in X$, such that $F(y) \succ F(x)$.

Practitioners do not prefer weakly efficient solutions, because the image of such solutions in the criteria space may be dominated by other feasible solutions, which is not desirable in the real world.

3.2 Epsilon-Constraint Methods

One of the well-known methods for solving multiobjective optimization problems is the ε-constraint method, which was introduced by Haimes [25] and extended by Chankong [12]. The principle of this approach is to optimize only one objective function; the other objective functions will be considered as new constraints.

Let us consider Problem (3). Then the corresponding ε-constraint problem is the following:

$$\begin{cases} \max f_i(x) & \\ s.t. \quad x \in X & \text{Original constraints} \\ \qquad f_j(x) \geq \varepsilon_j, \forall j = \overline{1, m} \ \ \& \ \ j \neq i & \end{cases} \tag{4}$$

where

ε_j represent lower limits that are often designated by the decision maker and which represent thresholds from which he accepts the solution.

By parametrical variation of the right-hand side (RHS) ε in Problem (4) corresponding to the new constraints, we obtain the front of efficient solutions.

Theorem 1 *The solution of the ε-constraint problem (4) is weakly efficient [36].*

Theorem 2 *A point x^* is efficient if it is the unique solution of the ε-constraint problem (4) for some $i \in 1, .., k$ with $\varepsilon_j = f_j(x^*)$ for $j = 1, .., k, j \neq i$ [36].*

4 An iterative method to generate non-dominated solutions

4.1 General description of the proposed algorithm

The method presented in this work can be seen as a variant of the ε-constraint method. Indeed, throughout the resolution process, we consider exclusively the linear function (Problem 5) and we add a constraint that serves as an upper bound to the second function (Elastic constraint). As this bound evolves

according to the solution found in step t, this allows us to minimize the risk according to the same principle as the ε-constraint method.

At each iteration, we solve the mixed integer linear programming problem $MILP_t$, presented as follows:

$$(MILP_t) : \begin{cases} \max r^T x \\ s.t. \quad e^T x = 1 \\ \qquad a_i y_i \leq x_i \leq b_i y_i, \forall i \in \{1, .., n\} \\ \qquad e^T y = K \\ \qquad y_i \in \{0, 1\} \qquad \forall i \in \{1, .., n\} \\ \qquad x \leq \varepsilon_{t-1} \qquad \text{Elastic constraint} \end{cases} \tag{5}$$

where:

ε_t represents the upper bound of the selected assets (how to determine this upper bound (elastic constraint) will be explained in the following section).

The optimization process stops as soon as a solution step x_t satisfies one of the following conditions:

1. $\|\sigma.x_t\|_2 \leq \xi$.
2. $\|x_{t+1} - x_t\|_2 \leq \xi$.

where ξ is a tolerance chosen close to 0; in our case, we put $\xi = 10^{-3}$.

4.2 Construction of an efficiency cut

At each iteration, we solve the problem to maximize the expected return, under the constraints described in Section 2. The second objective function intervenes in the form of an elastic constraint that follows the direction of improvement of the risk. It is calculated by applying the steepest descent method.

Our goal is to form a constraint to achieve the following inequality:

$$f(x_{t+1}) \leq f(x_t). \tag{6}$$

Let us consider the quadratic programming problem:

$$\min_{x \in \mathbb{R}^n} f(x) = x^T \sigma x \tag{7}$$

where

σ is a positive and symmetric $n \times n$-matrix.

Now, let $x_{t+1} = x_t + \alpha_t d_t$.

The direction d: We use the direction of the steepest descent method $d_t = -\nabla f(x) \neq 0$, which is a descent direction because:

$$(\nabla f(x))^T (-\nabla f(x)) = -\|\nabla f(x)\|^2 < 0.$$

In our case, we find

$$d_t = -2\sigma x_t \tag{8}$$

The steplength α^*: We use the exact line search method to calculate the descent step as follows:

$$\alpha_t^* = \arg\min_{\alpha \geq 0} f(x_t + \alpha_t d_t),$$

with

$$\begin{aligned} f(x_t + \alpha_t d_t) &= (x_t + \alpha_t d)^T \sigma (x_t + \alpha_t d_t), \\ &= x_t^T \sigma x_t + x_t^T \sigma \alpha_t d_t + \alpha_t d_t^T \sigma x_t + \alpha_t^2 d_t^T \sigma d_t, \\ &= x_t^T \sigma x_t + 2 x_t^T \sigma d_t \alpha_t + d_t^T \sigma d_t \alpha_t^2, \end{aligned}$$

$$\begin{aligned} \frac{\partial f(x_t + \alpha_t d_t)}{\partial \alpha_t} &= 2 x_t^T \sigma d_t + 2 d_t^T \sigma d_t \alpha_t, \\ &= 0, \end{aligned}$$

and so we find

$$\alpha_t^* = \frac{x_t^T \sigma d_t}{d_t^T \sigma d_t}.$$

The elastic constraint ($x_{t+1} \leq \varepsilon_t$): can then be adjusted as follows:

$$x_{t+1} \leq x_t + \frac{x_t^T \sigma d_t}{d_t^T \sigma d_t} d_t \tag{9}$$

where d_t is defined in Equation (8).

Despite the fact that the steepest descent method is known for the zigzag behavior of its iterates which makes its convergence slow [14], its use is carried out to find a direction to improve the second objective and a step-size to go in the direction of this improvement.

4.3 The proposed algorithm

The main steps of our proposed method are presented in Algorithm 1.

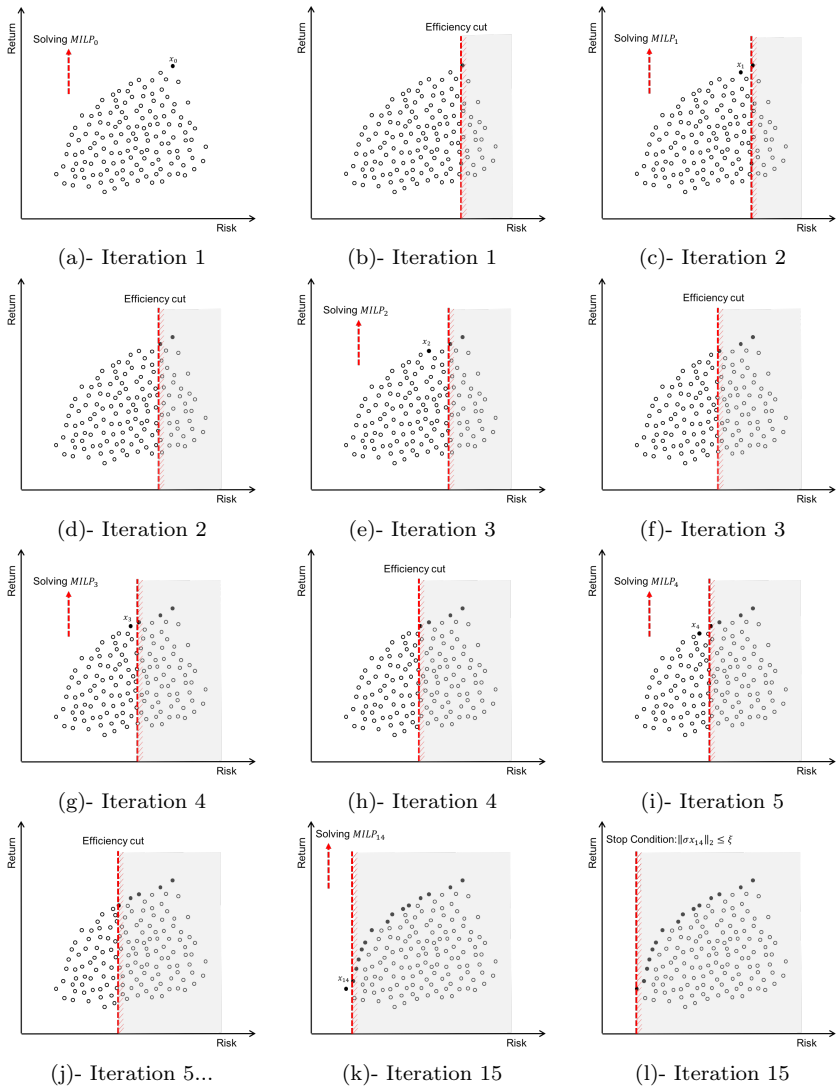
Algorithm 1: Algorithm for bi-objective constrained portfolio selection problems.

Data: A Constrained Portfolio Selection Problem (CPSP)
Result: ES: the set of weakly efficient solutions of (CPSP)
Step 0: *Initialization*;
  $t = 0$; // the first iteration;
  $ES = \emptyset$;
  $\varepsilon_t = (1, 1, ..., 1)^T \in \mathbb{R}^n$; // the upper bound of x_t;
  $\xi = 10^{-3}$; // the tolerance;
Step 1: *General step*;
  Solve $MILP_t$ defined by (11) using a Branch-and-Bound method;
  if $MILP_t$ *doesn't admit any solution* **then**
    Stop;
  else
    Let $(x, y)_t$ be the solution of $MILP_t$;
    $ES = ES \cup \{x_t\}$;
    Go to step 2;
  end
Step 2: *Efficiency cut construction*;
  $d_t = -2\sigma x_t$;
  $\alpha_t^* = \frac{x_t^T \sigma d_t}{d_t^T \sigma d_t}$;
  $\varepsilon_{t+1} = x_t + \alpha_t d_t$;
Step 3: *Checking the stop conditions*;
  if $\|\sigma x_t\|_2 \leq \xi$ *or* $\|x_{t+1} - x_t\|_2 \leq \xi$ **then**
    Stop;
  else
    Set $t = t + 1$;
    Go to Step 1;
  end

Observation 1 *In the case where $K = n$, i.e., all assets must be selected, the problem becomes a classical Markowitz problem with one budget constraint and more restricted bounds on the variables, i.e., $a \leq x \leq b$ instead of $0 \leq x \leq 1$, with $0 < a < b < 1$. The problem is then transformed into a bi-objective quadratic programming problem. For this purpose, we use the simplex method to solve $MILP_t$ in Step 1.*

Example 1 In Figure 1, we illustrate the behavior of the proposed method, by showing the most important steps: General step (Step 1 in Algorithm 1) and Efficiency cut construction (Step 2 in Algorithm 1). The points in Figure 1 represent the different images of the proposed portfolios. In Step 1 (Figure 1-(a), 1-(c), 1-(e),..., 1-(k)), we find the best portfolios having maximum return, after this, we add an efficiency cut to improve the risk of the next solution, which is Step 2 (Figure 1-(b), 1-(d), 1-(f),..., 1-(l)).

In Figure 1-(l), the results of applying our algorithm represent the front of the efficient solutions that will be selected, which is also called the Pareto front.



(a)- Iteration 1 (b)- Iteration 1 (c)- Iteration 2

(d)- Iteration 2 (e)- Iteration 3 (f)- Iteration 3

(g)- Iteration 4 (h)- Iteration 4 (i)- Iteration 5

(j)- Iteration 5... (k)- Iteration 15 (l)- Iteration 15

Fig. 1 An example of executing the proposed algorithm.

5 Algorithm validation and computational results

In this section, we present the theoretical validation of our method and give some results using real data from seven markets. At the end of this section, we present a comparison of our method with two exact methods and a meta-heuristic.

5.1 Theoretical validation

Theorem 3 *All solutions generated by this method are weakly efficient.*

Proof Consider the following bi-objective mixed integer quadratic optimization problem:

$$\begin{cases} \max\ f_1(x) = r^T x \\ \min\ \ f_2(x) = x^T \sigma x \\ s.t.\ \ x \in D_t \end{cases} \tag{10}$$

where

$$D_t = \{(x,y) \in \mathbb{R}^n \times \{0,1\}^n | e^T x = 1, e^T y = K, a_i y_i \leq x_i \leq b_i y_i, \forall i = \overline{1,n}, x_t \leq \varepsilon_{t-1}\}.$$

Let $D = D_0$ be the initial set of feasible solutions of Problem (10). Then suppose that a particular solution x_t^* is not weakly efficient, i.e., there exists another feasible point $z \in D$ that is "*better*" than x_t^*,
i.e. , $f_1(z) > f_1(x_t^*)$ and $f_2(z) < f_2(x_t^*)$.

a. if $f_1(z) > f_1(x_t^*)$ and $z \in D_t$:
As x_t^* is an optimal solution of Problem (5) at iteration t, then:

$$\begin{aligned} f_1(x_t^*) &= \max_{x \in D_t} f_1(x) \\ &\Longrightarrow f_1(x_t^*) \geq f_1(x), \forall x \in D_t \end{aligned}$$

and especially $f_1(x_t^*) \geq f_1(z)$ because $z \in D_t$. Therefore, $f_1(z) \not> f_1(x_t^*)$.

b. if $f_1(z) > f_1(x_t^*)$ and $z \notin D_t$:
Then $z \in D \backslash D_t$, which means that the solution z is in a region removed by a cutting step added at iteration j, with $j \in \{1, .., t-1\}$, which means that $f_2(z) \geq f_2(x_t^*)$ because of the decay of the efficiency cut defined by Equation (9), which guarantees Inequality (6). Therefore, $f_2(z) \not< f_2(x_t^*)$.
Consequently, all solutions generated by this method are weakly efficient.

■

Theorem 4 *If x_t^* is a unique solution of Problem (5) and Inequality (6) is strictly satisfied at step t then x_t^* is an efficient solution to Problem (2).*

Proof Consider Problem (10).
Let $D = D_0$ be the initial set of feasible solutions of Problem (10) and suppose that a particular solution x_t^* generated at iteration t is a unique solution to Problem (10) but not efficient, i.e., there exists another feasible point $z \in D$ which verifies $f_1(z) \geq f_1(x_t^*)$, and $f_2(z) \neq f_2(x_t^*)$.

a. if $f_1(z) \geq f_1(x_t^*)$ and $z \in D_t$:
Since x_t^* is a unique solution to Problem (10) and the inclusion $D_t \subset D$ holds, x_t^* is also a unique solution to Problem (5) at iteration t. But then $f_1(x_t^*) > f_1(x), \forall x \in D_t$ and especially $f_1(x_t^*) > f_1(z)$ because $z \in D_t$. Therefore, $f_1(z) \ngeq f_1(x_t^*)$.
b. if $f_1(z) \geq f_1(x_t^*)$, $f_2(z) \leq f_2(x_t^*)$ and $z \notin D_t$:
Then $z \in D \backslash D_t$, which means that the solution z is in a region removed by a cutting step added at iteration j, with $j \in \{1, .., t-1\}$, which means that $f_2(x_t^*) \leq f_2(z)$ because of the decay of the efficiency cut defined by Equation (9). This means that $f_2(x_t^*) = f_2(z)$ but since Inequality (6) is strict at step t, $f_2(z) \nleq f_2(x_t^*)$.

■

This approach finds weakly efficient solutions to a bi-objective quadratic optimization problem under cardinality constraints. The solutions are efficient if the auxiliary problem solutions are unique. Note that this method does not necessarily find all the efficient solutions of the problem. In the following proposition, we present a case in which efficient solutions are not removed by the constraints which we add at each step.

Proposition 1 *Let $MILP_t$ be the problem solved by our method at step t and let x_t^*, x_{t+1}^* be the solutions of $MILP_t$ and $MILP_{t+1}$, respectively. Consider Problem* (11) *formulated as follows:*

$$\begin{cases} \max\ f_1(x) = r^T x \\ s.t.\ \ x \in D_t \\ \qquad f_1(x) \geq f_1(x_{t+1}^*) + \xi \qquad (C1) \\ \qquad f_1(x) \leq f_1(x_t^*) - \xi \qquad (C2) \end{cases} \tag{11}$$

where

ξ is a small real number close to zero,
D_t is the set of feasible solutions at step t.

If Problem (11) has no solution, then there is no efficient solution removed by the constraint $x_{t+1} \leq \varepsilon_{t+1}$.

Proof If at step t such a solution exists, its return is between x_t and x_{t+1}, which is expressed by the constraints $(C1)$ and $(C2)$. If Problem (11) does not admit a solution, this means that no efficient solution is removed by the constraint added at step t.

■

As a consequence of the previous proposition, we conclude that this algorithm does not find all the efficient solutions of Problem (3) unless the condition of Proposition 1 is verified for all generated constraints added at each step of our algorithm.

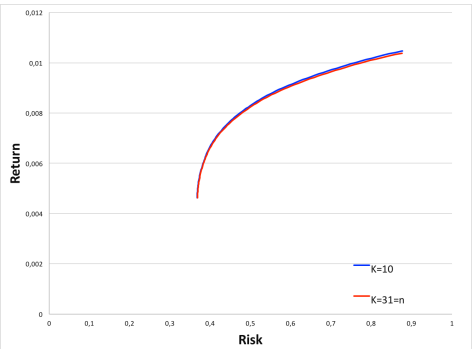
5.2 Computational results

In this section, we present the computational results obtained by performing experiments on a publicly available dataset. The presented study was performed on seven reference datasets, five of them being derived from [39], available in Beasley's OR Library [4]. These data provide the necessary input data for various assets in different stock indices: Hong Kong Hang Seng with 31 assets, German Dax 100 with 85 assets, British FTSE 100 with 89 assets, American S&P 100 with 98 assets, Japanese Nikkei 225 with 225 assets, and two additional datasets being described by Cesarane et al. in 2014 [10], available from [9]. These authors reported 263 weekly prices from March 2003 to March 2008 of American S&P 500 which contains 476 assets and European-American NASDAQ which includes 2196 assets.

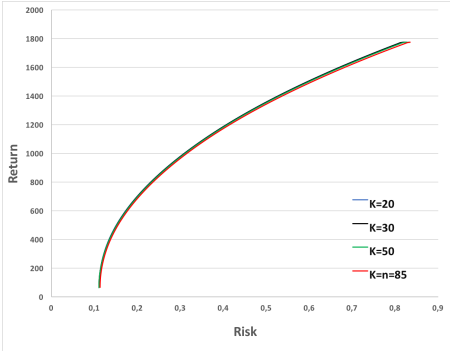
Figures 2 and 3 show the results of the execution of our method. Each figure represents the graph of the Pareto front containing the values of the returns of the compromise solutions according to the risk based on different cardinality values (K). We observe that the larger the value of K, the worse the obtained Pareto border. This can be explained by the reduced space of feasible solutions since more assets are forced to be strictly positive.

We also observe that this difference between the curves is not always significant; for example in the DAX market, the border is almost the same for different values of K. This is due to the fact that in the coefficients of the return function r_1 is equal to 1970 and all the other coefficients are smaller than 1. In this case, the solution consists of investing the maximum into the first asset to maximize the return and selecting the $K - 1$ assets at the minimum limit to satisfy the constraints of budget, cardinality, and quantity. Therefore, as shown by Figure 2, the obtained difference between investing 20 or 30 is not very significant with respect to the variation between the return on the first asset and the remaining 19 or 29 returns, respectively.

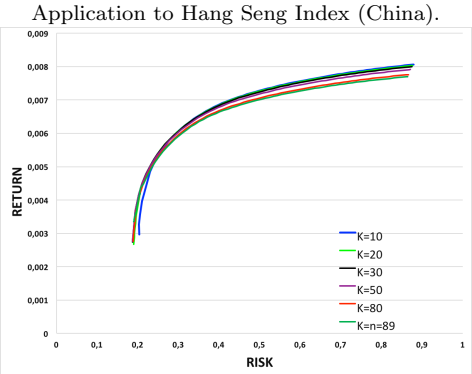
The difference between the curves representing the Pareto boundaries corresponding to the different values of K are very visible in the case of the Nikkei market, which can be explained by the fact that the spreads between the returns of the market's assets are not large. In general, we note that the smaller the number of assets selected (securities), the better the portfolios (better profits for the same level of risk). This can be explained by the fact that if the decision-maker is forced to choose exactly K assets, he is forced to include the worst income portfolios to satisfy the cardinality constraint.



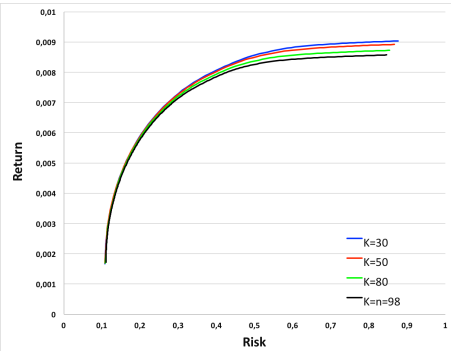
Application to Hang Seng Index (China).



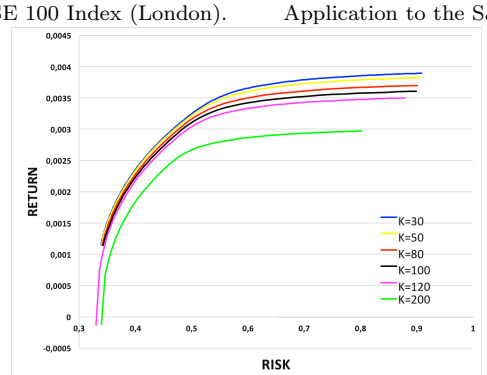
Application to DAX Stock Market (Germany).



Application to the FTSE 100 Index (London).



Application to the S&P 100 Index (U.S.A).

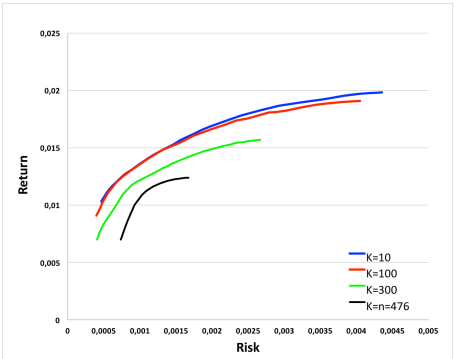


Application to the Nikkei index (Japan).

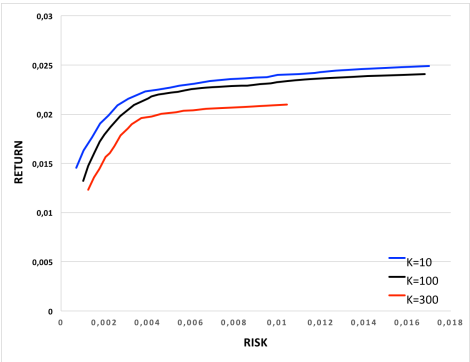
Fig. 2 Application of our method to different markets, with the database downloaded from [39] for various values of cardinality.

We can see that the curves obtained in the $S\&P500$ and $Nasdaq$ markets increase more slowly as the risk increases, which is very visible in the $Nasdaq$ market. After verifying the used dataset, we can explain this behavior by the fact that the values of the variance matrix representing the risk are very small.

The maximum value of the $S\&P500$ market risk matrix, for example, is 0.0067 whereas it is 1 in the DAX market.



Application to S&P500 (U.S.A).



Application to Nasdaq Stock Market (U.S.A).

Fig. 3 Application of our method to different markets, with the database downloaded from [9] for various values of cardinality.

5.3 Discussion and comparison

The proposed method has been programmed in Matlab, on a machine with an *Intel*® *Core*™ i7-2500 Processor (8GB RAM). The results of the execution of our proposed method are listed in Table 1. We additionally implemented the Pascoletti-Serafini approach and the ε-constraint method in their bi-objective form [17] using Matlab and IBM-Cplex 12. These methods find weakly efficient solutions to multiobjective programming problems. We use them to evaluate the quality of the proven solutions of the method we propose. Unfortunately, these two methods did not find a solution to large-scale programming problems as in our case: we had to wait more than 7 hours for the S&P 500 market without having any output. In the case of Nasdaq, the program stopped after 26 minutes with an error message indicating that the computer memory was full.

In order to test the effectiveness of our proposed method for large-scale programming problems, we have implemented the memetic method presented in [43]. For this experience, we used the same machine and the same test problem. We varied each time the number of authorized assets in a portfolio and compared the methods based on two parameters: the quality of the solutions provided and the execution time. For quantity constraints, we have set the lower bound to $a = 0.001$ and the upper bound to $b = 1$, i.e., an asset cannot have a part of less than 0.1% or more than 100%. Before starting the tests, we designed an instance generator and compared the results obtained with those obtained by the other two methods in terms of execution time. We ran our programs 10 times for each instance and averaged the results for each case.

These results are summarized in the Table 1, where "Iterative", "P-S", "ε-constraint" and "Memetic" refers to, respectively, the proposed method, the Pascoletti-Serafini method, ε-constraint method and memetic method. "Time" is the execution time (in seconds), "ES" is the number of solutions found.

Pascoletti-Serafini's method contains some parameters to initiate the algorithm. These parameters are chosen randomly. We tried our program several times and on several parameters before the experiments were performed. For each problem, the proposed method allowed to find the efficient set of weakly efficient solutions in a reasonable time.

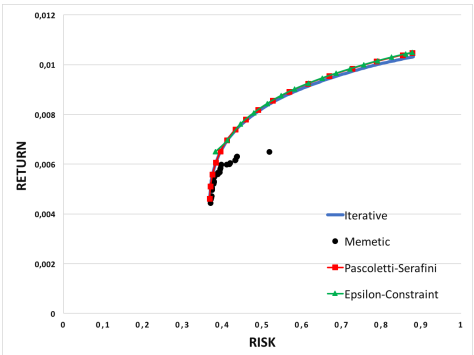
Table 1 Comparison between the proposed method and three other methods.

		Iterative		P-S		ε-constraint		Memetic	
Markets	K	Time	ES	Time	ES	Time	ES	Time	ES
Hang Seng	10	10.08	119	8.39	24	5.22	26	28.01	33
with	20	15.64	119	8.92	27	5.28	29	29.71	15
31 assets	31	7.32	105	3.23	19	4.28	21	27.19	15
DAX	10	10.15	108	182.27	17	152.28	21	43.46	31
with	30	17.97	111	83.36	20	82.19	28	45.27	23
85 assets	85	9.73	100	31.28	21	27.84	19	59.37	16
FTSE	10	4.28	181	329.63	38	268.98	34	93.27	26
with	30	5.74	175	293.19	41	208.72	58	92.43	18
89 assets	89	3.29	130	73.92	46	63.15	49	102.38	18
S&P100	10	3.06	111	548.81	14	527.91	12	113.50	23
with	30	5.35	109	293.01	23	231.04	37	132.93	14
98 assets	98	4.83	57	83.92	42	68.39	40	139.20	15
Nikkei	10	35.56	973	231.93	91	189.36	78	219.54	14
with	30	143.33	943	178.92	74	152.51	83	227.91	17
225 assets	225	24.39	221	127.30	32	93.28	64	282.12	18
S&P500	10	26.17	95	-	-	-	-	394.19	61
with	100	28.56	89	-	-	-	-	383.81	62
476 assets	476	17.28	22	-	-	-	-	362.34	57
Nasdaq	10	1544.21	194	-	-	-	-	474.10	24
with	30	511.74	265	-	-	-	-	418.02	22
2196 assets	300	316.18	162	-	-	-	-	582.64	14

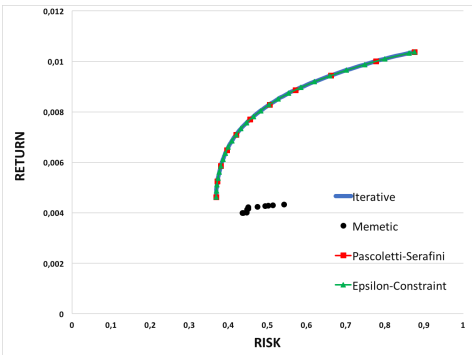
We observe that the Pascoletti-Serafini method and ε-constraint behave almost the same way. In fact, when K is small, the execution time for these two methods is very large. The number of solutions of the ε-constraint method is always higher than that obtained by the Pascoletti-Serafini method for the same execution time period. As we have already mentioned, these two methods have failed to produce solutions for the SP500 and Nasdaq markets due to their large dimensions. The memetic method is a metaheuristic adapted according to the algorithm proposed by [43]. The execution times and the number of solutions, corresponding to the memetic method, shown in Table 1 are the average of 10 executions.

The method proposed in this work finds a larger number of solutions in a shorter time than each of the other methods for small instances. For larger instances, this method finds a fairly large number of solutions compared to the memetic method, but the execution time increases considerably. However, the

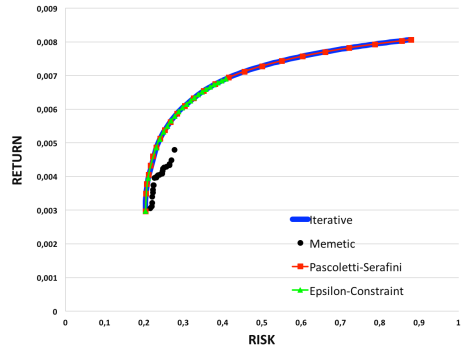
solutions found by our method are better than those of the memetic method, as we can see in Figure 4. A more detailed study on the major problems is presented in Table 2.



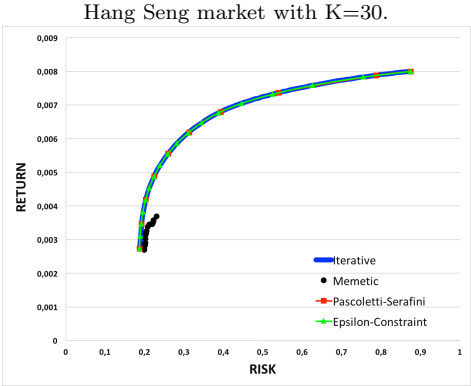
Hang Seng market with K=10.



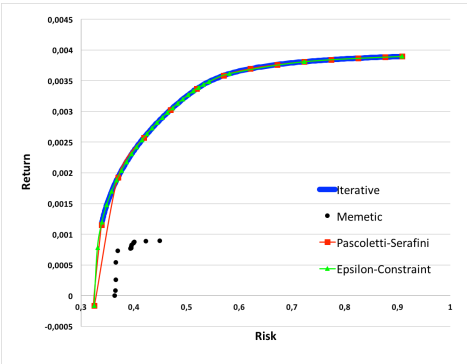
Hang Seng market with K=30.



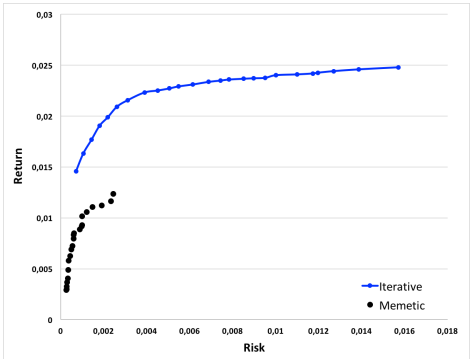
FTSE market with K=10.



Hang Seng market with K=30.



Nikkei market with K=30.



Nasdaq market with K=10.

Fig. 4 Comparison of the 4 methods according to the markets and the considered cardinality.

In Figure 4, a comparison of the four implemented methods is described. We can remark that the quality of the solutions found by our method and that of the Pascoletti-Serafini and ε-constraint methods are similar. We notice that the solutions found by the memetic method are not as good as those found by the other methods.

Table 2 Result of the execution according to generated problems.

		Iterative		P-S		ε-constraint		Memetic	
n	K	Time	ES	Time	ES	Time	ES	Time	ES
100	10	13.20	89.3	182.92	23.3	146.73	27.3	238.62	26.9
	20	14.74	92.1	92.32	27.8	39.36	30.2	277.46	18
	100	5.19	75.7	47.29	21	32.54	17	289.73	15.8
300	10	73.38	98.3	364.73	32.4	283.94	42.3	328.91	37.4
	30	74.92	63.5	288.92	14.9	237.1	16.2	364.28	46.2
	100	93.67	71.8	173.93	23.4	182.67	31.1	372.93	28
1000	10	638.66	82.3	-	-	-	-	538.92	26.6
	30	783.91	63.2	-	-	-	-	554.3	38.4
	100	281.04	58.3	-	-	-	-	583.02	27.3
3000	10	1638.92	78.4	-	-	-	-	673.91	17.9
	30	1780.36	58.3	-	-	-	-	648.01	15.3
	100	682.93	37	-	-	-	-	839.72	12.3
5000	10	4838.64	137.92	-	-	-	-	936.93	11.7
	100	3977.05	143.06	-	-	-	-	1037.61	14.93
	500	2102.94	127.3	-	-	-	-	893.60	19.47

Table 2 summarizes the results for randomly generated problems, all of which have the same portfolio specifications. The results found are the average of the 10 executions for each generated problem, where "n" is the dimension of the generated problems, "ES" and "$Time$" are, respectively, the mean number of found solutions and the mean execution time.

After having performed our tests presented in Figure 4 and Tables 1 and 2, we are in a position to confirm that the method presented in this paper gives results as good as the Pascoletti-Serafini method or the ε-constraint method in terms of quality and better results in terms of execution time. In fact, our method is capable of finding the solution even for large problems that reach up to 5000 assets, unlike the two others that cannot propose a solution for problems exceeding 500 assets.

We also notice that the memetic method finds points that are very far from the solutions found by our method. The execution time of this technique is very large compared to our method for small instances. For large instances, we can see that the execution time of our method exceeds the execution time of the memetic method, but this difference is not significant.

Finally, we can confirm that our proposed iterative method finds Pareto's efficient frontier more rapidly than exact methods with respect to the execution time. Our methods are also able to find the Pareto front for large instances in an acceptable time.

6 Conclusion

In this problem, risk and return are two opposing objectives that must be optimized simultaneously. We have extended the standard optimization model to include cardinality constraints that limit a portfolio to have a specified number of assets and quantity constraints that impose limits on the proportion of the portfolio to be held in a given asset. The obtained model is a bi-objective mixed integer quadratic programming problem. A novel multiobjective optimization algorithm is presented, which is based on a new efficient cut to find the cardinality constrained efficient frontier.

The experimental results reveal that the proposed algorithm can manage the budget, minimum purchase threshold, maximum limit and cardinality constraints simultaneously. In order to test the efficiency of the presented method, we compare it with the Pascoletti-Serafini method, ε-constraint and a memetic method. We find that our method produces a high quality of portfolio in reasonable time compared to these methods. The portfolio manager can compromise between return, risk, quantity, and a number of assets to determine the portfolio according to his needs. As to future work, we are working on including the time factor in the model and to study this problem over multiple time periods.

References

1. Saurabh Agarwal. Portfolio Selection Theories: Review, Synthesis and Critique. *Asian Journal of Management*, 5(1):1–7, 2014.
2. Emrah Altun and Huseyin Tatlidil. A Comparison of Portfolio Selection Models via Application on ISE 100 Index Data. 1558:1438–1441, 2013.
3. K.P. Anagnostopoulos and G. Mamanis. A portfolio optimization model with three objectives and discrete variables. 37(7):1285–1297, July 2010.
4. J. E. Beasley. OR-Library: Distributing Test Problems by Electronic Mail. 41(11):1069–1072, November 1990.
5. Dimitris Bertsimas and Romy Shioda. Algorithm for cardinality-constrained quadratic optimization. 43(1):1–22, May 2009.
6. Mehdi Beyhaghi and James P. Hawley. Modern portfolio theory and risk management: assumptions and unintended consequences. *Journal of Sustainable Finance & Investment*, 3(1):17–37, January 2013.
7. Daniel Bienstock. Computational study of a family of mixed-integer quadratic programming problems. *Mathematical Programming*, 74(2):121–140, August 1996.
8. Brian Borchers and John E. Mitchell. An improved branch and bound algorithm for mixed integer nonlinear programs. *Computers & Operations Research*, 21(4):359–367, April 1994.
9. Francesco Cesarone. http://w3.uniroma1.it/tardella/datasets.html, Accessed date: 12 January 2017.
10. Francesco Cesarone, Andrea Scozzari, and Fabio Tardella. Linear vs. quadratic portfolio selection models with hard real-world constraints. *Computational Management Science*, pages 1–26, May 2014.
11. T. J. Chang, N. Meade, J. E. Beasley, and Y. M. Sharaiha. Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research*, 27(13):1271–1302, November 2000.
12. Vira Chankong and Yacov Y Haimes. Optimization-based methods for multiobjective decision-making-an overview. *Large Scale Systems In Information And Decision Technologies*, 5(1):1–33, 1983.

13. Tunchan Cura. Particle swarm optimization approach to portfolio optimization. *Nonlinear analysis: Real world applications*, 10(4):2396–2406, 2009.
14. Yu-Hong Dai and Ya-Xiang Yuan. Alternate minimization gradient method. *IMA Journal of Numerical Analysis*, 23(3):377–393, 2003.
15. Adiel Teixeira de Almeida and Rudolf Vetschera. A note on scale transformations in the PROMETHEE V method. *European Journal of Operational Research*, 219(1):198–200, May 2012.
16. Guang-Feng Deng, Woo-Tsong Lin, and Chih-Chung Lo. Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization. *Expert Systems with Applications*, 39(4):4558–4566, March 2012.
17. Gabriele Eichfelder. *Adaptive scalarization methods in multiobjective optimization*, volume 436. Springer, 2008.
18. Edwin J Elton, Martin J Gruber, Stephen J Brown, and William N Goetzmann. *Modern portfolio theory and investment analysis*. John Wiley & Sons, 2009.
19. Jonathan E. Fieldsend, John Matatko, and Ming Peng. Cardinality Constrained Portfolio Optimisation. In Zheng Rong Yang, Hujun Yin, and Richard M. Everson, editors, *Intelligent Data Engineering and Automated Learning IDEAL 2004*, volume 3177, pages 788–793. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.
20. Antonio Frangioni and Claudio Gentile. Perspective cuts for a class of convex 0–1 mixed integer programs. *Mathematical Programming*, 106(2):225–236, 2006.
21. Gianluca Fusai and Andrea Roncoroni. *Implementing models in quantitative finance: methods and cases*. Springer Science & Business Media, 2007.
22. Luca Di Gaspero, Giacomo Di Tollo, Andrea Roli, and Andrea Schaerf. Hybrid metaheuristics for constrained portfolio selection problems. *Quantitative Finance*, 11(10):1473–1487, 2011.
23. Arthur M Geoffrion. Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3):618–630, 1968.
24. Nalan Gulpinar, Le Thi Hoai An, and Mahdi Moeini. Robust investment strategies with discrete asset choice constraints using DC programming. *Optimization*, 59(1):45–62, January 2010.
25. Yacov Y Haimes. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE transactions on systems, man, and cybernetics*, 1(3):296–297, 1971.
26. Kambiz Forqandoost Haqiqi and Tohid Kazemi. Ant colony optimization approach to portfolio optimization. In *Proc. 3rd International Conference on Financial Theory and Engineering*, pages 292–296, 2012.
27. Norbert J. Jobst, Michael D. Horniman, Cormac A. Lucas, and Gautam Mitra. Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints. *Quantitative finance*, 1(5):489–501, 2001.
28. Eva K. Lee and John E. Mitchell. Computational experience of an interior-point SQP algorithm in a parallel branch-and-bound framework. *Proceedings of High Performance Optimization Techniques 1997*, pages 97–08, 1997.
29. Bin Li and Steven CH Hoi. Online portfolio selection: A survey. *ACM Computing Surveys (CSUR)*, 46(3):35, 2014.
30. Duan Li, Xiaoling Sun, and Jun Wang. Optimal lot solution to cardinality constrained mean-variance formulation for portfolio selection. *Mathematical Finance*, 16(1):83–101, 2006.
31. Jun Li and Jiuping Xu. Multi-objective portfolio selection model with fuzzy random returns and a compromise approach-based genetic algorithm. *Information Sciences*, 220, January 2013.
32. Khin Lwin and Rong Qu. A hybrid algorithm for constrained portfolio selection problems. *Applied intelligence*, 39(2):251–266, 2013.
33. Khin Lwin, Rong Qu, and Graham Kendall. A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. 24:757–772, November 2014.
34. Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.
35. George Mavrotas and Olena Pechak. Combining Mathematical Programming and Monte Carlo Simulation to Deal with Uncertainty in Energy Project Portfolio Selection. In Fausto Cavallaro, editor, *Assessment and Simulation Tools for Sustainable Energy Systems*, number 129 in Green Energy and Technology, pages 333–356. Springer London, 2013.

36. Kaisa Miettinen. Nonlinear multiobjective optimization, volume 12 of international series in operations research and management science, 1999.
37. R. Moral-Escudero, R. Ruiz-Torrubiano, and A. Suarez. Selection of Optimal Investment Portfolios with Cardinality Constraints. In *IEEE Congress on Evolutionary Computation, 2006. CEC 2006*, pages 2382–2388. IEEE, 2006.
38. Walter Murray and Howard Shek. A local relaxation method for the cardinality constrained portfolio optimization problem. *Computational Optimization and Applications*, 53(3):681–709, 2012.
39. OR-LIBRARY. http://people.brunel.ac.uk/∼mastjjb/jeb/orlib/portinfo.html, Accessed date: 21 July 2016.
40. Adriano Pascoletti and Paolo Serafini. Scalarizing vector optimization problems. *Journal of Optimization Theory and Applications*, 42(4):499–524, 1984.
41. R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.
42. Ruben Ruiz-Torrubiano and Alberto Suarez. Hybrid Approaches and Dimensionality Reduction for Portfolio Selection with Cardinality Constraints. *Computational Intelligence Magazine, IEEE*, 5(2):92–107, May 2010.
43. Rubén Ruiz-Torrubiano and Alberto Suárez. A memetic algorithm for cardinality-constrained portfolio optimization with transaction costs. *Applied Soft Computing*, 36:125–142, 2015.
44. Dong X Shaw, Shucheng Liu, and Leonid Kopman. Lagrangian relaxation procedure for cardinality-constrained portfolio optimization. *Optimisation Methods & Software*, 23(3):411–420, 2008.
45. Patrick Smith, Matthew Ferringer, Ryan Kelly, and Inki Min. Budget-constrained portfolio trades using multiobjective optimization. *Systems Engineering*, 15(4):461–470, 2012.
46. Hamed Soleimani, Hamid Reza Golmakani, and Mohammad Hossein Salimi. Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. *Expert Systems with Applications*, 36(3):5058–5063, April 2009.
47. Ralph E. Steuer, Yue Qi, and Markus Hirschberger. Developments in multi-attribute portfolio selection. *Multiple criteria decision making*, 5:251–262, 2006.
48. Juan Pablo Vielma, Shabbir Ahmed, and George L. Nemhauser. A Lifted Linear Programming Branch-and-Bound Algorithm for Mixed-Integer Conic Quadratic Programs. *INFORMS Journal on Computing*, 20(3):438–450, August 2008.