The optimal security selection is a classical portfolio problem since the seminal work of Markowitz [1]. In the standard approach, the decision maker selects the securities in such a way that the portfolio expectation is maximized, under the constraint that risk (variance) must be kept under a fixed threshold [2]. The problem consists in picking the best amount of securities, with the aim of maximizing future returns. It is a typical multivariate problem: the only way to improve future returns is to increase the risk level that the decision maker is disposed to accept [3].

The portfolio selection problem is usually considered as a bi-criteria optimization problem where a reasonable trade-off between expected rate of return and risk is sought. In the classical Markowitz model future returns are random variables that can be controlled by the two parameters: a portfolio’s efficiency calculated by the expectation, and a risk, which is measured with the variance. The classical problem is formulated as a quadratic program with continuous variables and some side constraints.

While the original Markowitz model forms a quadratic programming problem, many attempts have been made to linearize the portfolio optimization procedure [4–10]. The linear program solvability is very important for applications to real-life financial and other decisions where the constructed portfolios have to meet numerous side constraints. Examples of them are minimum transaction lots, transaction costs or mutual funds characteristics etc. The introduction of these features leads to mixed integer program problems.

This paper presents a bi-criterion extension of the Markowitz portfolio optimization model, in which the variance has been replaced with the Value-at-Risk (VaR). The VaR is a quantile of the return distribution function [5].

The advantage of using VaR measure in portfolio optimization is that this value of risk is independent of any distribution hypothesis. It concern only downside risk, namely the

Bartosz Sawik*

**A Weighted-Sum Mixed Integer Program for Bi-Objective Dynamic Portfolio Optimization**

1. Introduction

The optimal security selection is a classical portfolio problem since the seminal work of Markowitz [1]. In the standard approach, the decision maker selects the securities in such a way that the portfolio expectation is maximized, under the constraint that risk (variance) must be kept under a fixed threshold [2]. The problem consists in picking the best amount of securities, with the aim of maximizing future returns. It is a typical multivariate problem: the only way to improve future returns is to increase the risk level that the decision maker is disposed to accept [3].

The portfolio selection problem is usually considered as a bi-criteria optimization problem where a reasonable trade-off between expected rate of return and risk is sought. In the classical Markowitz model future returns are random variables that can be controlled by the two parameters: a portfolio’s efficiency calculated by the expectation, and a risk, which is measured with the variance. The classical problem is formulated as a quadratic program with continuous variables and some side constraints.

While the original Markowitz model forms a quadratic programming problem, many attempts have been made to linearize the portfolio optimization procedure [4–10]. The linear program solvability is very important for applications to real-life financial and other decisions where the constructed portfolios have to meet numerous side constraints. Examples of them are minimum transaction lots, transaction costs or mutual funds characteristics etc. The introduction of these features leads to mixed integer program problems.

This paper presents a bi-criterion extension of the Markowitz portfolio optimization model, in which the variance has been replaced with the Value-at-Risk (VaR). The VaR is a quantile of the return distribution function [5].

The advantage of using VaR measure in portfolio optimization is that this value of risk is independent of any distribution hypothesis. It concern only downside risk, namely the
risk of loss. This index measures the loss in question in a certain way. Finally VaR is valid for all types of securities and therefore either involve the various valuation models or be independent of these models [11].

This portfolio optimization problem is formulated as a bi-objective mixed integer program. The portfolio selection problem considered is based on a dynamic model of investment, in which the investor buys and sells securities in successive investment periods. The problem objective is to dynamically allocate the wealth on different securities to optimize the weighted difference of the portfolio expected return and the probability that the return is not less than a required level.

The results of some computational experiments with the mixed integer programming approach modeled on a real data from the Warsaw Stock Exchange are reported. The input dataset consist of time series of the daily quotation of returns of securities from the Warsaw Stock Exchange.

2. Problem formulation

Let $n$ be the number of securities available in the market with historical quotations in $t$ investment periods, each consisting of $h$ historical periods.

Let $r_{ij}$ be the random variable representing the future daily return of security $j$ in historical time period $i$.

The portfolio optimization problem with Value-at-Risk constraint is formulated as the classic Markowitz approach, but with Value-at-Risk instead of variance as a risk measure.

The decision maker fixes the lower bound $r^{VaR}$ for successful returns – any investments whose Value-at-Risk is less than $r^{VaR}$ will be not acceptable.

Let $r^{Min}$ be the minimum return that can be observed in the market, for example the biggest possible loss of money invested in portfolio. In the worst case it is the whole amount of capital, so for instance it can be equal $-100\%$ [2, 12].

The bi-objective dynamic portfolio optimization model with Value-at-Risk is NP-hard problem even when future returns are described by discrete uniform distributions [13].

The five types of variables for each successive investment period are introduced in the model: a continuous wealth allocation variable that represents the percentage of wealth allocated to each security, a continuous wealth allocation variable for buying amount of each security, a continuous wealth allocation variable for selling amount of each security, a binary selection variable that prevents the choice of portfolios whose VaR is below the fixed threshold and a binary selection variable for selecting each security to the portfolio.

In the approach proposed in this paper, the portfolio optimization problem is formulated as weighting dynamic bi-objective mixed integer program, which allows commercially available software (e.g. AMPL/CPLEX [14]) to be applied for solving medium size, yet practical instances.

The problem formulation is presented below.
Maximize

\[
\sum_{k=1}^{t} \left( \beta_1 \left( \sum_{i=(k-1)h+1}^{kh} \sum_{j=1}^{n} p_i r_{ij} x_{jk} \right) - \beta_2 \alpha_k^{\text{VaR}} \right)
\]

subject to

\[
\sum_{j=1}^{n} r_{ij} x_{jk} - r_{\text{Min}} y_{ik} \leq \frac{r_{\text{VaR}} - r_{\text{Min}}}{r_{\text{VaR}}}, \quad i = (k-1)h+1, \ldots, kh, \quad k = 1, \ldots, t
\]

\[
\sum_{i=(k-1)h+1}^{kh} p_i (1 - y_{ik}) \leq \alpha_k^{\text{VaR}}, \quad k = 1, \ldots, t
\]
The objective (1) represents the weighted difference of the portfolio expected return and the probability that the return is not less than a required level. Constraints (2) and (3) prevent the choice of portfolios whose VaR is below the fixed threshold. Every time expected portfolio return is below $r^{VaR}$, then $y_{ik}$ must be equal to 0 and $1 - y_{ik} = 1$ in constraint (3). Therefore, all probabilities of events $i$ whose returns are below the $VaR$ threshold was summed up. If the result is greater than $\alpha^{VaR}$, then the portfolio is not feasible.

$$\sum_{j=1}^{n} x_{jk} = 1, \quad k = 1, \ldots, t$$  \hspace{1cm} (4)

Constraint (4) requires that in each investment period all capital must be allocated on different securities with positive expected return.

$$x_{j1}^{buy} = x_{j1}, \quad j = 1, \ldots, n: \sum_{i=1}^{h} p_{ir_{ij}} > 0$$  \hspace{1cm} (5)

$$x_{j1}^{sell} = 0, \quad j = 1, \ldots, n$$  \hspace{1cm} (6)

$$x_{jk} = x_{jk-1} + x_{jk}^{buy} - x_{jk}^{sell}, \quad j = 1, \ldots, n, \quad k = 2, \ldots, t$$  \hspace{1cm} (7)

Constraints (5), (6) and (7) are responsible for a dynamic balance among $x_{jk}^{buy}, x_{jk}, x_{jk}^{sell}$ for each successive investment period $k$.

$$\sum_{j=1}^{n} z_{jk} \geq v, \quad k = 1, \ldots, t$$  \hspace{1cm} (8)

Constraint (8) ensures that the number of stocks in optimal portfolio must be greater than or equal to the accepted number of assets in the selected portfolio.

$$\sum_{i=(k-1)h+1}^{kh} p_t \sum_{j=1}^{n} r_{ij} x_{jk} \geq r^{VaR}, \quad k = 1, \ldots, t$$  \hspace{1cm} (9)

Constraint (9) imposes the minimum portfolio expected return equal $r^{VaR}$ that the decision maker is prepared to accept for each successive investment period $k$.

$$x_{jk} \leq z_{jk}, \quad j = 1, \ldots, n: \sum_{i=(k-1)h+1}^{kh} p_i r_{ij} > 0, \quad k = 1, \ldots, t$$  \hspace{1cm} (10)

$$x_{jk}^{buy} \leq z_{jk}, \quad j = 1, \ldots, n: \sum_{i=(k-1)h+1}^{kh} p_i r_{ij} > 0, \quad k = 1, \ldots, t$$  \hspace{1cm} (11)

$$x_{jk}^{sell} \leq z_{jk}, \quad j = 1, \ldots, n, \quad k = 1, \ldots, t$$  \hspace{1cm} (12)
Constraints (10), (11) and (12) are responsible for relations between variables $x_{jk}, x_{jk}^{\text{buy}}, x_{jk}^{\text{sell}}$ and $z_{jk}$.

\[ \frac{z_{jk}}{100} \leq x_{jk}, \quad j = 1, \ldots, n : \sum_{i=(k-1)h+1}^{kh} p_{i}r_{ij} > 0, \quad k = 1, \ldots, t \]  
(13)

\[ \frac{z_{jk}}{100} \leq x_{jk}^{\text{buy}}, \quad j = 1, \ldots, n : \sum_{i=(k-1)h+1}^{kh} p_{i}r_{ij} > 0, \quad k = 1, \ldots, t \]  
(14)

Constraints (13) and (14) ensure the addition to portfolio and buying of some amount of security $j$ in successive investment period $k$.

\[ 0 \leq \alpha_k^{\text{VaR}} \leq 1, \quad k = 1, \ldots, t \]  
(15)

Constraint (15) defines continuous variable $\alpha_k^{\text{VaR}}$ – probability that return of investment is not less than $r_{k}^{\text{VaR}}$ of successive investment period $k$.

\[ x_{jk} \geq 0, \quad j = 1, \ldots, n : \sum_{i=(k-1)h+1}^{kh} p_{i}r_{ij} > 0, \quad k = 1, \ldots, t \]  
(16)

Constraint (16) defines continuous variable $x_{jk}$ – percentage of capital invested in successive investment period $k$ in security $j$. In addition, this formula eliminates securities with a non-positive expected return.

\[ x_{jk}^{\text{buy}} \geq 0, \quad j = 1, \ldots, n : \sum_{i=(k-1)h+1}^{kh} p_{i}r_{ij} > 0, \quad k = 1, \ldots, t \]  
(17)

\[ x_{jk}^{\text{sell}} \geq 0, \quad j = 1, \ldots, n, \quad k = 1, \ldots, t \]  
(18)

\[ y_{ik} \in \{0,1\}, \quad i = (k-1)h+1, \ldots, kh, \quad k = 1, \ldots, t \]  
(19)

\[ z_{jk} \in \{0,1\}, \quad j = 1, \ldots, n, \quad k = 1, \ldots, t \]  
(20)

Finally, constraints (17), (18), (19) and (20) define variables $x_{jk}^{\text{buy}}, x_{jk}^{\text{sell}}$ and $z_{jk}$ and also for eliminates securities with a non-positive expected return.

Variables $x_{jk}$ are percentage of capital invested in security $j$ in successive investment period $k$.

The combination of continuous variables $x_{jk}, x_{jk}^{\text{buy}}, x_{jk}^{\text{sell}}$ and binary variables $y_{ik}, z_{jk}$ leads NP-hard mixed integer programming problem [13]. If the number of historical observations $m$ is bounded by a constant, there are $2^m$ ways of fixing the variables $y_{ik}, z_{jk}$ for each successive investment period $k$. 


3. Computational results

In this section numerical examples and some computational results are presented to illustrate possible applications of the proposed formulations of this multi-period optimization model. The examples are modeled on a real data form the Warsaw Stock Exchange.

Suppose that \( n = 241 \) securities with historical quotations in \( t = 12 \) investment periods, each of \( h = 335 \) days, in total 4020 samples.

The eighteen years horizon from 30th Jan 1991 to 30th Jan 2009 – consist of \( m = 4020 \) historic daily quotations divided into \( t = 12 \) investment periods (\( h = 335 \) daily quotations each), with the selection of \( n = 241 \) input securities for portfolio, quoted each day in the historical horizon. Probability of realization for expected securities returns is the same for each day and summed up for whole period to one.

The computational experiments have been performed using AMPL programming language [13] and the CPLEX v.11 solver (with the default settings) on a laptop with Intel® Core 2 Duo T9300 processor running at 2.5 GHz and with 4 GB RAM. Computational time range is from a few seconds to minutes.

Tables 2–5 presents the solution results for bi-objective dynamic portfolio – weighting approach.

### Table 2
The solution results for bi-objective dynamic portfolio – weighting approach \( \beta_1 = 0.5, \beta_2 = 0.5 \)

<table>
<thead>
<tr>
<th>Security ( x[j,k] )</th>
<th>ALMA</th>
<th>ALCHEMIA</th>
<th>DEBICA</th>
<th>APATON</th>
<th>ARTMAN</th>
<th>ECHO</th>
<th>FORTISPL</th>
<th>Bioton</th>
<th>ARTHAM</th>
<th>JEZEN</th>
<th>HRENA</th>
<th>KABLE</th>
<th>ENERGIOPOL</th>
<th>CEZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portfolio Return</td>
<td>0.881823</td>
<td>0.980000</td>
<td>1.000000</td>
<td>0.010616</td>
<td>0.908384</td>
<td>0.908384</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
</tr>
</tbody>
</table>

### Table 3
The solution results for bi-objective dynamic portfolio – weighting approach \( \beta_1 = 0.1, \beta_2 = 0.9 \)

<table>
<thead>
<tr>
<th>Security ( x[j,k] )</th>
<th>ALMA</th>
<th>ALCHEMIA</th>
<th>DEBICA</th>
<th>APATON</th>
<th>ARTMAN</th>
<th>ECHO</th>
<th>FORTISPL</th>
<th>Bioton</th>
<th>ARTHAM</th>
<th>JEZEN</th>
<th>HRENA</th>
<th>KABLE</th>
<th>ENERGIOPOL</th>
<th>CEZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portfolio Return</td>
<td>0.881823</td>
<td>0.980000</td>
<td>1.000000</td>
<td>0.010616</td>
<td>0.908384</td>
<td>0.908384</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
<td>0.010000</td>
</tr>
</tbody>
</table>

* CPU seconds for proving optimality on a laptop with Intel® Core 2 Duo T9300 processor running at 2.5 GHz and with 4 GB RAM /CPLEX v.11
The solution results for bi-objective dynamic portfolio – weighting approach $\beta_1 = 0.7$, $\beta_2 = 0.3$

Table 4

<table>
<thead>
<tr>
<th>$\beta_1$</th>
<th>$\beta_2$</th>
<th>$\gamma_{\text{Var}}$</th>
<th>$\gamma_{\text{Min}}$</th>
<th>-100</th>
<th>$\tau_{\text{BH}}$</th>
<th>15(2,335) quotations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>0.024353</td>
<td>0.007091</td>
<td>0.005971</td>
<td>0.003194</td>
<td>0.024428</td>
</tr>
<tr>
<td>0.1</td>
<td>0.9</td>
<td>0.035346</td>
<td>0.026575</td>
<td>0.035873</td>
<td>0.034199</td>
<td>0.044206</td>
</tr>
</tbody>
</table>

Table 5

The solution results for bi-objective dynamic portfolio – weighting approach $\beta_1 = 0.9$, $\beta_2 = 0.1$

Table 6

<table>
<thead>
<tr>
<th>$\beta_1$</th>
<th>$\beta_2$</th>
<th>$\gamma_{\text{Var}}$</th>
<th>$\gamma_{\text{Min}}$</th>
<th>-100</th>
<th>$\tau_{\text{BH}}$</th>
<th>15(2,335) quotations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>0.024353</td>
<td>0.007091</td>
<td>0.005971</td>
<td>0.003194</td>
<td>0.024428</td>
</tr>
<tr>
<td>0.1</td>
<td>0.9</td>
<td>0.035346</td>
<td>0.026575</td>
<td>0.035873</td>
<td>0.034199</td>
<td>0.044206</td>
</tr>
</tbody>
</table>

Table 7

Problem size after presolving

<table>
<thead>
<tr>
<th>Number of constraints</th>
<th>Number of binary variables</th>
<th>Number of linear variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>12637</td>
<td>6674</td>
<td>8209</td>
</tr>
</tbody>
</table>

* CPU seconds for proving optimality on a laptop with Intel® Core 2 Duo T9300 processor running at 2.5 GHz and with 4 GB RAM /CPLEX v.11
In the tables, column “MIP simplex iteration” shows the number of mixed integer programming simplex iterations until presented solution. Column “B-&-B” shows the number of searched nodes in the branch and bound tree until presented solution. Column “CPU” shows CPU seconds required for proving optimality on a laptop with Intel® Core 2 Duo T9300 processor running at 2.5 GHz and with 4 GB RAM using the solver CPLEX v.11.

4. Conclusions

The presented extension of the quadratic Markowitz model, in which the variance has been replaced with the Value-at-Risk leads to NP-hard mixed integer program problem. The short-selling variables may take on non-negative values only due to the dynamic balance constraint introduced. The computational experiments modeled on a real data from the Warsaw Stock Exchange have indicated that the approach is capable of finding optimal solutions for medium size problems in a reasonable computation time using commercially available software for mixed integer programming.

The total computation time ranges from a few seconds to minutes or even hours depending on the number of historical quotations in the optimization problem.

References

