Abstract

The focus of this paper is to present an intuitive end-user Decision Support System (DSS) for portfolio selection based on Mean-Variance (M-V) Model of portfolio selection by Markowitz [1952, 1991]. The DSS utilizes a Goal Linear Programming (GLP) model for fulfilling the investor's objectives and preferences in terms of rate of return, risk and asset allocation and diversification in order to reach an optimum solution. The DSS reported is flexible enough and could be extended through goal extension, extending the goal achievement requirements by adding more goals and using other models within the computation intelligence paradigm such as ANN, Fuzzy, Rough Sets, Genetic Algorithm (GA), etc. The DSS is implemented within an objective-oriented paradigm using Microsoft VB.NET. The implementation of the GLP model for the portfolio selection was done using the Extended Large-Scale LP/QP Solver within Solver Platform SDK from Frontline Systems, Inc. (www.solver.com).

1. Introduction

Over the last few decades since the introduction of the Mean-Variance (M-V) Model of portfolio selection by Markowitz [1], [2], and Sharpe [3], [4], significant number of research studies has been reported on Portfolio Selection Theory and its application. These studies show various solutions using techniques within the Operations Research (OR) paradigm such as Multi-Objective and Goal Programming, computational intelligence techniques such as Fuzzy Logic, RST, Neural Network and Machine Learning, or a hybrid approaching combining OR techniques with computational intelligence ones.

In this paper we present an intuitive, iterative and interactive environment that would allow the investor to present his/her investment objectives, create various scenarios using a what-if analysis, execute these scenarios and then replot and evaluate the scenarios in order to select the best scenario that would fulfill his/her objectives. These objectives are defined in terms of return required, risk to be taken and the preference for diversification in terms of investment in a stock or a sector. The representation and fulfilments of these goals and objectives are achieved using a goal linear programming that would lead to an optimum solution. Extension to this basic and intuitive system is easily achievable through:

- Goal extension, extending the goal achievement requirements by adding more goals.
- Using Other Models within the computation intelligence paradigm such as ANN, Fuzzy, Rough Sets, etc.

The remainder of this paper is organized as follows. Section 2 covers the literature review, Section 3 covers Problem Description and Formulation. Section 4 presents the System Architecture, Implementation and Functionality, Section 5 presents the Experimentation And Results and finally section 6 covers the Conclusion and Future Research.

2. Literature Review

As indicated above, following the introduction of the Mean-Variance (M-V) Model of portfolio selection by Markowitz [1], [2], and Sharpe [3], [4], there exists an extensive literature on Portfolio Selection Theory and its application. In this section, we cover selected articles covering solutions and techniques, such as Multi-Objective and Goal Programming [4][5], and some of the recent work utilizing computational intelligence techniques such as Fuzzy Logic, RST, Neural Network and Machine Learning, or a hybrid approaching combining OR techniques with computational intelligence[6],[7], and [8] as shown below. For a categorized bibliographic study on the application of the techniques of multiple criteria decision making (MCDM) to problems and issues in finance see Steuer [5]. The bibliography is classified based on the methodological approaches Goal programming, Multiple objective programming, Multi-attribute utility analysis, Multi-criteria decision analysis, and the analytic hierarchy process and the application areas covered are capital budgeting, working capital management, portfolio analysis, etc. Some research studies are reviewed below.

In his article Ben Abdelaziz, et al [9], proposes a chance constrained compromise programming model (CCCP) as a deterministic transformation to multi-objective stochastic programming portfolio model. CCCP is based on CP and chance constrained programming (CCP).
models. The proposed program is uses the Tunisian Stock Exchange market.

Ballestero, et al [10] indicates that in order to prevent the potential flaws that past data a good indicator for the future performance and the necessity to mathematically determine the investor’s utility function, they propose a methodology whose initial phase is to filter portfolios that are inefficient from a historical perspective and uses a decision table constructed by considering multiple scenarios assuming strict uncertainty. The cells in the decision tables measure consequences by a multi-criteria linear performance index of simulated future returns. The propose approach was tested in two studies using data from the stock exchanges in Frankfurt and Vienna.

Lin and Hsiech [11] showed that based on the concepts of decision support system (DSS), an integrated framework that incorporates fuzzy theory into strategic portfolio selection was developed. The framework provides managers with a flexible, expandable and interactive DSS to select projects for portfolio management.

Ji, et al [12] proposed a stochastic linear goal programming model for multistage portfolio management is proposed. The model takes into account both the investment goal and risk control at each stage. A scenario generation method is proposed that acts as the basis of the portfolio management model. Scenarios for multistage portfolio management are generated by incorporating this single-stage method with the time-series model for the asset returns.


Petkov, et al. [14] presents the combination Multiple Criteria Decision Making (MCDM) and techniques from soft systems approaches for decision support at particular stages of complex problem solving.

Samara, et al [15] presents a multicriteria decision support system for Athens Stock Exchange. The system evaluates the stocks based on the method of fundamental analysis ratio. The system utilizes multicriteria analysis methodologies in order to rank the stocks by placing the best stock first and the worst last.

Tseng [16] reports a real time dss for information gathering and managing of investment portfolio using an Object-Oriented Bayesian Knowledge Base (OOBKB) which was design for the creation of decision models to guide the information gathering and to produce investment recommendations.

Garlappi [17] reports of the development of a model for an investor with multiple priors and aversion to ambiguity the multiple priors was characterized by a "confidence interval" around the estimated expected returns and ambiguity aversion was modeled via a minimization over the priors. Garlappi [18] proposed an integrated methodological framework for the evaluation of Mutual Fund performance. The proposed methodology is based on the combination of discrete and continuous multicriteria decision aid (MCDA) methods for MFs selection and composition.

Li and et al. [19] indicate that since the future returns cannot be predicted based on the historical data, a novel portfolio selection model in a hybrid uncertain environment in which the returns of securities are assumed to be fuzzy random variables and then following the ideas of mean variance model was proposed.

Ko [20] consider portfolio selection as a resource allocation problem in a finance market and as such investor’s asset optimization necessitate the distribution of a set of capital (resources) among a set of entities (assets) with the trade-off between risk and return. They present a resource allocation neural network model that would optimize investment weight of portfolio.

Other research studies using computation intelligence are Lin and Liu [21] which presents three possible models for portfolio selection problems with minimum transaction lots, and devises corresponding genetic algorithms to obtain the solutions, Shyng, et al. [7] proposes a novel Forward Search and Backward Trace (FSBT) technique based on Rough Set Theory (RST), and Huang [8] proposes a technique based upon Fuzzy C-Means (FCM) classification theory and related fuzzy theories for choosing an appropriate value of the Variable Precision Rough Set (VPRS) threshold parameter.

3. The Problem Description and Formulation

3.1. The Problem Description

The general goal linear programming model formulation is presented in this section followed by a detailed description of the specific goal linear programming model that is applied to the portfolio selection. Finally, a detailed description of the necessary input data required in this application is also covered. The goal linear programming model presented here encode the requirements of an investor for portfolio selection. The model contains different goals with different priorities.

3.1.1. General Goals Linear Programming (GLP) Model Formulation

A goal is a desired result. It may be underachieved, fully achieved, or overachieved. Relative emphasis applied through managerial action contributes to the degree of goal achievement. Symbolically, 1 unit of effort applied to activity $x_j$ might contribute an amount
$. \quad a_{ij}$ toward the achievement of the $i^{th}$ goal. The target level for the $i^{th}$ goal, $b_i$, is fully achieved if
\[
\sum_{j=1}^{n} a_{ij} x_j = b_i
\]
to allow for underachievement or overachievement, let
\[
d_i^- = \text{negative deviation from the } i^{th} \text{ goal (underachievement)}
\]
\[
d_i^+ = \text{positive deviation from the } i^{th} \text{ goal (overachievement)}
\]
from this, for the $i^{th}$ can be stated in general terms as
\[
\sum_{j=1}^{n} a_{ij} x_j + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \ldots, m
\]
It is required that one or both of the deviational variables $(d_i^-, d_i^+)$ be zero in the solution, since it is not possible for both underachievement and overachievement to occur at the same time.

The goal programming model must be able to incorporate goal attendant with both ranking and weighting as appropriate. Let
\[
P_k = \text{Ranking coefficient for all deviations having } k^{th} \text{ priority of being avoided.}
\]
\[
w_k^- = \text{Relative weight of the } d_i^- \text{ in the } k^{th} \text{ rank}
\]
\[
w_k^+ = \text{Relative weight of the } d_i^+ \text{ in the } k^{th} \text{ rank}
\]
With $m$ goals, the goal linear programming model may be formulated mathematically as requiring the minimization of the linear weighted ranking function
\[
\sum_{i=1}^{m} \sum_{k} P_k (w_k^- d_i^- + w_k^+ d_i^+)
\]
Subject to the linear constraints
\[
\sum_{j=1}^{n} a_{ij} x_j + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \ldots, m
\]
\[
x_j, d_i^-, d_i^+ \geq 0 \quad i = 1, 2, \ldots, m, \quad j = 1, 2, \ldots, n
\]
Goal programming seeks satisfactory level of goal attainment that represents the best possible overall outcome. This aim can be realized if the several goal are stated and formulated mathematically in a manner that permit appropriate consideration in the goal programming model. Therefore, the decision maker must determine which goal are most important and assign an ordinal value (priority), $P_k$, to each goal or group of goals.

Further, within a given goal, there may be sub goals of unequal importance which must be given different weights, $w_{ik}^-, w_{ik}^+$. Solving any goal programming model involves achieving highest priority goal, first, before any of the lower priority goals are considered. Once the highest priority goal is attained to fullest extent possible, the goal programming model proceeds to find a satisfactory level to next highest priority goal, and so on. However, it is not always possible to achieve every goal to the extent desired by the decision maker. But the advantage of goal linear programming over ordinary linear programming is that it seeks, within the given set of constraints, to minimize the deviation from the established goals targets, while the ordinary linear programming seeks to minimize or maximize certain goal directly.

### 3.1.2. Goal Linear Programming Formulation for Portfolio Selection

#### Notations
- $A$ = Total amount of money in Kuwaiti Dinar (KD) that the investor wants to invest in his/her portfolio.
- $X_{ij}$ = The amount of money that will be invested in stock $i$ in sector $j$.
- $j_m$ = the number of stocks in sector $j$.
- $i_j$ = The return percent of stock $i$ in sector $j$.
- $\beta_{ij}$ = The risk of stock $i$ in sector $j$.
- $\beta$ = The market risk.
- $R$ = The portfolio returns that the investor expects.
- $L$ = Large number.

#### Goal 1: Amount Invested
The total amount the investor wants to invest exactly $A$ KD (KD stands for Kuwaiti Dinar which is equivelant to approximately $3.4) and the priority for this goal is $P_1$. This goal can be represent as follows:
\[
\sum_{j=1}^{m} \sum_{i=1}^{n} X_{ij} x_j + d_i^- - d_i^+ = A
\]
and the objective is to minimize $d_i^-$ and $d_i^+$.

#### Goal 2: Return
The total return of the portfolio the investor desires should not be no less tha $R\%$ and the priority for this goal would be $P_2$. This goal can be represent by
\[ \sum_{j=1}^{m_j} R_{xy} + d_{2j} - d_2^+ = R \times A \quad (2) \]

And the objective is to minimize \( d_2^- \).

**Goal 3: Risk**

The total portfolio risk should not be more than the average risk or \( \beta \) of the market and the priority for this goal would be \( P_3 \). This goal can be represent by
\[ \sum_{j=1}^{m_j} \beta_{xy} + d_3^- - d_3^+ = \beta \times A \quad (3) \]
And the objective is to minimize \( d_3^- \).

**Goal 4: Stocks Allocation**

This goal relates to the portion of investment in each stock. This goal related to the portion of investment in each sector. In developed markets where the number of stocks are in thousands, the portion of investment in each stocks is between 2%-10% while in emerging markets that portion of investment is between 5%-25% depending on the number of companies listed in the stock exchange. For Kuwaiti market we have decided that the portion be 10%. The priority for this goal would be \( P_4 \) and the goal can be represented as follows:
\[ X_{xy} + d_{4ij}^- - d_{4ij}^+ = 0.10A \quad i = 1,2,\ldots,m \]
\[ and \quad j = 1,2,\ldots,n \]
And the objective is to minimize \( d_{4ij}^- \), \( i = 1,2,\ldots,m \) and \( j = 1,2,\ldots,n \).

**Goal 5: Sectors Allocation**

This goal related to the portion of investment in each sector. In developed markets where the number of sectors is greater than 10 sectors, the portion of investment in each sector is between 5%-20% while in emerging market where the number of sector is less than 10, this portion is between 5%-40% and these min and max percentages were used for the Kuwaiti market. The priority for this goal would be \( P_5 \) and the goal can be represent as follows:
\[ \sum_{i=1}^{m_j} X_{ij} + d_{5ij}^- - d_{5ij}^+ = 0.05A \quad j = 1,2,\ldots,n \]
\[ \sum_{i=1}^{m_j} X_{ij} + d_{6ij}^- - d_{6ij}^+ = 0.40A \quad j = 1,2,\ldots,n \]
\[ \sum_{i=1}^{m_j} X_{ij} + d_{5ij}^- - d_{5ij}^+ = 0.40A \quad j = 1,2,\ldots,n \]
\[ \sum_{i=1}^{m_j} X_{ij} + d_{6ij}^- - d_{6ij}^+ = 0.40A \quad j = 1,2,\ldots,n \]
\[ \sum_{i=1}^{m_j} X_{ij} + d_{5ij}^- - d_{5ij}^+ = 0.40A \quad j = 1,2,\ldots,n \]
And the objective is to minimize \( d_{5j}^- \) and \( d_{6j}^- \).

**Goal 6: Maximization of Return**

This goals allows for the maximization of return beyond the preferred level set in goal 2. \( L \) is a very large unattainable number. The priority for this goal would be \( P_6 \) and the goal can be represent as follows:
\[ \sum_{j=1}^{m_j} R_{xy} + d_7^- - d_7^+ = L \quad (7) \]
And the objective is to minimize \( d_7^- \).

**GLP:**

\[ \text{Minimize } P_1(d_1^+ + d_1^-) + P_2d_2^- - d_2^+ + P_3d_3^- + P_4n_{i=m_j}d_{ij} + P_5(n_{j=1} + n_{j=1}d_{ij}) + P_6d_7^- \]
Subject to:
\[ P_1: \sum_{j=1}^{m_j} X_{ij} + d_1^- - d_1^+ = A \]
\[ P_2: \sum_{j=1}^{m_j} R_{xy} + d_2^- - d_2^+ = R \times A \]
\[ P_3: \sum_{j=1}^{m_j} \beta_{xy} + d_3^- - d_3^+ = \beta \times A \]
\[ P_4: X_{ij} + d_{4ij}^- - d_{4ij}^+ = 0.10A \quad i = 1,2,\ldots,m \]
\[ \text{and } j = 1,2,\ldots,n \]
\[ P_5: \sum_{j=1}^{m_j} X_{ij} + d_{5j}^- - d_{5j}^+ = 0.05A \quad j = 1,2,\ldots,n \]
\[ P_6: \sum_{j=1}^{m_j} R_{xy} + d_7^- - d_7^+ = L \]

An End-User Decision Support System for Portfolio Selection
4. The System Architecture, Implementation and Functionality

4.1. System Architecture and Functionality

Figures 1 shows the main components and architecture of the decision support system. The data management component is responsible to handle the database requirements of the DSS. The model management is responsible to managing the models used in the DSS in case of using multiple models or approaches for the portfolio selection in the DSS. Currently the model management handle a GLP model for portfolio selection as described in the previous section. The rest of the components of the DSS are described as follows along with their functionality:

![Decision Support System Components](image)

4.1.1. Data Preparation

Step 1: Data Cleansing
Fixing The Data: This function consists of loading the data from excel files that was captured from KSE site into a master record containing all the companies within KSE. And deleting any daily transaction that contain missing data or incomplete data.

Step 2: Computation
A group of activities consisting of the following functions:

1. Active Stock Selection: Selecting stocks with certain number of trading days per year. Stocks that are traded (i.e. their volumes per day was greater than zero) for certain number of days per year are selected. For KSE, the cut of point was 120 trading days per year. Using this cut of point we were able to select 59 companies in 6 sectors.
2. Return Computation: (see [22] and [23])
   a. Return computation consists of Computation of Weighted Holding Period Yield (WHPY) for each stock for certain number of years is computed as follows:
      \[
      WHPY_j = \frac{\sum_{i=1}^{n} (Yearly \ HPY_i \cdot Yearly \ Total \ Cap_i)}{\sum_{i=1}^{n} Yearly \ Total \ Cap_i}
      \]
      where \(WHPY_j\) is Weighted Holding Period Yield for the \(j^{th}\) stock, \(m\) is the number of years, \(Yearly \ HPY_i\) is the yearly Holding Period Yield for the \(i^{th}\) year, \(Yearly \ Total \ Cap_i\) is the total sales for that particular year (sum of daily value for the entire year). And where the general formula for computing Yearly HPY, is as follows:
      \[
      HPY = \frac{Current \ Period \ Closing \ Price}{Previous \ Period \ Closing \ Price} - 1
      \]
   b. Computation of the Market HPY
3. Risk Computation: The risk computation consists of BETA Computation for each stock using the following formula (Monthly, Yearly, and Summary)
   - Risk (BETA) Computation
     Regression Coefficient (Rate of change in Y with regard to X):
     Sharpe’s (1964) beta regression is computed for each stock and for the market as a whole using the following formula
     \[
     Beta_j = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}, \quad i = 1, 2, ..., n
     \]
     where \(n\) is the number of days, \(x_i\) is the Market HPY for \(i^{th}\) day, \(\bar{x}\) is the market mean HPY for \(n\) days, \(y_i\) is the stock HPY for \(i^{th}\) day and \(\bar{y}\) is the stock mean HPY for the same period.
4.1.2. Scenario Creation

Scenario creation allows the user to enter the main parameters that are necessary to create various scenarios and where by what-if analysis or sensitivity analysis becomes possible. These parameters, as shown in Figure 3, are as follows:

1. Investment Amount
2. Return
3. Risk
4. Percentage of Allocation for Each Stock
5. Minimum Percentage Allocation for Each Sector
6. Maximum Percentage Allocation for Each Sector

These parameters are stored in a unique record in the database.

4.1.3. Scenario Execution

Scenario Execution reads the parameters for the selected scenario from the database, create an instance of the GLP model, set its parameters and executes the model. The result of the execution of the model is stored in a unique record. And particular scenario record is updated with the Return Achieved (as specified above). Figure 4 shows the scenario execution.

4.1.4. Scenario Reporting and Evaluation

Scenarios are evaluated based on the extend to which investor’s preferences have been met and the total return achieved. If scenario was not acceptable, Scenario Creation would be reinvoked with different parameters values and the model would be executed again. Scenario(s) that would fulfill the goals and the constrains as defined by the users and with the highest Return Achieved, as computed bellow, are selected and its output would be send to a spreadsheet, e.g. MS Excel, for further analysis.

\[
\text{Return Achieved} = \sum_{i=1}^{\text{con}} W_{HPY} \times \text{AmntInvestInStock}_i
\]
4.2. Implementation

The Portfolio Selection Decision Support System was implemented within an object-oriented paradigm using VB.Net platform. For data management, MS ACCESS was used. ACCESS is widely available on every PC with MS Office installed. ACCESS is very easy to use and easy to manage and exporting Access databases to more powerful DBMSs such SQL Server or Oracle Server is very easy as well. The Portfolio Selection Goal Programming Model was implemented using Premium Solver SDK from Frontuir Company (www.solver.com). Premium Solver SDK consists of a set of object-oriented APIs callable from a VB.Net environment and provides the same functionality as that of EXCEL SOLVER. The Premium SOLVER version used allows for the creation of models with unlimited number of variables and constraints. Figures 2, 3, 4, and 5 show screen shots of the system functionality.

5. Experimentation

5.1. Data Set Characteristics

Kuwait Stock Exchange(KSE) was established in 1962 and it is the oldest stock exchange in the Gulf Cooperative Council (GCC) region with around 189 listed companies distributed sectorwise as follows:

1. Banking: 9
2. Investment: 46
3. Insurance: 7
4. Real Estate: 36
5. Industrial: 28
6. Services: 57
7. Food: 6

The daily movements of each stock related to these 7 sectors were downloaded from KSE site for the period 2000 to 2008 and imported from the EXCEL files to the ACCESS DBMS. However, lack of market index data for the years prior to 2002, forced us to limit the coverage of our dataset to the years 2002 to 2008. A part from data cleansing that we have performed which eliminated daily stock movements with missing or incomplete data, we have also eliminated from the dataset those companies with less than 120 days of trading per year. In our selection criteria, a day is considered a trading day if the volume, i.e. no. of transactions, was greater than zero and the decision for selecting a cutoff point of 120 trading days, was made from the trading data of each stock. With a cut of point of 120 tradings only 60 companies were selected from 6 sectors from 189 companies registered in KSE in 7 sectors. A higher cut off point would have given us a smaller number of companies. A 60 companies sample, in our opinion, represents a relatively large enough sample of companies representing all sector except one sector as shown in Table 1.

Table 1. Active Stocks

<table>
<thead>
<tr>
<th>No.</th>
<th>Sector</th>
<th>Stk No</th>
<th>Ticker</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Banking</td>
<td>101</td>
<td>NBK</td>
</tr>
<tr>
<td>2</td>
<td>Banking</td>
<td>102</td>
<td>GBK</td>
</tr>
<tr>
<td>3</td>
<td>Banking</td>
<td>103</td>
<td>CBK</td>
</tr>
<tr>
<td>4</td>
<td>Banking</td>
<td>104</td>
<td>ABK</td>
</tr>
<tr>
<td>5</td>
<td>Banking</td>
<td>105</td>
<td>BKME</td>
</tr>
<tr>
<td>6</td>
<td>Banking</td>
<td>106</td>
<td>KIB</td>
</tr>
<tr>
<td>7</td>
<td>Banking</td>
<td>107</td>
<td>BURG</td>
</tr>
<tr>
<td>8</td>
<td>Banking</td>
<td>108</td>
<td>KFIN</td>
</tr>
<tr>
<td>9</td>
<td>Investment</td>
<td>201</td>
<td>KINV</td>
</tr>
<tr>
<td>10</td>
<td>Investment</td>
<td>202</td>
<td>FACIL</td>
</tr>
<tr>
<td>11</td>
<td>Investment</td>
<td>203</td>
<td>IFA</td>
</tr>
<tr>
<td>12</td>
<td>Investment</td>
<td>204</td>
<td>NINV</td>
</tr>
<tr>
<td>13</td>
<td>Investment</td>
<td>205</td>
<td>KPROJ</td>
</tr>
<tr>
<td>14</td>
<td>Investment</td>
<td>206</td>
<td>AINV</td>
</tr>
<tr>
<td>15</td>
<td>Investment</td>
<td>209</td>
<td>SECH</td>
</tr>
<tr>
<td>16</td>
<td>Investment</td>
<td>210</td>
<td>IIC</td>
</tr>
<tr>
<td>17</td>
<td>Investment</td>
<td>211</td>
<td>SGC</td>
</tr>
<tr>
<td>18</td>
<td>Investment</td>
<td>213</td>
<td>KFC</td>
</tr>
<tr>
<td>19</td>
<td>Investment</td>
<td>214</td>
<td>KMEFI</td>
</tr>
<tr>
<td>20</td>
<td>Investment</td>
<td>216</td>
<td>AIG</td>
</tr>
<tr>
<td>21</td>
<td>Investment</td>
<td>217</td>
<td>TID</td>
</tr>
<tr>
<td>22</td>
<td>Investment</td>
<td>219</td>
<td>ALOLA</td>
</tr>
<tr>
<td>23</td>
<td>Investment</td>
<td>220</td>
<td>ALMAL</td>
</tr>
<tr>
<td>24</td>
<td>Investment</td>
<td>221</td>
<td>GIH</td>
</tr>
<tr>
<td>25</td>
<td>Real Estate</td>
<td>402</td>
<td>URC</td>
</tr>
<tr>
<td>26</td>
<td>Real Estate</td>
<td>403</td>
<td>NRE</td>
</tr>
<tr>
<td>27</td>
<td>Real Estate</td>
<td>405</td>
<td>PEARL</td>
</tr>
<tr>
<td>28</td>
<td>Real Estate</td>
<td>406</td>
<td>TAM</td>
</tr>
<tr>
<td>29</td>
<td>Real Estate</td>
<td>409</td>
<td>MREC</td>
</tr>
<tr>
<td>30</td>
<td>Real Estate</td>
<td>410</td>
<td>ARABREC</td>
</tr>
<tr>
<td>31</td>
<td>Real Estate</td>
<td>411</td>
<td>UREC</td>
</tr>
<tr>
<td>32</td>
<td>Real Estate</td>
<td>412</td>
<td>ERESCO</td>
</tr>
<tr>
<td>33</td>
<td>Real Estate</td>
<td>413</td>
<td>MABANEE</td>
</tr>
<tr>
<td>34</td>
<td>Real Estate</td>
<td>414</td>
<td>INJAZZAT</td>
</tr>
<tr>
<td>35</td>
<td>Industrial</td>
<td>501</td>
<td>NIND</td>
</tr>
<tr>
<td>36</td>
<td>Industrial</td>
<td>503</td>
<td>KCEM</td>
</tr>
</tbody>
</table>
5.2. Analysis and Results

Ten Scenarios were created and executed with various parameters as shown in Table 2. Table 2 also shows the Return Achieved for each scenario. Figures 6, 7, 8, 9, and 10 depicting 5 of the scenarios generated. The figures show the parameters, Return Achieved, and Investment Result, i.e. the output of the execution of the model showing each sector, the stock ticker, stock WHPY, the stock Beta, and the amount invested in the stock. In executing Scenario 1, goal 6, which allows for maximum return achievement, was invoked. In executing the rest of the scenarios, i.e. 2-10, goal 6 was passed.

Table 2. Scenarios Parameters and Return Achievements
6. Conclusion and Future Research

The paper presents an intuitive end-user Decision Support System (DSS) for portfolio selection based on Mean-Variance (M-V) Model of portfolio selection by Markowitz [1952, 1991]. The DSS utilizes a Goal Linear Programming (GLP) model for fulfilling the investor’s objectives and preferences in terms of rate of return, risk and asset allocation and diversification in order to reach an optimum solution. The DSS is implemented within an objective-oriented paradigm using Microsoft VB.NET and the implementation of the GLP model for the portfolio selection was done using the Extended Large-Scale LP/QP Solver within Solver Platform SDK from Frontline Systems, Inc. (www.solver.com).

Ten scenarios were generated using stock movements of 60 active companies covering the period between 2002-2008. Selection of a scenario was based on the degree of the goals fulfilsment and the level of return achievement.

Future research could be focused on developing and utilizing other models within the computation intelligence paradigm and a hybrid model consisting of OR and CI techniques.

References


