Using Genetic and Particle Swarm Algorithms to Select and Optimize Portfolios of Companies Admitted to Tehran Stock Exchange

Abbas Toloie- Eshlaghy
Corresponding Author, Industrial Management Department
Science and Research Branch, Islamic Azad University, Tehran, Iran
E-mail: toloie@gmail.com
Tel: +98 912 310 87 56

Ahmad Abdolahi
M.A. and Faculty Member of Golestan High Education Institute, Golestan, Iran
E-mail: ahmabdollahi@gmail.com

Motahare Moghadasi
M.A. and lecturer in Golestan Branch, Islamic Azad University, Golestan, Iran
E-mail: Moghadasi@yahoo.com

Alireza Maatofi
Financial management Department
Gorgan Branch, Islamic Azad University, Gorgan, Iran
E-mail: alirezamaetoofi@gmail.com

Abstract
Selecting and optimizing stock portfolio is one of the main concerns of investors and dealers in the stock market. The present study aims to select an optimized portfolio using genetic and particle swarm algorithms and comparing the related results. The statistical sample included 146 companies active in the Tehran Stock Exchange. Considering monthly share prices of the companies between 2001 and 2009, a model was developed to select and optimize their stock portfolios. In this line, through adding a number of the real world limitations to the mean-semi variance base model of Markowitz, the developed model was prepared and its related algorithms were designed using MATLAB software. Finally, the algorithms were carried out for different sizes of stock portfolios. Results of the study prove high stability and optimization for both genetic and particle swarm algorithms in different frequencies and acceptable times. Also, results of the study indicate that the particle swarm algorithm has higher speed compared to the genetic algorithm, both in number of generations and implementation time.

Keywords: Optimizing stock portfolio, genetic algorithm, particle swarm algorithm, semi variance, risk and efficiency.

1. Introduction
The process of selecting and optimizing stock portfolio is one of the main concerns of investors and dealers in the stock market. Stock portfolio optimization and diversification are the two tools to expand and understand financial markets and decision-making. Introducing the stock portfolio selection theory
of Harry Markowitz was the key success and achievement in this regard. It is said that Harry Markowitz introduced the stock portfolio optimization theory and it is for sure that the heart of the stock portfolio optimization issue is an investor his/her profit is dependent on his/her expected efficiency and stock portfolio risk which is derived by variance.

But, the fact that has not been paid so attention was that Markowitz showed interest toward another definition of risk, which was semi variance. In fact, Markowitz allocated a chapter of his book to semi variance, so that he said: “It seems that analysis based on semi variance creates better portfolios than analysis based on variance.” Part of the corrected edition of his book in 1991 reads: “Semi variance is the most justified method for estimating risk.”

The Markowitz model can be solved using mathematical programming models. But, with adding limitations of the real world, high number of assets, weight limitations of stocks and so on, the search space will become very large and discontinuous, which will make using mathematical models impossible. So, inventive algorithms such as genetic, neural network, ant’s algorithm, etc. become very important. In recent decades, inventive algorithms have been widely used in solving stock portfolio optimization problems due to their high capabilities of these algorithms to solve the problems. Considering the successful performance of inventive algorithms in solving optimization problems, such algorithms can provide investors with appropriate methods to optimize their stock portfolios. So, the main purpose of the present study is to select and optimize stock portfolios once using the genetic algorithm and once using the particle swarm algorithm and then comparing related results.

Many researches have been conducted in this regard, but a few of them have compared the two algorithms. Filho (2006) conducted a study to compare the particle swarm algorithm with the genetic algorithm in line with risk management in a limited stock portfolio optimization problem, if short-term selling is not allowed. Results of the study showed that both methods can find appropriate solution in acceptable times. But, the particle swarm algorithm showed to be more rapid than the genetic algorithm. Sensitivity of the particle swarm algorithm compared to the initial location of particles was higher than the genetic algorithm. Mishra et al. (1009) carried out a comparative study on four inventive algorithms to gain efficient weights of stock optimization problem. They used four inventive algorithms of single front (PSFGA), strengthened Pareto algorithm (SPEA2), non dominated sorting genetic algorithm (NSGA) and multiple objective particle swarm optimization (MOPSO) algorithm. Final results showed higher efficiency of the particle swarm algorithm in comparison to other algorithms.

In another research by Tunçhancı (2009) used the particle swarm algorithm to solve the stock portfolio optimization problem. He considered the mean-variance model with limitations of integers and used the particle swarm algorithm for solving the optimization problem. Then, he compared results of the algorithm with results of genetic, cooling simulation and tabu search algorithms. Results showed efficiency and success of the particle swarm algorithm in the stock portfolio optimization issue. Xu and Chen (2009) decided to solve the stock portfolio selection problem, which is a complicated linear programming issue, with the least transaction cost. So, they introduced a developed particle swarm algorithm to solve the problem.

In conclusion, a numerical example was used to show efficiency of the introduced method and the related results were compared with results of implementing the genetic algorithm. The results showed that the developed particle swarm algorithm was apparently better than the standard genetic algorithm. Chang (2009) compared the genetic and the particle swarm algorithms as an efficient way to build stock portfolios with equal weights. The genetic and the particle swarm algorithms were used to fins the optimum capital allocation for building stock portfolios with equal weights. Related results showed that the stock portfolio which was gained using the particle swarm algorithm was better than algorithms which were gained using the genetic algorithm.
2. Genetic Algorithm

The genetic algorithm is an inventive method of searching which uses the natural evolvement and survival theories for solving related problems. Using the word survival implies a negative meaning. Maybe it implies the jungle law and the idea that the most powerful always survives. In fact, it is better to say that the most appropriate is selected in the nature. The genetic algorithm was firstly introduced by Holland and colleagues in the University of Michigan. The Holland’s article was titled “adaptability in natural and artificial systems.” Goldberg extended the method in 1989. The genetic algorithm is a programming technique which uses genetic evolvement as a pattern to solve problems.

The problem which is to be solved is the input and solutions are coded based on a pattern. The fitness assessment function will evaluate every proposed solution. Most of the solutions are selected randomly. The basic idea of the genetic algorithm is the transfer of hereditary traits by genes. The genetic algorithm is a probable searching method which acts on a population of potential answers to produce better estimates of an answer (chromosome).

The operation method of the genetic algorithm is charmingly simple and understandable. It says that the man believes that animals have been evolved based on the same principle. To explain the evolvement process of animals, it should be said that the genetic algorithm starts to work by a string of initial answers, named chromosome, and a set named the initial population.

Each chromosome is connected to a fitness function which indicates the optimal level of the answer. Through applying the three operators of selection, crossover and mutation on chromosomes, the answer set is converged toward the best generation.

The genetic algorithm is a mathematical tool which has eased stock portfolio optimization based on the modern portfolio theory. In 1990, Koza and Goldberg created vast improvements in the genetic algorithm, so that the algorithm was rapidly use din a broad range of fields, from music to horse farming. Financial applications of the genetic algorithm were also developed at that time. In 1994, Bauer presented a useful summary of the genetic algorithm applications and a significant improvement in stock portfolio optimization using the genetic algorithm was obtained was introduced by Streichert in 2003. Although basics and findings of the theory are not new, but it can still solve many problems in special fields such as stock portfolio optimization.

The dynamic behavior of the stock market and its specific complications on one hand and the ability of the genetic algorithm to crate optimized structures on the other hand have led to efforts toward modeling the algorithm for stock markets and creating optimized stock portfolios.

The simplest case of using the genetic algorithm in stock portfolio optimization includes five stages:

- In the first stage the investor randomly selects a number of portfolios from among portfolios in the stock market. Each portfolio introduces a chromosome and each stock in the portfolio introduces one of the chromosome’s genes.
- In the second stage each portfolio is assessed based on a number of criteria such as capital gain, risk, efficiency and so on compared with other market indices and each portfolio is given a specific score. The criteria which are used in this stage are shown as a mathematical function named the target function or the optimized function and are applied on each of the portfolios (chromosomes).
- In the third stage, chromosomes (portfolios) with higher scores are allowed to be produced more and chromosomes with lower scores are eliminated. In this stage, selection, crossover and mutation operators are used to produce new populations.
- In this stage, weaker populations are newly eliminated and are replaced with stronger populations. The program is repeated again and again to finally create the optimized stock portfolio.
- In this stage, if the halt condition is met, the program will be stopped and the selected portfolio is introduced as the answer and the optimized stock portfolio. Otherwise, the procedure will be repeated as of the stage 2.
3. Particle Swarm Algorithm

The particle swarm algorithm was introduced by Kennedy and Eberhart in 1995. The particle swarm algorithm directs searching using a set of random solutions proportional to particles (birds). Moreover, to each potential solution, which is called particles, a random acceleration is allocated. This algorithm operates parallel like the genetic algorithm. It means that it is able to search for answer spaces simultaneously in different places. The ability to solve problems with complicated answer environment and discontinuous functions is among the algorithm’s advantages. In the particle swarm algorithm, a group of birds are searching for food in a vast area and the food is located just in a certain point and the birds are unaware of the issue. They only know their distance from the food.

In this algorithm, each solution is called a bird or a particle. Each particle has a fitness value which is gained by its fitness function. The bird which is nearer to the food has a higher fitness value. In the particle swarm algorithm, each particle flies in a multidimensional space and adjusts its position considering its own and its neighbors’ experiences and selects its route considering direction and route:

- The best position
- The best position of neighbors in each repetition

A population from potential solutions is sorted in the frame of vectors with real values. In each generation, a solution is updated through adding another vector which is called the fitness vector. The fitness is changed during the optimization process. Different strategies are used in this algorithm. One of the strategies is the bumping strategy which does not allow particles (birds) to exit pre-defined borders. Each particle which quits borders will be set a volume once again. Amnesia strategy is another strategy in which particles are allowed to exit the borders, but values which are gained out of the borders will not be considered as the optimized values and are forgotten. The penalty function is another strategy which considers a penalty for each value out of the borders which in fact neutralizes the optimized value defined out of the borders. Random positioning is another strategy. Due to reducing the searching space, it does not contain waste of time and energy, but we may be involved in local optimum values.

4. Research Modeling

The Markowitz model is one of the most applied models for stock portfolio selection.

\[
\min z = \sum_{i=1}^{N} \sum_{j=1}^{n} \omega_i \omega_j \delta_{ij}
\]

s.t.:

\[
\sum_{i=1}^{n} \omega_i \mu_i \geq R
\]

\[
\sum_{i=1}^{n} \omega_i = 1
\]

\[
\omega_i \geq 0, i = 1,2,3,\ldots,N
\]

So that R is the favorable gain of the investor. The model is solved based on different R values and a resulted answer obtained from the target function, which is the risk is depicted in a chart along with equivalent Rs of the model. So, the chart is called the efficient border. In the above model, \( \delta_{ij} \) is the covariance of share i,j and \( \omega_i, \omega_j \) is the share’s weight of i,j, \( \mu_i \) is the share gain mean, and R shows a certain level of the efficiency.

With the aim of closing the model to the real market and making the model applicable and finally directing investors of the stock market to a reliable side, the model was developed. Through entering \( \lambda \) in the target function it was attempted to insert risk and efficiency in the target function and minimize risk and maximize efficiency. In fact, \( \lambda \) is a weighting parameter which varies in [0, 1] range and is by which the value of investor to risk or efficiency is applied. But the main limitation and the
The weak point of the model is the inability to optimize the portfolio selection problem under integer limitations. The integer limitation is added to the model based on the following formula:

$$\sum_{i=1}^{n} z_i = k$$

Based on this limitation, if is invested in the share i, \( z_i \) will be equal to one and if it is not invested in the share, \( z_i \) will be zero. In this formula, \( k \) is the number of shares in which the investor wants to have in the stock portfolio and invest in them. So, the developed model of selecting and optimizing stock portfolio was offered as:

$$\max = \lambda \sum_{i=1}^{\infty} \omega_i \mu_i - (1-\lambda) \sum_{i=1}^{\infty} \omega_i \omega_j \delta_{ij}$$

s.t.:

$$\sum_{i=1}^{\infty} \omega_i = 1$$

$$\sum_{i=1}^{n} z_i = k$$

$$\omega_i \geq 0, i = 1,2,3,\ldots,n$$

$$z_i \in \{0,1\}$$

Using variance or its square as a risk criterion is facing some problems. This criterion will be acceptable for a capital which has a normal distribution and is dealt in an efficient market. If the two specifications do not exist for the capital, using variance will face problem, so other criteria are proposed for risk, including semi variance.

Considering this criterion, only random efficiencies which are lower than the mean will be used in calculating risk. In fact, this definition of risk the deviation from expected efficiency will be dangerous if the random efficiency is higher than expected efficiency. We put zero as the difference of the two. So, the semi variance formula will be:

$$\text{semi var} = \frac{1}{n-1} \sum_{i=1}^{n} (\min([r_i - \bar{r}],0))^2$$

Using semi variance instead of variance in the above model we will reach a new model which is the developed model of mean-semi variance model which is offered as follows:

$$\max = \lambda \sum_{i=1}^{\infty} \omega_i \mu_i - (1-\lambda) \sum_{i=1}^{\infty} \omega_i \omega_j \text{semi cov}_{i,j}$$

s.t.:

$$\sum_{i=1}^{\infty} \omega_i = 1$$

$$\sum_{i=1}^{n} z_i = k$$

$$\text{semi cov}_{i,j} = \frac{1}{T} \sum_{i=1}^{T} \min([r_{ij} - \bar{r}],0) \text{*} \sum_{i=1}^{T} \min([r_{ij} - \bar{r}],0)$$

$$\omega_i \geq 0, i = 1,2,3,\ldots,n$$

$$z_i \in \{0,1\}$$

$$t = 1,2,3,\ldots,T$$

5. Methodology

The target society of the study includes all companies admitted to the Tehran Stock Exchange. So, from among all companies which were in the list of the stock market in 2009, 146 companies were selected as samples of the target society. The used variables were the monthly prices of share prices from 2001 to 2009. It has been tried to use appropriate algorithms to help optimize the selected stock portfolio. In this algorithm, the steady mutation operator with the rate of 0.5 has been used. The
number of generations of 2,000 and the population of each generation is 20. This algorithm has been written using the MATLAB software. It has been tried to use the particle swarm algorithm with the best possible strategy. Since the bumping strategy is faster than the amnesia and penalty and is slower than the random replacement strategy, so we have used this strategy in designing the algorithm. The number of generations is 2,000 and the suitable population for this algorithm is between 20 and 30. This algorithm has been written using the MATLAB software.

6. Empirical Results of Research
6.1. Studying Algorithm’s Stability

- Genetic algorithm

The algorithm stability test is one of the most important tests which should be conducted. The issue that the algorithm gives the same answers and the uniqueness of the optimized answer should be tested. To this end, one of the algorithms was selected as sample and was carried out several times. The resulted answers were compared. The selected stock portfolio was considered for testing the 10-stock portfolio. Results of the algorithm repetition are shown in Table 1 and Figure 1.

Table 1: Studying stability of the genetic algorithm in 10 times of implementation

<table>
<thead>
<tr>
<th>Objective function Run1</th>
<th>Objective function Run2</th>
<th>Objective function Run3</th>
<th>Objective function Run4</th>
<th>Objective function Run5</th>
<th>Total mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0154</td>
<td>0.018</td>
<td>0.0184</td>
<td>0.0178</td>
<td>0.0179</td>
<td>0.01694</td>
</tr>
<tr>
<td>0.0184</td>
<td>0.0179</td>
<td>0.0129</td>
<td>0.0159</td>
<td>0.0168</td>
<td>Variance</td>
</tr>
<tr>
<td>0.000003</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Studying stability of genetic algorithm in 10 times of implementation

- Particle swarm algorithm:

To study the stability of the particle swarm algorithm, one of the algorithms was selected as the sample and run several times. The results were compared. The selected stock portfolio was considered for testing the 10-stock portfolio. Results of the algorithm repetition are shown in Table 2 and Figure 2.

Table 2: Studying stability of the particle swarm algorithm in 10 times of implementation

<table>
<thead>
<tr>
<th>Objective function Run1</th>
<th>Objective function Run2</th>
<th>Objective function Run3</th>
<th>Objective function Run4</th>
<th>Objective function Run5</th>
<th>Total mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0193</td>
<td>0.0219</td>
<td>0.0223</td>
<td>0.0217</td>
<td>0.0218</td>
<td>0.02078</td>
</tr>
<tr>
<td>0.0223</td>
<td>0.0218</td>
<td>0.0168</td>
<td>0.0198</td>
<td>0.0201</td>
<td>Variance</td>
</tr>
<tr>
<td>0.000003</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results apparently show a small difference among results of repetitions. The 0.000003 variance is very low which shows high stability of the algorithm in different runs for 1000 repetitions.

6.2. Effect of Portfolio Size on Values of Target Function, Mean, Portfolio Efficiency and Variance

- **Genetic algorithm**
  
  As it can be seen in the Table 3, the genetic algorithm is able to optimize stock portfolios in different sizes, but the target function value does not change so high. With increasing sizes of portfolios, risk and efficiency values are reduced proportionally and with reducing the sizes, the values are increased.

<table>
<thead>
<tr>
<th>Portfolio size</th>
<th>10 Pieces</th>
<th>20 Pieces</th>
<th>30 Pieces</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective function value</td>
<td>0.0196</td>
<td>0.0207</td>
<td>0.0203</td>
</tr>
<tr>
<td>Portfolio variance</td>
<td>0.008006</td>
<td>0.0089</td>
<td>0.0065</td>
</tr>
<tr>
<td>Portfolio efficiency</td>
<td>0.047492</td>
<td>0.0025</td>
<td>0.0016</td>
</tr>
</tbody>
</table>

- **Particle swarm algorithm**
  
  As it can be seen in the Table 4, the particle swarm algorithm is able to optimize stock portfolios in different sizes, but the target function value does not change so high. With increasing sizes of portfolios, risk and efficiency values are reduced proportionally and with reducing the sizes, the values are increased.

<table>
<thead>
<tr>
<th>Portfolio size</th>
<th>10 Pieces</th>
<th>20 Pieces</th>
<th>30 Pieces</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective function value</td>
<td>0.0219</td>
<td>0.0217</td>
<td>0.0207</td>
</tr>
<tr>
<td>Portfolio variance</td>
<td>0.011938</td>
<td>0.008</td>
<td>0.0085</td>
</tr>
<tr>
<td>Portfolio efficiency</td>
<td>0.00542</td>
<td>0.0025</td>
<td>0.0015</td>
</tr>
</tbody>
</table>

6.3. Comparing Genetic and Particle Swarm Algorithms

Results of the study show that the particle swarm algorithm is converged faster in low number of generations compared to the genetic algorithm.
Figure 3: Convergence trend in 10-stock portfolio GA

Figure 4: Convergence trend in 20-stock portfolio GA

Figure 5: Convergence trend in 30-stock portfolio GA

Figure 6: Convergence trend in 10-stock portfolio PSO
**Figure 7:** Convergence trend in 20-stock portfolio PSO

![Convergence trend in 20-stock portfolio PSO](image)

**Figure 8:** Convergence trend in 30-stock portfolio PSO

![Convergence trend in 30-stock portfolio PSO](image)

**Table 5:** Comparing target function, variance, efficiency and time of running particle swarm and genetic algorithms

<table>
<thead>
<tr>
<th>Portfolio size</th>
<th>PSO Objective function value</th>
<th>PSO Portfolio variance</th>
<th>PSO Portfolio efficiency</th>
<th>PSO Run time</th>
<th>GA Objective function value</th>
<th>GA Portfolio variance</th>
<th>GA Portfolio efficiency</th>
<th>GA Run time</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Pieces</td>
<td>0.0219</td>
<td>0.011938</td>
<td>0.00542</td>
<td>4.2</td>
<td>0.0196</td>
<td>0.008006</td>
<td>0.047492</td>
<td>38.4</td>
</tr>
<tr>
<td>20 Pieces</td>
<td>0.0217</td>
<td>0.008</td>
<td>0.0025</td>
<td>19.1</td>
<td>0.0207</td>
<td>0.0089</td>
<td>0.0025</td>
<td>214.3</td>
</tr>
<tr>
<td>30 Pieces</td>
<td>0.0207</td>
<td>0.0085</td>
<td>0.0015</td>
<td>32.1</td>
<td>0.0203</td>
<td>0.0065</td>
<td>0.0016</td>
<td>290</td>
</tr>
</tbody>
</table>

**Figure 9:** Comparing target function, variance, efficiency and time of running particle swarm and genetic algorithms

![Comparing target function, variance, efficiency and time of running particle swarm and genetic algorithms](image)
7. Conclusion
The present study concentrates on stock portfolio selection and optimization problem using inventive algorithms. In line with realizing needs and goals of investors and complications of financial markets, variance as a risk factor in the mean-variance model of Markowitz was replaced with semi variance. By adding some limitations, the mean-semi variance developed model was used as a base for the study. Then, the genetic and the particle swarm algorithms were used for the stock portfolio selection and optimization problem. Finally, a numerical example was given to show empirical results of the algorithms. Results of the study show efficiency of both the genetic and the particle swarm algorithms in the stock portfolio selection and optimization problem. But, the particle swarm algorithm was much faster in speed of convergence and timing compared to the genetic algorithm.

References


