Applied Mathematical Sciences, Vol. 8, 2014, no. 65, 3215 - 3227
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ams.2014.43203

A Heuristic Crossover for Portfolio Selection

Joseph Ackora-Prah

Department of Mathematics
Kwame Nkrumah University of Science and Technology, Kumasi, Ghana

Samuel Asante Gyamerah

Department of Mathematics
Kwame Nkrumah University of Science and Technology, Kumasi, Ghana

Perpetual Saah Andam

Department of Mathematics
Kwame Nkrumah University of Science and Technology, Kumasi, Ghana

Abstract

We propose the suitability of Heuristic Crosover in Genetic Algorithm (GA) for the selection of an optimal portfolio of stocks from the Ghana Stock Exchange. The appropriate choice of an optimal portfolio is the principal problem of both the portfolio manager and the investor. In this paper, we formulate a model which includes practical constraints (floor-ceil and cardinality constraints) which the Markowitz unconstrained Mean-Variance method does not consider in the selection of optimal portfolio. We use heuristic crossover to optimize the risk-return trade-off and achieve an optimal solution for the portfolio selection and the allocation of weights to each portfolio.

Keywords: Portfolio Selection, Genetic Algorithm, Heuristic Crossover

1 Introduction

In Portfolio Selection, we choose a weighted group of assets from a large number of available securities so as to maximize the expected return to a given risk rate. It is the wish of every investor to minimize the total risk whilst maximizing the return. The total risk of an investment in a portfolio can be reduced if different stocks are added to the portfolio [5]. This means that the change in price of some stocks can compensate for the changes of other stocks. The investor decides on the stocks and the proportions in the portfolio in order to maximize the expected return. To solve this problem, *Markowitz (1952)* proposed the *Modern Portfolio Theory, (MPT)*, which states that by choosing a combination of assets to invest in, an investor could get higher returns with the same risk rate [6].

The selection of optimal portfolio using GA have sprung out in recent years. John Henry Holland (1975) developed the general GA as a stochastic optimization algorithm based on the mechanisms of natural selection and genetics [4]. In GA, an initial population of chromosomes are generated randomly. In each generation, the fitness of every individual in the population is evaluated and multiple individuals are stochastically selected from the current population based on their fitness. By using crossover and mutation, a new population is formed. The new population is then used in the next iteration until the stopping criteria is met. Usually, the algorithm terminates when either a maximum number of generations is achieved, or an acceptable fitness level has been reached for the population.

Applications of GA to portfolio selection problems have developed recently. *Pereira (2000)*, explained why GA's are more efficient and robust compared to other search heauristics [8]. *Chang et al. (2000)* considered the problem associated with the search of an efficient frontier related to the standard mean-variance portfolio optimization model by adding constraints that limit a portfolio to a specified number of assets [2]. *Zhang et al. (2006)* applied GA to solve the portfolio selection problem in which there exist both probability constraint on the lowest return rate and the lower-upper bounds constraints [16]. In a study done on 50 Supreme Tehran Stock Exchange Companies, *Pandari (2012)* used GA in selecting the best portfolio in order to optimize the rate of return, return skewness, liquidity and Sharpe ratio [7]. *Sefiane and Benbouziane (2012)* tried GA on a five(5) stock asset portfolio successfully [12].

Markowitz mean-variance model has been criticised not only for its main assumptions, but also does not take into consideration some important features like trading constraints of portfolio performance in real life. In this paper, we

look at two common trading constraints which are not handled by Markowitz, namely floor-ceil and bounding constraints.
Several researchers have attempted to solve this constrained mean-variance model by a variety of techniques. However, these techniques are computationally more complex than the standard mean-variance model. In this paper, we show that portfolio selection problems containing floor-ceil and boundary constraints can be successfully solved by GA. Again, we indicate that the type of crossover used for the GA has a great impact on the optimal portfolio.

2 Preliminaries

We consider a portfolio with a wide range of investments and verify whether the portfolio risk can be reduced if more assets are added to the portfolio based on the following theorems with proofs:

Theorem 2.1 *Specific risk can be reduced by diversification if the returns on assets in a portfolio are uncorrelated.*

Proof
We consider portfolio with n assets and assume $W_i = \frac{1}{n}$ and $\rho_{ij} = 0, \forall i, j$.

$$\sigma_p^2 = \sum_{i=1}^{n} \sigma_i^2 \left(\frac{1}{n}\right)^2 = \frac{\sum_i^n \sigma_i^2}{n^2}$$

$$\bar{\sigma} = \frac{\sigma_i}{n}$$

We assume the risk of all assets equal to the highest risk, σ_{max}. As $n \to \infty$, $\sigma_p^2 \to 0$ and $\bar{\sigma_i^2} = constant$,

$$\frac{\sum \sigma_{max}^2}{n^2} \leq \frac{\sum \sigma_{max}}{n}$$

$$\frac{n\sigma_{max}^2}{n^2} \leq \frac{n\sigma_{max}}{n}$$

$$\frac{\sigma_{max}^2}{n} \leq \sigma_{max}$$

Theorem 2.2 *Market risk cannot be reduced by diversification.*

Proof
We consider a portfolio with n assets and assume that $W_i = \frac{1}{n}, \sigma_{ii} = s$ and $\sigma_{ij} = a$ for every i and j.

$$\sigma_p = \frac{\sum_i^n s^2}{n^2} + \frac{\sum_{i \neq J} a}{n^2}$$

$$\sigma_p = \frac{ns^2}{n^2} + n(n-1)\frac{a}{n^2}$$

$$\sigma_p = \frac{s^2}{n} + a + \frac{a}{n}$$

$$\sigma_p = a + \frac{s^2 - a}{n}$$

No matter the increament of n (the number of assets), the risk will never be less than the average covariance,a.

2.1 Crossover

We consider crossover as a genetic operator which combines 2 parent chromosomes from the current population to produce offsprings. In crossover, we swap the genetic information of selected chromosomes to obtain better and more adapted individuals. The crossover operation is of various types namely: One-Point Crossover, Two-Point Crossover, Uniform Crossover, Arithmetic Crossover, and Heuristic Crossover. We note that the type of crossover used in GA has an impact on the process and hence the solution.

For constrained optimization problems, arithmetic crossover is appropriate. Here, the offsprings are created using the following equations:
$Offspring\ A = \alpha * Parent\ A + (1 - \alpha) * Parent\ B$
$Offspring\ B = (1 - \alpha) * Parent\ A + \alpha * Parent\ B$. Where α is a random weighting factor chosen before each crossover operator.

Heuristic crossover uses the fitness values of two parent chromosomes to ascertain the direction of the search. It moves from worst parent to slightly best parent. The offspring are created according to the equation:
$Offspring\ A = BestParent + \beta * (BestParent - WorstParent)$
$Offspring\ B = BestParent$. Where β is a random number between 0 and 1. This crossover type is good for real-valued genomes.

In single or double point crossover, genomes that are near each other survive together and those that are far apart are separated. Uniform Crossover eliminates this effect. Each gene in this crossover type has an equal chance of coming from either parent.

3 Portfolio Selection Model

We consider a portfolio with a weighted composition of n weights $(W_1, W_2, ..., W_n)$, where W_i is the proportion of the investors' total invested wealth on asset i, r_i a random variable which is the expected return of asset i and n as the total number of asset. The return of the portfolio r_p is given by:

$$r_p = W_1 r_1 \, + \, W_2 r_2 \, + \, W_3 r_3 \, + \, ... + W_n r_n \, = \, \sum_{i}^{n} W_i r_i$$

$$E(r_p) \, = \, E\left[\sum_{i=1}^{n} W_i r_i\right] \, = \, \sum_{i=1}^{n} W_i E(r_i)$$

We let $E(r_p) \, = \, R$ and $E(r_i) \, = \, \mu_i$

$$\Rightarrow \, R \, = \, \sum_{i=1}^{n} W_i \mu_i$$

$$\sigma_p^2 \, = \, E\left[(r_p - \mu_p)^2\right]$$

$$\sigma_p^2 \, = \, \sum_{i=1}^{n} W_i^2 \sigma_i^2 \, + \, 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} W_i W_j \sigma_{ij}$$

But $var(r_i) = \sigma ii$,

$$Var(P) = \sum_{i=1}^{n} \sum_{j=1}^{n} W_i W_j \sigma_{ij}$$

We introduce our model which includes practical constraints and a risk aversion parameter λ $(0 \, \leq \, \lambda \, \leq \, 1)$ that describes the sensitivity of the investor as indicated below:

$$\text{Minimize} \quad \lambda \left[\sum_{i=1}^{n} \sum_{j=1}^{n} W_i W_j \sigma ij\right] - (1 - \lambda) \left[\sum_{i=1}^{n} W_i \mu_i\right] \tag{1}$$

Subject to

$$\sum W_i = 1$$

$$\sum z_i = k$$

$$\varepsilon_i z_i \leq W_i \leq \sigma_i z_i, \quad i = 1, 2, 3, \cdot \cdot \cdot, n.$$

where the constraints above are the budget constraint, cardinality and boundary (floor and ceiling) constraints respectively. Budget constraints ensures that the investor allocate exactly 100% to a portfolio. Cardinality constraint ensures that exactly K assets are held in the portfolio. Floor constraint (lower-bound constraint) is introduced to avoid administration cost for very small holdings while ceiling constraint (upper-bound constraint) avoids excessive exposure to a specific asset.

3.1 The Data

We randomly selected 5 companies from Ghana stock exchange (GSE) market. The historical prices from these companies for a period of 6 years, $2007-2012$ were taken and their log-normal return calculated. We evaluated the portfolio variance and the portfolio average return using the annual average return data below:

Table 1: Portfolio annual average returns from GSE

Year	STOCK 1	STOCK 2	STOCK 3	STOCK 4	STOCK 5
2012	-0.002447224	0.000543547	0.000324513	0.00071918	0.000225605
2011	1.36E-05	-6.17E-05	-0.00071292	-0.002628853	-0.000658221
2010	0.000764859	0.000689607	-4.27E-12	0.00119565	0.002395387
2009	-0.000431354	-0.002004711	-4.27E-12	0.000382691	-0.000723365
2008	0.000661889	0.000533048	-0.001825326	0.001103673	0.000174978
2007	0.001245823	0.0016855	0.000340635	0.000678282	0.001297666

The covariance matrix and mean return for each asset portfolio are given in the tables below:

Table 2: The Covariance matrix

	STOCK 1	STOCK 2	STOCK 3	STOCK 4	STOCK 5
STOCK 1	1.75E-06	5.12E-07	-1.69E-07	1.15E-07	6.81E-07
STOCK 2	5.12E-07	1.52E-06	9.47E-07	4.02E-07	9.85E-07
STOCK 3	-1.69E-07	9.47E-07	1.11E-06	9.88E-08	7.73E-07
STOCK 4	1.15E-07	4.02E-07	9.88E-08	2.07E-06	9.80E-07
STOCK 5	6.81E-07	9.85E-07	7.73E-07	9.80E-07	1.45E-06

Table 3: The mean returns, r_i for each asset

	STOCK 1	STOCK 2	STOCK 3	STOCK 4	STOCK 5
mean return	-3.12E-05	0.00023167	-0.00064554	0.00024283	0.00045283

4 Crososover Portfolio Function

Vector of real valued variables for each weight in the portfolio is the most common way and easiest to represent a portfolio in a GA since the transformation from the genome to the problem solution is simple. The output for the system is a portfolio which comprises of 5 weights using the respective functions below:

4.1 Arithmetic Crossover

```
function[f] = portx(Wts)
global expRet expCov
A=0.5;
format long
Wts
Ret=expRet*(Wts')
Var=Wts*expCov*(Wts')
f =(A*(Wts*expCov *(Wts')))-((1-A)*(expRet*(Wts')))

% Constraint definitions used by |ga|
nvars=5;   %number of variables
Aeq = ones(1,5); beq = 1;     % weights sum to 1
lb  = zeros(1,5);             % weights are positive
ub  = ones(1,5);              % weights are below one
PopulationSize=50; EliteCount=3; CrossoverFraction=0.9;

function [x,fval,exitflag,output,population,score] = genetics(nvars...
Aeq,beq,lb,ub,PopulationSize_Data,EliteCount_Data,CrossoverFraction_Data)
global nvars Aeq beq lb ub PopulationSize EliteCount CrossoverFraction
% Start with the default options
options = gaoptimset;
% Modify options setting
options = gaoptimset(options,'PopulationSize', PopulationSize_Data);
options = gaoptimset(options,'EliteCount', EliteCount_Data);
options = gaoptimset(options,'CrossoverFraction', CrossoverFraction_Data);
options = gaoptimset(options,'SelectionFcn', @selectionroulette);
options = gaoptimset(options,'CrossoverFcn', {  @crossoverarithmetic [] });
options = gaoptimset(options,'MutationFcn', @mutationadaptfeasible);
options = gaoptimset(options,'Display', 'off');
options = gaoptimset(options,'PlotFcns', {  @gaplotbestf @gaplotbestindiv ...
@gaplotdistance });
[x,fval,exitflag,output,population,score] = ...
ga(@portx,nvars,[],[],Aeq,beq,lb,ub,[],[],options);
```

4.2 Heuristic Crossover

```
function [x,fval,exitflag,output,population,score] = genetics(nvars...
Aeq,beq,lb,ub,PopulationSize_Data,EliteCount_Data,CrossoverFraction_Data)
global nvars Aeq beq lb ub PopulationSize EliteCount CrossoverFraction
% Start with the default options
options = gaoptimset;
```

```
% Modify options setting
options = gaoptimset(options,'PopulationSize', PopulationSize_Data);
options = gaoptimset(options,'EliteCount', EliteCount_Data);
options = gaoptimset(options,'CrossoverFraction', CrossoverFraction_Data);
options = gaoptimset(options,'SelectionFcn', @selectionroulette);
options = gaoptimset(options,'CrossoverFcn', {  @crossoverheuristic [] });
options = gaoptimset(options,'MutationFcn', @mutationadaptfeasible);
options = gaoptimset(options,'Display', 'off');
options = gaoptimset(options,'PlotFcns', {  @gaplotbestf @gaplotbestindiv ...
@gaplotdistance });
[x,fval,exitflag,output,population,score] = ...
ga(@portx,nvars,[],[],Aeq,beq,lb,ub,[],[],options);
```

4.3 Uniform Crossover

```
function [x,fval,exitflag,output,population,score] = genetics(nvars...
Aeq,beq,lb,ub,PopulationSize_Data,EliteCount_Data,CrossoverFraction_Data)
global nvars Aeq beq lb ub PopulationSize EliteCount CrossoverFraction
% Start with the default options
options = gaoptimset;
% Modify options setting
options = gaoptimset(options,'PopulationSize', PopulationSize_Data);
options = gaoptimset(options,'EliteCount', EliteCount_Data);
options = gaoptimset(options,'CrossoverFraction', CrossoverFraction_Data);
options = gaoptimset(options,'SelectionFcn', @selectionroulette);
options = gaoptimset(options,'CrossoverFcn', {  @crossoverscattered [] });
options = gaoptimset(options,'MutationFcn', @mutationadaptfeasible);
options = gaoptimset(options,'Display', 'off');
options = gaoptimset(options,'PlotFcns', {  @gaplotbestf @gaplotbestindiv ...
@gaplotdistance });
[x,fval,exitflag,output,population,score] = ...
ga(@portx,nvars,[],[],Aeq,beq,lb,ub,[],[],options);
```

4.4 Results

All simulations in this work were executed using MATLAB. Illustrated below are the results obtained via the 3 crossover techniques:

4.4.1 Arithmetic Crossover

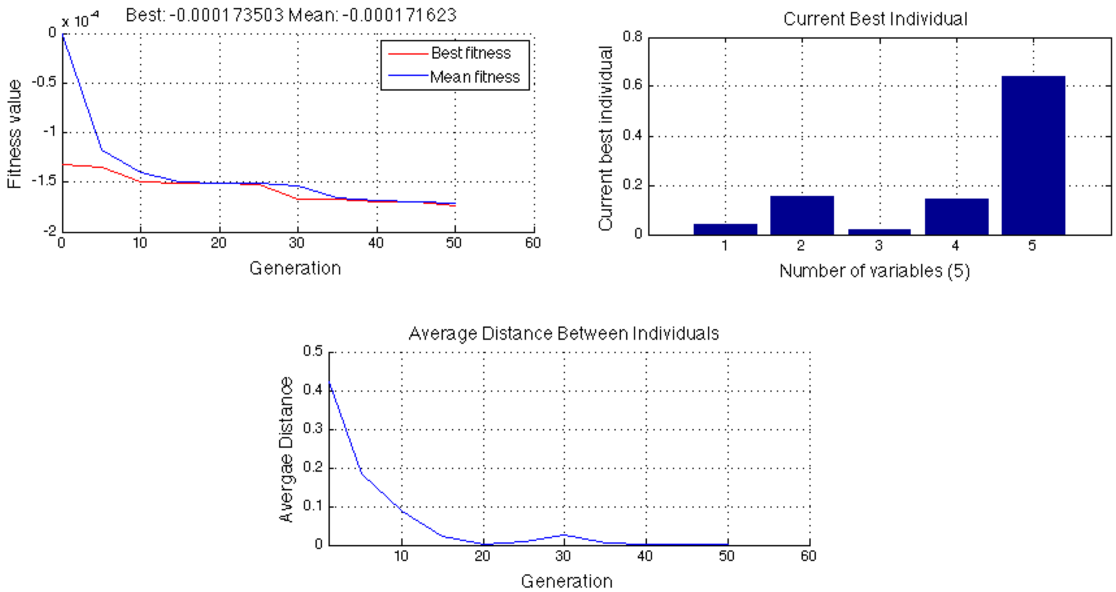

Figure 1: The variations of the GA functions according to generation under Arithmetic crossover technique

Objective function value = -1.735049154920278e-04
Average Return of Portfolio = 3.481543945314851e-04
Variance of Portfolio = 1.144563547429597e-06
Portfolio Weights :
Weight 1= 0.038056758381922
Weight 2= 0.155418010272420
Weight 3= 0.019858002318151
Weight 4= 0.144597019644021
Weight 5= 0.642712653783081

4.4.2 Heuristic Crossover

Objective function value = -2.100619086026940e-04
Average Return of Portfolio = 4.214555303246128e-04
Variance of Portfolio = 1.331713119224894e-06
Portfolio Weights :
Weight 1= 0.000150118405472
Weight 2= 0.029845672108102
Weight 3= 0.000000003905862

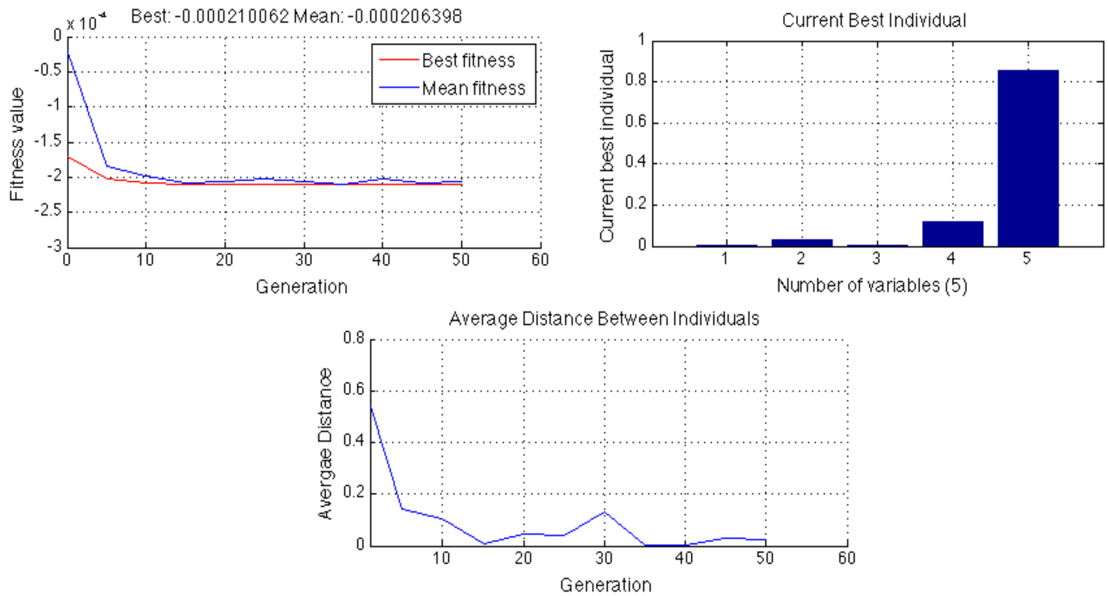

Figure 2: The variations of the GA functions according to generation under Heuristic crossover technique

Weight 4= 0.117642837123397
Weight 5= 0.852361368457167

4.4.3 Uniform Crossover

Objective function value = -2.057425150293183e-04
Average Return of Portfolio = 4.128158217522129e-04
Variance of Portfolio = 1.332791693576277e-06
Portfolio Weights :
Weight 1= 0.010833630409913
Weight 2= 0.055925224208046
Weight 3= 0.012315752652018
Weight 4= 0.043848245554371
Weight 5= 0.877805105388565

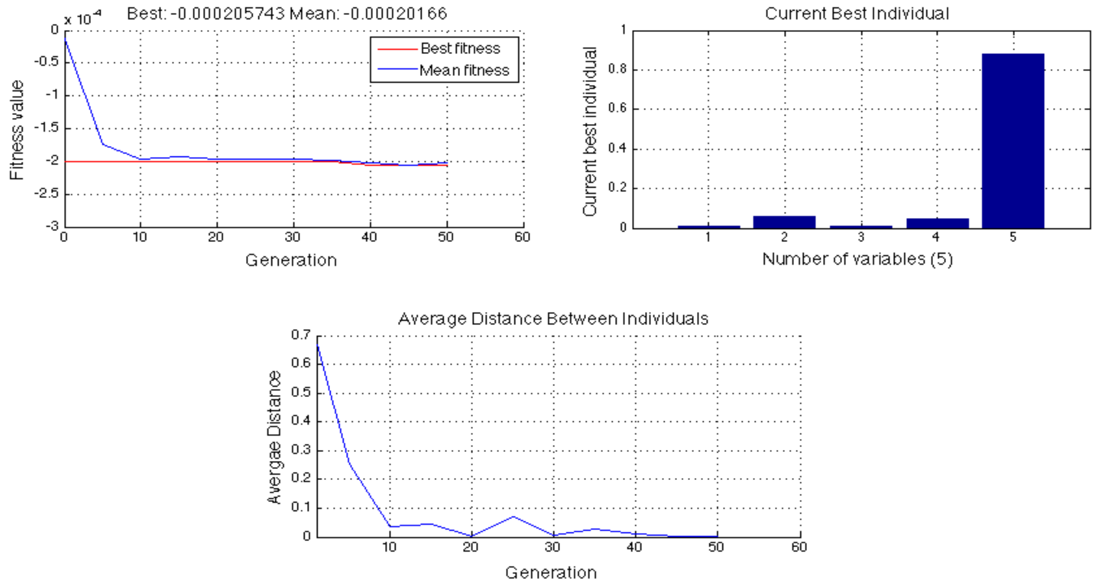

Figure 3: The variations of the GA functions according to generation under uniform crossover technique

4.5 Discussion

In the implementation of Genetic Algorithm, we formulated an objective fitness function to evaluate which among the three crossover techniques (Arithmetic, Heuristic and Uniform) score less on the fitness scale.
We indicate that, the Heuristic crossover led to the best choice of weights, return and risk. The Heuristic crossover performed less on the fitness scale with $-2.100619086026940e-04$, whereas the Arithmetic and Uniform had $-1.735049154920278e-04$ and $-2.057425150293183e-04$ respectively.
The Heuristic crossover had the highest return of $4.214555303246128e-04$ and weights ($W_1 = 0.000150118405472, W_2 = 0.029845672108102,$
$W_3 = 0.000000003905862, W_4 = 0.117642837123397, W_5 = 0.852361368457167$).
An investor wishing to get this maximum return and its associated lower risk should invest 0.02% of his total wealth in Stock 1, about 2.98% of his total wealth in Stock 2, 11.76% of his total wealth in Stock 4, and 85.24% of his total wealth in Stock 5 and can choose not to invest in stock 3.
The Arithmetic crossover had the lowest return of $3.4815439453148510e-04$ and the lowest risk of $1.144563547429597e-06$.

5 Conclusion

We applied Arithmetic, Heuristic, and Uniform crossover on portfolio of stocks from Ghana Stock Exchange which includes practical constraints (floor-ceil and boundary constraints). The results we obtained showed that the Heuristic crossover gives better results than the two other crossovers with a maximum return $4.214555e-04$ and a minimum risk of $1.331713e-06$. Our results show that Heuristic crossover is very useful when an investor wants to allocate his total wealth in an investment to yield a maximum return and lesser risk.

Acknowledgements. We are grateful to the Almighty God and the Department of Mathematics, Kwame Nkrumah University of Science and Technology for providing us resources to help complete this research successfully.

References

[1] A. Fernandez and S. Gomez, Portfolio selection using neural networks, *Computers & Operations Research,* **34(4)** (2007), 1177 - 1191.

[2] T.J. Chang, N. Meade, J. Beasley, and Y. Sharaiha, Heuristics for cardinality constrained portfolio optimisation, *Computers and Operations Research,* **(27)** (2000), 1271 - 1302.

[3] D.O. Hebb, *The Organization of Behavior,* Wiley, New York, 1949.

[4] J. H. Holland, *Adaptation in natural and artificial systems, University of Michigan Press,* 1975.

[5] H. Markowitz, Analysis in portfolio choice and capital markets Oxford, Basil Blackwell, 1987.

[6] H. Markowitz, Portfolio selection *Journal of Finance,* **7(1)** (March 1952), 77 - 91.

[7] A.R. Pandari, A. Azar, and A.R. Shavazi, Genetic algorithms for portfolio selection problems with non-linear objectives, *African Journal of Business Management,* **6(20)** (2012), 6209-6216.

[8] Robert Pereira, Genetic Algorithm Optimization for Finance and Investments, *MPRA Paper,* **8610***,* University Library of Munich, Germany, (February, 2000).

[9] Felix Roudier, *Portfolio optimization and genetic algorithms,* Master's thesis, Department of Management, Technology and Economics, Swiss Federal Institute of Technology (ETM), Zurich, 2006.

[10] A. Popov, Genetic Algorithms for optimization-programs for Matlab, 2005.

[11] B. Samanta and T. K. Roy, Multi-objective portfolio optimization model, *Tamsui Oxford Journal of Mathematical Sciences,* **21(1)** (2005), 55 - 70.

[12] S. Sefiane. and M. Benbouziane, Portfolio selection using genetic algorithm, *Journal of Applied Finance and Banking,* **2(9)** (2012), 143 - 154.

[13] P. Sinha, A. Chandwani, and T. Sinha, Algorithm of construction of optimum portfolio of stocks using genetic algorithm *MPRA Paper,* **48204**, (July, 2013).

[14] S.M. Wang, J.C. Chen, H.M. Wee, and K.J. Wang, Non-linear stochastic optimization using genetic algorithm for portfolio selection, *International Journal of Operations Research,* **3(1)** (2006), 16 - 22.

[15] X. Yao, Y. Liu, and G. Lin, Evolutionary programming made faster, *IEEE Transactions on Evolutionary Computation,* **3(2)** (1999), 82-102.

[16] W.G. Zhang, W. Chen, and Y.L. Wang, The adaptive genetic algorithms for portfolio selection problem., *International Journal of Computer Science and Network Security,* **6(1)** (2006), 196 - 200.

Received: March 20, 2014