Robust and Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithms

Sudhansu Kumar Mishra

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Rourkela-769008 (INDIA)

Ph.D

Thesis

Robust and Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithms

Sudhansu Kumar Mishra

2012

Robust and Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithms

Thesis Submitted in Partial Fulfillment of the Requirements
forthe Degree of

Doctor of Philosophy

by

Sudhansu Kumar Mishra

Under the Supervision of
Prof. G. Panda (FNAE, FNASc.)

and

Prof. S. Meher

Department of Electronics and Communication Engineering

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

Rourkela-769008 (INDIA)

2012

...dedicated to my loving parents

CERTIFICATE

This is to certify that the thesis entitled "**Robust and Constrained Portfolio Optimizations using Swarm Intelligence Techniques**" submitted to the National Institute of Technology, Rourkela (INDIA) by **Mr.Sudhansu Kumar Mishra**, Roll No. 508EC101 for the award of the degree of Doctor of Philosophy, in the department of Electronics and Communication Engineering, is a record of bonafide research work carried out by him under our supervision. We believe that the thesis fulfills part of the requirements for the award of degree of Doctor of Philosophy. The results embodied in the thesis have not been submitted for award of any other degree.

To the best of our knowledge, Mr. Mishra bears a good moral character and decent behavior.

Acknowledgement

Abstract

Optimization plays an important role in many areas of science, management, economics and engineering. Many techniques in mathematics and operation research are available to solve such problems. However these techniques have many shortcomings to provide fast and accurate solution particularly when the optimization problem involves many variables and constraints. Investment portfolio optimization is one such important but complex problem in computational finance which needs effective and efficient solutions. In this problem each available asset is judiciously selected in such a way that the total profit is maximized while simultaneously minimizing the total risk. The literature survey reveals that due to non availability of suitable multiobjective optimization tools, this problem is mostly being solved by viewing it as a single objective optimization problem.

Multiobjective solution techniques have been introduced in literature to solve portfolio optimization problem. In recent past many evolutionary/ swarm computing techniques have been proposed and have successfully been applied to many engineering, science and finance problems. Further, multiobjective versions of these algorithms have also been reported in the literature to efficiently solve the multiobjective problems.

When number of constraints are present, the portfolio optimization problem becomes complex and needs effective solution. Further, the existing multiobjective computing methods also require suitable modification to suit to portfolio optimization problem. The existing methods cannot be applied to plan future portfolio optimization strategy, as required future data is not available. New multiobjective algorithms are also needed to efficiently solve the portfolio optimization problems. The portfolio optimization problem becomes more challenging when some data become uncertain and contaminated with outliers. These issues have been addressed in this thesis and satisfactory solution of each of these problems has been provided. In all cases multiobjective evolutionary algorithms (MOEAs) have been successfully applied.

This thesis has proposed and suitably applied four MOEAs for solving the multiobjective optimization problem associated with constraints. The performance of these algorithms has been evaluated and compared using three error measures, six performance metrics, Pareto front, computational time and nonparametric statistical testing. For

comparison, the results have also been obtained by formulating the problem as a single objective problem. The results demonstrate that the proposed algorithms are capable of identifying good Pareto solutions maintaining adequate diversity for different market indices.

The Markowitz mean-variance portfolio optimization and many other models use the mean of the past return as expected return. They also assume that the time series of returns of each stock follows a normal distribution. However these time series often depart from normality and exhibits kurtosis and skewness and thus make the variance of returns an inappropriate measure of risk. Hence there is a need to develop an efficient approach which will free from this assumption and is capable to predict the future expected return. In the thesis a new mean-variance model has been proposed in which, the expected return and risk are predicted using a low complexity functional link artificial neural network (FLANN) structure. Four multiobjective swarm intelligence technique has been applied to solve the portfolio optimization problem considering various constraints and their performance has been compared. The results demonstrate that the proposed model provides improved performances in terms of diversity and coverage of Pareto solutions.

Actually the stock values are highly uncertain due to political crises or turmoil in global markets. As a result the stock parameters deviate heavily from its actual value. Under such condition, the estimation of the expected return and risk becomes poor and hence leads to inferior optimization performance. To alleviate this shortcoming, a minimum volume ellipsoid (MVE) methodology using core set and Lagrange multipliers is proposed to handle outliers present in the stock market data. Simulation results show that the proposed method exhibits good portfolio strategy in the presence of market uncertainties.

In many situations portfolio optimization is needed for future data. Further, the present data may be contaminated with outliers. This difficult problem has been addressed and satisfactory solution has been provided using robust prediction, mean variance model, MVE and MOEA based method.

Keywords: Portfolio Optimization, Multiobjective Optimization, Efficient Frontier, Non-dominated Sorting, Cardinality Constraint, Outliers, Minimum Volume Ellipsoid.

Contents

Particulars   Page no.

Chapters

1. Introduction

2. Multiobjective Evolutionary Algorithms and Performance Metrics for Portfolio   Optimization

3. Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithms

4. Prediction based mean-variance Model for Multiobjective Portfolio Optimization

5. Novel Robust Multiobjective Portfolio Optimization Schemes

6. Prediction Based Robust mean-variance Model for Constraint Portfolio Optimization

7. Conclusion

List of Figures

List of Tables

List of Acronyms

SA: simulated annealing

TS: Tabu search

GA: genetic algorithm

MOEAs: Multiobjective evolutionary algorithms

POF: Pareto-optimal front

GOEF: Global optimal efficient front.

MOP: Multiobjective optimization problem (MOP)

PSO: Particle swarm optimization

BFO: Bacteria foraging optimization

MSE: Mean squares error

PAS: Portfolio asset selection

PO: Portfolio Optimization

NS-MOPSO: Non-dominated sorting multiobjective particle swarm optimization

MOBFO: Multiobjective bacteria foraging optimization

MOEA/D: Decomposition based multiobjective evolutionary algorithm

MODM: Multiobjective decision making.

FLANN: Functional link artificial neural network

PBMV: Prediction based mean variance

MVE: Minimum volume ellipsoid

Chapter 1

Introduction

Chapter

1

1.1 Introduction

In recent past, applications of different swarm and evolutionary computation techniques in diversified domains have gained popularity in wide area ranging from engineering and computer science to the field of finance, ecology, sociology and medicine. Chen and Kuo [1.1] have reported several popular articles in the area of evolutionary computing application to economics and finance.

The taxonomy of applications of swarm and evolutionary computation in economics and finance has been provided by Chen [1.2], which includes (1) investment portfolio optimization (2) financial time series (3) stock ranking (4) risk-return analysis and (5) economic modeling. In fact, all these applications are inherently multiobjective in nature. The use of swarm and evolutionary algorithms for solving multi-objective optimization problem has emerged as a potential field of research in recent years.

In this thesis, different multiobjective evolutionary algorithms (MOEAs) have been studied and successfully employed to solve problems related to portfolio optimization with special emphasis on portfolio constraints. The optimization problem varies from simple portfolios held by individuals to huge portfolios managed by professional investors. The portfolio consists of stocks, bank investments, real estate holdings, bonds, treasury bills etc.

The objective is to find an optimal set of assets to invest on, as well as the optimal amount of investment for each asset. This optimal selection of assets and weighting of each asset is a multi-objective problem where the total profit of investment has to be maximized and total risk has to be minimized. There are also different constraints under which the optimization task is to be carried out depending on the type of problem to be solved. For example, the weights normally have lower and upper bounds as well as many other practical constraints. This is the so-called optimal investment portfolio that one wishes to obtain by using optimization techniques. The recently developed swarm and evolutionary computation algorithms have been effectively used for solving many multiobjective problems in a single run giving a set of desired solutions. Hence suitable choice and applications of multiobjective evolutionary algorithms (MOEAs) have potential future to handle different challenges in constraint portfolio optimization problem which is inherently a multiobjective problem.

1.2 Background and scope of the thesis

The problem of portfolio optimization has always been a challenging task for researchers, investors and fund managers. Markowitz has devised a quantitative framework for the selection of a portfolio [1.3],[1.4].   In this framework, the percentage of each available asset is selected in such a way that the total profit of the portfolio is maximized while total risk is minimized simultaneously. The sets of portfolios of assets that yield minimum risk for a given level of return form the efficient frontier. The optimal solution for the standard form of the Markowitz portfolio asset selection problem, which is classified as a quadratic programming model, can be solved through exact methods such as active set methods, interior point techniques etc.

However, portfolio optimization is very complicated as it depends on many factors such as preferences of the decision makers, resource allocation and growth in sales, liquidity, total turnover, dividend and several other factors. Some authors have also added some practical constraints such as floor, ceiling, cardinality etc. to Markowitz model that makes it more realistic. Inclusion of these constraints to the portfolio optimization problem makes it intractable even for small instances. With these constraints it is a mixed integer programming problem with quadratic objective functions. The traditional optimization methods used to solve this problem are trapped in local minima solutions. To overcome this problem different efficient heuristic methods have been developed.

An overview of the literature on the application of evolutionary computation to the portfolio selection problem has been discussed in [1.5]. These methods consist of simulated annealing (SA) [1.6], Tabu search (TS) and genetic algorithm (GA) [1.7]. The PSO (particle swarm optimization) technique has been applied in [1.8] to solve cardinality constrained portfolio and the results have been compared with those obtained by using GA, TS and SA. Improved PSO (particle swarm optimization) algorithms have also been proposed in [1.9] for portfolio problem with transaction costs. The PSO algorithm has been applied to solve constrained portfolio selection problem with bounds on holdings (minimum buy in threshold and maximum limit in combination), cardinality, minimum transaction lots and sector capitalization constraint [1.10]. Hanhong *et al.* [1.11] has applied the PSO technique to solve different restricted and unrestricted risky investment portfolios and compared it with GA.

Portfolio optimization problem is intrinsically a multiobjective problem having conflicting objectives i.e. risk and return. But in the aforementioned studies, the problem has been viewed as a single objective optimization problem by considering the overall objective

as a weighted sum of objectives. Such a formulation yields multiple solutions by suitably varying the associated weights. The main advantage of these approaches is that it reduces the multiobjective problem to a scalar optimization problem and any single objective metaheuristics algorithm can then be applied. However, solving this multiobjective problem with these SOEAs (single objective evolutionary algorithms) methods require the repeated use of an optimization technique to find one single solution on the efficient frontier per run. Hence it is a time consuming process to get the entire Pareto front. Furthermore, a uniform set of weight does not guarantee a uniformly distributed set of efficient points [1.12]. To achieve a diversity of solutions along the efficient frontier is of immense importance since certain trade-off portfolios of interest may be missed if they are concentrated in a small area of the efficient frontier. One more shortfall of this approach is that it cannot find all efficient points as shown in [1.7]. In addition, if practical constraints are considered the problem becomes extremely difficult to solve by using such method.

To overcome these shortcomings many researchers have applied multiobjective evolutionary algorithms (MOEAs) to solve the problem. One of the main advantages of MOEAs is that it gives a set of possible solutions in a single run called as Pareto optimal solution in a reasonable amount of time [1.12, 1.13]. Pareto ant colony optimization (PACO) has been introduced for solving the portfolio selection problem [1.13] and the performance has been compared with other heuristic approaches (i.e., Pareto simulated annealing and the nondominated sorting genetic algorithm) by means of computational experiments with random instances. Some authors have also used few MOEAs to solve the portfolio optimization problem with many practical constraints [1.14, 1.15].

5

Since the introduction of the mean-variance portfolio optimization model by Harry Markowitz, considerable research attention has been paid on model simplifications and the development of different risk measures such as semi-variance, mean absolute deviation and variance with skewness model. All these techniques use the mean of the past return as expected return. These models are built upon some fundamental assumptions which are based on a distortion-free normally distributed series of returns [1.16]. However, these assumptions fails as the distribution of series of return deviates from normalcy due to kurtosis and skewness [1.17],[1.18]. Hence the development of a model free from such assumptions is still a challenging field of research.

Markowitz theory helps to diversify the asset allocation. But there are some evidences which indicate that diversification does not help in reducing the total risk when the global markets face with some crises such as the incident of September, 11 or the recent turmoil in global markets which started from the financial sector. The value of stock in these conditions may be considered as outliers. During the last two decades, the idea of quality estimation, making the optimization robust under such conditions has become an interesting area of research. Hence robust optimization aims to find solutions to a given optimization problems with uncertain data. Different researchers have applied different robust optimization techniques to solve portfolio selection problem in this uncertain condition [1.19, 1.20]. However in these optimization techniques, the program dimension increases exponentially as the size of the problem i.e. number of assets present in the portfolio optimization increases. The difficulties become more pronounced when the numbers of constraints become more. In addition, if heavy turmoil on the input data occurs i.e. input data is contaminated with outliers, the optimization problem become more complex to get

the final solution. Therefore, there is a need to develop robust portfolio optimization techniques which can efficiently handle the outliers present in the financial data.

In many situations it is required to invest the fund in future where the future data are not available and the present data are uncertain due to the presence of outliers. In such scenario future stock has to be predicted and the expected return and variance is to be calculated accordingly. Such complex problem needs a solution by involving robust prediction followed by efficient optimization.

The above cited burning issues need attention and appropriate solutions. Hence the scope of the present thesis is to address these issues and suggest appropriate methodologies based on multiobjective optimization techniques to provide satisfactory solutions.

1.3 Multiobjective optimization: basic concepts and brief overview

Multiobjective optimization deals with simultaneous optimization of multiple objective functions which are conflicting in nature. A multiobjective optimization problem (MOP) is defined as the problem of computing a vector of decision variables that satisfies the constraints and optimize a vector function whose elements represent the objective functions. The generalized multiobjective minimization problem [1.21, 1.22] is formulated as

$$Minimize \ \ f(\vec{x}) = \left(f_1(\vec{x}), f_2(\vec{x}),..., f_M(\vec{x}) \right) \tag{1.1}$$

subject to constraints:

$$g_j(\vec{x}) \geq 0, \ j = 1, 2, 3....., J \tag{1.2}$$

$$h_k(x) = 0, \ k = 1, 2, 3 \ldots \ldots, K \tag{1.3}$$

where $\vec{x}$ represents a vector of decision variables

$$\vec{x} = \{x_1, x_2 \ldots, x_N\}^T \tag{1.4}$$

The search space is limited by

$$x_i^L \leq x_i \leq x_i^U, \ i = 1, 2, 3 \ldots \ldots, N \tag{1.5}$$

The notations x_i^L and x_i^U represent the lower and upper acceptable values respectively for the variable x_i. N and M represent the number of decision variables and number of objective functions.

Pareto Dominance: Any solution vector $\vec{u} = \{u_1, u_2, \ldots \ldots u_K\}^T$ is said to dominate over $\vec{v} = \{v_1, v_2, \ldots \ldots, v_k\}^T$ if and only if

$$f_i(\vec{u}) \leq f_i(\vec{v}) \quad \forall i \in \{1, 2, \ldots, M\}$$

$$f_i(\vec{u}) < f_i(\vec{v}) \quad \exists i \in \{1, 2, \ldots \ldots, M\} \tag{1.6}$$

Those solutions which are not dominated by other solutions for a given set are considered as non-dominated solutions.

Pareto-optimal front (POF): The front obtained by mapping these non-dominated solutions is called Pareto-optimal front (POF).

$$POF = f(\vec{x}) = \left\{ \left(f_1(\vec{x}), f_2(\vec{x}), \ldots \ldots, f_k(\vec{x}) \right) | \vec{x} \in p \right\} \tag{1.7}$$

where p is the set of non-dominated solutions.

The generalized concept of Pareto front was introduced by Pareto in 1986 [1.23].

Pareto Optimality: A point $\vec{x}^* \in \Omega$ is Pareto optimal if for every $\vec{x} \in \Omega$ and $I = \{1,2,3,....,k\}$ either $\forall_{i \in I} \left(f_i(\vec{x}) = f_i(\vec{x}^*) \right)$ or, there is at least one $i \in I$ such that $f_i(\vec{x}) > f_i(\vec{x}^*)$. The symbols f and Ω represent the objective function and the feasible region $(\Omega \in S)$ of the whole search space S respectively. In other words, $\vec{x}^*$ is Pareto optimal if there exists no feasible vector $\vec{x}$ which would decrease some criteria without causing a simultaneous increase in at least one other criterion.

Pareto optimal set: For a given MOP $\vec{f}(x),$ the Pareto optimal set p^* is defined as,

$$p^* := \{x \in \Omega \mid \neg \exists x' \in \Omega, \vec{f}(x') \leq \vec{f}(x)\} \qquad (1.8)$$

The solution of a MOP is a set of vectors which are not dominated by any other vector, and which are Pareto-equivalent to each other. This set is known as the Pareto-optimal set.

1.4 Portfolio optimization problem

Two main objective of portfolio optimization is the maximization of return and minimization of risk. In Markowitz model [2] for portfolio selection, variance is used as a measure of risk which is mathematically expressed as

$$\sigma_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \sigma_{ij} \qquad (1.9)$$

where, σ_{ij} is the covariance between assets i and j, σ_p^2 is the variance of portfolio and N denotes the number of assets available. w_i and w_j (weighting of asset) is the proportion of the portfolio held in asset i and j respectively.

The portfolio return is represented as

9

$$r_p = \sum_{i=1}^{N} w_i r_i \tag{1.10}$$

where r_i is the expected return of the asset i and r_p is the expected return of the portfolio.

In addition, constraints like budget, cardinality, ceiling and may be considered for effective PO. Hence, with the presence of two objectives as shown in (1.9) and (1.10), the problem of portfolio optimization is transformed to multiobjective optimization problem.

1.5 Motivation behind the research work

A lot of research ideas have gone into the development of heuristic algorithms based on a range of swarm intelligence techniques over the past few decades to analyze various problems in portfolio optimization. There are some significant issues in the portfolio optimization problem which needs to be addressed and resolved.

- The Portfolio optimization problem satisfying a set of constraints such as budget, floor, ceiling and cardinality is a challenging problem. These constraints have been handled by the conventional statistical and heuristic techniques using both single and multiobjective optimization. However, these techniques fail to get efficient solutions when the number of constraint increases. Hence it is required to use suitable multiobjective swarm intelligence algorithms to solve the portfolio optimization problem with more number of constraints.

- Since the introduction of the mean-variance portfolio optimization model by Harry Markowitz, considerable research attention has been made on model simplifications and the development of different risk measures. All these techniques use the mean of the past return as expected return. Hence there is a need to develop efficient ways of approach which would directly predict the future return and would be considered as

expected return.

- There is a need to develop robust portfolio optimization techniques which can efficiently handle the outliers present in the financial data.

- In many situations it is required to invest the fund in future where the future data are not available and the present data are uncertain due to the presence of outliers. In such scenario future stock has to be predicted and the expected return and variance are to be estimated. Such complex problem needs potential solution by devising robust prediction method followed by efficient optimization.

Based on the aforementioned motivations, the concept of the research work of this thesis was born. These above cited problems have been addressed in the thesis and some satisfactory solutions to each of them have been provided using multiobjective evolutionary computational techniques.

1.6 Objective of the thesis

The objective of the present research work is to propose few MOEAs for solving Portfolio optimization problem. In essence the objectives of the research work carried in the present thesis are:

- To formulate the portfolio optimization problem as a multiobjective optimization problem and to successfully apply the multiobjective PSO and bacteria foraging optimization (BFO) algorithms to solve the investment portfolio problem.

- To employ multiobjective swarm intelligence based strategies for portfolio optimization when practical constraints are present..

- To develop a methodology for future portfolio management, by generating future stock data, through prediction using artificial neural network.

- To develop multiobjective swarm intelligence based robust portfolio management technique to handle the outliers present the stock data.

- To develop improved and robust swarm intelligence techniques for portfolio management for future investment when outliers are present in input data.

1.7 Structure and Chapter Wise Contribution of the Thesis
Chapter 1: Introduction

This chapter contains an introduction to the portfolio optimization problem, its importance, the motivation behind the proposed research work and a condensed version of chapter wise contribution made in the thesis. Finally, the overall conclusion of the investigation and scope for further research work have also been outlined.

Chapter 2: Multiobjective Evolutionary Algorithms and Performance Metrics for Portfolio Optimization

The classical statistical and heuristic optimization techniques are ineffective for solving constrained portfolio optimization problem. This shortcoming has motivated the researchers to develop multiobjective evolutionary techniques to solve the problem effectively. Some well known MOEAs which have been reported in the literature are Pareto envelope based selection algorithm (PESA), Pareto-archived evolution strategy (PAES), PESA-II, strength Pareto evolutionary algorithm (SPEA), SPEA2, Micro Genetic Algorithm (Micro-GA). This chapter also outlines adaptive Pareto-archived evolution strategy (APAES) and nondominated sorting genetic algorithm-II (NSGA-II) . Two novel MOEAs, based on non-dominated sorting such as nondominated sorting multiobjective particle swarm optimization (NS-MOPSO) algorithm and multiobjective bacteria foraging

optimization (MOBFO) have been proposed in the thesis for portfolio optimization purpose. Two algorithms based on decomposition such as decomposition based particle swarm multiobjective evolutionary algorithm (P-MOEA/D) and decomposition based bacteria foraging multiobjective evolutionary algorithm (B-MOEA/D) have been proposed and suitably used for effectively solving constrained portfolio optimization problem.

Chapter 3: Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithm

This chapter addresses a realistic portfolio optimization problem with budget, floor, ceiling and cardinality constraints by formulating it as a multiobjective multiconstrained optimization problem. This problem has been solved by using proposed NS-MOPSO, MOBFO, P-MOEA/D and B-MOEA/D algorithms. Other MOEAs such as PESA-II, SPEA-II, Micro-GA, APAES, NSGA-II and 2LB-MOPSO have also been applied to the same problem for comparison purpose. The performance of these MOEAs has been evaluated and has been compared with that obtained by the single objective genetic algorithm (GA), Tabu search (TS), Simulated annealing (SA) and particle swarm optimization (PSO). The mean Euclidean distance, variance of return error and mean return error are used as performance measure. The performance of the MOEAs is also evaluated using six statistical metrics such as generation distance, inverted generation distance, spacing, diversity and convergence metrics and error ratio. The comparison is also made using Pareto front and computational time. Nonparametric statistical analysis using the Sign test and Wilcoxon signed rank test are also performed to demonstrate the pairwise comparison of MOEAs. The simulation studies are carried out for four different constrained conditions. From the simulation results it is clear that the investor does not have to invest money on all available assets rather to

invest in fewer assets (around 10 percent) to explore wide risk- return areas. The portfolio manager has the option to make a tradeoff between risks, return and number assets to decide the portfolio according to the requirement.

Chapter 4: Prediction based mean-variance Model for Multiobjective Portfolio Optimization

This chapter consists of two parts. The first part deals with a novel prediction based portfolio optimization model. In the second part, the performance of proposed prediction based portfolio optimization model is evaluated and compared with the mean-variance model.

The novel prediction based portfolio optimization model has been proposed in this chapter which differs from the mean-variance model, (i) In prediction based mean variance (PBMV) model, the expected return of each stock is its predicted return unlike that in mean-variance model, where the expected return is the mean of past returns. (ii) The individual risk of each stock and the risk between each pair of stocks are obtained from the variance and covariance of the time series of the errors of prediction, instead of from the variance and covariance of the time series of return. (iii) In PBMV model the normal variable of interest is the error of prediction of the return of stocks, while in the mean-variance model the normal variable of interest is the return of the stocks.

An efficient single layer neural network called as functional link artificial neural network (FLANN) is used for prediction which is trained with evolutionary computing. The inputs to the network are some financial and economic variables which are judiciously selected by using evolutionary algorithms. The FLANN structure is used for predicting the expected return and corresponding risk using the proposed model.

The new risk and return is calculated for each of the stock present in the market which is the predicted output of the FLANN. These are taken as two objectives to be optimized using efficient MOEAs. The results are obtained with real life data from the Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 stock indices. Experimental results show that the prediction based portfolio optimization model outperforms the conventional Markowitz model.

Chapter 5: Novel Robust Multiobjective Portfolio Optimization Schemes

In this chapter, the minimum volume ellipsoid (MVE) methodology is adapted to handle uncertainty of the stock market data. The source of uncertainty is the outliers present in the stock data which occurs due to unexpected situations. We can easily differentiate the data without outliers from unexpected data by clustering the good data using MVE method. The MVE is formed covering approximately 90 percent of the data (assuming 10 percent of the data are corrupted by outliers). In order to make the method computationally efficient, the MVE is formed by using core set and Lagrange multipliers. Thereafter, the weight factor is calculated by taking the parameters associated with the ellipsoid. Then the data are modified by multiplying with the weight factor. The weight factor is designed in such a way that it does not change the data those are present inside the ellipsoid, but those are present outside are diminished according to the weight factors. Then the desired parameters such as risk and return are calculated from the weighted data. The performance is obtained using real life data from the Hang Seng, DAX 100, FTSE 100, S&P 100, Nikkei 225 and BSE stock indices. Simulation results reveal that the proposed method exhibits good portfolio strategy in the presence of market uncertainties.

Chapter 6: Prediction Based Robust mean-variance Model for Constraint Portfolio Optimization

This chapter deals with future investment of the fund where the future data are not available and the present data are uncertain due to the presence of outliers. In order to predict the future data, the FLANN is used as prediction model. The inputs to the FLANN are technical indicators which are judiciously selected after modifying the real data by multiplying with suitable weighted factors. This FLANN structure is used for prediction of future data, which is further used for portfolio selection using the Markowitz model. The same model is again used for prediction of future return, which is subsequently used for portfolio selection using the proposed PBMV model. This approach helps in mitigating the effect of outliers in the stock data as well as provides very good portfolio strategy for future investment. A subset of 20 stocks from Hang-Seng, DAX 100, FTSE 100, S&P 100, Nikkei 225 and BSE-500 index between December 2008 to January 2012 have been selected for the present study.

Chapter 7: Conclusion and Future Work

In this chapter the overall contribution of the thesis is reported. Two novel multiobjective optimization algorithms approach based on bacteria foraging optimization and particle swarm optimization have been proposed and applied to the portfolio asset selection problem by formulating it as a multiobjective problem with many practical constraints. From the simulation results it is found that the portfolio manager has the option to make a trade-off between risk, return and number assets, to decide the portfolio according to the requirement. A new methodology is introduced for improved portfolio optimization

using predicted values obtained by artificial neural network. Improved and robust swarm intelligence techniques for portfolio management have been introduced.

The future research problems are outlined in this chapter for further investigation on the same/related topics. Incorporation of advanced local search operators into the proposed algorithm can been done which is expected to allow better exploration and exploitation of the search space. The proposed algorithm can also be tested using other real world constraints like round-lot, turnover and trading. The proposed multiobjective optimization algorithm may be applied in many other financial applications such as asset allocation, risk management and option pricing.

References

[1.1] S. H. Chen and T. W. Kuo, "Evolutionary computation in economics and finance: A bibliography", Evolutionary Computation in Economics and Finance, Heidelberg: Physica-Verlag, Heidelberg New York, 2002, pp. 419–455, iSBN 3-7908-1476-8.

 [1.2] S-H. Chen (Ed.), "Evolutionary Computation in Economics and Finance" Springer. Heidelberg, 2002. .

[1.3] H. M. Markowitz, "Portfolio selection", Journal of Finance, vol. 7, no.1, pp. 77-91, 1952.

[1.4] H.M. Markowitz, "Portfolio Selection: efficient diversification of investments", New York: Yale University Press. John Wiley & Sons, 1991.

[1.5] G. DI. Tollo, A. Roli, "Metaheuristics for the Portfolio Selection Problem", International Journal of Operations Research, vol.5, no.1, pp. 13-35, 2008.

[1.6] Y. Crama, M. Schyns, "Simulated annealing for complex portfolio selection problems", European Journal of Operational Research, vol.150, no.3, pp. 546-571, 2003.

[1.7] T. J. Chang, N. Meade, J. E. Beasley, Y. M. Sharaiha, "Heuristics for cardinality constrained portfolio optimization", Computers & Operations Research, vol.27, no.13, pp. 1271–1302, 2000.

[1.8] Tunchan Cura, "Particle swarm optimization approach to portfolio optimization", Nonlinear Analysis: Real World Applications, vol.10, no.4, pp. 2396–2406, 2009.

[1.9] J. Gao, Z. Chu, "A new particle swarm optimization based on MATLAB for portfolio selection problem", International Journal of Modeling, Identification and Control, vol. 9, no. 1/2, pp. 206 -211, 2010.

[1.10] H. R. Golmakani, M Fazel, "Constrained portfolio selection using particle swarm optimization", Expert System with Applications, vol. 38, no.7, pp. 8327-8335, 2011.

[1.11] Hanhong Zhu, Yi Wang, Kesheng Wang, Yun Chen, "Particle swarm optimization (PSO) for the constrained portfolio optimization problem", Expert System with Applications, vol. 38, no.8, pp.10161-10169, 2011.

[1.12] A. Mukherjee, R. Biswas, K. Deb and A. P. Mathur, "Multi-objective Evolutionary algorithms for the Risk-return trade-off in bank loan management", International transaction in Operations Research, vol. 9 no.5, pp. 583-597, 2002.

[1.13] K. Doerner, W. Gutjahr, R. Hartl, C. Strauss, C. Stummer, "Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection", Annals of Operations Research, vol. 131 no. 1-4, pp.79–99, 2004.

[1.14] R. Armananzas and J. A. Lozano, "A Multiobjective Approach to the Portfolio Optimization Problem", in 2005 IEEE Congress on Evolutionary Computation (CEC2005), vol. 2. Edinburgh, Scotland: IEEE Service Center, September 2005, pp. 1388-1395.

[1.15] S. C. Chiam, A.Al. Mamun and Y. L. Low, "A realistic approach to evolutionary multiobjective portfolio optimization", IEEE Congress on Evolutionary Computation (CEC) 2007, pp. 204-211.

[1.16] E. J. Elton, M. J. Gruber, S.J.Brown,W.N.Goetzmann, "Modern Portfolio Theory and investment Analysis", seventh. ed., Wiley, NewYork, 2007.

[1.17] E. F. Fama, "Portfolio analys is in a stable Paretian market", Management Science 11(3SeriesA) (1965) 404–419.

[1.18] S. J. Kon, "Models of stock returns—a comparison", The Journal of Finance 39 (1) (1984)147–165.

[1.19] A. Ben-Tal, A. Nemirovski, "Robust convex optimization", Mathematics of Operations Research, vol. 23, pp.769–805, 1998.

[1.20] D. Goldfarb, G. Iyengar, "Robust portfolio selection problems", Mathematics of Operations Research, vol.28, no. 1, pp.1–38, 2003.

[1.21] E. Zitzler, L.Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach". IEEE Trans Evolutionary Computing, vol.3. pp. 257–71, 1999.

[1.22] C. A. C. Coello, T.P.Gregorio. S. L. Maximino, "Handling Multiple objectives with particle swarm optimization", IEEE transaction on evolutionary computation,vol.8.no.3,pp. 256-279,2004.

[1.23] Vilfredo Pareto, Cours d'Économie Politique. Lausanne, F. Rouge, Volume I and II, 1896.

Chapter 2

Multiobjective Evolutionary Algorithms and
Performance Metrics for Portfolio Optimization

Chapter

2

A multiobjective optimization problem involves several conflicting objectives and has a set of Pareto optimal solutions. By initializing a population of solutions, multiobjective evolutionary algorithms (MOEAs) are able to approximate the Pareto optimal set in a single run. The MOEAs have attracted a lot of research effort in last few decades and are still one of the hottest research areas in the field of evolutionary computing. In this chapter, a brief and update overview of several MOEAs have been presented. Few application areas of MOEAs have also been dealt. Four novel MOEAs have been proposed and suitably oriented for solving portfolio optimization problem.

2.1 Introduction

Many real world optimization problems involve multiple objectives. Evolutionary algorithms (EAs) are able to approximate the whole Pareto front (PF) of a multiobjective optimization problem (MOP) in a single run due to their population based nature.

Schaffer [2.1] in 1985 introduced a multiobjective evolutionary algorithm called as vector evaluated genetic algorithms (VEGA). After his work, a lot of research effort has been made to apply EAs for solving multiobjective optimization problem. The research work on MOEAs in different aspect has been surveyed by many researchers. The survey based on generic methodologies are discussed in [2.2]-[2.5]. Similarly, some survey is based on different fields of application of MOEAs, such as engineering problems [2.6],[2.7], scheduling problems [2.8], economic and financial problems [2.9], automatic cell planning problems [2.10] and traveling salesman problems [2.11] etc. Comprehensive survey has been done by Aimin Zhou *et al.* on the development of MOEAs in 2011 [2.12]. According to algorithmic frameworks the MOEAs may be categorized as MOEAs based on non-dominated sorting, decomposition-based, memetic type and indicator based MOEAs etc. [2.12].

Different non-dominated sorting based approach includes nondominated sorting genetic algorithm (NSGA) [2.13], strength Pareto evolutionary algorithm (SPEA) [2.14], Pareto-archived evolution strategy (PAES) [2.15], Pareto envelope based selection algorithm (PESA) [2.16] etc. In this approach, the reproduction and selection operators of the MOEA guide the population iteratively towards non-dominated regions by preserving the diversity to get the Pareto optimal set. Decomposition based multiobjective evolutionary algorithm

(MOEA/D) [2.17] is based on conventional aggregation where an MOP is decomposed into a number of scalar objective optimization problems (SOPs).

The MOEAs based on the decision maker (DM's) preference was introduced by Fonseca and Fleming [2.18] in 1993. Due to the conflicts of the objectives in MOPs, the total number of Pareto optimal solutions might be very large. However, the investor may be interested in some of the preferred solutions instead of all the Pareto optimal solutions. DM provides the preference information in order to guide the search towards the preferred solution in the Pareto front (PF). Based on the role of the DM in the solution process, multiobjective optimization can be classified into *a priori, a posteriori* and interactive methods [2.19]. If the preference information is given before the search process, it is called as *a priori* method. Similarly, *a posteriori* method uses the preference information of DM after the search process. In an interactive method, the intermediate search results are presented to the DM so that one can provide the preference information for guiding the search process. Greenwood *et al.* have combined preference information in the survival criteria with Pareto ranking to solve MOPs [2.20]. Branke and Deb have incorporated the preference information into NSGA-II by modifying the definition of dominance and using a biased crowding distance based on weights [2.21]. Deb *et al.* have proposed a progressively interactive MOEA where an approximate value function is progressively generated after every generation [2.22]. Thiele *et al.* have used the DM's preferences expressed interactively in the form of reference points [2.23].

Zitzler and Künzli have suggested a general indicator based evolutionary algorithm (IBEA) to solve MOPs [2.24, 2.25]. Such MOEAs use indicators such as generational distance and hypervolume to guide the search for getting Pareto solution. The quality of an

approximated Pareto front could be measured by these scalar indicators. Basseur and Zitzler proposed an indicator-based model for handling uncertainty, in which each solution is assigned a probability in the objective space [2.25].

The MOEAs can also be categorized as memetic MOEAs where hybridization of global search and local search occur. Ishibuchi and Murata have proposed one of the first memetic MOEAs [2.26] in 1998 where the algorithm uses a local search method after applying the classical variation operators. In [2.27], Knowles and Corne have proposed a memetic Pareto archived evolution strategy to solve MOPs. The algorithm introduces a Pareto ranking based selection method and couples it with a partition scheme in objective space. Jaszkiewicz [2.28] has suggested a multiobjective genetic local search (MOGLS) algorithm for the multiobjective 0/1 knapsack problem.

The MOEAs can also be categorized in terms of generic methodologies such as genetic algorithm, particle swarm optimization, bacteria foraging optimization etc. The pioneering work in the practical application of genetic algorithm to MOP is the vector evaluated genetic algorithm (VEGA) [2.1]. For similar applications, a number of algorithms based on genetic algorithm such as non-dominated sorting genetic algorithm (NSGA) [2.13], niched Pareto genetic algorithm (NPGA) [2.29], genetic algorithms for multiobjetive optimization (MOGA) [2.18], SPEA [2.14], SPEA2 [2.30], PAES [2.15], PESA [2.16], PESA-II [2.31], NSGA-II [2.32], DMOEA [2.33], PAES [2.15], APAES [2.34] and Micro-GA [2.35] have been proposed in the literature. In the recent past, multiobjective bacteria foraging technique have been reported in [2.36]-[2.39] with different variations. Another bio-inspired technique based on particle swarm optimization to solve multiobjctive problem (MOP) known as muliobjective particle swarm optimization (MOPSO) has been proposed by

Coello *et al.* [2.40]-[2.41]. Non-dominated sorting particle swarm optimization (NSPSO) is reported in [2.42]. Some other variants of multiobjective particle swarm optimization techniques such as TV-MOPSO [2.43], FMOPSO [2.44], FCPSO [2.45], MOCPSO [2.46] and QPSO [2.47] have been proposed to solve the MOP. In [2.48], a multiobjective comprehensive learning particle swarm optimizer (MOCLPSO) has been presented. In [2.49], a two-*lbests* based multiobjective particle swarm optimizer (2LB-MOPSO) technique has been reported. A Pareto-frontier differential evolution (PDE) algorithm is dealt in [2.50]. A multiobjective differential evolution algorithm with diversity enhancement strategies is available in [2.51]. In [2.52], [2.53], a multiobjective immune system has been employed to deal with dynamic multiobjective problems with constraints. In [2.54], a multiobjective immune system has been proposed to find Pareto optimal robust solutions for bi-objective scheduling problems.

In the present study, the main objective is to solve different challenges of portfolio optimization problem which are inherently a multiobjective in nature. In this chapter two non-dominated sorting based MOEAs such as non-dominated sorting multiobjective particle swarm optimization (NS-MOPSO) and multiobjective bacteria foraging optimization has been proposed and suitably oriented for solving portfolio optimization problem. Two MOEAs algorithm based on decomposition such as decomposition based particle swarm multiobjective evolutionary algorithm (P-MOEA/D) and decomposition based multiobjective bacteria foraging optimization (B-MOEA/D) have also been proposed to solve the same problem. In this chapter, these four algorithms have been explained in details. Six other peer non-dominated sorting based algorithms such as PESA-II, SPEA-II, Micro-GA, APAES, NSGA-II and 2LB-MOPSO have also been applied to same problems,

are dealt in brief. Different performance metrics such as generation distance (GD), inverted generation distance (IGD), spacing (S), diversity metric (Δ), convergence metric (C) and error ratio (ER), which have been used to compare the performance of different algorithms are discussed in the chapter.

2.2 Overview of existing MOEAs

A majority of MOEAs in both the research and the application areas are Pareto-dominance based which are mostly the same frameworks as that of NSGA-II [2.32]. However, decomposition based multiobjective evolutionary algorithm (MOEA/D) is a recent multiobjective evolutionary algorithmic framework which is successfully applied to different fields [2.17]. Some of the peer MOEAs of both of these categories, which have been successfully applied to other fields and suitably tuned to suit for portfolio optimization problem are briefly explained in this section.

2.2.1 Non-dominated sorting based MOEAs

The non-dominated sorting based MOEAs involve two populations of individuals. The first population, or archive/external population, used to retain the ''best'' solutions are found during the search. The second population is the normal population of individuals, sometimes used to store the offspring population and in some other times it takes part in the reproduction process. The archive is updated by the "best" individuals based on information from both the population and hence elitism is ensured.

In these algorithms, a selection operator based on Pareto domination and a reproduction operator are used. The operator of the MOEAs guides the population iteratively towards non-dominated regions by preserving the diversity to get the Pareto optimal set. The

evaluation operator leads to population convergence towards the efficient frontier and helps to preserve the diversity of solutions along the efficient frontier. However, the method by which they achieve these two fundamental goals differs. Both goals are achieved by assigning a rank and a density value to every solution. The MOEAs provide first priority to non-dominance and second priority to diversity. The main difference between the algorithms lies in their fitness assignment techniques. The popular fitness assignment strategies are alternating objectives-based fitness assignment such as the VEGA [2.1] and domination-based fitness assignment such as SPEA 2 [2.30], NSGA-II [2.32] etc. The MOEAs which are based on nondominated sorting such as PESA-II, SPEA 2, Micro-GA, APAES, NSGA II and 2LB-MOPSO have been explained in brief.

(a) The PESA-II Algorithm

Corne *et al.* have proposed [2.16] Pareto envelope-based selection algorithm for solving multiobjective optimization problem. In this algorithm, the newly generated solutions B_t are incorporated into the archive one by one. A candidate child from newly generated solutions enters the archive when it is non-dominated within B_t, or it is not dominated by any current member of the archive. If the addition of a solution renders the archive over-full, then a mating selection is carried out by employing crowding measure. The crowding distance measurement is done over the archive members. Each individual in the archive is associated with a particular hyper-box. It has a squeeze factor which is equal to the number of other individuals from the archive which present in the same hyper box. The environmental selection criteria is based on this crowding measure and used for each

individual from the archive. The PESA-II algorithm proposed in [2.31] by incorporating region based selection and shows improved performance over PESA.

(b) The SPEA 2 Algorithm

In SPEA 2 mating selection is used which is based on fitness measure and it uses binary tournament operator [2.30]. It emphasizes non-dominated individuals by using a technique, which combines the dominance count and dominance rank method. Each individual is assigned a raw fitness value that specifies the number of individuals it dominates and also the number of individuals by which it is dominated. The density information is incorporated to the raw fitness by adding a value which is equal to the inverse of the kth smallest Euclidean distance to the kth nearest neighbor plus two. The archive updation is performed according to the fitness values associated with each of the individuals in the archive. Then, the updated operator returns all non-dominated individuals from the combined set of archive and the current pool. There are two possibilities, if the archive size is less than the pre-established size, the archive is completed with dominated individuals from the current pool otherwise some individuals are removed from the archive using the truncation operator. This operator is based on the distance of an individual to its nearest neighbor.

(c) The Micro-GA Algorithm

The micro-GA algorithm employs a small population and involves a reinitialization process [2.35]. Initially the random population is generated which is fed to the population memory. It is divided in two parts, replaceable and non replaceable portion. The non replaceable portion of the population memory remains unchanged during the entire run and provides the required diversity. But the other portion undergoes changes after each cycle.

The micro-GA uses three forms of elitism such as (i) it retains non-dominated solutions found within the internal cycle (ii) it uses a replaceable memory whose contents is partially refreshed at certain intervals and (iii) it replaces the population by the best solutions found after a full internal cycle of the micro-GA.

(d) The APAES Algorithm

Knowles and Corne [2.15] have suggested a simple evolutionary algorithm called Pareto Archived Evolution Strategy (PAES). In this algorithm one parent generates one offspring by mutation. The offspring is compared with the parent. If the offspring dominates the parent, the offspring is accepted as the next parent and the iteration continues. If the parent dominates the offspring, the offspring is discarded and the new mutated solution is generated which becomes the new offspring. If the offspring and the parent do not dominate each other, a comparison set of previously non-dominated individuals is used. For maintaining population diversity along the Pareto front, an archive of non-dominated solutions is considered. Newly generated offspring is compared with the members of archive to verify whether it dominates any of them. If it dominates, then the offspring enters the archive and is accepted as a new parent. The dominated solutions are eliminated from the archive. If the offspring does not dominate any member of the archive, both parent and offspring are checked for their nearness with the solution of the archive. If the offspring resides in the least crowded region in the parameter space among the members of the archive, it is accepted as a parent and a copy is added to the archive. The APAES proposed by M Oltean *et al.* [2.34] can be considered as an adaptive representation of the standard PAES. When the current solution dominates the mutated solution for a consecutive fixed number of times, it indicates that the representation of current solution has no potential for exploring

the search space from the place where it belongs. Therefore, the representation of the current solution must be changed in order to ensure a better exploration.

(e) NSGA-II algorithm

Dev and Pratab [2.32] have proposed NSGA-II for solving MOPs. The NSGA-II algorithm starts from a random population and utilizes some operators for uniform covering of Pareto set. The NSGA-II algorithm for multi-criteria optimization contains three main operators (i) a non-dominated sorting (ii) density estimation and (iii) a crowded comparison. To guide the individuals towards the efficient frontier, dominance depth method is adopted by NSGA-II. It classifies the solutions in several layers, based on the position of fronts containing the individuals. The crowding distance mechanism is employed to preserve the diversity of solutions which calculates the volume of the hyper-rectangle defined by the two nearest neighbors. Based on these values, the update operator returns the best individuals from the combination of archive and the population. Individuals with the lower rank and higher crowding distance would fill the archive. The three main characteristics of NSGA-II are (i) Non-dominated sorting algorithm is having the lower computational complexity than that of its predecessor NSGA. The maximum number of computational complexity of NSGA-II algorithm is $O(mN^2)$, where N is the population size and m is the number of objectives (ii) Elitism is maintained and (iii) No sharing parameter needs to be chosen because sharing is replaced by crowded-comparison to reduce computations.

(f) 2 LB-MOPSO Algorithms

In the next chapter, we have employed another most recently proposed evolutionary MO algorithm called the Two-*lbests* based multi-objective particle swarm optimizer (2LB-

MOPSO) [2.49] for solving portfolio optimization problem. This algorithm uses two local bests instead of one personal best and one global best to lead each particle. In order to select the first lbest for a particle, an objective is first randomly selected followed by a random selection of a bin of the chosen objective. Within this bin, the archived member with the lowest front number and among these with the highest crowding distance is selected as the first *lbest*. The second *lbest* is selected from a neighboring non empty bin with the lower front number and the smallest Euclidean distance in the parameter space to the first lbest. As each particle's velocity is adjusted by the two *lbests* from two neighboring bins, the flight of each particle will be in the direction of the positions of two *lbests* and orientated to improve upon the current solutions. A pair of *lbests* is assigned to a particle and the number of iterations the particle fails to contribute a solution to the archive is counted. If the count exceeds a predefined threshold, the particle is re-assigned to another pair of *lbest.* The two local bests are close to each other and help to enhance the local search ability of the algorithm.

2.2.2 The decomposition based MOEAs

The decomposition based multiobjective evolutionary algorithm (MOEA/D) [2.17] is another way of approach for solving the multiobjective problem which differs from non-dominated sorting algorithm. In this approach the multiobjective optimization problem is decomposed into a number of scalar objective optimization problems (SOPs). The objective of each SOP, called subproblem, is a weighted aggregation of the individual objectives.

2.2.3 The constraint handling in MOEAs

Although MOEAs have more extensively been investigated within the context of unconstrained and bound constrained MOPs, various general constraints are involved when solving real-world problems. Typically, the search space Ω of a constrained MOP can be formulated as follows [2.12]

$$\Omega = \begin{cases} g_j(x) = g_j(x_1, x_{2,...}, x_n) \leq 0 & j = 1, 2, ..., J \\ h_k(x) = h_k(x_1, x_2, ..., x_n) = 0 & k = 1, 2, ..., K \\ x_i^L \leq x_i \leq x_i^U & i = 1, 2, ..., n \end{cases} \tag{2.1}$$

where $g_j(x)$ and $h_k(x)$ are inequality and equality constraint functions, respectively. Generally, equality constraints are transformed into inequality forms, and then combined with inequality constraints using

$$G_j(x) = \begin{cases} \max\{g_j(x), 0\} & j = 1, 2, ..., J \\ \max\{|h_{j-J}(x)| - \delta, 0\} & j = J+1, J+2, ..., J+K \end{cases} \tag{2.2}$$

where δ is a tolerance parameter for the equality constraints. Due to the presence of constraints, the search space is partitioned into feasible and infeasible regions.

Coello [2.40] classifies the constraints handling methods into five categories: (1) penalty functions (2) special representations and operators (3) repair algorithms (4) separate objective and constraints (5) hybrid methods. A constrained dominance concept has been introduced by Deb *et al.* [2.23] to handle constraints in multiobjective problems. A solution x dominates a solution y if (i) x is feasible, while y is infeasible (ii) both are infeasible and x has less constraint violation than y or (iii) both are feasible and x dominates y. The solutions are ranked using the non-constrain-dominated method while the superiors are

selected to evolve. The handling of different practical constraints in portfolio optimization problem is explained in the next chapter. In the simulation the inequality constraint is considered as a soft constraint and repair operator is used to adjust the weight so as to meet it instead of transforming it in to inequality form.

2.3 The particle swarm optimization for the design of MOEAs

Kennedy and Eberhart [2.58] realized that an optimization problem can be formulated by mimicking the social behavior of a flock of birds flying across an area looking for food. This observation and inspiration by the social behavior exhibited by flocks of birds and schools of fish resulted the invention of a novel optimization technique called particle swarm optimization (PSO). Particle swarm optimization algorithms optimize an objective function by conducting a population based stochastic search. The population comprises potential solutions, called particles. These particles are randomly initialized and freely fly across the multi-dimensional search space. During flight, each particle updates its velocity and position based on the best experience of its own and the best experience of the entire population. The updating rule enables particles to move toward the desired region with a higher objective value.

In PSO [2.58] each solution is represented by a particle and the ith particle is given by $X_i = (x_{i1}, x_{i2}, x_{i3}, ..., x_{id})$, where d is the dimension of the search space. The ith particle of the swarm population has its best position $P_i = (p_{i1}, p_{i2}, ..., p_{id})$, that yields the highest fitness value. The global best position $P_g = (p_{g1}, p_{g2}, ..., p_{gd})$, is the position of the best particle that gives the best fitness value in the entire population. $V_i = (v_{i1}, v_{i2}, ..., v_{id})$ be the

current velocity of ith the particle. Particles communicate with each other and for a fully connected topology the position and velocity of each particle in next iteration are mathematically expressed as:

$$V_{id}(t) = wv_{id}(t-1) + C_1 r_1 (p_{id} - x_{id})(t-1) + C_2 r_2 (p_{gd} - x_{id})(t-1) \qquad (2.3)$$

$$x_{id}(t) = x_{id}(t-1) + \chi v_{id}(t) \qquad (2.4)$$

where $d = 1,2,...,D$ and $i = 1,2,...,N$. The size of swarm population is N. χ is a constriction factor which controls and constricts the magnitude of velocity. w is the inertia weight parameter to control exploration or exploitation in the search space. It can be a linear or nonlinear function of time or a positive constant [2.58]. r_1 and r_2 are two random values called as acceleration constants within range [0, 1].

2.3.1 Decomposition based particle swarm MOEAs

The MOEA/D decomposes the multiobjective optimization problem into N scalar optimization subproblems. It solves these subproblems simultaneously by evolving a population of solutions. At each generation, the population is composed of the best solution found so far for each subproblem. The neighborhood relations among these subproblems are defined basing on the distances between their aggregation weight vectors. A subproblem is a neighbor of another subproblem if its weight falls close to that of the other. Each subproblem is optimized in the MOEA/D by using information mainly from its neighboring subproblems. In this case each individual subproblem keeps one solution in its memory, which could be the best solution found so far for the subproblem.

The MOEA/D optimizes N scalar optimization problems rather than directly solving MOP as a whole. Therefore, it employs scalar optimization methods as each solution is

associated with a scalar optimization problem. The issues of fitness assignment and diversity

maintenance are easier to handle in the framework of MOEA/D. Several improvements on

MOEA/D have been reported in [2.55] and has been applied to a number of application

areas [2.56, 1.57].

The MOEA/D provides flexibility of using any decomposition approach, into its

framework for solving the MOPs. These approaches include the weighted sum approach,

Tchebycheff approach and the Boundary intersection approach [2.17]. If weighted sum

approach is applied to MOEA/D algorithm, it considers a convex combination of different

objectives. Mathematically it is expressed as

$$Maximize \ g^{ws}\left(x/\lambda\right) = \sum_{i=1}^{m} \lambda_i f_i(x) \tag{2.5}$$

$$subjected \ to \ x \in \Omega$$

where $\lambda = (\lambda_1,...,\lambda_m)^T$ be the weight vector i.e. $\lambda_i \geq 0$ for all $i = 1,...,m$ and

$$\sum_{i=1}^{m} \lambda_i = 1 \tag{2.6}.$$

λ is a coefficient vector in the objective function and x is the variable to be optimized.

Different weight vectors λ is used in the above scalar optimization problem to generate a set

of different Pareto optimal vectors.

Hence, the multiobjective optimization problem is decomposed into a number of

scalar objective optimization problems, called subproblem, is a weighted aggregation of the

individual objectives. In the proposed method the individual objective is optimized using

particle swarm optimization for designing decomposition based particle swarm

multiobjective evolutionary algorithm (P-MOEA/D).

2.3.2 Non-dominated sorting multiobjective particle swarm (NS-MOPSO)

In classical PSO, each particle tries to maximize its food substance obtained by moving across the multi-dimensional search space by updating its velocity and position. It is the only objective that governs the search process. But in the course of moving, it may face constraint like favorable temperature condition and it is expected that swarm should not move to a region of unfavorable temperature. If the temperature constraint is incorporated by adding a penalty function to the actual nutrient concentration then the approach leads to single objective constraint optimization. The food concentration and favorable temperature can also be considered as two separate objectives. Individual particle tries to optimize these two objectives simultaneously and can be applied to multiobjective optimization problem. PSO is extended to MOPSO in order to deal with the multiobjective problem in [2.40]. In our proposed NS-MOPSO the concept of non-dominated sorting is incorporated in MOPSO satisfying both the objectives and constraints. Those swarms whose locations represent non-dominated solutions are classified as the optimal Pareto front 1 (OPF1) and the remaining swarms are classified into higher OPFs. In this way the complete population is ranked based on Pareto dominance criteria. The locations in lower OPF1 are rich in food and the locations of higher OPFs are poor in food content. Each particle updates its velocity and position based on the best experience of its own and the best experience of the particles with lower OPF. The updating rule enables particles to move toward the lower optimal Pareto front.

The constraint handling is carried out based on the approach given by Deb *et al.* [2.32]. In this approach the normalized sum of constraint violations for all individuals are calculated. Then the individuals are classified according to the overall constraint violation. In between any two individuals if the overall violation of both of them is zero then the

ordinary ranking assignment is applied. Otherwise the individual with the lowest (or null) overall violation dominates the other one. In this proposed study of NS-MOPSO based portfolio optimization, the position of each particle represents a weight vector associated with different assets. The two fitness functions (risk and return) evaluate the fitness value for each particle.

Pseudo-code of NS-MOPSO algorithm

Step 1: Initialization of parameters

N : Population size and store the population in a list *PSOList*:

X_i: The current position of the ith particle within specified variable range

V_i : The current velocity of the ith particle within specified variable range and it has

probability of 0.5 being specified in a different direction.

The personal best position P_i is set to X.

V_{UPP} and V_{LOW} : Upper and lower bounds of the decision variable range.

MaxIterations: Maximum number of iterations.

Step 2: Evaluate each particle in the population.

Step 3: Iteration count loop: $t = t + 1$

Step 4: Identify particles that give non-dominated solutions in the population and store them in a list *NonDomPSOList*.

Step5: Calculate crowding distance value for each particle.

Step6: Resort the *NonDomPSOList* according to crowding distance values.

Step7: Number of particles: $i = i + 1$ (step through *PSOList*).

- Select randomly a global best P_g for the ith particle from a specified top part (e.g. top 5 %) of the sorted *NonDomPSOList.*

- Calculate the new velocity V_i and the new X_i based on (7) and (8) respectively.

- Add the ith particles P_i and the new X_i to a temporary population, stored in *NextPopList.*

   At this stage the P_i and X_i coexist and the size of *NextPopList* is $2N$.

Step 8: If $i < N$, go to the next particle $(i+1)$ (step 7).

Step 9: Identify particles that give non-dominated solutions from *NextPopList* and store them in *NonDomPSOList.* Particles other than non-dominated ones from *NextPopList* are stored in a list *NextPopListRest.*

Step10: Empty *PSOList* for next iteration step.

Step11: Select random members of *NonDomPSOList* and add them to *PSOList* (not to exceed the number of particles (N). Assign rest of *NonDomPSOList* as *NonDomPSOListRest*.

Step 12: If *PSOList* size $<$ Number of particles (N)

- Identify non-dominated particles from *NonDomPSOListRest* and store them in *NextNonDomList.*

- Add member of *NextNonDomLis*t to *PSOList.*

- If still the *PSOList* size $< N$, copy *NextPopListRest* to *NextPopListRestCopy*, then vacant *NextPopListRest.*

- Assign the vacant *NextPOPListRest* with the remaining particles other than non-dominated ones from *NextPopListRestCopy.*

Step 13: If *PSOlist* size < Number of particles (*N*), go to (step 12).

Step 14: If *t < MaxIterations*, go to the next iteration (step 3)

2.4 Bacteria foraging optimization algorithm for the design of multiobjective evolutionary algorithms

The evolutionary algorithms rely on the cooperative behavior of insects, birds etc. It is a fact that animals with poor foraging strategies are extinguished and those who have successfully foraging strategies survive from generation to generation and are reshaped into good ones. This idea was used by Bremermann [2.59] and subsequently by Passion [2.36] to develop bacteria foraging optimization algorithm. The way bacteria search for high gradient nutrient regions may be viewed as an optimization process. Each bacterium tries to maximize its obtained energy per each unit of time of the foraging process and avoid noxious substances. In addition the swarms communicate among individuals. The swarm behavior dealt in [2.36] is summarized as:

1. At first the bacteria are randomly placed in the region of nutrients. Subsequently they move towards high nutrient regions.

2 Those bacteria that are located in the region with noxious substances die and those at low-nutrient region disperse.

3. Bacteria with convenient region split and reproduce and tend to move towards high nutrient region.

4. The bacteria disperse to look for new nutrient region.

The E-coli bacteria of our intestine have a foraging strategy with four processes such as chemotaxis, swarming, reproduction, and elimination and dispersal. The detailed analysis of this concept is presented in [2.36],[2.37].

Let the parameters used are:

N : Number of bacteria used in the search space.

p : Dimension of the search space.

N_s : Swimming length.

N_c : Number of iterations in a chemotactic loop.($N_c > N_s$)

N_{re} : Number of reproduction.

N_{ed} : Number of elimination and dispersal events.

p_{ed} : The probability of elimination and dispersal.
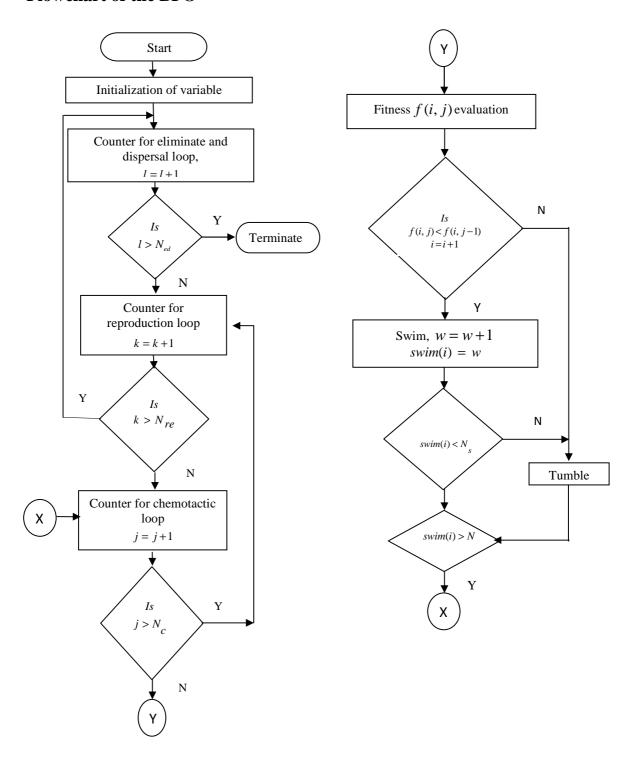
1. Chemotaxis: This process comprises of swimming and tumbling. Depending upon the rotation of flagella it decides whether to move in a predefined direction called swimming or in a different direction called tumbling. The direction of movement after a tumble can be expressed as

$$\theta(i, j+1, k, l) = \theta(i, j, k, l) + c(i).\phi(j)$$

(2.7)

Where $\theta(i, j, k, l)$ represents the position of ith bacterium at j^{th} chemotactic, k^{th} reproduction and l^{th} elimination and dispersal step. $c(i)$ and $\phi(j)$ denote the step size taken in random direction specified by the tumble and an unit length in random direction.

2. Swarming: The bacterium that has discovered the optimum path for the food tries to attract other bacteria. This process makes the bacteria bundle into groups and hence move as concentric patterns of groups with high bacterial density.

3. Reproduction: Half of the least healthy bacteria die and each of the healthy ones splits into two bacteria and are placed in the same location. This process makes the population of bacteria constant.

4. Elimination and dispersal: The life of population of bacteria changes either by consumption of nutrients or due to other environmental influence. This in turn destroys the chemotactic progress and at time it helps to place bacteria near good food source. This process facilitates in reducing the behavior of stagnation.

Flowchart of the BFO

Flow graph for bacteria foraging algorithm

2.4.1 Decomposition based bacteria foraging MOEA

The multiobjective optimization problem is decomposed into a number of subproblem which is a weighted aggregation of the individual objectives. The individual objective can be optimized using bacteria foraging optimization. The proposed multiobjective optimization algorithm is named as decomposition based bacteria foraging multiobjective evolutionary algorithm (B-MOEA/D).

2.4.2 Multiobjective bacteria foraging optimization (MOBFO) algorithm

In BFO, each bacterium tries to maximize its nutrient substance obtained and attempts to avoid noxious substances. In addition to this objective if it faces constraint like favorable temperature condition, then it is expected that bacterium should not move to a region of unfavorable temperature. The nutrient concentration and favorable temperature can be considered as two separate objectives. Individual bacterium tries to optimize these two objectives simultaneously and can be applied to multiobjective problem.

The BFO is extended to MOBFO in order to deal with the multiobjective problem [2.39]. In the proposed (MOBFO) the bacterial location represents the value of decision variables within the range of search space. The fitness values of all the variables which represent the amount of nutrients present in the environment are computed. All bacteria form a colony and are located at random positions. Applying a fast non-dominated sorting procedure [2.32] the current positions are grouped in different Pareto fronts. Those bacteria whose locations represent non-dominated solutions, are classified within the first set of optimal Pareto front 1 (OPF1) and the remaining bacteria are classified into higher OPFs. In this way the whole of the population is ranked according to Pareto dominance criteria. The

locations in OPF1 are rich in nutrients and therefore the bacteria present there have enough nutrients to eat. The locations of higher OPFs are poor in nutrient content.

During chemotaxis the bacteria in OPF1 compare the non-dominated classification of their current location with the previous ones. Hence these bacteria reach with any of the two possible movements. If both the previous and current locations are rich in nutrients (OPF1), the bacteria take a very small step in a random direction (tumble). However if the present location is rich in nutrient the bacteria take a swim. The bacteria present at higher OPF get a signal from bacteria present at OPF1 that at their location the nutrient is high. Each bacterium present at higher OPF selects randomly a strong bacterium from lower rank and moves towards its rich location, by taking a swimming step. The reproduction step consists of sorting bacteria based on their fitness function values and discarding half of them with the worst values with a higher front and lower crowding distance and duplicating the other half. Elimination and dispersal operations are carried out on bacterium with some probability and disperse it to a random location keeping the swarm size constant.

In MOBFO based portfolio optimization, the position of each bacterium represents a weight vector associated with different assets. The two fitness functions (risk and return) are evaluated for each bacterium. The constraint handling is based on the approach given by Deb *et al.* [2.32].

Pseudo-code of MOBFO algorithm

Step 1: initialization of parameters

N : Population size.

p : Dimension of the search space.

N_c : Number of chemotactic loop ($N_c > N_s$).

N_s : Number of swimming loops.

The chemotaxis loop consists of swimming and tumble for which ($N_c > N_s$).

The number of swimming loop depends on the situation which a bacterium faces at the time of chemotaxis.

N_{re} : Number of reproductions.

N_{ed} : Number of elimination and dispersal events.

p_{ed} : Probability of elimination and dispersal.

$C(i)$: Size of the step taken in the random direction specified by the tumble.

M : Number of objective functions.

Initialize the parameters: Ranks of all the bacterium to $1, m = 1, \ j = k = l = 0$

Step 2: Elimination and dispersal loop: $l = l + 1$

Step 3: Reproduction loop: $k = k + 1$

Step 4: Chemotactic loop: $j = j + 1$

Step 5: Objective functions: $m = m + 1$

Step 6: Number of bacteria $i = i + 1$

Compute the fitness function $f^m(\theta(i, j, k, l))$.

Tumbling /swimming decision:

- Tumble: Generate $\Delta(i)$ which is an unit vector towards another bacterium belonging to a front whose rank is lower. The index of the new bacterium is

chosen at random. Suppose n^{th} bacterium is chosen at random and it belongs

to a lower rank front compared to i^{th} bacterium.

Then,

$$\Delta(i) = \theta(n, j, k, l) - \theta(i, j, k, l) \tag{2.8}$$

Else,

Generate a random vector $\Delta(i)$ with each element $\Delta_r(i)$ where $r = 1, 2, \ldots, p$, a random

number on $[0, 1]$.

- Move: let $\theta(i, j+1, k, l) = \theta(i, j, k, l) + C(i) \dfrac{\Delta(i)}{\sqrt{\Delta^T(i).\Delta(i)}}$ (2.9)

  Compute $f^m(i, j+1, k, l)$

$$f_{new}^m(i, j+1, k, l) = f_{old}^m(i, j+1, k, l) + \Delta f^m(\theta(i, j+1, k, l), P(j+1, k, l)) \tag{2.10}$$

Where $\Delta f^m(\theta, p(j, k, l))$ are the cost function values of objectives to be added to the actual

cost function.

- Swim:
  - Let $q = 0$ (counter for swim length)

  - While $q < N_s$ (If climbed down is incomplete)

    Let $q = q + 1$

    - If $f_{new}(i, j+1, k, l) < f_{last}$ (if performance is

      improving)

      Now let $f_{last} = f_{new}(i, j+1, k, l)$ and

      $$\theta(i, j+1, k, l) = \theta(i, j, k, l) + c(i) \dfrac{\Delta(i)}{\sqrt{\Delta^T(i).\Delta(i)}}$$

Use $\theta(i, j+1, k, l)$ to compute the

new $f^m(i, j+1, k, l)$.

- Else, let $m = N_s$

o End of while statement.

Step 7: If $i < N$, go to the next bacterium $(i+1)$ (step 6).

Step 8: Store these new as well as the old positions in the memory which are ordered on the basis of non-dominated sorting. Only numbers of better ranked positions are retained from the sorted pool to be used in the next iteration $(j+1)$. If $j < N_c$, go to step 4 which indicates chemotactic operation to continue since the life of the bacteria is not over.

Step 9: Reproduction: Reproduction step consists of selecting half of the bacteria with a higher front and lower crowding distance and then eliminating the lower half. The remaining half is duplicated to maintain a fixed population size. For the given l and for each $i = 1,2,....,N$, $f_{final}^m = \min_{j \in \{1,2,...N_c\}} \{f_{nw}^m(i, j, k, l)$ represents the health of bacterium i.

If $k < N_{re}$, go to step 3 which means that the process has not reached the number of specified reproduction steps.

Step 10: Elimination-dispersal: Eliminate and disperse bacteria chosen with probability P_{ed} to a random location on the optimization space but retaining the bacteria population space.

If $l < N_{ed}$, then go to step 2.

Else stop

The bacteria foraging optimization steps during non-dominated sorting are outlined as

Begin

      Initialize input parameters

      Create a random initial swarm of bacteria $\theta(i,j,k,l)$, $\forall i, i = 1,2,....,N$

      Evaluate $f^{m}(\theta(i,j,k,l))$, $\forall i, i = 1,2,....,N$

  For $l = 1$ to N_{ed} Do

    For $k = 1$ to N_{re} Do

      For $j = 1$ to N_{c} Do

        For $i = 1$ to N Do

          For $m = 1$ to M Do

            Perform the chemotactic step tumble-swim or tumble-tumble operations for all bacteria and for all objectives $\theta(i,j,k,l)$ .Evaluate the cost functions of all the bacteria and for all the objectives.

          end for

        end for

      end for

Perform the reproduction step by eliminating the half worst bacteria with higher front and with lower crowding distance and duplicating the other half.

    end for

Perform the elimination-dispersal step for all bacteria with probability $0 \leq P_{ed} \leq 1$.

    end for

end

2.5 Performance measure metrics

The main objective of MOEAs is to provide solutions satisfying three objectives: (i) minimal distance to the standard efficient front or global optimal Pareto front (GOPF) (ii) good distribution (iii) maximum spread. The final Pareto optimal front obtained from different MOEAs is compared using performance metrics proposed by many researches [2.2], [2.13], [2.14]. Six different metrics defined in the sequel are used during the investigation for measuring the performance quality is given as:

(a) Generation distance (GD)

It estimates the distance of elements of non-dominated vectors found, from those standard efficient frontier [2.2] and is mathematically expressed as

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \qquad (2.11)$$

where n is the number of vectors in the set of obtained non-dominated solutions. d_i is the Euclidean distance between each of these and the nearest member of the standard efficient frontier. If $GD = 0$, all the candidate solutions are in standard efficient frontier. The smaller the value of GD the closer is the solution to the standard efficient frontier.

(b) Inverted generation distance (IGD)

This indicator [2.2] is used to measure how far the elements of the standard efficient fronts are from the non-dominated vectors found by the proposed algorithm. If $IGD = 0$, all the candidate solutions are in the global optimal Pareto front covering all its extensions.

(c) Spacing (S)

It measures the spread of candidate solution throughout the non-dominated vectors found. This metric [2.30] is mathematically expressed as

$$S \overset{\Delta}{=} \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \left(\bar{d} - d_i \right)^2} \qquad (2.12)$$

where $d_i = \min_j \left(\left| f_1^{i}(\vec{x}) - f_1^{j}(\vec{x}) \right| + \left| f_2^{i}(\vec{x}) - f_2^{j}(\vec{x}) \right| \right)$ and $i, j = 1,2,...,n$

$\bar{d} =$ mean of all d_i and n is the number of non-dominated vectors found so far. A value of

zero for this metric indicates all members of the Pareto front currently available are equidistantly spaced.

(d) Diversity metric (Δ)

This metric (Δ) measures the extent of spread i.e. how evenly the points are distributed among the approximation set in the objective space [2.13]. This metric does not require any standard efficient frontier and has a relation with Euclidean distance between solutions. It is defined as

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \bar{d} \right|}{d_f + d_l + (N-1)\bar{d}} \tag{2.12}$$

where d_i is the Euclidean distance between consecutive solutions in the obtained non-dominated set of solutions. $\bar{d}$ is the average of these distances d_i. d_f and d_l are the Euclidean distance between the extreme solutions and the boundary solutions of the obtained non-dominated set. N is the number of solutions on the best non-dominated front. If there are N solutions then there are $N-1$ consecutive distances. The low value indicates better diversity of the non-dominated solution. Its value for most widely and uniformly spread out set of non-dominated solutions is zero.

(e) Convergence metric (C)

This metric compares the quality of two non-dominated set. This matrix is computed without taking standard efficient frontier into consideration. Let A and B be two different sets of non-dominated solutions then the C metric [2.14] is mathematically expressed as

$$C(A,B) = \frac{\left|\{b \in B \mid \exists a \in A : a \preceq b\}\right|}{|B|} \tag{2.14}$$

where a and b are candidate solutions of set A and B respectively. The function C maps

the order pair (A, B) to the interval [0, 1]. If $C(A, B) = 1$, all the candidate solutions in B are

dominated by at least one solution in A. Similarly, if $C(A, B) = 0$, no candidate solutions in

B is dominated by any solution in A.

(f) Error ratio (ER)

This metric is introduced by Veldhuizen and Lamont [2.2] to indicate the percentage

of candidate solutions those are not the member of the global optimal Pareto front.

$$ER = \frac{\sum\limits_{i=1}^{n} e_i}{n} \tag{2.15}$$

where n is the number of vectors in the current set of non-dominated vectors available.

If $e_i = 0$, vector i is a member of the global optimal Pareto front and if , the reverse is true

which indicates that the candidate solutions vectors generated by the algorithm belong to the

GOPF of the problem.

2.6 Conclusion

In this Chapter, six MOEAs which have earlier been applied in PO problem are discussed in brief. Two non-dominated sorting based MOEAs such as NS-MOPSO and MOBFO have been discussed in details. Two decomposition based MOEA algorithm such as decomposition based particle swarm MOEA (P-MOEA/D) and decomposition based bacteria foraging MOEA (B-MOEA/D) have also been discussed. In P-MOEA/D and B-MOEA/D the objective of each subproblem has been optimized using PSO and BFO respectively. These four proposed MOEAs have been successfully applied to solve portfolio optimization problems in subsequent chapters.

References

[2.1] J. D. Schaffer, "Multiple objective optimizations with vector evaluated genetic algorithms. In Genetic Algorithms and their Applications", Proceedings of the International Conference on Genetic Algorithm, Lawrence Erlbaum, 1985, pp. 93-100.

[2.2] D.A.Van Veldhuizen,G.B.Lamont, "Multiobjective evolutionary algorithms: analyzing the state-of-the-art, Evolutionary Computation", vol.8. no.2. pp. 25–147, 2000.

[2.3] C.A.Coello Coello,"A short tutorial on evolutionary multiobjective optimization", in: 1st International Conference on Evolutionary Multi-Criterion Optimization, EMO 2001, in: LNCS, vol. 1993, pp. 21–40, 2001.

[2.4] A.Chinchuluun, P.Pardalos,"A survey of recent developments in multiobjective optimization", Annals of Operations Research, vol.154. no.1, pp. 29–50, 2007.

[2.5] M.Gong, L.Jiao, D.Yang, W.Ma, "Research on evolutionary multi-objective optimization algorithms", Journal Software, vol.20. no.2, pp. 271–289, 2009.

[2.6] J. Andersson, "A survey of multiobjective optimization in engineering design", Tech. Rep. LiTH-IKP-R-1097, Department of Mechanical Engineering, Linköping University, 2000.

[2.7] R.T. Marler, J.S. Arora, "Survey of multi-objective optimization methods for engineering", Structural and Multidisciplinary Optimization, vol. 26. no.6, pp. 369–395, 2004.

[2.8] D. Lei, "Multi-objective production scheduling: a survey", International Journal of Advanced Manufacturing Technology, vol.43. no.9-10, pp. 926–938, 2009.

[2.9] M.G.C. Tapia, C.A. Coello Coello, "Applications of multi-objective evolutionary algorithms in economics and finance: a survey", in: IEEE Congress on Evolutionary Computation, CEC 2007, 2007, pp. 532–539.

[2.10] F. Luna, J.J. Durillo, A.J. Nebro, E. Alba, "Evolutionary algorithms for solving the automatic cell planning problem: S survey, Engineering Optimization", vol.42. no.7, pp.671-690, 2010.

[2.11] T. Lust, J. Teghem, "The multiobjective traveling salesman problem: a survey and a new approach", in: Advances in Multi-Objective Nature Inspired Computing, in: SCI, vol. 272, pp.119-141, 2010.

[2.12] Aimin Zhou, Bo-Yang Qub, Hui Li, Shi-Zheng Zhaob, Ponnuthurai Nagaratnam Suganthan, Qingfu Zhangd, "Multiobjective evolutionary algorithms: A survey of the state of the art", Swarm and Evolutionary Computation",vol.1.no.1, pp.32-49, 2011.

[2.13] N. Srinivas, K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms". Evolutionary Computation", vol.2, no.3, pp 221-248, 1994.

[2.14] Zitzler E, Thiele L, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach". IEEE Trans. on Evolutionary Computation, vol. 3 no.4, pp. 257-271, 1999.

[2.15] J. D. Knowles and D. W. Corne, "Approximating the non-dominated front using the Pareto archived evolution strategy". Evolutionary Computing, vol.8, no.2, pp.149-172, 2000.

[2.16] D.W. Corne, J.D.Knowles, M.J.Oates, "The Pareto envelope-based selection algorithm for multiobjective optimization". In: Proceedings of Sixth International Conference on Parallel Problem Solving from Nature, Berlin. Springer-Verlag, 2000. pp. 839-848.

[2.17] Q. Zhang and H. Li, "MOEA/D: A Multi-objective Evolutionary Algorithm Based on Decomposition", IEEE Trans. on Evolutionary Computation, vol.11.no.6, pp.712-731, 2007.

[2.18] C.M. Fonseca, P.J. Fleming, "Genetic algorithms for multiobjective optimization: formulation, discussion and generalization", in: 5th International Conference Genetic Algorithms, 1993, pp. 416–423.

[2.19] K. Miettinen, "Nonlinear multiobjective optimization", Kluwer Academic Publishers, 1999.

[2.20] G. Greenwood, X. Hu, J. D'Ambrosio, "Fitness functions for multiple objective optimization problems: combining preferences with Pareto rankings", in: Foundations of Genetic Algorithms, vol. 4, 1997, pp. 437–455.

[2.21] J. Branke, K. Deb, "Integrating user preferences into evolutionary multiobjective optimization", Tech. Rep. KanGAL 2004, Indian Institute of Technology, 2004

[2.22] K. Deb, A. Sinha, P. Korhonen, J. Wallenius, "An interactive evolutionary multiobjective optimization method based on progressively approximated value functions", Tech. Rep. Kan GAL 2009005, Indian Institute of Technology, 2009.

[2.23] L. Thiele, K. Miettinen, P.J. Korhonen, J.M. Luque,"A preference-based evolutionary algorithm for multi-objective optimization", Evolutionary Computation", vol.17.no.3, pp. 411-436, 2009.

[2.24] E. Zitzler, S. Künzli, "Indicator-based selection in multiobjective search", in: Parallel Problem Solving from Nature, PPSN VIII, in: LNCS, vol. 3242, pp.832–842, 2004.

[2.25] M. Basseur, E. Zitzler, "Handling uncertainty in indicator-based multiobjective optimization", International Journal of Computational Intelligence Research, vol.2, no.3, pp.255–272, 2006.

[2.26] H. Ishibuchi, T. Murata, "A multiobjective genetic local search algorithm and its application to flowshop scheduling", IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews, vol.28. no.3, pp.392–403, 1998.

[2.27] J.D. Knowles, D.W.Corne, "M-PAES: a memetic algorithm for multiobjective optimization", IEEE Congress on Evolutionary Computation, CEC 2000, 2000, pp. 325–332.

[2.28] A. Jaszkiewicz, "On the performance of multiple-objective genetic local search on the 0/1 knapsack problem—a comparative experiment", IEEE Transactions on Evolutionary Computation, vol.6. no.4. pp.402-412, 2002.

[2.29] J. Horn, N.Nafpliotis, D.E.Goldberg, "A niched pareto genetic algorithm for multiobjective optimization",In: Proceedings of the first IEEE Conference on Evolutionary Computation, IEEE World Congress on Computation Intelligence, 27-29 June, Orlando, FL, USA, vol.1, pp. 82-87, 1994.

[2.30] E. Zitzler, M. Laumanns, L. Thiele. "SPEA2: Improving the strength Pareto evolutionary algorithm". Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.

[2.31] D.W. Corne, N.R.Jerram, J. D. Knowles, M.J.Oates, J.Martin, "PESA-II: region-based selection in evolutionary multiobjective optimization". In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), 2001, pp. 283-290.

[2.32] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II". IEEE Transactions on Evolutionary Computation, vol. 6, no.2, pp. 182–197, 2002.

[2.33] G. Yen, H. Lu, "Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation". IEEE Transactions on Evolutionary computation,vol.7.no.3, pp-253-274, 2003

[2.34] M. Oltean, C. Grosan, A. Abraham, A. Koppen, "Multiobjective optimization using adaptive pareto archive evolution strategy", 5th Int. Conf. on Intelligent Systems Design and Applications, Piscataway, NJ: IEEE Press, 8-10 Sept. 2005, pp.558-563.

[2.35] Coello C A C, Toscano G, "A micro-genetic algorithm for multiobjective optimization", Lecture Notes in Computer science. 1993.

[2.36] K M Passino, "Biomimicry of bacterial foraging for distributed optimization and control", IEEE Control System Magazine, vol.22, no. 3, pp. 52-67, 2002.

[2.37] M Tripathy and S. Mishra , "Bacteria foraging-based solution to optimize both real power loss and voltage stability limit". IEEE Transactions on Power Systems, vol.22, no.1, pp. 240-248, 2007.

[2.38] Guzman M A, Delgado A, Carvalho J D, "A bacterial chemotaxis multiobjective optimization algorithm". Rio de Janeiro, Brazil , 01 -05 June 2008.

[2.39] B K Panigrahi, V R Pandi, R Sharma and S Das, "Multiobjective bacteria foraging algorithm for electrical load dispatch problem", Energy Conversion and Management,vol.52, no.2, pp. 1334-1342, 2011.

[2.40] C. A. C Coello, M. S Lechunga, "MOPSO: A proposal for multiple objective particle swarm optimization", IEEE world congress on computational intelligence, Hawaii, 12–17 May 2002, pp. 1051–1056.

[2.41] C. A. C Coello, Gregorio Toscano Pulio and Maximino Salazar Lechuga, "Handling Multiple objectives with paticle swarm optimization", IEEE transaction on evolutionary computation, vol.8, no.3, pp .256-279, 2004.

[2.42] Li. X. "A non-dominated sorting particle swarm optimizer. Proceeding of Genetic and Evolutionary Computation", Conference 2003 (GECCO'03), Lecture Notes in Computer Science, Chicago, USA, pp. 37-48, 2003.

[2.43] P. Kumar, Bandyopadhyay, S. Kumar, "Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients". Information Sciences, vol.177, no.22, pp.5033-5049, 2007.

[2.44] L Wang, C Singh., "Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization". Electrical Power System Research, vol. 77 no. 12, pp.1654-1664, 2007.

[2.45] S. Agrawal. B. K. Panigrahi, M.K.Tiwari, "Multiobjective particle swarm algorithm with fuzzy clustering for electrical power dispatch", IEEE Transaction. Evolutionary Computing, vol.12 , no. 5, pp. 529–541, 2008.

[2.46] Cai J, Ma X, Li Q, Li L and Peng H, "A multi-objective chaotic particle swarm optimization for environmental/economic dispatch". Energy Convers Manage, vol. 50, no.55, pp. 1318–1325, 2009.

[2.47] Lu S, Sun C, Lu Z, "An improved quantum-behaved particle swarm optimization method for short-term combined economic emission hydrothermal scheduling", Energy Convers Manage, Vol.51, no.3, pp. 561–571, 2010.

[2.48] V.L. Huang, P.N. Suganthan, J.J. Liang, "Comprehensive learning particle swarm optimizer for solving multiobjective optimization problems, International Journal of Intelligent Systems", vol.21. no.2, pp. 209-226, 2006.

[2.49] S.Z. Zhao, P.N. Suganthan, "Two-lbests based multi-objective particle swarm optimizer, Engineering Optimization", vol.43. no.1, pp.1–17, 2011.

[2.50] R. Sarker, H. Abbass, "Differential evolution for solving multi-objective optimization problems", Asia Pacific Journal of Operational Research, vol.21. no.2, pp.225–240, 2004.

[2.51] B.Y. Qu, P.N. Suganthan, "Multi-objective differential evolution with diversity enhancement", Journal of Zhejiang University Science A, vol.11, pp.538–543, 2010.

[2.52] Z. Zhang, "Immune optimization algorithm for constrained nonlinear multiobjective optimization problems", Applied Soft Computing, vol.7.no.3, pp.840–857, 2007.

[2.53] Z. Zhang, "Multiobjective optimization immune algorithm in dynamic environments and its application to green house control", Applied Soft Computing, vol.8. no.2, pp.959-971, 2008.

[2.54] X. Zuo, H. Mo, J. Wu, "A robust scheduling method based on a multi-objective immune algorithm", Information Sciences, vol.179. no.19, pp.3359–3369, 2009.

[2.55] H. Li, Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II", IEEE Transactions on Evolutionary Computation, 2009; 13 (2): 284–302.

[2.56] P.C. Chang, S.H. Chen, Q. Zhang, J.L. Lin, "MOEA/D for flowshop scheduling problems", in: IEEE Congress on Evolutionary Computation, CEC 2008; 1433–1438.

[2.57] S. Pal, B. Qu, S. Das, P.N. Suganthan, "Optimal synthesis of linear antenna arrays with multi-objective differential evolution", Progress in Electromagnetics Research B 21, 2010; 87–111.

[2.58] R.C. Eberhart, J. Kennedy, "A new optimizer using particle swarm theory", in: 6th International Symposium on Micromachine and Human Science, 1995; 39–43.

[2.59] H.Bremermann, "Chemotaxis and optimization". J. Franklin Institute, pp. 397-404, 1974.

[2.60] Schott J R, "Fault tolerant design using single and multicriteria genetic algorithm optimization", M.S. thesis, Dept. Aeronautics and Astronautics, Massachusetts Inst. Technol, Cambridge, MA, May 1995.

Chapter 3

Constrained Portfolio Optimization using Multiobjective Evolutionary Algorithm

Chapter

3

This chapter addresses a realistic portfolio optimization problem as a multiobjective optimization problem by considering budget, floor, ceiling and cardinality as constraints. Four novel multiobjective evolutionary optimization algorithms, two based on non-dominated sorting and two based on decomposition have been employed to solve the problem efficiently. The performance of the proposed algorithms is compared with four single objective evolutionary algorithms such as genetic algorithm (GA), tabu search (TS), simulated annealing (SA) and particle swarm optimization (PSO) as well as a set of competitive multiobjective algorithms. The comparisons are based on three performance measures, six performance metrics, Pareto front and computational time. Nonparametric statistical analysis using the Sign test and Wilcoxon signed rank test has also carried out to demonstrate the pairwise comparison. On examining the performance metrics it is observed that the proposed MOEAs are capable of identifying good Pareto solutions maintaining adequate diversity in the presence of cardinality.

3.1. Introduction

The task of portfolio optimization is a very challenging and interesting problem in computational finance and has received attention of many researchers in the last few decades. The portfolio contains stocks, bank investments, real estate holdings, bonds, treasury bills etc. Markowitz has set up a quantitative framework for the selection of a portfolio [3.1], [3.2]. In this framework, the percentage of each available asset is selected in such a way that the total profit of the portfolio is maximized while total risk is minimized simultaneously. Hence the portfolio optimization problem is inherently a multiobjective problem. The portfolio optimization is very complicated as it depends on many factors such as preferences of the decision makers, resource allocation, growth in sales, liquidity, total turnover, dividend and several other factors. Some authors have also added some practical constraints such as floor, ceiling, cardinality etc. to Markowitz model that makes it more realistic. Inclusion of these constraints to the portfolio optimization problem makes it intractable even for small instances. With these constraints, it becomes a mixed integer programming problem with quadratic objective functions. Researchers have tried to solve the constrained portfolio optimization problem using (a) classical/exact method such as active set methods, interior point techniques (b) heuristics approach such as single objective heuristic approach and multiobjective heuristic approach.

(a) Classical method

Bienstock [3.3] in 1996 have presented a 'branch and cut algorithm' for the exact solution of the cardinality constrained portfolio optimization problem. Shaw *et al.* [3.4] have

used a 'lagrangean relaxation based procedure' for solving the cardinality constrained portfolio optimization problem using the exact/classical method. Recently Vielma *et al.* [3.5] have proposed a "branch-and-bound algorithm" by using classical method for solving cardinality constrained portfolio optimization problem based on a lifted polyhedral relaxation of conic quadratic constraints. Bertsimas and Shioda [3.6] have introduced an approach for the cardinality constrained portfolio optimization problem using "Lemkes pivoting algorithm". In 2010 Gulpinar *et al.* [3.7] have applied "difference of convex functions programming" for getting the exact solution of the cardinality constrained portfolio optimization problem. Considering the floor and cardinality constraint, Li *et al*. [3.8] have solved the portfolio optimization problem. However, these classical/traditional optimization methods meant for solving this cardinality constrained portfolio optimization problem are likely to be trapped to local minima solutions. Hence there is a need to propose new approach which avoids this limitation to the extent possible.

(b) Heuristic approach

To overcome the shortcomings of the classical methods, different efficient heuristic methods are developed. Chang *et al.* [3.9] in 2000 have presented three heuristic algorithms based on genetic algorithm, Tabu search and simulated annealing for finding the cardinality constrained efficient frontier. This may be considered to be the first heuristic approach to solve cardinality constrainted portfolio optimization problem. Computational results are presented for five test problems of five different stock indices such as Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 having 31, 85, 89, 98 and 225 assets respectively. These data are publicly available from OR-Library maintained by Prof. Beasley [3.33]. In our study also we have used these data. Many researchers have followed the work of Chang

et al. [3.9] for solving the same problem using different metaheuristics. This approach can be divided into two categories such as single objective or multiobjective metaheuristic.

(i) Single objective heuristic approach.

Fernandez and Gomez have applied a Hopfield neural network along with three heuristics GA, SA and Tabu Search to the portfolio optimization problem [3.10]. Pai and Michel (2009) have applied a clustering approach for choosing the assets in the portfolio, thereby eliminating the cardinality constraint [3.11]. Crama and Schyns have proposed a simulated annealing approach to the constrained portfolio optimization problem, (that includes cardinality, turnover and trading as parameters) [3.12]. Derigs and Nickel have also used simulated annealing based metaheuristic to solve the portfolio management problem [3.13]. Particle swarm optimization has been applied to solve portfolio optimization problem in [3.14]. Genetic algorithm [3.15] has been applied to solve the portfolio optimization problem considering different constraints such as minimum transaction lots and cardinality. Chang *et al.* [3.16] in 2009 have used three other measures of risk such as semi-variance, mean absolute deviation and variance with skewness for modeling of MOEA using GA.

The aforementioned models are most popular approach to solve portfolio optimization problem considering the overall objectives as a weighted sum of the two objectives. However, solving this multiobjective problem with these single objective evolutionary algorithms (SOEAs) require the repeated use of an optimization technique to find one single solution on the efficient frontier per run. Hence it is a time consuming process to get the entire Pareto front. Furthermore, a uniform set of weight does not guarantee a uniformly distributed set of efficient points [3.17]. The diversity of solutions along the efficient frontier is of much importance as certain trade-off portfolios of interest

may be missed if they are concentrated in a small area of the efficient frontier. One more shortfall of this approach is that, it cannot find all efficient points [3.14].

(ii) Multiobjective heuristic approach

To overcome the shortcomings of single objective optimization approach, many researchers have applied multiobjective evolutionary algorithms (MOEAs) to solve the problem that does not require any weight parameter. One of the main advantages of MOEAs is that it gives a set of possible solutions called as a Pareto optimal solution in a single run and in a reasonable amount of time [3.17]. Pareto ant colony optimization (PACO) has been introduced in [3.18] for solving the portfolio selection problem and compared its performance has been compared with other heuristic approaches such as Pareto simulated annealing and the non-dominated sorting genetic algorithm. Mishra *et al.* [3.19], [3.20] have applied different MOEAs to solve portfolio optimization problem considering only budget constraint. The literature survey reveals that the cardinality constraint has been addressed in [3.21], and [3.22]. The floor, ceiling and cardinality constraints have been dealt with in [3.23]. However, all these aforementioned studies lack of generality and in depth analysis in examining how the presence of these constraints affects the decision of the portfolio manager. Hence the portfolio optimization problem satisfying a set of constraint is a challenging problem for researchers. In the proposed work the combined presence of practical constraints such as budget, floor, ceiling and cardinality is considered to make the portfolio optimization problem more realistic. Anagnostopoulos and Mamanis [3.24] have adopted a tri-objective view of the problem and have applied three multiobjective evolutionary optimization algorithms such as NSGA-II, SPEA2 and the PESA. In 2011 the same authors compare the effectiveness of five state-of-the-art multiobjective evolutionary

algorithms (MOEAs) together with a steady state evolutionary algorithm on the mean–variance cardinality constrained portfolio optimization problem (MVCCPO) [3.25]. Burbank *et al.* [3.26] have used a multiobjective evolutionary algorithm in conjunction with the critical line algorithm of Markowitz. They have included a constraint (involving additional zero-one variables) based on the German investment law.

In the aforementioned studies a particular case of constraint condition has been analyzed but in-depth analysis of different combination of constraints is not considered and hence it lacks generality. In most cases the inequality in the cardinality restriction has been replaced by an equality restriction. Hence handling of these constraints is very challenging and there is a need to apply efficient MOEAs algorithm for achieving efficient solution.

This chapter addresses the portfolio optimization problem considering budget, floor, ceiling and cardinality constraints. Two multiobjective evolutionary algorithms (MOEAs) based on non-dominated sorting such as NS-MOPSO and MOBFO as discussed in previous chapter have been applied to the portfolio optimization problem. Two MOEA algorithms based on decomposition (MOEA/D) such as decomposition based particle swarm evolutionary algorithm (P-MOEA/D) and decomposition based bacteria foraging evolutionary algorithm (B-MOEA/D) have also been proposed for solving the same problem. The performance of these algorithms is compared with some peer MOEAs algorithms such as PESA-II [3.27], SPEA-II [3.28], Micro-GA [3.29], APAES [3.30], NSGA-II [3.31], and 2LB-MOPSO [3.32]. The performance obtained from the study is also compared with those of single objective evolutionary algorithms such as genetic algorithm (GA), tabu search (TS), simulated annealing (SA) and particle swarm optimization (PSO) identical to [3.9]. The comparisons of the performance include, three error measures, six

performance metrics, Pareto front and computational time. Nonparametric statistical test such as the Sign test and Wilcoxon signed rank test are also performed to demonstrate the performance of proposed algorithms.

3.2. Portfolio optimization problem with different practical constraints

As discussed in Chapter-1 the variance of Markowitz model [3.2] are mathematically expressed as:

$$\sigma_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \sigma_{ij} \tag{3.1}$$

where, σ_{ij} is the covariance between assets i and j, σ_p^2 is the variance of portfolio and N denotes the number of assets available, w_i and w_j (weighting of asset) is the proportion of the portfolio held in asset i and j respectively. The portfolio return is represented as:

$$r_p = \sum_{i=1}^{N} w_i r_i \tag{3.2}$$

where, r_i is the expected return of the asset i, r_p is the expected return of the portfolio, subjected to constraints. These constraints are:

(a) Budget constraint

$$\sum_{i=1}^{N} w_i = 1 \tag{3.3}$$

Eq. (3.3) shows the budget constraint which ensures that the sum of the weights associated with each asset is equal to one i.e. all the available money is invested in the portfolio. The budget constraint is an equality linear constraint. This constraint makes the

portfolio optimization problem a convex problem and hence is referred to as convex constraint.

(b) Floor constraint

It is expressed as:

$$a_i z_i \leq w_i, \ 0 \leq a_i \leq 1 \tag{3.4}$$

$$where \qquad z_i = \begin{cases} 1, & for \ w_i > 0 \\ 0, & otherwise \end{cases} \tag{3.5}$$

The decision variable z_i is 1 or 0 depending upon an asset $i(i = 1,2,...., N)$ is held or not respectively. a_i is the lowest limit on the proportion of any asset that can be held in a single portfolio if it selected. It is the lower limit on the proportion of each asset that can be held in a single portfolio. It prevents excessive administrative cost for very small holdings which have insignificant influence on the performance of the portfolio. It is called as minimum proportion constraint or floor constraint.

(c) Ceiling constraint

It is expressed as:

$$w_i z_i \leq b_i, \ 0 \leq b_i \leq 1 \tag{3.6}$$

$$where \qquad z_i = \begin{cases} 1, & for \ w_i > 0 \\ 0, & otherwise \end{cases} \tag{3.7}$$

The decision variable z_i is 1 or 0 depending upon an asset $i(i = 1,2,...., N)$ is held or not respectively. b_i is the maximum limit on the proportion of any asset that can be held in a single portfolio if it will be selected. It is the highest limit on the proportion of each asset

that can be held in a single portfolio. It prevents the excessive exposure to any portfolio

which is a part of the institutional diversification policy. It is called as ceiling constraint.

(d) Cardinality constraint

This constraint specifies the number of assets that a portfolio can hold. The

cardinality constraint K denotes the number of assets a portfolio manager can invest money

out of N available assets. The decision variable z_i is 1 or 0 depending upon an asset

$i(i = 1,2,....,N)$ is held or not respectively.

$$\sum_{i=1}^{N} z_i = K \tag{3.7}$$

This equation ensures that exactly K assets of N available assets are held.

It also specifies the maximum and minimum number of assets that a portfolio can

hold and is expressed as:

$$K_L \leq \sum_{i=1}^{N} z_i \leq K_U \tag{3.8}$$

It implies that the number of assets in the portfolio lies between K_L and K_U $(K_L \neq K_U)$.

In this model the risk is formulated using covariance. An equivalent formulation can

be obtained using correlations because the covariance between the returns of assets i and j

is equal to the product of the standard deviations in return for assets i and j multiplied by

the correlation between returns for assets i and j.

The Markowitz unconstrained model is shown in (3.1) to (3.3) with

$0 \leq w_i \leq 1, i = 1,2,3...,N$. Considering all constraints from (3.1) to (3.7) the problem

becomes a quadratic mixed-integer program (QMIP) which has been solved by Chang *et al.*

in [3.9]. Most of the research works on MOEAs solve the QIPM by relaxing the equality

constraint of (3.7) to inequality constraint i.e. $\sum_{i=1}^{N} z_i \leq K$. But the portfolio problem is

solved for a fixed K asset or for a range of K assets as shown in (3.7) and (3.8)

respectively. Hence with the presence of two objectives as shown in (3.1) and (3.2) and

constrains shown in (3.3) to (3.8) the problem of portfolio optimization becomes a

multiobjective one and the aim is to find all non-dominated set of solutions.

3.2.1 Single objective formulation of portfolio optimization

This multiobjective optimization problem is usually solved with single objective

solution techniques. The most popular approach considers the overall objectives as a

weighted sum of these two objectives [3.9] and can be expressed mathematically as:

$$V = \lambda \left[\sigma_p^2 \right] - (1-\lambda) \left[r_p \right]$$
$$= \lambda \left[\sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \sigma_{ij} \right] - (1-\lambda) \left[\sum_{i=1}^{N} w_i r_i \right] \tag{3.9}$$

Now the only objective to be minimized is $'V'$. The efficient portfolios from the

minimum variance portfolio ($\lambda = 1$) to the maximum return portfolio ($\lambda = 0$) can be found

out by repeatedly varying the parameter value λ and solving a sequence of optimization

problems (for each λ). Hence such a formulation yields non-dominated solutions by suitably

varying the λ factor from 0 to 1 with a small increment of 0.02. The main advantage of

these approaches is that it reduces the multiobjective problem to a scalar optimization

problem and any single objective metaheuristics algorithm can then be applied. In this

chapter four single objective evolutionary algorithms (SOEAs) such as the PSO, GA, TS and

SA have been applied for solving the multiobjective portfolio optimization problem identical

to those dealt in [3.9].

However, solving the portfolio optimization problem is a time consuming process to get the entire Pareto front. A uniform set of λ does not guarantee a uniformly distributed set of efficient points [3.17].

3.2.2. Multiobjective formulation of portfolio optimization

The portfolio optimization problem which is inherently a multiobjective problem can be efficiently solved by using the MOEAs.

(a) Formulation for non-dominated sorting based MOEAs

The multiobjective portfolio optimization problem can be solved by MOEAs based on non-dominated sorting which do not combine the two objectives to obtain the Pareto optimal solution set. Here the two objectives are taken individually and try to optimize both simultaneously.

The main objective is to maximize return r_p and minimize risk σ_p^2. The proposed NS-MOPSO and MOBFO are suitably oriented in such as to minimize the two objectives. To express both the objectives in minimization form, the second objective r_p is expressed as $-r_p$. In addition to these objectives, different practical constraints mentioned in (3.3) to (3.8) are also considered. Accordingly portfolio problem is expressed as

$$\textit{Minimize } \sigma_p^2 \textit{ and } -r_p \textit{ simultaneously considering all constraint} \qquad (3.10)$$

Hence with the presence of this multiple objectives and constraints, the problem becomes a multiobjective minimization problem. By solving this, a set of efficient solutions called the efficient frontier is obtained. This is a curve lies between the global minimum risk

portfolio and the maximum return portfolio. In this study, this efficient frontier is termed as

Pareto front.

(b) Formulation for decomposition based MOEAs (MOEA/D)

As discussed in Chapter-2 in the decomposition based MOEA (MOEA/D) [3.33]

approach the multiobjective optimization problem is decomposed into a number of scalar

objective optimization problems (SOPs). The optimal solution to the scalar optimization

problem is expressed as:

$$Maximize \; g^{ws}\left(x/\lambda\right) = \sum_{i=1}^{m} \lambda_i f_i(x) \tag{3.11}$$

$$Subjected \; to \; x \in \Omega$$

In portfolio optimization problem the number of objectives m is two i.e. risk and

return. For applying MOEA/D, the portfolio optimization problem can be expressed as:

$$Maximize \; g\left(w/\lambda\right) = \sum_{i=1}^{2} \lambda_i f_i(w) \tag{3.12}$$

where $\lambda_i \geq 0$ for all $i = 1,2$ and $\sum_{i=1}^{2} \lambda_i = 1$, subjected to $x \in \Omega$, λ is a coefficient vector of

the objective function and x is the variable to be optimized. The two functions $f_1(x)$ and

$f_2(x)$ are to be maximized. To generate a set of different Pareto optimal vectors, one can

use different weight vectors λ in the above scalar optimization problem. In a single run, a

set values of λ is utilized and using the neighborhood concept the complete set of solutions

on the Pareto front is obtained.

Since the objective is to maximize return r_p and minimize risk σ_p^2. The same may be

expressed in maximization form as $-\sigma_p^2$. In addition to these objectives, different practical

constraints mentioned in (3.3) to (3.8) are also considered. Accordingly portfolio problem is expressed as:

$$Maximize \; -\sigma_p^2 \; and \; r_p \; considering \; all \; constraints \; together \qquad (3.13)$$

Hence in the presence of this multiple objectives and constraints, the problem becomes a multiobjective maximization problem. Individual objectives are optimized using any single objective heuristic optimization technique. In the thesis work we have applied PSO and BFO to optimize it. The constraints are handled in the same way in case of conventional PSO or BFO algorithm. A set of Pareto solution is obtained by solving (3.13) in a single run.

3.3. Simulation study

The algorithms are coded in MATLAB and were run on a PC with Intel Core2 Duo 3.0 GHz with 4 GB RAM.

3.3.1. Data Collection

The test data, which have been used in [3.9], were obtained from OR-Library (Beasley, 1996) available in [3.34]. The data corresponds to weekly prices between March 1992 and September 1997 from different well known indices of Hang Seng in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA and Nikkei 225 in Japan. The numbers of different assets for the above benchmark indices are 31, 85, 89, 98 and 225 respectively. In each data set the return of individual assets and the correlation between assets are given. The covariance between the assets, evaluated from the correlation matrix, is used for calculating the risk of portfolio. The standard efficient frontiers (Global optima

Pareto front) for each of these data sets are available in files PORTEF-1 to PORTEF-5 [3.34].

At first, four assets from Hang-Seng stock indices are considered. The mean return, standard deviation and the correlation matrix among these four assets are shown in Table-3.1.

Table-3.1. The mean return, standard deviation and correlation matrix for first four assets of Hang-Seng stock indices

Asset	Mean return	Standard deviation	Correlation Matrix			
			1	2	3	4
1	.001309	.043208	1	.562289	.746125	.707857
2	.004177	.040258		1	.625215	.570407
3	.001487	.041342			1	.757165
4	.004515	.044896				1

3.3.2. Solution representation and encoding

In order to allow for a fair comparison, we have chosen all algorithms to have the same solution representation. We have implemented the hybrid representation proposed by Streichert *et al.* [3.35] which seems to be more appropriate for portfolio optimization. In hybrid representation, two vectors are used for defining a portfolio: a binary vector that specifies whether a particular asset participates in the portfolio, and a real-valued vector used to compute the proportions of the budget invested in the assets:

$$\Delta = \{z_1, \ldots, z_n\}, \quad z_i = \{0, 1\}, i = 1, \ldots, n.$$
$$W = \{w_1, \ldots, w_n\}, \ 0 \leq w_i \leq 1, i = 1, \ldots, n \tag{3.8}$$

3.3.3. Constraints satisfaction

To meet the budget constraint, the simplest strategy is to normalize the weights so that the total sum of weights will be equal to one. This can be mathematically shown as:

$$New\ \ w_i = \frac{w_i . z_i}{\sum_{i=1}^{n} w_i . z_i} \tag{3.9}$$

To satisfy the cardinality constraint, the following repair condition for W is applied. If the number of assets in the portfolio i.e., the number of 1's in Δ of (3.8), exceeds the maximum allowed, those assets that have the minimum weight in W is deleted (by changing its value from 1 to 0 in Δ).

If the floor and ceiling constraint are included, then the weight values are to be within a specific range. For this case, the simple strategy of normalizing the total weights to one so as to meet the budget constraint is no longer applicable, since the normalized weights might not be within the limits.

Hence the fitness evaluation for the proposed representation needs to be modified. The modified fitness evaluation has to be initialized with an empty portfolio where assets are to be added iteratively. However, the various values in the weight vector will have to be adjusted to the floor and ceiling constraint. This can be represented as:

$$w_{adjusted} = a_i . z_i + \frac{w_i . z_i}{\sum_{i=1}^{n} w_i . z_i} \left(b_i . z_i - \sum_{i=1}^{n} a_i . z_i \right),\ \ i = 1,\ldots,n. \tag{3.10}$$

If weight has to adjust for budget and floor constraint and there being no restriction on the upper limit (ceiling constraint) then the adjusted portfolio weight can be computed using the following equation:

$$w_{adjusted} = a_i . z_i + \frac{w_i . z_i}{\sum_{i=1}^{n} w_i . z_i} \left(1 - \sum_{i=1}^{n} a_i . z_i \right), \ i = 1, \ldots, n. \tag{3.11}$$

Similarly, if weight has to adjust for budget and ceiling constraint and there being no restriction on lower limit (floor costraint) then the adjusted portfolio weight can be computed using the following equation:

$$w_{adjusted} = b_i . z_i - \frac{w_i . z_i}{\sum_{i=1}^{n} w_i . z_i} \left(b_i . z_i \right), \ i = 1, \ldots, n. \tag{3.12}$$

3.3.4. Parameters used in the simulation of MOEAs

Identical schemes for all tested algorithms are used in order to ensure a fair comparison. For selecting the parents, binary tournament selection is used for all genetic algorithms based MOEAs. For reproducing the offspring population, the uniform crossover operator is applied in each string of the chromosome. In uniform crossover two selected individuals generate a single child and its value for each array is selected with equal probability from one or another parent. The children were considered also for mutation having some probability which is mentioned in next section.

The conceptual framework for parameter tuning of different evolutionary algorithm is presented in [3.37]. For all the six MOEAs the population size and number of generations are taken as 100 and 10000 respectively. For the MOEAs based on genetic algorithm such as PESA-II, APAES, Micro-GA, SPEA2 and NSGA-II one chromosome represents one set of

weights of assets and each gene represents weight of one asset. In NS-MOPSO, 2LB-MOPSO and P-MOEA/D the position of each particle represents a weight vector associated with different assets. In MOBFO and B-MOEA/D the position of each bacterium represents the weight given to one asset. The dimensions of search space depend on the number of assets of the stock. After several experiments with different parameters, the final parameters of fine-tuned algorithms are mentioned below.

PESA-II: The internal and external population size is taken as 50, uniform crossover is taken having rate of 0.8. It has a mutation rate of $1/L$, where L refers to the length of the chromosome string that encodes the decision variables. The grid size i.e. the number of division per dimension is set at 10.

APAES: The number of times the current solution dominates the mutated solutions is fixed at 20.The crossover is uniform and is fixed at 0.8. Mutation rate is taken as 0.05.

SPEA 2: The crossover is taken as uniform. The crossover and mutation rate is taken as 0.8 and 0.05 respectively. The archive size is fixed at 50.

NSGA-II: The uniform crossover and mutation rates are taken 0.08 and 0.05 respectively.

Micro-GA: An external memory of 100 individuals, 5 percent of non-replaceable memory and 25 subdivisions of the adaptive grid are used. The crossover rate of 0.9 and mutation rate of $1/L$ (L = length of the chromosomic string) are chosen for this algorithm.

NS-MOPSO: Velocity having probability of 0.5 being specified in a different direction. The upper and lower bounds of the decision variable range V_{UPP} and V_{LOW} are fixed at 0.06 and 0.5 respectively.

MOBFO: Values of various parameters for the proposed MOBFO algorithm are provided in Table-3.2.

Table-3.2. Parameters of MOBFO

N	p	N_c	N_{re}	N_{ed}	p_{ed}	$C(i)$	M
100	31	100	100	100	0.15	0.10	2

2LB-MOPSO: The parameter $w = 0.862$, $C_1 = C_2 = 2.05$. Each objective function range in the external archive is divided into a number of bins i.e. n_bin and it is set to 10.

P-MOEA/D: Each subproblem of P-MOEA/D has been optimized using particle swarm optimization. The parameter $w = 0.862$ and $C_1 = C_2 = 2.05$.

B-MOEA/D: Each subproblem of B-MOEA/D has been optimized using bacteria foraging optimization. The values of various parameters used are provided in Table-3.3.

Table-3.3 Parameters of B-MOEA/D

N	N_s	N_c	N_{re}	N_{ed}	p_{ed}
100	50	100	100	100	0.15

3.3.5. Nonparametric statistical tests for comparing algorithms

The interest in nonparametric statistical analysis has grown recently for comparing evolutionary and swarm intelligence algorithms [3.37]. The pairwise comparisons are the simplest kind of statistical test which can be applied within the framework of an experimental study. Such tests compare the performance of two algorithms when applied to a common set of problems. In this Chapter the Sign test and Wilcoxon signed rank test

[3.37] are carried out to compare the performance pairwise. In simulation work the two tests are carried out by comparing all the MOEAs algorithms with the NS-MOPSO algorithms.

The Sign test requires counting the number of wins achieved either by NS-MOPSO or by the comparison algorithm. The Wilcoxon signed rank test is analogous to the paired t-test in nonparametric statistical procedure [3.37]. The aim of Wilcoxon signed rank test is to detect the difference between the behavior of two algorithms.

3.3.6. Experimental results

The standard efficient fronts for five stock indices such as Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 are depicted in Figs 3.1-3.5. which show the tradeoff between risk (variance of return) and return (mean return).
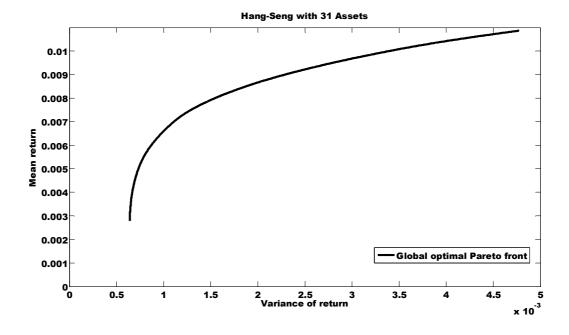
Fig.3.1.Global optimal Pareto front for Hang-Seng, stock indices
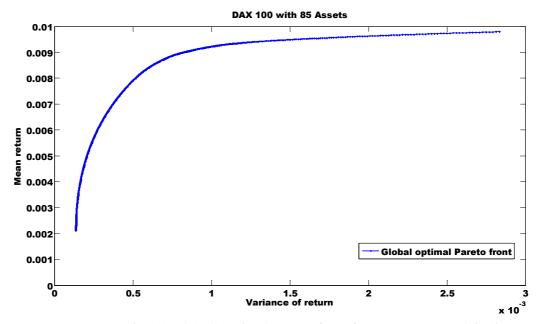
Fig.3.2.Global Optimal Pareto front for DAX 100 stock indices
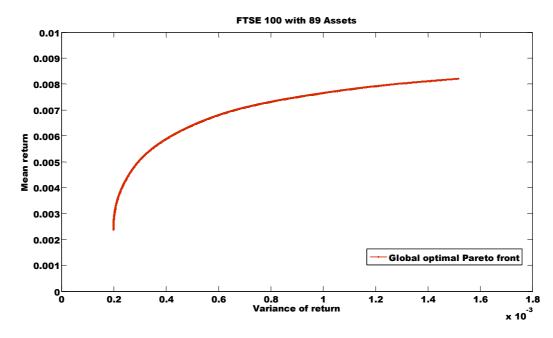
Fig.3.3. Global Optimal Pareto front for FTSE 100 stock indices
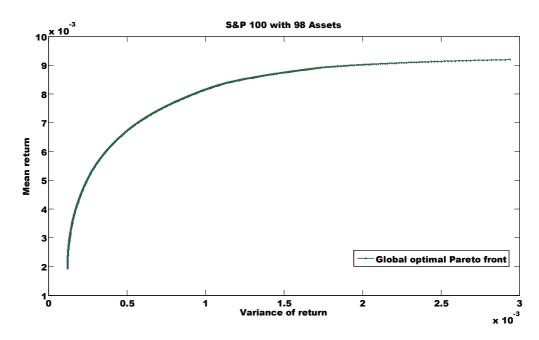
Fig.3.4. Global Optimal Pareto front for S&P 100 stock indices
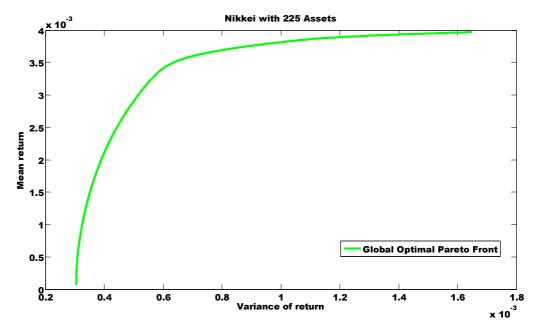
Fig.3.5. Global Optimal Pareto frontier for and Nikkei 225 stock indices

The effects of four different practical constraints such as budget, floor, ceiling and cardinality on portfolio have been analyzed by examining the resultant Pareto front achieved. The theoretical implementation of the constraint is that it limits the portfolio size and hence influences the level of return and the possible risk. The experiments have been carried out to study four distinct cases of constraint conditions.

Case 1: Budget constraint

Case 2: Fixed cardinality with budget constraint

Case3: Budget, floor, ceiling and cardinality constraint

Case 4: Variable cardinality with budget, floor and ceiling constraint.

Case 1: Budget constraint

Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 benchmark indices have 31, 85, 89, 98 and 225 assets respectively. In our experiment for testing we have applied ten MOEAs to Nikkei 225 stock indices as it has the highest number of assets to test them. The frontiers obtained have been shown in Fig.3.6.

It is evident that the MOBFO is capable of providing better solutions in comparison to other five algorithms, as its Pareto front is closer to the standard efficient frontier. The Pareto front obtained from NS-MOPSO, P-MOEA/D and B-MOEA/D algorithm are comparable with each other and better than other six competitive MOEAs.
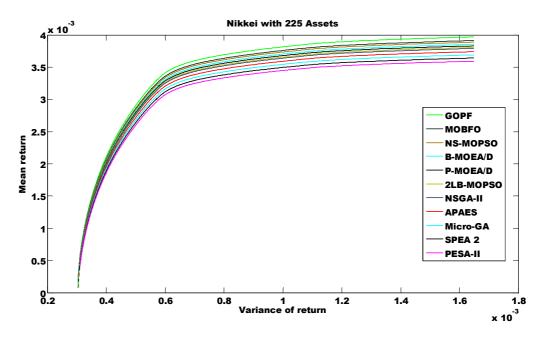
Fig.3.6.Global optimal Pareto front and ten MOEAs efficient frontiers

For Nikkei 225 stock indices

Further, the performance of ten different MOEAs is evaluated using six different metrics such as S, GD, IGD, Δ, ER and C metrics. Each algorithm is applied to Nikkei 225 market for 25 independent runs. The maximum, minimum, average and standard deviation value of S, GD, IGD, Δ and ER metrics for 25 independent runs are calculated and are shown in Table-3.4.

The smallest value of standard deviation obtained by the 2LB-MOPSO algorithm indicates better consistency compared to other algorithms. The mean value of five metrics S, GD, IGD, Δ and ER for different MOEAs in graphical form are shown in Figs.3.7 to 3.11.

Table-3.4.Comparison of performance evaluation metrics obtained using different MOEAs

Algorithm		PESA-II	SPEA 2	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO	MOBFO
S	Max.	3.21E-5	7.43E-6	7.12 E-6	6.86 E-6	6.54 E-6	5.12 E-6	5.93 E-6	5.99 E-6	5.38 E-6	5.22E-6
	Min.	1.87E-5	5.23E-6	4.54 E-6	4.12 E-6	3.98-6	1.88-6	2.51E-6	2.38 E-6	2.32 E-6	2.33E-6
	Avg.	2.33E-5	6.36E-6	5.87 E-6	5.12 E-6	4.74E-6	3.53E-6	3.62E-6	3.93 E-6	3.48 E-6	3.45-6
	Std.	0.58E-5	1.58E-6	1.21 E-6	1.01 E-6	1.53E-6	0.82E-6	0.87E-6	0.98 E-6	0.76 E-6	0.85E-6
GD	Max.	2.54E-2	2.01E-3	7.20 E-3	6.21 E-4	7.23E-4	2.01E-4	2.63E-4	2.92 E-4	2.12 E-4	2.16E-4
	Min.	1.01E-2	0.89E-3	4.32 E-3	4.10 E-4	5.23E-4	1.02E-4	1.65E-4	1.36 E-4	1.02 E-4	1.10E-4
	Avg.	1.76E-2	1.02E-3	5.45 E-3	5.23 E-4	6.72E-4	1.36E-4	1.76E-4	1.73 E-4	1.58 E-4	1.45E-4
	Std.	0.42E-2	0.28E-3	1.36 E-3	1.31 E-4	1.48E-4	0.32E-4	0.57E-4	0.52 E-4	0.38 E-4	0.36E-4
IGD	Max.	11.2 E-3	10.8 E-3	10.2 E-3	2.32 E-3	9.98 E-4	8.52 E-4	9.10 E-4	9.21 E-4	8.45 E-4	8.30 E-4
	Min.	7.32 E-3	7.02 E-3	6.98 E-3	0.98 E-3	7.02 E-4	6.80 E-4	7.67 E-4	7.76 E-4	6.35 E-4	6.45 E-4
	Avg.	9.83 E-3	9.37 E-3	8.32E-3	1.28 E-3	8.72E-4	7.20 E-4	8.20 E-4	8.23 E-4	7.15E-4	7.10 E-4
	Std.	2.77 E-3	2.35 E-3	2.08E-3	3.20E-4	2.18E-4	1.64 E-4	2.02 E-4	2.07 E-4	1.81E-4	1.74 E-4
Δ	Max.	6.78E-1	4.34E-1	4.12 E-1	3.99 E-1	3.34E-1	2.32E-1	2.34E-1	2.41 E-1	2.43 E-1	2.45E-1
	Min.	4.23E-1	2.89E-1	2.76 E-1	2.54 E-1	1.89E-1	1.02E-1	1.20E-1	1.22 E-1	1.20 E-1	0.99E-1
	Avg.	5.93E-1	3.86E-1	3.27 E-1	3.02 E-1	2.96E-1	1.42E-1	1.45E-1	1.49 E-1	1.34 E-1	1.33E-1
	Std.	1.48E-1	0.93E-1	0.86 E-1	0.78 E-1	0.78E-1	0.44E-1	0.46E-1	0.48 E-1	0.45 E-1	0.47E-1
Er	Max.	0.54	0.50	0.58	0.44	0.35	0.26	0.29	0.28	0.24	0.23
	Min.	0.35	0.40	0.34	0.27	0.20	0.16	0.14	0.15	0.15	0.16
	Avg.	0.44	0.42	0.41	0.37	0.26	0.20	0.21	0.22	0.19	0.18
	Std.	0.18	0.19	0.16	0.12	0.08	0.07	0.07	0.08	0.06	0.06
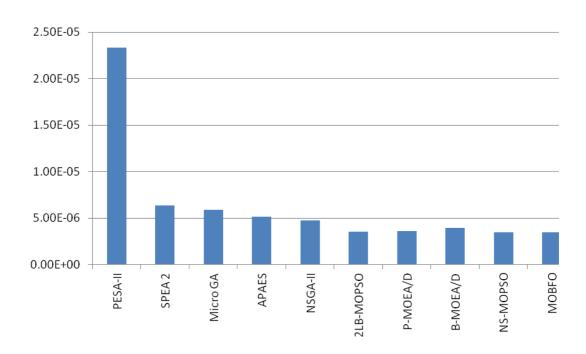
Fig.3.7 Average value of S metric for MOEAs algorithms
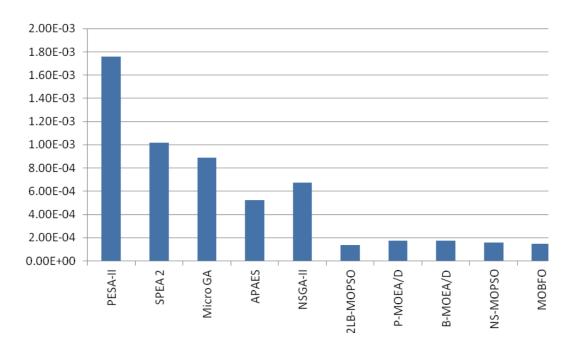
Fig.3.8 Average value of GD metric for MOEAs algorithms
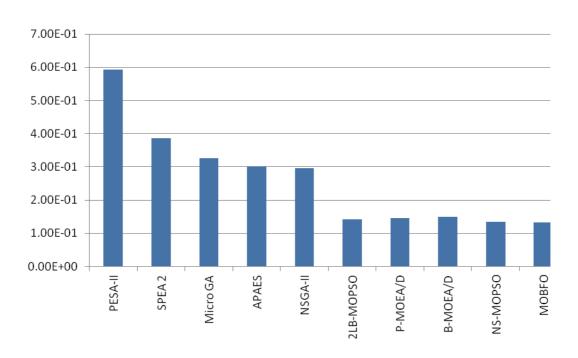
Fig.3.9 Average value of *IGD* metric for MOEAs algorithms
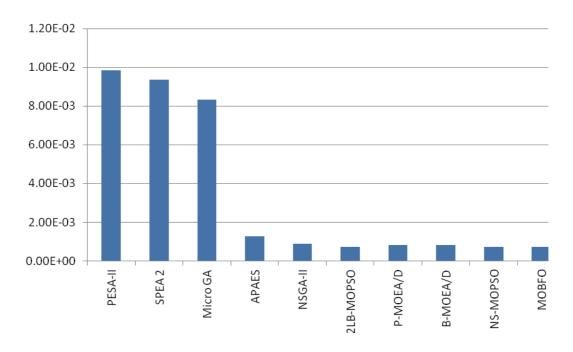
Fig.3.10 Average value of Δ metric for MOEAs algorithms
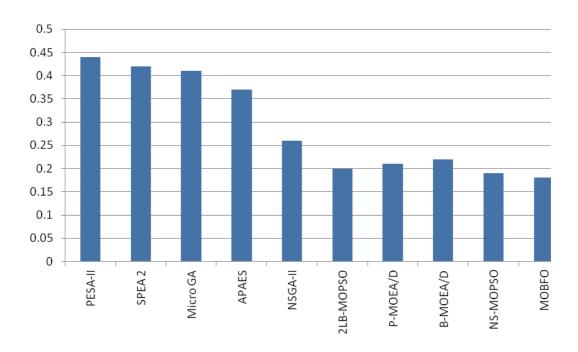
Fig.3.11 Average value of *ER* metric for MOEAs algorithms

The convergence (C) metrics for all the ten MOEAs are listed in Table-3.5. It clearly shows that most of the solutions obtained by NS-MOPSO and MOBFO dominate the solutions obtained by other MOEAs. The results of 2LB-MOPSO, P-MOEA/D and B-MOEA/D are almost comparable with each other.

The computational time is also evaluated for each algorithm based on the same hardware platform. The CPU times for Nikkei 225 data set of all algorithms are shown in Table-3.6 which indicates the decomposition based MOEAs (MOPEA/D) such as P-MOEA/D and B-MOEA/D are comparable with each other and take much less time as compared to others. Among all the algorithms the SPEA 2 takes maximum time. The execution times of these algorithms are also calculated for other stock indices and are shown in Table-3.7.

Table.3.5 Comparison of (C) metrics obtained using different MOEAs

	PESA-II	SPEA 2	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO	MOBFO
PESA-II	—	0.3810	0.3620	0.2600	0.2230	0.2111	0.2010	0.1905	0.1900	0.1880
SPEA-II	0.6280	—	0.4200	0.3400	0.3280	0.2821	0.2651	0.2621	0.2620	0.2480
Micro-GA	0.6400	0.4400	-----	0.3610	0.3422	0.3012	0.2910	0.2821	0.2410	0.2300
APAES	0.6988	0.6377	0.6100	-----	0.3888	0.3421	0.3328	0.3220	0.3164	0.3122
NSGA-II	0.8530	0.7620	0.7399	0.4600	—	0.3528	0.3432	0.3411	0.3400	0.3230
2LB-MOPSO	0.8721	0.8432	0.8211	0.7021	0.5155	-----	0.3533	0.3411	0.3213	0.2811
P-MOEA/D	0.8810	0.8532	0.8221	0.7322	0.5411	0.4321	----	0.4231	0.3544	0.3012
B-MOEA/D	0.8932	0.8621	0.8302	0.7412	0.5722	0.4822	0.3988	------	0.3744	0.3211
NS-MOPSO	0.9090	0.8920	0.8600	0.7900	0.6800	0.4962	0.4522	0.4412	-----	0.3522
MOBFO	0.9166	0.9012	0.8700	0.8012	0.7243	0.5101	0.4866	0.4711	0.3900	—

Table.3.6.Comparison of CPU time required among MOEAs for Nikkie-225

Algorithms	PESA-II	SPEA 2	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO	MOBFO
CPU Time inseconds	4820	4960	4825	4905	4760	4720	3100	3050	4700	4650

Table-3.7.Comparison of CPU time in seconds among different markets using MOEAs

Algorithms		PESA-II	SPEA-II	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO	MOBFO
CPU Time	Hang-Seng	685	708	689	700	675	673	443	436	671	664
	DAX-100	1606	1653	1608	1608	1586	1570	1033	1016	1566	1550
	FTSE-100	1621	1669	1623	1647	1601	1585	1048	1031	1582	1565
	S&P-100	1641	1680	1644	1668	1617	1600	1070	1052	1602	1586

The nonparametric statistical test such as Sign test and Wilcoxon signed rank test are carried out for pairwise comparisons of MOBFO algorithms with other MOEAs. The critical number of wins needed to achieve both $\alpha = 0.05$ and $\alpha = 0.1$ levels of significance is shown in Table-3.8. An algorithm is significantly better than other if its performance is better on at least the cases presented in each row.

Table-3.8. Critical values for the two-tailed signtest at $\alpha = 0.05$ and $\alpha = 0.1$.

Cases	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$\alpha = 0.05$	5	6	7	7	8	9	9	10	10	11	12	12	13	13	14	15	15	16	17	18	18
$\alpha = 0.01$	5	6	6	7	7	8	9	9	10	10	11	12	12	13	13	14	14	15	16	16	17

The results of the Sign test for pairwise comparisons among proposed MOBFO and other algorithms while taking the S metric as the wining parameter (i.e. lower value of S means win) are shown in Table-3.9. From the results it is clear that the MOBFO shows significant improvement over PESA-II, SPEA-II, Micro-GA, APAES, and NSGA-II algorithm with a level of significance $\alpha = 0.05$ and over NSGA-II, with a level of significance $\alpha = 0.1$. Similarly for Δ metric the result of Sign test is shown in Table-3.10.

This test can be conducted using other metrics as winning parameters. The Wilcoxon signed rank test is carried out by calculating R^+ and R^- and then using well-known statistical software package SPSS. Table-3.11 shows the R^+, R^-, z, Asymp. sig (2-tailed), Exact sig. (2-tailed), Exact sig. (1-tailed) and point of probability computed for all the pairwise comparisons with MOBFO considering S metric as winning parameter and applying to Nikkie 225 market indices. The result of the Wilcoxon signed rank test for another metric Δ is shown in Table-3.12. Win

Table-3.9. Critical values for the two-tailed Sign test at $\alpha = 0.05$ and $\alpha = 0.1$ using S metric as winning parameter.

MOBFO	PESA-II	SPEA 2	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO
Wins(+)	22	21	19	18	17	16	15	14	13
Losses(-)	3	4	6	7	8	9	10	11	12
Detected differences	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.01$	—	—	—	—

Table-3.10.Critical values for the two-tailed Sign test at $\alpha = 0.05$ and $\alpha = 0.1$ using Δ metric as winning parameter

MOBFO	PESA-II	SPEA 2	Micro-GA	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	NS-MOPSO
Wins(+)	22	21	20	19	18	17	16	14	13
Losses(-)	3	4	5	6	7	8	9	11	12
Detected differences	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.05$	$\alpha = 0.01$	—	—	—

Table-3.11.Wilcoxon Signed test using S metric as winning parameter and applying different MOEAs to Nikkie 225 market indices

Comparison	R^+	R^-	z	Asymp.sig(2-tailed),	Exact sig. (2-tailed),	Exact sig. (1-tailed)	pointof probability
MOBFO with PESA-II	252	73	-2.410	0.016	0.014	0.007	0.000
MOBFO with SPEA-II	231	94	-1.845	0.065	0.065	0.033	0.001
MOBFO with Micro-GA	222	103	-1.602	0.109	0.112	0.056	0.002
MOBFO with APAES	217	108	-1.468	0.142	0.146	0.073	0.002
MOBFO with NSGA-II	211.5	113.5	-1.319	0.187	0.193	0.096	0.002
MOBFO with 2LB-MOPSO	208	117	-1.225	0.220	0.227	0.114	0.003
MOBFO with P-MOEA/D	186	139	-0.633	0.527	0.538	0.269	0.004
MOBFO with B-MOEA/D	160	165	-0.067	0.946	0.953	0.476	0.005
MOBFO with MOPSO	168	152	-0.148	0.882	0.810	0.445	0.005

Table-3.12. Wilcoxon signed test using Δ metric as winning parameter and

applying different MOEAs to Nikkie 225 market indices

Comparison	R^+	R^-	z	Asymp.sig(2-tailed),	Exact sig. (2-tailed),	Exact sig. (1-tailed)	pointof probability
MOBFO with PESA-II	274.50	50.50	-3.018	0.003	0.006	0.005	0.000
MOBFO with SPEA-II	246.00	79.00	-2.250	0.024	0.023	0.012	0.000
MOBFO with Micro-GA	234.00	91.00	-1.926	0.054	0.054	0.027	0.001
MOBFO with APAES	214	111	-1.387	0.165	0.170	0.085	0.002
MOBFO with NSGA-II	196	129	-0.902	0.367	0.377	0.188	0.004
MOBFO with 2LB-MOPSO	194	131	-0.848	0.396	0.407	0.203	0.004
MOBFO with P-MOEA/D	191.5	133.5	-0.781	0.435	0.445	0.223	0.004
MOBFO with B-MOEA/D	185	140	-0.542	0.632	0.642	0.342	0.005
MOBFO with NS-MOPSO	186.5	168.5	-0.162	0.872	0.879	0.440	0.005

Case 2: Cardinality with budget constraint

The effect of cardinality constraints K is studied in this section. The Pareto fronts

obtained by applying MOBFO for Nikkei 225 data set having different cardinalities are

presented in Fig.3.12. K is set at 20 and is increased to 180 at a step of 20. The portfolio

manager has the option to make a trade-off between risk and returns for different values of

K. The maximum, minimum, average and standard deviation values of various performance

metrics are shown in Table-3.13 It is observed that when K increases these metrics values also increase. Table-3.14 lists the results of convergence (C) metric. It shows that the final solutions obtained at $K = 20$ dominate the solutions obtained at $K = 180$. The CPU time for various values of K are shown in Table-3.15. It reveals that the computation time increases with an increase in the value of K. From the Fig.3.12, it is clear that Pareto fronts become shorter with increase in K values. Hence the proposed algorithm is able to obtain a near optimal solution efficiently by investing lower number of assets i.e. approximately 10 percent of available assets. The Pareto front of MOBFO is also calculated for other stock indices for different K and are depicted in Figs.3.13 -3.16.
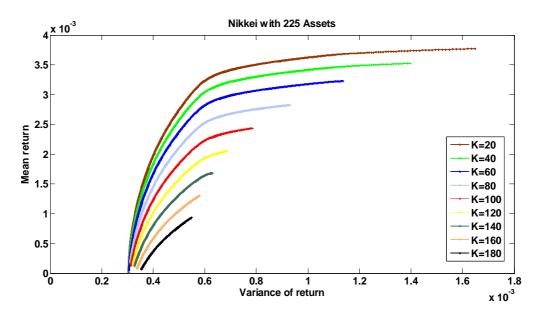
Fig.3.12. MOBFO efficient frontier for different cardinality for Nikkei 225 data
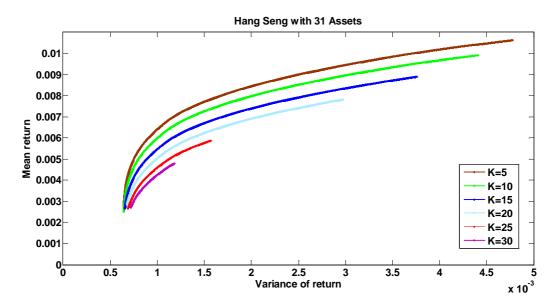
Fig.3.13.MOBFO efficient frontier for different cardinality for Hang-Sang data

Fig.3.14.MOBFO efficient frontier for different cardinality for DAX 100 data
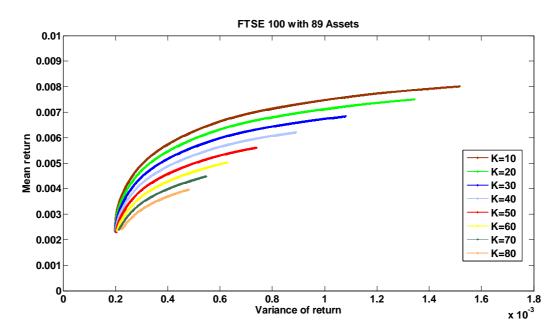
Fig.3.15.MOBFOefficient frontier for different cardinality for FTSE 100 data
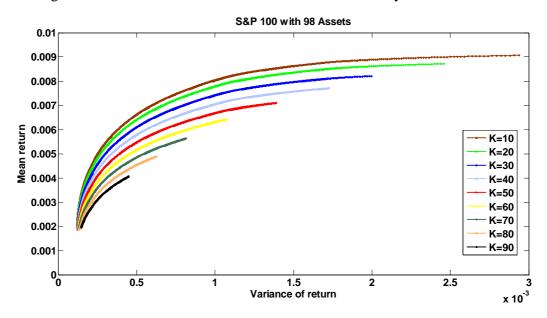
Fig.3.16.MOBFO efficient frontier for different cardinality for S & P 100 data

The Pareto front of the NS-MOPSO, P-MOEA/D, B-MOEA/D algorithms for different market having a different cardinality constraint for Nikkei 225 data set are shown in Figs.3.17 - 3.19.
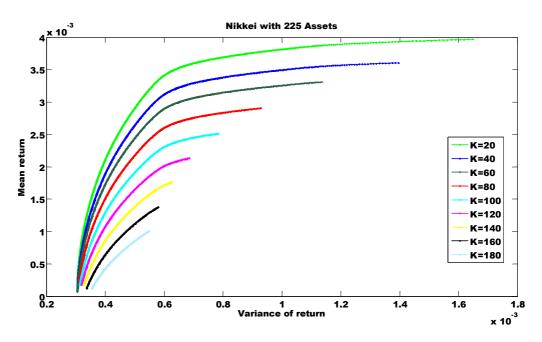
Fig.3.17. NS-MOPSO efficient frontier for different cardinality for Nikkei 225 data

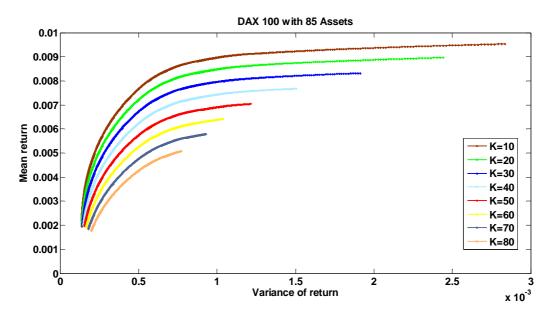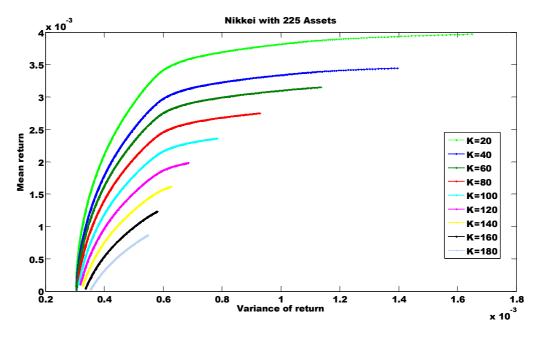Fig.3.18. P-MOEA/D efficient frontier for different cardinality for Nikkei 225 data
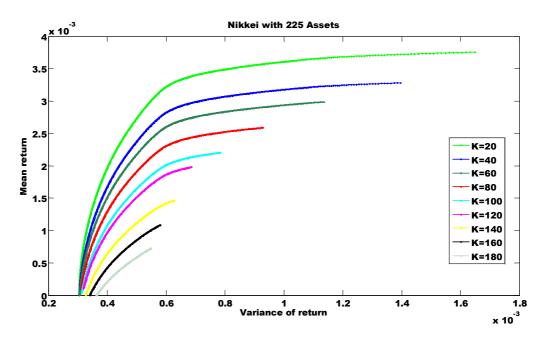
Fig.3.19. B-MOEA/D efficient frontier for different cardinality for Nikkei 225 data

Table-3.13.Comparison of results of performance evaluation metrics for different cardinality constraints.

Cardinality Constraint			$K=0$	$K=20$	$K=40$	$K=60$	$K=80$	$K=100$	$K=120$	$K=140$	$K=160$	$K=180$
Matric Values	S	Max.	5.21 E-6	7.21 E-6	9.21 E-5	5.32 E-5	6.86E-5	9.21 E-5	1.94 E-4	5.45 E-4	1.11 E-3	2.21 E-3
		Min.	2.32 E-6	3.45 E-6	5.32 E-6	3.45 E-5	5.32 E-5	7.65 E-5	0.98 E-4	3.42 E-4	0.88 E-3	1.67 E-3
		Avg.	3.43E-6	5.64E-6	7.77E-6	4.43E-5	6.45E-5	8.88E-5	1.21E-4	4.24 E-4	1.01 E-3	1.90E-3
		Std.	0.85E-6	1.41E-6	2.12E-6	1.15E-5	1.98E-5	2.52E-5	0.25 E-4	1.6 E-4	0.25 E-3	0.47 E-3
	GD	Max.	2.16 E-4	3.90 E-4	4.2 E-4	5.6 E-4	6.7 E-4	7.7 E-4	9.10 E-4	1.92 E-3	5.24 E-3	6.28 E-3
		Min.	1.10E-4	2.6 E-4	3.5 -4	4.1 E-4	5.7 E-4	6.2 E-4	8.21 E-4	1.08 E-3	4.51 E-3	5.42 E-3
		Avg.	1.45E-4	2.20 E-4	3.9 E-4	4.6 E-4	6.3 E-4	6.8 E-4	8.9 E-4	1.62 E-3	5.01 E-3	5.91E-3
		Std.	0.36 E-4	0.49 E-4	0.91 E-4	1.2 E-4	1.7 E-4	1.9 E-4	2.4 E-4	0.42E-3	1.01 E-3	1.47 E-3
	\` IGD	Max.	8.23 E-4	9.11 E-4	1.2 E-3	2.31 E-3	2.98 E-3	3.42 E-3	4.71 E-3	5.83 E-3	7.19 E-3	8.12 E-3
		Min	6.02 E-4	7.52 E-4	0.8 E-3	1.79 E-3	2.09 E-3	2.92 E-3	3.95 E-3	4.28 E-3	6.51 E-3	7.03 E-3
		Avg.	7.05 E-4	8.41 E-4	1.01 E-3	2.01 E-3	2.71 E-3	3.02 E-3	4.05 E-3	4.50 E-3	6.99 E-3	7.49 E-3
		Std.	1.76 E-4	2.11 E-4	0.25 E-3	0.51 E-3	0.61 E-3	0.75 E-3	1.01 E-3	1.20 E-3	1.77 E-3	1.82 E-3
	Δ	Max.	2.45 E-1	3.12 E-1	3.91 E-1	4.21 E-1	4.98 E-1	5.42 E-1	5.99 E-1	6.51 E-1	7.51 E-1	8.61E-1
		Min.	0.99E-1	2.50 E-1	3.01 E-1	3.87 E-1	3.92 E-1	3.98 E-1	5.01 E-1	6.01 E-1	7.01 E-1	8.12 E-1
		Avg.	1.33 E-1	1.84 E-1	2.29 E-1	3.45 E-1	3.75 E-1	4.25 E-1	4.55 E-1	5.61 E-1	6.62 E-1	7.71 E-1
		Std.	0.47E-1	0.46 E-1	0.57 E-1	0.81 E-1	1.32 E-1	1.61 E-1	1.30 E-1	1.40 E-1	1.71E-1	1.91 E-1
	Er	Max.	0.23	0.29	0.35	0.45	0.51	0.55	0.61	0.67	0.71	0.75
		Min	0.16	0.22	0.28	0.38	0.42	0.47	0.52	0.58	0.60	0.62
		Avg.	0.18	0.24	0.30	0.39	0.41	0.49	0.54	0.61	0.64	0.68
		Std.	0.06	0.08	0.09	0.11	0.12	0.14	0.15	0.17	0.19	0.21

Table-3.14. Comparison of results of convergence metric (C) for budget and cardinality constraints for Nikkei 225 Stock using MOBFO

Cardinality Constraint	$K=20$	$K=40$	$K=60$	$K=80$	$K=100$	$K=120$	$K=140$	$K=160$	$K=180$
$K=20$	—	0.2610	0.3400	0.4660	0.5970	0.6580	0.7200	0.7810	0.8600
$K=40$	0.0890	—	0.3020	0.4220	0.5690	0.6260	0.7060	0.7670	0.8420
$K=60$	0.0840	0.2420	—	0.3840	0.5322	0.5860	0.6810	0.7420	0.8280
$K=80$	0.0810	0.2250	0.2620	—	0.5020	0.5590	0.6640	0.7220	0.8020
$K=100$	0.0770	0.2040	0.2420	0.3680	—	0.5220	0.6430	0.7040	0.7840
$K=120$	0.0740	0.1880	0.2240	0.3440	0.4740	—	0.6210	0.6810	0.7600
$K=140$	0.0710	0.1560	0.1990	0.3260	0.4420	0.4920	—	0.6620	0.7380
$K=160$	0.0670	0.1250	0.1640	0.3010	0.4170	0.4480	0.5920	—	0.7040
$K=180$	0.0590	0.1080	0.1280	0.2790	0.03820	0.4200	0.5680	0.6390	—

Table.3.15.Comparison of mean value of CPU time in seconds for MOBFO

Number of Cardinality	$K=20$	$K=40$	$K=60$	$K=80$	$K=100$	$K=120$	$K=140$	$K=160$	$K=180$
CPU Time in second	4910	5320	5730	6190	6680	7020	7490	7830	8440

Case3: Budget, floor, ceiling and cardinality constraint

The effect of combined presence of all the constraints is examined in this section. The cardinality constraint is taken as $K=10$, the floor constrain has been set at $a_i = 0.01$

and the ceiling constraint is fixed at $b_i = 1$ with all available money has to be invested (budget constraint)**.** The performance of all MOEAs has been compared with the results obtained using single objective GA, TS, SA and PSO as given in [3.6] by evaluating three error measures such as Euclidian distance, variance of return error and mean return error. The experimental results of Table-3.16 demonstrate that the proposed NS-MOPSO algorithm outperforms all single and multiobjective algorithms for stock with higher number of assets i.e. Nikkei 225 with 225 assets. MOBFO gives quite better performance for the stock indices such as Hang-Sang, DAX 100, FTSE 100 and S&P 100 which are having lesser number of assets than Nikkei 225. Experimental results show that the performance of 2LB-MOPSO, P-MOEA/D and B-MOEA/D algorithms are almost comparable to each other.

Case 4: Variable cardinality with budget, floor, ceiling and cardinality constraints

Let us assume the portfolio is having the minimum buy in threshold and maximum limit constraint within the range {1 % to 10%}. The different ranges of cardinality constraint i.e. {10 to 15}, {15 to 20} and {20 to 25} are taken. The Pareto fronts obtained by the MOBFO algorithm for these conditions are shown from Figs.3.20 -3.22.

Table-3.16. Experimental results for three error measure of all algorithms to five markets

Index	Assets	Error	GA	TS	SA	PSO	PESA-II	SPEA2	APAES	NSGA-II	2LB-MOPSO	P-MOEA/D	B-MOEA/D	MOPSO	MOBFO
Hang Seng	31	Mean Euclidian distance	0.0040	0.0040	0.0040	0.0049	0.0044	0.0042	00041	0.0041	0.0040	0.0040	0.004	0.0040	0.0040
		Variance of return error	1.6441	1.6578	1.6628	2.2421	1.5233	1.4877	1.3912	1.3266	1.2981	1.2965	1.2961	1.2840	1.2712
		Mean return error(%)	0.6072	0.6107	0.6238	0.7427	0.7620	0.6899	0.6652	0.6472	0.6182	0.6212	0.6121	0.6021	0.6015
DAX 100	85	Mean Euclidian distance	0.0076	0.0082	0.0078	0.0090	0.0098	0.0084	0.0082	0.0077	0.0075	0.0077	0.0076	0.0075	0.0074
		Variance of return error	7.2180	9.0390	8.5485	6.8588	9.2819	8.2432	7.5422	7.1211	6.7562	6.8271	6.7723	6.7543	6.7421
		Mean return error(%)	1.2791	1.9078	1.2817	1.5885	2.2212	1.5922	1.4352	1.2634	1.2532	1.2691	1.2681	1.2671	1.2511
FTSE 100	89	Mean Euclidian distance	0.0020	0.0021	0.0021	0.0022	0.0024	0.0022	0.0022	0.0021	0.0019	0.0021	0.0022	0.0019	0.0018
		Variance of return error	2.8660	4.0123	3.8205	3.0596	5.2381	3.7652	3.2311	2.9871	2.8114	2.9122	2.8813	2.8120	2.7911
		Mean return error(%)	0.3277	0.3298	0.3304	0.3640	0.4023	0.3652	0.3522	0.3329	0.3248	0.3271	0.3259	0.3250	0.3211
S&P 100	98	Mean Euclidian distance	0.0041	0.0041	0.0041	0.0052	0.0056	0.0049	0.0047	0.0042	0.0040	0.0041	0.0041	0.0040	0.0039
		Variance of return error	3.4802	5.7139	5.4247	3.9136	7.0122	5.4323	4.5362	3.7629	3.4635	3.4773	3.4771	3.4763	3.4751
		Mean return error(%)	1.2258	0.7125	0.8416	1.4040	2.4232	1.2109	0.9812	0.7321	0.7001	0.7032	0.7028	0.7021	0.7020
Nikkei	225	Mean Euclidian distance	0.0093	0.0010	0.0010	0.0019	0.0101	0.0032	0.0017	0.0010	0.0008	0.0009	0.0008	0.0008	0.0007
		Variance of return error	1.2056	1.2431	1.2017	2.4274	3.0986	2.0421	1.9811	1.1232	0.9866	0.9888	0.9880	0.9876	0.9872
		Mean return error(%)	5.3266	0.4270	0.4126	0.7997	1.2314	0.8654	0.6754	0.4325	0.3267	0.3252	0.3249	0.3244	0.3211
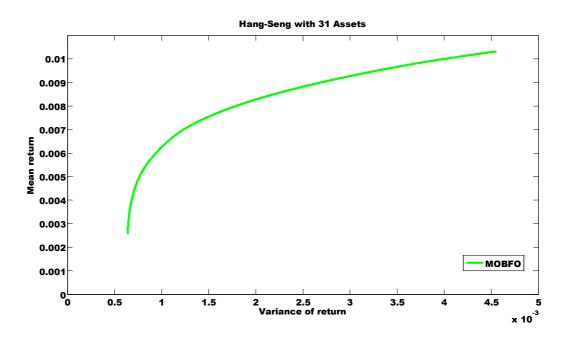
Fig.3.20. Pareto front obtained MOBFO for floor constraint (1%) and ceiling constraint

(10%) and cardinality {10 to 15} to Hang Sang data
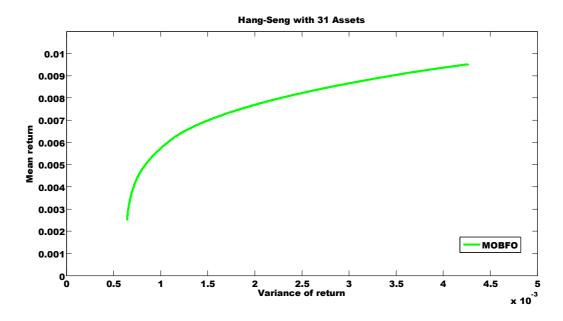
Fig.3.21. Pareto front obtained by MOBFO for floor constraint (1%) and ceiling constraint

(10%) and cardinality {15 to 20} to Hang Sang data

Hang-Seng with 31 Assets
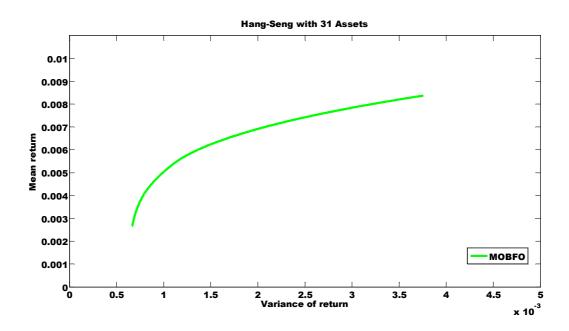
Fig.3.22. Pareto front obtained by MOBFO for floor constraint (1%) and ceiling constraint (10%) and cardinality {20 to 25} to Hang Sang data

The Pareto fronts obtained by applying the ten MOEAs algorithms in the case having a cardinality range {10 to 15} are shown in Fig.3.23


Hang-Seng with 31 Assets
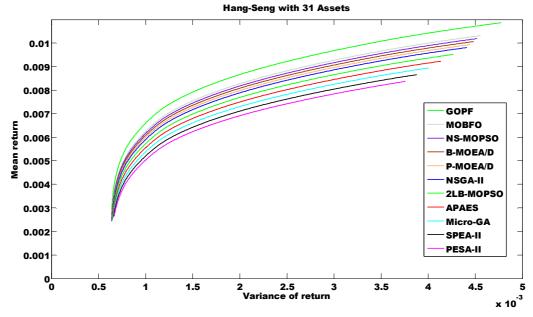
Fig.3.23.Pareto front obtained by ten MOEAs for floor constraint (1%) and ceiling constraint (10%) and cardinality {10 to 15} to Hang Sang data

3.4 Conclusion and further work

The effects of four different practical constraints such as budget, floor, ceiling and cardinality constraints on portfolio have been analyzed by examining the resultant Pareto front achieved.

Two novel multiobjective algorithms based on non-dominated sorting and two algorithms based on decomposition based framework have been suitably applied to realistic portfolio optimization problems with budget, floor, ceiling and cardinality constraints by formulating it as a multiobjective optimization problem. The performances of the proposed approaches are evaluated by comparing with four single objective evolutionary algorithms and a set of competitive MOEAs. The comparisons include the evaluation of three error measures, six performance metrics, Pareto optimality and computational complexity. By examining different values of performance metrics obtained it is concluded that the Pareto solutions obtained by different approaches are comparable with each other. Experimental results reveal that the proposed algorithms are able to adequately handle budget, floor, ceiling and cardinality constraint simultaneously. From the simulation results it is clear that the investor does not have to invest money on all available assets rather to invest in fewer assets i.e. approximately 10 percent of available assets, to explore wide risk-return area. The portfolio manager has the option to make a tradeoff between risk and return for different cardinality constraints to decide the portfolio according to the requirement. In particular, the MOBFO algorithm gives best Pareto solutions maintaining adequate diversity.

The statistical analysis such as Sign test and Wilcoxon signed rank test are also performed for pairwise comparison of MOBFO with other algorithms. The simulation

results demonstrate significant improvement of MOBFO over PESA-II, SPEA 2, Micro-GA, APAES and NSGA-II algorithm with a level of significance $\alpha = 0.05$ and over 2LB-MOPSO, with a level of significance $\alpha = 0.1$.

Future research work on the topic includes incorporation of advanced local search operators into the proposed algorithm which is expected to allow better exploration and exploitation of the search space. To assess the strengths and weaknesses of non-dominated sorting based or decomposition based MOEAs frameworks further investigation is needed. The performance of proposed method can also be evaluated considering other real world constraints like round-lot, turnover and trading. The same multiobjective optimization algorithm can also be applied to other financial applications such as asset allocation, risk management and option pricing.

References

[3.1] H .M. Markowitz."Portfolio Selection", Journal of Finance, vol.7, pp.77-91,1952.

[3.2] H.M. Markowitz."Portfolio Selection: efficient diversification of investments". New York: Yale University Press. John Wiley & Sons, 1991.

[3.3] D. Bienstock, "Computational study of a family of mixed-integer quadratic programming problems". Mathematical Programming, vol. 74, pp.121–140, 1996.

[3.4] D.X. Shaw, S. Liu, L. Kopman, "Lagrangian relaxation procedure for cardinality-constrained portfolio optimization". Optimization Methods & Software,vol. 23,pp- 411–420, 2008.

[3.5] J.P.Vielma, S.Ahmed, G.L.Nemhauser, "A lifted linear programming branchand-bound algorithm for mixed-integer conic quadratic programs". INFORMS Journal on Computing, vol.20, pp.438–450, 2008.

[3.6] D. Bertsimas, R. Shioda, "Algorithm for cardinality-constrained quadratic optimization". Computational Optimization and Applications, vol.43, pp.1–22, 2009.

[3.7] N. Gulpinar, L.T.H An, M. Moeini, "Robust investment strategies with discrete asset choice constraints using DC programming". Optimization,vol.59, pp.45–62, 2010.

[3.8] Li, D., Sun, X., Wang, J., "Optimal lot solution to cardinality constrained mean-variance formulation for portfolio selection". Mathematical Finance, vol.16, pp.83–101, 2006.

[3.9] T. J. Chang, N. Meade, J. E. Beasley, Y. M. Sharaiha. "Heuristics for cardinality constrained portfolio optimization". Computers & Operations Research, vol.27.no.13, pp. 1271–1302, 2000.

[3.10] Fernandez, A., Gomez, S, "Portfolio selection using neural networks". Computers & Operations Research, vol.34, pp.1177–1191, 2007.

[3.11] Pai, G.A.V., Michel, T., 2009. "Evolutionary optimization of constrained k-means clustered assets for diversification in small portfolios", IEEE Transactions on Evolutionary Computation vol.13, pp.1030–1053, 2009.

[3.12] Crama, Y., Schyns, M., "Simulated annealing for complex portfolio selection problems", European Journal of Operational Research, vol.150, pp-546–571, 2003.

[3.13] Derigs, U., Nickel, N.-H., "On a local-search heuristic for a class of tracking error minimization problems in portfolio management", Annals of Operation Research, vol.131, pp. 45–77, 2004.

[3.14] Tunchan Cura. "Particle swarm optimization approach to portfolio optimization", Nonlinear Analysis: Real World Applications,vol.10.no.4, pp.2396–2406, 2009.

[3.15] Soleimani, H., Golmakani, H.R., Salimi, M.H., "Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm". Expert Systems with Applications vol.36, pp.5058–5063,2009.

[3.16] Chang.T.-J., Yang S.C, Chang. K.-J, "Portfolio optimization problems in different risk measures using genetic algorithm". Expert Systems with Applications. vol.36, pp.10529–10537, 2009.

[3.17] A. Mukherjee, R. Biswas, K. Deb and A. P. Mathur, "Multi-objective Evolutionary algorithms for the Risk-return trade-off in bank loan management", International transaction in operations research, vol.9, 583-597,2002.

[3.18] K. Doerner, W. Gutjahr, R. Hartl, C. Strauss, C. Stummer. "Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection". Annals of Operations Research, vol.131.pp.79–99,2004.

[3.19] S. K. Mishra, G. Panda and S. Meher. "Multiobjective Particle Swarm Optimization Approach to Portfolio Optimization", IEEE World Congress on Nature and Biologically Inspired Computing (NaBIC), Coimbatore, India, 2009,pp.1612-1615.

[3.20] S. K. Mishra, G. Panda, S. Meher, R. Majhi. "Comparative Performance Evaluation of Multiobjective Optimization Algorithm for Portfolio Management", International Journal of Computational Vision and Robotics, Inderscience publisher, vol.1.no.2,pp.236-247,2010.

[3.21] H. Keller and D. Maringer. "Optimization of cardinality constrained portfolios with a hybrid local search algorithm", OR Spectrum,vol.25,no.4,pp.481-495,2004.

[3.22] R. Armananzas and J.A. Lozano. "A multiobjective approach to the portfolio optimization problem", IEEE World Congress on Evolutionary Computation,vol.2.pp.1388-1395,2005.

[3.23] S. C. Chiam, A.Al. Mamun and Y. L. Low, "A realistic approach to evolutionary multiobjective portfolio optimization", IEEE Congress on Evolutionary Computation (CEC) 2007, pp.204-211.

[3.24] K. P. Anagnostopoulos, G. Mamanis. 2010, "A portfolio optimization model with three objectives and discrete variables". Computers & Operations Research vol.37, pp.1285–1297, 2010.

[3.25] K.P. Anagnostopoulos, G. Mamanis, "The mean–variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective

evolutionary algorithms", Expert Systems with Applications, vol. 38, issue 11, pp. 14208–14217, October 2011

[3.26] J. Branke, B.Scheckenbach, M. Stein, K. Deb, H. Schmeck, " Portfolio optimization with an envelope-based multi-objective evolutionary algorithm". European Journal of Operational Research, vol.199, issue 3, pp.684–693, December 2009.

[3.27] David W. Corne, Nick R. Jerram, Joshua D. Knowles, Martin J. Oates, J. Martin, "PESA-II: region-based selection in evolutionary multiobjective optimization". In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO) 2001, pp. 283-290.

[3.28] E. Zitzler, M. Laumanns and L. Thiele. "SPEA2: Improving the strength Pareto evolutionary algorithm". Technical Report 103, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, May 2001.

[3.29] Coello C A C, Toscano G, "A micro-genetic algorithm for multiobjective optimization", Lecture Notes in Computer science. 1993.

[3.30] M Oltean, C Grosan, A Abraham, and A Koppen, "Multiobjective optimization using adaptive pareto archive evolution strategy", 5th Int. Conf. on Intelligent Systems Design and Applications, Piscataway, NJ: IEEE Press, 8-10 Sept. 2005, pp.558-563.

[3.31]] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II". IEEE Transactions on Evolutionary Computing, vol. 6, no.2, pp. 182–197, 2002.

[3.32] S. Z. Zhao and P. N. Suganthan, "Two-lbests Based Multi-objective Particle Swarm Optimizer", Engineering Optimization, vol.43.no.1, pp.1-17, 2011.

[3.33] Q. Zhang and H. Li, "MOEA/D: A Multi-objective Evolutionary Algorithm Based on Decomposition, IEEE Trans. on Evolutionary Computation", vol.11.no.6, pp.712-731, 2007.

[3.34] http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html

[3.35] Streichert, F., Ulmer, H., & Zell, A. (2004b). "Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem", Proceedings of congress on evolutionary computation (vol. 1, pp. 932–939). Portland, USA: IEEE Press.

[3.36] A. E. Eiben, S. K. Smit, "Parameter tuning for configuring and analyzing evolutionary algorithms", Swarm and Evolutionary Computation, vol.1.no.1, pp.19-31, 2011.

[3.37] J. Derrac, S. García, D. Molina, F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms", Swarm and Evolutionary Computation, vol.1 no.1, pp.3-18, 2011.

Chapter 4

Prediction based mean-variance Model for Multiobjective Portfolio Optimization

Chapter

4

In this chapter a novel prediction based mean-variance (PBMV) model has been proposed to solve constrained portfolio optimization. In this model, the expected return and risk are predicted using a low complexity functional link artificial neural network (FLANN) structure. Four swarm intelligence based MOEAs using PBMV model have been applied to solve the portfolio optimization problem considering various constraints. The performance of MOEAs obtained using the proposed model is compared with that obtained using Markowitz mean-variance model. The performance is based on six performance metrics as well as Pareto front. In addition to this, in the present study the nonparametric statistical analysis using Sign test and Wilcoxon signed rank test are also carried out to compare the performance of algorithms pairwise. From the simulation results it is observed that the proposed PBMV model approach is capable of identifying good Pareto solutions maintaining adequate diversity and is comparable with the Markowitz model. The predicted value of risk and return are subsequently used by four MOEAs to achieve the Pareto solution.

4.1. Introduction

The mean-variance model, proposed by Harry Markowitz [4.1], is a landmark in Modern Portfolio Theory (MPT). In the past few decades, this model has extensively been studied in the field of portfolio optimization. Recently, several authors have tried to improve this model by applying some model simplification techniques or by proposing models having different risk measure such as semi-variance, mean absolute deviation and variance with skewness model [4.2]. The fundamental assumptions of these models have been described in [4.3]. It has been observed that in most of the models, the expected return of portfolios is given by the linear combination of the participations (weighting) of the stocks in the portfolio and its expected returns (the mean returns). The portfolio risk measure of these models varies from Markowitz mean-variance model but is based on the moments about the mean of the linear combination of the participations and time series of returns of its stocks.

The fundamental assumptions of these models include (i) the time series of returns of each stock follows a normal distribution (ii) mean of past stock's return is taken as expected future return (iii) variance taken as a measure of the stock's risk and (iv) the covariance of each pair of time series is considered as a measure of joint risk of each pair of stocks. But the fundamental assumptions of the above models have been threatened by real world data because of the following reasons. These are (i) distributions of the series of returns often depart from normality which exhibits kurtosis and skewness [4.4], [4.5] and make the variance of the returns an inappropriate measure of risk [4.6] (ii) use of mean of past stock's returns imposes a low pass filtering effect on the dynamic behaviour of the stock markets [4.7].

Hence the development of a model, free from those shortcomings is a still challenging field of research. There is need to develop an efficient model which would directly predict the expected return. Accurate prediction of future data/information such as stock parameter (return) is a promising and interesting field of research and has lot of importance for commercial applications. However the prediction of stock return is not an easy task, because the stock market indices are essentially dynamic, non-linear, complicated, nonparametric, and chaotic in nature [4.8]. The time series of stock parameters are also noisy and random [4.9]-[4.10]. In addition, stock market's movements are affected by many macro-economical factors [4.11] such as political events, firms' policies, general economic conditions, investors' expectations, institutional investors' choices, movement of other stock market, psychology of investors, etc.

A good number of research papers have been reported in the field of stock market prediction. Researchers have studied various macro-economic factors to discover the extent of correlation that may exist with the changes in the stock prices and have extracted the trends in the market using past stock prices and volume information. Technical analysts and researchers have believed that there are recurring patterns in the market behavior, which can be identified and predicted. In the last few decades, different adaptive models have been developed for forecasting financial parameters. These models can be broadly divided into statistical models and soft-computing models. One of the well known statistical methods used for this purpose is auto regressive integrated moving average (ARIMA) [4.12]. The recent advancement in the field of soft and evolutionary computing leads to a new dimension in the field of financial forecasting. Different soft computing approaches using variants of artificial neural networks (ANNs) have been introduced by many researchers in

this field. These include radial basis function (RBF) [4.13], recurrent neural network (RNN) [4.14], multilayer perceptron, multi branch neural networks (MBNN) [4.15] and local linear wavelet neural networks (LLWNN) [4.16]. These variants of ANN have gained popularity due to their inherent capabilities to approximate any nonlinear function to a high degree of accuracy, less sensitivity to error term assumptions and tolerance to noise, chaotic components etc.[4.17]. Most artificial neural network (ANN) based models use historical stock index data and technical indicators [4.12] to predict market data.

In most cases, it has been observed that the development and testing of the model involve large computational complexity as well as more prediction and testing time but lacks in prediction accuracy. Majhi *et al.* have proposed functional link ANN (FLANN) based model for prediction of exchange rates [4.19]. They have reported that their simple model provides improved performance compared to models proposed earlier. The same authors have also achieved improved performance of this model by considering various statistical parameters such as technical indicators based on historical data and fundamental economic factors [4.20]. The basic structure and training algorithm for FLANN have been dealt with, in Section 4.2. Recently two different adaptive algorithms such as PSO and clonal-PSO have been introduced to update the weights of the prediction model [4.21]. The prediction performance has been shown to be better than other methods.

In this chapter we have chosen the FLANN structure for prediction of return and is trained with evolutionary computing. The inputs to the network are some financial and economic variables such as moving average, mode and median of input parameters. The right combinations of these features are obtained by using evolutionary algorithms. The network parameters are also trained using evolutionary algorithms. The corresponding risk

of the predicted return is calculated. Considering these two conflicting objectives the Portfolio optimization problem can be formulated as a multiobjective optimization problem and is solved by using MOEAs algorithm.

4.2. Evolutionary functional link artificial neural network

The functional link ANN is a novel single layer neural network proposed by Pao [4.22]. The structure of the FLANN is very simple. It is a flat net with no hidden layer. Therefore, the computation is few and the learning algorithm used in this network is simple. The functional expansion of the input to the network effectively increases the dimensionality of the input vector and hence the hyper-planes generated by the FLANN provide greater discrimination capability in the input pattern space [4.23]. It is capable of forming arbitrarily complex decision regions by generating nonlinear decision boundaries [4.24]. Here, the input has been enhanced by using nonlinear function. This nonlinear functional expansion of the input pattern may be trigonometric, exponential, power series or Chebyshev type. A number of research papers on system identification and control of nonlinear systems, noise cancellation and channel equalization have been reported in recent times [4.25] using FLANN. These experiments have demonstrated that the FLANN has adequate potential to give satisfactory results to problems with highly non-linear and dynamic data. It has been shown that the FLANN can be conveniently used for functional approximation and pattern classification with faster convergence rate and lesser computational complexity than a multi layer perceptron (MLP) structure.

4.2.1 FLANN as a forecasting system.

The block diagram of a FLANN forecasting system is shown in Fig.4.1.
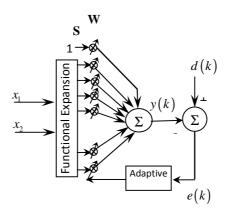
Fig. 4.1 Structure of the FLANN

Let **X** is the input vector of size $N \times 1$ which represents N numberof elements; the nth element is given by:

$$\mathbf{X}(n) = x_n, \quad 1 \le n \le N \tag{4.1}$$

Each element undergoes nonlinear expansion to form M elements such that the resultant matrix has the dimension of $N \times M$. This nonlinear expansion of each element may be trigonometric, exponential, power series or Chebyshev type. If the functional expansion of the element x_n is carried out using power series expansion it will be expressed as:

$$S_i = \begin{cases} x_n & for\ i = 1 \\ x_n^l & for\ i = 2,3,4,...M \end{cases} \tag{4.2}$$

where $l = 1, 2, 3.........., M$

For trigonometric expansion, the expanded elements are expressed as:

$$s_i = \begin{cases} x_n & for\ i = 1 \\ \sin(l\pi x_n) & for\ i = 2,4,....,M \\ \cos(l\pi x_n) & for\ i = 3,5,....,M+1 \end{cases} \tag{4.3}$$

118

where $l = 1, 2, \cdots, M/2$. The Chebyshev polynomials are the set of orthogonal polynomials defined as the solution to the Chebyshev differential equation. These higher Chebyshev polynomials may be generated using a recursive formula given as

$$T_{n+1} = 2xT_n(x) - T_{n-1}(x) \tag{4.4}$$

The first few Chebyshev polynomials are given by

$$T_o(x) = 1$$
$$T_1(x) = x$$
$$T_2(x) = 2x^2 - 1$$
$$T_3(x) = 4x^3 - 3x$$
$$T_4(x) = 8x^4 - 8x^2 + 1$$
$$T_5(x) = 16x^5 - 20x^3 + 5x \tag{4.5}$$

Each element undergoes nonlinear expansion to form M elements such that the resultant matrix has the dimension of $N \times M$. In matrix notation, the expanded elements of the input vector $\mathbf{E}$, is denoted by $\mathbf{S}$ of size $N \times (M+1)$. The bias input to the FLANN structure is *unity*. So an extra *unity* value is suitably added to the $\mathbf{S}$ matrix and the dimension of the $\mathbf{S}$ matrix becomes $N \times Q$, where $Q = (M + 2)$.

Let the weight vector be represented by W with Q elements given by $W = [w1 \ w2 \ w3 ... wq]$. The output $y(k)$ at instant k is given as

$$y(k) = \sum_{i=1}^{Q} s_i \, w_i \tag{4.6}$$

In matrix notation the output is obtained as

$$\mathbf{Y} = \mathbf{S} \cdot \mathbf{W}^T \tag{4.7}$$

The error term K^{th} instant is computed as

$$e(k) = d(k) - y(k) \tag{4.8}$$

where $d(k)$ is the predicted value.

4.2.2. Learning Algorithms of FLANN network

There are varieties of learning algorithms which are employed to train different adaptive models. The performance of these models depends on the rate of convergence, training time, computational complexity involved and minimum mean square error achieved after training. The learning algorithms may be broadly classified into two categories (a) derivative based and (b) derivative free. The derivative based algorithms are least mean squares (LMS), recursive least squares (RLS) and back propagation (BP). The derivative free algorithms are mainly based on evolutionary computation such as GA, PSO and BFO. In this section the details of these two categories of learning algorithms are outlined.

(a) Derivative based Algorithms

Referring to Fig. 4.1 in Section 4.2.1, the error signal $e(k)$ at k^{th} iteration can be computed as follows:

Let $\xi(k)$ denote the cost function at iteration k and is given by

$$\xi(k) = \frac{1}{2} \sum_{j=1}^{P} e_j^2(k) \tag{4.9}$$

where P is the number of nodes at the output layer.

The update equation for weight vector by applying least mean squares (LMS) algorithm [4.29] is given by

$$w(k+1) = w(k) - \frac{\mu}{2}\hat{\nabla}(k) \tag{4.10}$$

where $\hat{\nabla}(k)$ is an instantaneous estimate of the gradient of ξ with respect to the weight vector $w(k)$ and is computed as

$$\hat{\nabla}(k) = \frac{\partial \xi}{\partial w} = -2\,e(k)\frac{\partial y(k)}{\partial w} = -2\,e(k)\frac{\partial [w(k)s(k)]}{\partial w} = -2e(k)s(k) \tag{4.11}$$

Substituting the values of $\hat{\nabla}(k)$ in equation 4.10 we get

$$w(k+1) = w(k) + \mu\,e(k)s(k) \tag{4.12}$$

where μ denotes the step-size $(0 \le \mu \le 1)$, which controls the convergence speed of the LMS algorithm. It is called as learning rate of LMS algorithm. This is the weight update formula for FLANN structure train with LMS [4.15], [4.18].

(b) Derivative free algorithms/Evolutionary computing based algorithms

Evolutionary computing algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), bacteria foraging optimization (BFO) etc. can also be used for training the network [4.18]. For training the weights using bacteria foraging optimization (BFO), the weights of the FLANN are considered as the bacteria and initially their values are set to random numbers. A population of such bacteria is chosen to represent the initial solutions of the model. Each bacterium updates its values using the BFO principle by way of minimizing the mean square error (MSE) as the cost function. The details of training of weight using PSO and CPSO are presented in [4.20]. The weights are considered as particles and gene while training the network using PSO and GA respectively.

4.3. Development of the prediction based mean-variance (PBMV) model

This section proposes a prediction based portfolio optimization model called as the prediction based mean-variance (PBMV) model. It uses predicted returns as expected returns instead of using the mean of past returns. Furthermore instead of using the variance of the returns it uses the variance of the errors of the predicted return as risk measure. An investment is planned over a time period and its performance is measured using its return that quantifies the wealth variation. The one period stock return at time t is defined as the difference between the price of the stock at time t and the price at time $t-1$, divided by the price at time $t-1$. Mathematically it is expressed as:

$$R_t = (P_t - P_{t-1})/P_{t-1} , \qquad t \geq 1 \tag{4.13}$$

where R_t is the one-period stock return at time t, and P_t and P_{t-1} are the stock prices at times t and $t-1$, respectively. The series of N past returns of a stock R_s, which is N period series return is defined as

$$R_S = (R_1, R_2, \ldots, R_N) \tag{4.14}$$

The prediction of stock return is a nonlinear task and can be achieved using an adaptive predictor. In this chapter we have used a FLANN structure as the predictor which is explained in the previous section. The training of FLANN is performed using bacteria

foraging optimization. Further Chebybyshev type nonlinear functional expansion of the input pattern is used as it provides better forecasting results

The predicted and the actual return may be represented as:

$$R_t = \hat{R}_t + E_t \qquad (4.15)$$

where R_t and $\hat{R}_t$ be the actual return and predicted return at time t respectively. E_t is the prediction error at time t and is defined as

$$E_t = R_t - \hat{R}_t \qquad (4.16)$$

The time series of N errors of prediction is represented as:

$$E = (E_1, E_2, \dots\dots, E_N) \qquad (4.17)$$

For a non-biased predictor, the series of errors of prediction must be statistically independent and identically distributed (iid), with mean and variance given by

$$E_{mean} = 0 \qquad (4.18)$$

$$\sigma_E^2 = \frac{1}{n-1} \sum_{t=1}^{N} E_t^2 \qquad (4.19)$$

The prediction-based portfolio optimization model is based on the assumptions that the mean of the errors of prediction is zero and the errors of prediction have normal distribution. The variance of the errors of prediction σ_E^2 reflects the uncertainty about the realization of the predicted return and is used in the model as a measure of the individual risk of each stock (the higher the variance, the higher is the risk).

A portfolio is a collection of N stocks and the corresponding weightage (participations). The participation, of each asset is w_i, $i = 0,1,2\dots N$. where $0 \le w_i \le 1$ represents the fraction of the portfolio value invested in the stock i such that

$$\sum_{i=1}^{N} w_i = 1 \qquad (4.20)$$

It shows the budget constraint which ensures that the sum of the weights associated with each asset is equal to one which means all the available money is invested in the portfolio. The predicted return of the portfolio, or portfolio expected return, R_p, is the linear combination of the participations and predicted returns of the stocks of the portfolio and may be expressed as

$$R_p = \sum_{i=1}^{N} w_i \hat{R}_i \qquad (4.21)$$

The portfolio risk is the variance of the joint Normal distribution of the linear combination of the participations and prediction errors of the stocks of the portfolio

$$V = \hat{\sigma}_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \gamma_{Eij} \qquad (4.22)$$

where $\hat{\sigma}_p^2$ is the total portfolio risk and is equal to the variance of the linear combination of the participations and prediction errors of the stocks of the portfolio. N is the number of stocks in the portfolio. w_i and w_j are the participating stocks i and j of the portfolio respectively. γ_{Eij} is the interactive prediction risk of stocks i and j, which is the covariance of the errors of prediction of the stocks i and j.

The prediction based portfolio optimization model can be formulated as single objective maximization of V.

$$\begin{aligned} V &= \lambda \left[\hat{\sigma}_p^2 \right] - (1-\lambda) \left[R_p \right] \\ &= \lambda \left[\sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \gamma_{Eij} \right] - (1-\lambda) \left[\sum_{i=1}^{N} w_i \hat{R}_i \right] \end{aligned} \qquad (4.23)$$

Hence such a formulation yields non-dominated solutions by varying the $\lambda(0 \leq \lambda \leq 1)$ factor. But in the present case, this problem is viewed as a dedicated multiobjective problem and is solved by using two MOEAs. It does not combine the two objectives to obtain the Pareto optimal solution set. Here the two objectives are taken individually and the algorithm tends to optimize both the objectives simultaneously. In the proposed work the two objectives are expressed as minimization problem. To express both the objectives in minimization form, the second objective R_p is expressed as $-R_p$. Accordingly the given portfolio problem is expressed as:

$$\textit{Minimize both } \hat{\sigma}_p^2 \textit{ and } -R_p \textit{ simultaneously.} \qquad (4.24)$$

Thus the novel prediction based mean-variance (PBMV) portfolio optimization model differs from the Markowitz mean-variance model as (a) in prediction based portfolio optimization model, the expected return of each stock is its predicted return. But in the case of Markowitz mean-variance model, the expected return is taken as the mean of past returns. (ii) In PBMV model the individual risk of each stock and the risk between each pair of stocks are obtained from the variance and covariance of the time series of the errors of prediction. But in the case of Markowitz model it is the variance and covariance of the time series of return. (iii) In prediction based portfolio optimization model the normal variable of interest is the error of prediction of the return of stocks, while in the case of Markowitz model the normal variable of interest is the return of the stocks.

4.4. Simulation studies

For simulation all the algorithms are coded in MATLAB and run on a PC with Intel Core 2 Duo 3.0 GHz with 4 GB RAM.

4.4.1. Data Collection

The experiments have been conducted with a set of benchmark data available online and obtained from OR-Library [4.26]. The data correspond to weekly prices between March 1992 and September 1997 from different well known indices such as Hang Seng in Hong Kong, DAX 100 in Germany, FTSE100 in UK,S&P 100 in USA and Nikkei225 in Japan. This weekly price can also be found out from [4.27]. The numbers of different assets for the above benchmark indices are 31, 85, 89, 98 and 225 respectively. Using each data the mean return of individual assets is calculated from the weekly price. The data set PORT-1 and PORT-5 correspond to the correlation between assets for five markets respectively. Covariance between the assets, evaluated from the correlation matrix, can be used for calculating the risk of portfolio. The data (risk and corresponding tradeoff return) for standard efficient frontiers for the five stocks can be found from PORTEF-1 to PORTEF-5[4.26] which correspond to Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 stock indices respectively.

4.4.2. The problem approach

(a) Using Markowitz mean-variance model

The raw weekly prices of all the stocks (assets) of five market indices are collected. The weekly return is calculated mathematically from this weekly price. The time series of expected return of any asset can be found by calculating the mean of past returns mathematically. The individual risk of each stock and the risk between each pair of stocks are obtained from the variance and covariance of the time series of return. The FLANN network is not used for this model as it does not need prediction to find out the expected return.

(b) Using proposed mean-variance model

The raw weekly prices of all the stocks (assets) of five market indices are collected. The weekly returns is calculated mathematically from this weekly price. Then the FLANN forecasting network is used to predict the weekly return by taking the calculated previous weekly return as input parameter. Some statistical variables such as moving averages, mode and median of input is also provided to the network. It is then expanded using Chebyshev functional expansions and evolutionary computation is used to adjust the weight parameters so that effective prediction is achieved.

(c) Constraint portfolio optimization using MOEAs

By applying the two models, the risk and return of all the assets are found out. After calculating the return and risk, the portfolio optimization task is carried out by using some efficient multiobjective evolutionary algorithms (MOEAs). Two MOEAs based on particle swarm optimization such as non-dominated sorting particle swarm optimization(NS-MOPSO) and decomposition based particle swarm multiobjective evolutionary algorithm (P-MOEA/D) have been applied to solve the portfolio optimization problem. Similarly another two algorithms based on bacteria foraging optimization such as multiobjective bacteria foraging optimization (MOBFO) and decomposition based bacteria foraging multiobjective evolutionary algorithm (B-MOEA/D) have been applied to the same problem.

4.4.3. Experimental results

The Pareto front corresponding to five market indices can be found in PORTEF-1 to PORTEF-5 [4.26], called as standard efficient front or global optimal Pareto front (GOPF). The GOPF for Hang-Seng stock is depicted in Fig.4.2. It shows the tradeoff between risk (variance of return) and return (mean return).
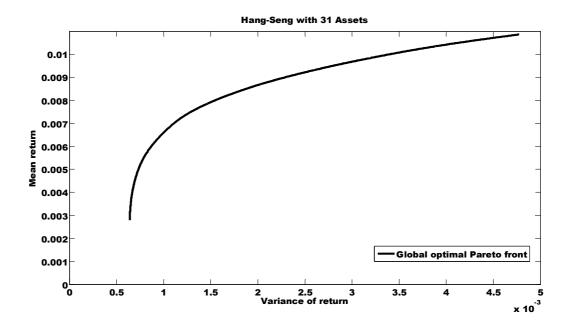
Fig.4.2 Global optimal Pareto front for Hang-Seng, stock indices

The Pareto fronts obtained by different algorithms for Hang-Seng stock using the proposed PBMV model is shown in Figs.4.3 to 4.7. It is compared with GOPF and Pareto front obtained using Markowitz model.
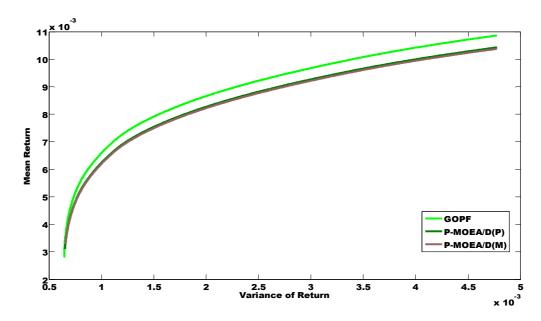
Fig.4.3 The GOPF and Pareto front by P-MOEA/D for Hang-Seng using Markowitz and PBMV model
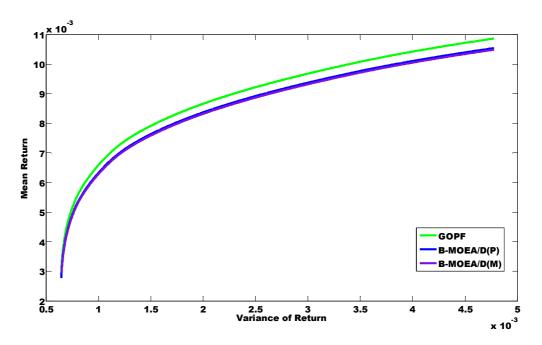
Fig.4.4 The GOPF and Pareto front by B-MOEA/D for Hang-Seng using Markowitz and PBMV model
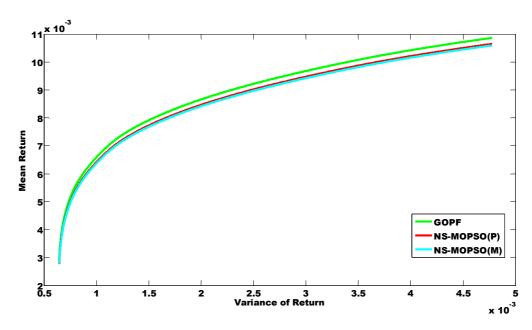
Fig.4.5. The GOPF and Pareto front by NS-MOPSO for Hang-Seng using Markowitz and PBMV model
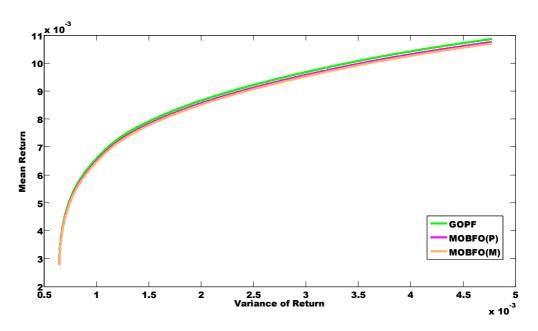
Fig.4.6 The GOPF and Pareto front by MOBFO for Hang-Seng using Markowitz and PBMV model
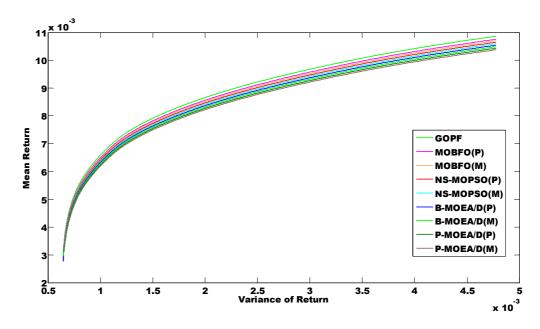
Fig.4.7 The GOPF and Pareto front by four algorithms applying two models

It is evident from the results that all the algorithms are capable of providing good solutions using the proposed PBMV model. The Pareto curve obtained by applying PBMV model is more close to GOPF.

In our proposed PBMV model risk is calculated by taking the covariance of time series of the error of prediction of stock. The risk can also be calculated using the covariance of time series of predicted return. The Pareto fronts obtain by taking risk as covariance of time series of predicted return is shown in figure 4.8.
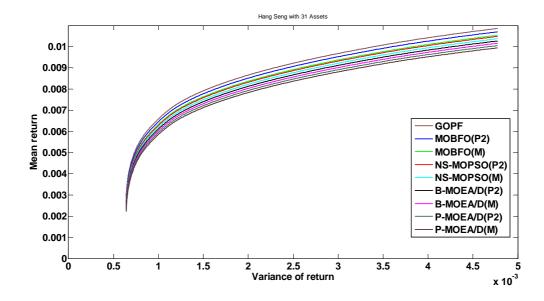
Fig.4.8. The GOPF and Pareto front by four algorithms applying two models

Further, the performance of these MOEAs is assessed by using five different metrics such as the S, GD, IGD, Δ and Er. The algorithms are run for 25 times and then the maximum, minimum, average and standard deviation of these metrics are calculated and the corresponding results are shown in Table-4.1.

Table-4.1 Comparison of performance evaluation metrics obtained using different MOEAs.

| Algorithm		P-MOEA/D (M)	P-MOEA/D (P)	B-MOEA/D (M)	B-MOEA/D (P)	NS-MOPSO (M)	NS-MOPSO (P)	MOBFO (M)	MOBFO (P)
S	Max.	5.93E-6	5.67E-6	5.99 E-6	5.78 E-6	5.38 E-6	5.04 E-6	5.22E-6	4.98 E-6
	Min.	2.51E-6	2.05E-6	2.38 E-6	2.08 E-6	2.32 E-6	2.02 E-6	2.33E-6	1.99 E-6
	Avg.	3.62E-6	3.43E-6	3.93 E-6	3.65 E-6	3.48 E-6	3.11 E-6	3.45E-6	3.05-E6
	Std.	0.87E-6	0.79E-6	0.98 E-6	0.88 E-6	0.76 E-6	0.59 E-6	0.85E-6	0.73 E-6
GD	Max.	2.63E-4	2.13E-4	2.92 E-4	2.23 E-4	2.12 E-4	1.89 E-4	2.16E-4	1.88 E-4
	Min.	1.65E-4	1.15E-4	1.36 E-4	1.06 E-4	1.02 E-4	0.86 E-4	1.10E-4	0.96 E-4
	Avg.	1.76E-4	1.45E-4	1.73 E-4	1.43 E-4	1.58 E-4	1.45 E-4	1.45E-4	1.27 E-4
	Std.	0.57E-4	0.53E-4	0.52 E-4	0.46 E-4	0.38 E-4	0.29 E-4	0.36E-4	0.27 E-4
IGD	Max.	9.10 E-4	8.50 E-4	9.21 E-4	8.81 E-4	8.45 E-4	7.98 E-4	8.30 E-4	7.88 E-4
	Min.	7.67 E-4	7.05 E-4	7.76 E-4	7.26 E-4	6.35 E-4	5.98 E-4	6.45 E-4	6.02 E-4
	Avg.	8.20 E-4	8.01 E-4	8.23 E-4	7.98 E-4	7.15E-4	6.75 E-4	7.10 E-4	6.80 E-4
	Std.	2.02 E-4	1.89 E-4	2.07 E-4	1.88 E-4	1.81E-4	1.75 E-4	1.74 E-4	1.35 E-4
Δ	Max.	2.34E-1	2.13E-1	2.41 E-1	2.15 E-1	2.43 E-1	2.06 E-1	2.45E-1	1.99 E-1
	Min.	1.20E-1	1.01E-1	1.22 E-1	1.02 E-1	1.20 E-1	0.91 E-1	0.99E-1	0.90E-1
	Avg.	1.45E-1	1.25E-1	1.49 E-1	1.38 E-1	1.34 E-1	0.99 E-1	1.33E-1	1.03E-1
	Std.	0.46E-1	0.36E-1	0.48 E-1	0.43 E-1	0.45 E-1	0.38 E-1	0.47E-1	0.36 E-1
Er	Max.	0.29	0.27	0.28	0.27	0.24	0.18	0.23	0.17
	Min.	0.14	0.13	0.15	0.14	0.15	0.14	0.16	0.13
	Avg.	0.21	0.20	0.22	0.20	0.19	0.16	0.18	0.15
	Std.	0.07	0.06	0.08	0.07	0.06	0.05	0.06	0.04

The convergence metrics (C) for these MOEAs are demonstrated in Table-4.2 It is found that most of the solutions obtained by the MOBFO algorithm with proposed PBMV model dominate the solutions obtained from others.

Table-4.2. Comparison of results of C metric obtained using different MOEAs

	P-MOEA/D(M)	P-MOEA/D(P)	B-MOEA/D(M)	B-MOEA/D(P)	NS-MOPSO(M)	NS-MOPSO(P)	MOBFO(M)	MOBFO(P)
P-MOEA/D(M)	—	0.2910	0.2720	0.2621	0.2430	0.2231	0.2110	0.2054
P-MOEA/D(P)	0.3180	—	0. 2910	0.2800	0.2680	0.2531	0.2351	0.2121
B-MOEA/D(M)	0.3620	0.3400	—	0.3210	0.2822	0.2612	0.2410	0.2321
B-MOEA/D(P)	0.4228	0.4077	0.3800	—	0.3288	0.2921	0.2728	0.2520
NS-MOPSO(M)	0.4530	0.4320	0.3999	0.3600	—	0.3428	0.3232	0.2811
NS-MOPSO(P)	0.4721	0.4632	0.4211	0.3821	0.3455	—	0.3533	0.3111
MOBFO(M)	0.4910	0.4732	0.4321	0.3922	0.3511	0.3521	—	0.3231
MOBFO(P)	0.5032	0.4821	0.4402	0.4288	0.3822	0.3722	0.3688	—

The nonparametric statistical test such as the Sign test and Wilcoxon signed ranks rest are carried out for pair wise comparisons of the performance of two algorithms [4.28]. The critical number of wins needed to achieve both $\alpha = 0.05$ and $\alpha = 0.1$ levels of significance is shown in Table-3.8 in Section 3.3.6. An algorithm is significantly better than other if its performance is better on at least the cases presented in each row.

The results of the Sign test for pairwise comparisons among proposed MOBFO(P) i.e. MOBFO with PBMV model and other algorithms while taking the S metric as the wining parameter (i.e. lower value of S means win) and applying to Heng-Seng stock are shown in Table-4.3. From the results it is clear that the MOBFO (P) shows improvement over P-MOEA/D with a level of significance $\alpha = 0.01$. This test can also be conducted using other metrics as winning parameters.

Table-4.3. Critical values for the two-tailed Sign test at $\alpha = 0.05$ and $\alpha = 0.1$ using S metric as winning parameter.

MOBFO(P)	P-MOEA/D(M)	P-MOEA/D(P)	B-MOEA/D(M)	B-MOEA/D(P)	NS-MOPSO(M)	NS-MOPSO(P)	MOBFO(M)
Wins(+)	17	16	16	15	15	14	13
Losses(-)	8	9	9	10	10	9	12
Detected differences	$\alpha = 0.01$	—	—	—	—	—	—

The Wilcoxon signed rank test is carried out by calculating R^+ and R^- and then using well-known statistical software package SPSS. Table-4.4 shows the R^+, R^-, z, Asymp. sig (2-tailed), Exact sig. (2-tailed), Exact sig. (1-tailed) and point of probability computed for all the pairwise comparisons with MOBFO(P) considering S metric as winning parameter and applying to Hang-Seng market indices. The result of the Wilcoxon signed rank test for other metrics can be tested for this case.

Table-4.4. Wilcoxon Signed test using S metric as winning parameter and applying different MOEAs to Hang-Seng market indices

Comparison	R^+	R^-	z	Asymp.sig (2-tailed),	Exact sig. (2-tailed),	Exact sig. (1-tailed)	Point of probability
MOBFO (P) with P-MOEA/D (M)	192	133	-0.794	0.427	0.437	0.219	0.004
MOBFO (P) with P-MOEA/D (P)	189	136	-0.714	0.475	0.486	0.243	0.004
MOBFO (P) with B-MOEA/D(M)	182	143	-0.711	0.465	0.498	0.251	0.004
MOBFO (P) with B-MOEA/D(P)	185.5	139.5	-0.619	0.536	0.546	0.273	0.004
MOBFO (P) with MOPSO (M)	180	145	-0.617	0.543	0.557	0.279	0.004
MOBFO (P) with MOPSO(P)	156	168	-0.162	0.872	0.879	0.440	0.005
MOBFO (P)with MOBFO (M)	168	157	-0.148	0.882	0.890	0.445	0.005
MOBFO with B-MOEA/D	192	133	-0.794	0.427	0.437	0.219	0.004
MOBFO with MOPSO	189	136	-0.714	0.475	0.486	0.243	0.004

From the results it is concluded that the MOBFO(P) i.e. MOBFO algorithm with PBMV model show improved performance compared to its counterpart. Similarly all the MOEAs can also be applied to other stock indices such as DAX 100, FTSE 100, S&P 100 and Nikkei 225 using both models.

The presence of cardinality constraints K is also studied here. The Pareto fronts obtained by applying MOBFO (P) for Hang-Seng data set having different cardinalities are presented in Fig.4.9. K is set at 5 and is increased to 30 at a step of 5. The Pareto fronts become shorter with increase in K values. Hence the proposed algorithm is able to obtain a near optimal solution efficiently by investing lower number of assets. The portfolio manager has the option to make trade-off between risk and returns for different values of K. Similarly the NS-MOPSO, P-MOEA/D, B-MOEA/D algorithms can be applied to different market having a different cardinality constraint.

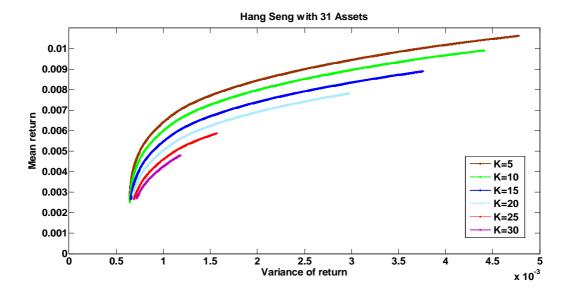Fig.4.9 The Pareto front obtained from MOBFO for Hang-Seng using proposed PBMV model for cardinality constraint condition.

The proposed algorithm is also applied for BSE-500 (Bombay Stock Exchange) of India. The raw weekly prices of 50 stocks (assets) from 500 stocks are collected [4.27].
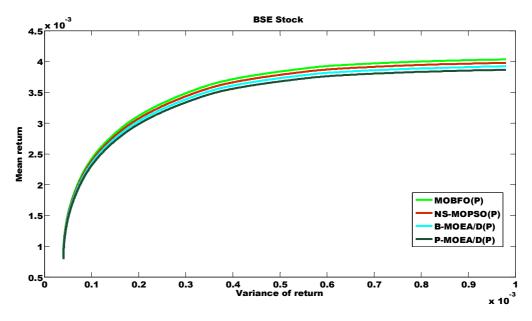

Fig.4.10 The Pareto front obtained from NS-MOPSO and MOBFO for BSE stock using PBMV and Markowitz model.

4.5. Conclusion

A novel prediction based mean variance (PBMV) model has been proposed in the chapter and four efficient MOEAs have been successfully employed to solve the portfolio optimization problem. In the proposed model the return is predicted with a low complexity single layer neural network. The performance of the proposed PBMV model and the Markowitz model have been evaluated and compared using six performance metrics. This evaluation involves experiments with real data from the five Hang-Seng, DAX 100, FTSE100, S&P and Nikkei 225 and Bombay Stock Exchange (BSE-500) data. In addition to this, in the present study the Sign test and Wilcoxon Signed rank test are carried out to compare the performance of the algorithms. From the simulation results it is observed that the PBMV model is capable of identifying good Pareto solutions maintaining adequate diversity and the performance is comparable with the well known Markowitz mean-variance model. Further study in this field may include performance evaluation of the MOEAs using the proposed model considering some real world constraints like ceiling, floor, round-lot, turnover etc. The same multiobjective optimization algorithm can also be applied to other financial applications such as asset allocation, risk management and option pricing.

Reference

[4.1] H. M. Markowitz, "Portfolio selection", Journal of Finance, vol. 7 no.1, pp. 77-91, 1952.

[4.2]Tun-Jen Chang, Sang-Chin Yang b, Kuang-Jung Chang, "Portfolio optimization problems in different risk measures using genetic algorithm", Expert Systems with Applications, Vol. 36 pp. 10529–10537,2009.

[4.3] E. J. Elton, M. J. Gruber, S. J. Brown, W. N. Goetzmann, "Modern portfolio theory and investment analysis", seventhed, Wiley, New York, 2007.

[4.4]E. F. Fama, "Portfolio analysis in a stable Paretian market", Management Science, vol.11,pp.404–419,1965.

[4.5] S. J. Kon, "Models of stock returns—a comparison", The Journal of Finance, vol.39. no.1, pp. 147–165, 1984.

[4.6] W. F. Sharpe, G. J. Alexander, J. V. Bailey, "Investments", sixthed, Prentice-Hall, Upper Saddle River, New Jersey,1999.

[4.7] Fabio D. Freitas, Alberto F. DeSouza , AilsonR. de Almeida, "Prediction-based portfolio optimization model using neural networks", Neuro computing, Vol. 72 pp. 2155–2170,2009.

[4.8] T.Z. Tan, C. Quek, G.S. Ng, "Brain Inspired Genetic Complimentary Learning for Stock Market Prediction", IEEE congress on evolutionary computation, vol. 3, pp. 2653-2660, 2-5th Sep., 2005.

[4.9] K.J. Oh, K-j. Kim, "Analyzing stock market tick data using piecewise non linear model." Expert System with Applications, vol. 22, issue 3, pp. 249-255, April 2002.

[4.10] Y. Wang, "Mining stock prices using fuzzy rough set system" Expert System with Applications, vol. 24, issue 1, pp. 13-23, Jan. 2003.

[4.11] Y. Wang, Predicting stock price using fuzzy grey prediction system" Expert System with Applications, vol. 22, pp. 33-39, Jan. 2002.

[4.12] M. Schumann and T. Lohrbach, "Comparing artificial neural networks with statistical methods within the field of stock market prediction", Proc. Of Twenty-Sixth Hawaii International Conference on System Sciences, vol. 4, pp. 597-606, 1993.

[4.13] J. Hanm, N. Kamber, "Data Mining: Concepts & Techniques", San Francisco; Morgan Kanfmann Publishers, 2001.

[4.14] E.W. Saad , D.V. Prokhorov, D.C. Wunsch, "Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks", IEEE Transactions of Neural Network, vol. 9, issue 6, pp. 1456-1470, Nov.1998.

[4.15] T. Yamashita, K. Hirasawa, J. Hu, "Application of multi-branch neural networks to stock market prediction", International joint conference on neural networks, Montreal, Canada, 31st July-4th Aug., 2005,vol.4, pp. 2544-2548.

[4.16] Y. Chen, X. Dong, Y. Zhao, "Stock index modeling using EDA based local linear wavelet neural network", International conference on neural networks and brain, 13th-15thOct., 2005, vol. 3, pp. 1646-1650.

[4.17] T. Masters, "Practical Neural Network recipes in C++", Academic Press, New York, 1993.

[4.18] Kyoung-jaeKim, "Artificial neural networks with evolutionary instance selection for financial forecasting", Expert System with Applications, vol.30, issue 3, pp. 519-526, April, 2006.

[4.19] Ritanjali Majhi, G. Panda, G. Sahoo, "Efficient prediction of exchange rates with low complexity artificial neural network models", Expert Systems with Applications, vol.36, pp.181-189, 2009.

 [4.20] Ritanjali Majhi, G. Panda, G. Sahoo, "Development and performance evaluation of FLANN based model for forecasting of stock markets", Expert Systems with Applications, vol. 36, pp.6800-6808, 2009.

[4.21] Ritanjali Majhi, G. Panda, G. Sahoo, "On the development of improved adaptive models for efficient prediction of stock indices using clonal-PSO (CPSO) and PSO techniques", Int. J. Business Forecasting and Marketing Intelligence, Vol. 1, No. 1, PP.50-67, 2008.

 [4.22]Y. H. Pao,"Adaptive Pattern Recognition and Neural Networks", Reading, MA : Addison-Wesley, 1989.

[4.23] Y. H. Pao, S. M Phillips and D. J. Sobajic, "Neural Net Computing and intelligent control systems", Int. Journal on Contr., vol. 56, issue 2, pp. 263-28, 1992.

[4.24] Y. H. Pao,"Adaptive Pattern Recognition and Neural Networks", Addison Wesley, Reading, Massachusetts, 1989.

[4.25] J. C. Patra, R. N Pal, B. N. Chatterji and G. Panda, "Identification of non-linear & dynamic system using functional link artificial neural network" IEEE Transactions on System, Man & Cybernetics – Part B; Cybernetics, vol. 29, issue 2, pp. 254-262, 1999.

[4.26] http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html

[4.27] http://in.finance.yahoo.com/q/hp?s=%5EHSI

[4.28] J. Derrac, S. García, D. Molina, F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms", Swarm and Evolutionary Computation,Vol.1 no.1, pp.3-18, 2011.

[4.29] B. Widrow and S.D. Sterns,"Adaptive Signal Processing" Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1985.

[4.30] S. Haykin, "Adaptive Filter Theory", 4th edition, Pearson Education Asia, 2002.

Chapter 5

Novel Robust Multiobjective Portfolio Optimization Schemes

Chapter

5

In this chapter the minimum volume ellipsoid (MVE) methodology is adopted to handle uncertainty of the stock market data. The uncertainty is in form of outliers present in the stock data and occurs at random samples. The uncertainties may be due to incidence like Sep 11 or sudden fall of oil price or any political crises. The value of stock at that unexpected situation may be called as uncertain or unexpected stock data. Firstly, the MVE is formed covering the data that are not corrupted by outliers. The unexpected data can easily be differentiated from other data by clustering using MVE method. In order to make the method computationally efficient, the MVE is formed by using the core set and Lagrange multipliers. Secondly, the weight factor is calculated by taking the parameters associated with the ellipsoid. Then the unexpected data are modified by multiplying the weighting factor with it and the desired parameters such as risk and return are calculated from the weighted data. The trade-off (Paretro curve) between this new estimated return and risk parameters are found out by using some efficient MOEAs using both Markowitz mean-variance model and prediction based mean-variance (PBMV) model as proposed in the previous chapter. Simulation results reveal that the proposed method exhibits good portfolio strategy in the presence of these market uncertainties.

5.1 Introduction

The mean-variance model, proposed by Harry Markowitz [5.1], is a landmark in modern portfolio theory (MPT). Subsequently some other methods such as semi-variance, mean absolute deviation and variance with skewness model are also used for portfolio optimization problem [5.2]. The idea of designing a model by suitably modifying the conflicting objectives is also investigated by Lin *et al.* [5.3]. All these frameworks require the knowledge of stock values from which these models estimate the expected return and calculate the corresponding risk. However the stock values are highly uncertain. This uncertainty may be due to incidents like September 11, any political crises or the recent turmoil in global markets which started from the financial sector. These uncertain factors make the stock value uncertain and deviate heavily from its actual value. The value of stock indices due to these types of unexpected situation may be considered as uncertain or unexpected stock data. These uncertain values of stock may be called as outliers. Hence, inaccuracy creeps in while estimating the return and risk by using such contaminated stock values.

Most of the aforementioned models consider the estimated parameters as the actual parameters without considering these types of uncertainty which limits the versatility of these models. Hence, the problem of portfolio optimization becomes more challenging and complicated under these uncertain conditions.

In the last two decades, robust optimization under such conditions has become an interesting area of research. Soyster *et al.* [5.4] were first to introduce the idea of robust optimization. In general, the robust optimization aims to find the solutions to a given optimization problem with uncertain parameters. The authors in [5.5] have developed a new

robust methodology using interior point based algorithm to find the robust solution. They have also applied a robust method to some portfolio optimization problems and have shown that the final optimal solution remains feasible against the uncertainty on different input parameters.

Robust optimization has been applied to portfolio selection problem to alleviate the sensitivity of optimal portfolios to statistical errors in the estimates of the parameters. Goldfarb and Iyengar [5.6] have considered a factor model for the random portfolio returns and have proposed some statistical procedures to construct the uncertainty sets for the parameters. Bertsimas and Pachamanova [5.7] have investigated the viability of different robust optimization approaches for multi-period portfolio selection. Recently robust optimization has been applied to different fields including finance and industrial problems [5.8-10].

In these studies, the robust optimization models treat the asset returns as uncertain coefficients and map the level of risk aversion of the investor to the level of tolerance of the total error in asset return estimation. However in these robust optimization techniques, the program dimension increases exponentially as the size of the problem i.e. number of assets present in the portfolio optimization increases. The difficulties become more pronounced when the number of constraints becomes more. Therefore, there is a need to develop robust portfolio optimization techniques which can handle efficiently the outliers present in the financial data.

In this Chapter, we propose a new framework using the MVE methodology for achieving robust portfolio optimization. The MVE is formed by using the core set and Lagrange multipliers. Some weight factor is calculated by taking the parameters associated

with the ellipsoid. Then the data are modified by multiplying each of them with the weight factor. The weight factor is designed in such a way that it does not change the data those are present inside the ellipsoid. The magnitude of data those are present outside the ellipsoid are suitably decreases. Then the desired parameters such as risk and return are calculated from the weighted data. The trade-off (Pareto curve) between this new estimated return and risk parameters are found out by using some efficient MOEAs. In this present study, the portfolio optimization problem with practical constraints has been solved by applying four MOEAs such as MOPSO, MOBFO, MOEA/D-P and MOEA/D-B algorithms and using both the Markowitz mean-variance model and proposed PBMV models.

5.2 Development of robust portfolio optimization under uncertainties

Since the data of the market do not changes fast with time, all the data points remain close to each other forming a cluster in multidimensional space. However, in the presence of uncertainty, the market data points deviate from its normal deviation. In multidimensional space these unexpected data remain away from the clustered data. Moreover, every unexpected datum also remains away from each other depending on the strength of the outliers. So the first objective is to suitably modify the uncertain data.

In order to achieve this, the minimum volume ellipsoid (MVE) method [5.11-13] is applied to get an ellipsoid covering healthy (good) data points. This covering of finite data set using MVE is a convex optimization [5.14] problem which is formulated as:

$$Q, c = \arg \min_{Q,c} \log Det(Q_{k+1})$$
$$s.t \ \ (x_{k+1} - c_{k+1})' Q_{k+1}^{-1} (x_{k+1} - c_{k+1}) \leq 1$$
$$k = 1,...,n \tag{5.1}$$

where Q_{k+1}, c_{k+1} are the spreading matrix and the center associated with the ellipsoid. x_{k+1},

$k = 1,2,3...,n$ are the good market data set. The above convex optimization problem can be solved by using interior point method [5.15]. However this method requires large amounts of computational complexity. In order to avoid this, the Lagrange multipliers based approach has been used. The Khachiyan's algorithm [5.16],[5.17] is one such method to calculate the MVE using the Lagrange multipliers. To apply the Khachiyan's algorithm the data should be symmetric across the origin. In order to make this, firstly the data number is increased to two times by first collecting the data and then multiplying by 1 and -1.Then 1 or -1 is padded according to the data is multiplied by 1 or -1 respectively. By this way n numbers of data points change to $2n$ number of data points with one extra dimension which is symmetric to the origin. Mathematically, it is given by

$$S' = \{\pm y_1,......,\pm y_n\} \tag{5.2}$$

where y_i and $-y_i$ are

$$+ y_1 = \left[+1\times(x_1)',1\right]'$$
$$- y_1 = \left[-1\times(x_1)',-1\right]' \tag{5.3}$$

Since the new data points are symmetric with respect to the origin, the center of the MVE to be formed lies at the origin. Hence the original ellipsoid is related to this new MVE to be formed as:

$$MVEE\,(S) = MVEE\,(S') \cap H \tag{5.4}$$

where $MVEE\,(S)$ means the minimum volume ellipsoid enclosing S original points and $MVEE\,(S')$ is the minimum volume ellipsoid enclosing S' new high dimensional points. and H is mathematically expressed as

$$H = \left\{ x \in R^{d+1} : x_{d+1} = 1 \right\} \tag{5.5}$$

Now the formulation of the MVE for the data of high dimension is given by

$$Q^P = \arg \min_{Q^P} \; - \log Det(Q^P)$$

$$s.t \;\; y_i' Q^P y_i \le 1, \; i = 1, ..., n \tag{5.6}$$

Now the Lagrange multipliers [5.18] based MVE can easily be found from the optimum point which is obtained by taking the the Karush–Kuhn–Tucker (KKT) condition. This is given mathematically as

$$-\left(Q^* \right)^{-1} + \Pi \left(Z^* \right) = 0$$

$$z^* \left(1 - y_i' Q^* y_i \right) = 0, \; i = 1, ..., n$$

$$y_i' Q^* y_i \le 1$$

$$z^* \ge 0 \tag{5.7}$$

where $\Pi : \Re^n \to \Re^{(d+1) \times (d+1)}$
which is given by

$$\Pi \left(z \right) := \sum z_i \, y_i \, y_i' \tag{5.8}$$

In order to achieve the optimum point based on the KKT condition the duality problem is

$$\max_{u} \log \det \Pi(u)$$

$$s.t. \; e'u = 1$$

$$u \ge 0 \tag{5.9}$$

This dual problem is the maximization of a concave function. So the optimum MVE covering S is given as

$$MVEE \, (S) = \left\{ x \in \Re^d : [1/(d+1)] \, [x' \; 1] \, \Pi \left(u^* \right)^{-1} \begin{bmatrix} x \\ 1 \end{bmatrix} \ge 1 \right\} \tag{5.10}$$

where , $\Pi(u^*)$ is defined in terms of the parameter of the MVE of the original problem in the following way

$$\Pi \left(u^* \right) = \begin{bmatrix} PU*P' & Pu^* \\ 0 & 1 \end{bmatrix} \tag{5.11}$$

Then applying the Schur complement [5.19] to the (5.11), we obtain

148

$$\Pi\left(u^{*}\right) = \begin{bmatrix} 1 & Pu^{*} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \left(PU*P - Pu^{*}\left(Pu^{*}\right)\right) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \left(Pu^{*}\right)' & 1 \end{bmatrix} \quad (5.12)$$

Then the inverse of $\Pi(u^{*})$ is given as

$$\Pi\left(u^{*}\right)^{-1} = \begin{bmatrix} 1 & 0 \\ -\left(Pu^{*}\right)' & 1 \end{bmatrix} \begin{bmatrix} \left(PU*P - Pu^{*}\left(Pu^{*}\right)'\right)^{-1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -Pu^{*} \\ 0 & 1 \end{bmatrix} \quad (5.13)$$

The original MVE problem is related to the parameter of the new MVE problem of higher dimension by the formula given in (5.14)

$$MVEE\left(S\right) = \in_{Q^{*}C^{*}} := \left\{ x \in \Re^{d} : \left(x - c^{*}\right)' Q^{*} \left(x - c^{*}\right) \leq 1 \right\} \quad (5.14)$$

where,

$$Q^{*} := \left(1/d\right) \left(PU^{*}P' - Pu^{*}\left(Pu^{*}\right)' \right)^{-1}$$

$$c^{*} = Pu^{*} \quad (5.15)$$

The method of duality problem expressed in (5.9) is based on the entire data set. It is a fact that the MVE of any number of data set with dimensions L can be the obtained from $2L$ number of points which may occur at the circumference of the data set. These 2L points are subset of original n number of points. To find out this $2L$ number of points the Gram-Schmidt Orthogonalisation procedure is used in the Chapter. In this procedure, a vector is randomly selected and then all the points are projected upon that vector. Furthermore, only two points are selected that are having large and small magnitudes of projected values. These two points are used to find the new vector which is passing through these two points. Subsequently, another vector is selected which is perpendicular to this new vector. Another two points are found out by applying similar projection based approach. This procedure of finding the orthogonal vector and the points based on projection is repeated number of times of the dimensions i.e. L number of times. The number of points obtained by this procedure

is called the core set of the data points. The pseudo code of this algorithm is dealt next.

The pseudo code:

$$
\begin{aligned}
&For\ i = 1:n \\
&\quad if\ n \leq 2d \\
&\quad\quad \chi_0 \leftarrow S \\
&else \\
&\quad while \\
&\quad \Re^d / \Psi \neq \phi \\
&\quad Pick\ an\ arbitrary\ direction\ b^i \in R^d \\
&\quad orthogonal\ component\ of\ \Psi \\
&\quad \alpha \leftarrow \underset{k=1\ldots,n}{\arg\max}\ \left(b^i\right)' p^k,\ \chi_0 \leftarrow \left\{\chi_0 \cup \left\{p^\alpha\right\}\right\} \\
&\quad \beta \leftarrow \underset{k=1\ldots,n}{\arg\max}\ \left(b^i\right)' p^k,\ \chi_0 \leftarrow \left\{\chi_0 \cup \left\{p^\beta\right\}\right\} \\
&\quad \Psi \leftarrow Span\left(\Psi, \left\{p^\beta - p^\alpha\right\}\right) \\
&end
\end{aligned}
\tag{5.16}
$$

The next objective is to find out the MVE from the core set. Thus the computational complexity of the MVE decreases. The dual problem of the original MVE problem taking only the core set point is given by

$$
\max_{v \in R^n}\ \sum_{k=1}^{n} v_k \left(y_k\right)' \Pi\left(u^i\right)^{-1} y_k
$$
$$
s.t\ e\ v = 0,\ v \geq 0.
\tag{5.17}
$$

Now the (5.17) is solved by using Khachiyan's algorithm and is given by

$$
k^j := \max_{k=1\ldots,n}\ y_k'\ \Pi\left(u^i\right)^{-1} y_k
\tag{5.18}
$$

$$
u^{i+1} := \left(1 - \beta^i\right) u^i + \beta^i\ e_j
\tag{5.19}
$$

$$\beta^i := \arg\max_{\beta \in [0,1]} \log \det \Pi\left((1-\beta)u^i + \beta e_j\right)$$

$$= \left[k^i - (d+1)\right] \bigg/ \left[(d+1)\left(k^i - 1\right)\right] \tag{5.20}$$

Thus a MVE is formed covering the data set without outlier inside of it. The data contaminated with outliers are present outside of it. The next objective is to find out the weight factor for every data point.

The data presented inside the MVE remains as they are and the data away from the MVE are provided lesser importance. The points remaining far away from the ellipse are assigned lesser weightage than those which are situated nearer to it. In order to calculate the required weight values the Mahalanobis distance [5.23] is found out by using the parameters of the MVE and is given by

$$M(x_i) = \sqrt{(x_i - c_i)^T Q^{-1}(x_i - c_i)} \tag{5.21}$$

Now the weight factor corresponding to x_i is given as

$$w_i = \min\left\{1, \frac{\chi_{p,n-1}}{\sqrt{(u_i - c_i)^T Q^{-1}(u_i - c_i)}}\right\} \tag{5.22}$$

To obtain the modified data, w_i is multiplied to each of it. It is clear from (5.22) that w_i is 1 for those data present inside the ellipsoid and this factor computed from (5.22) is less than 1 for those data outside the ellipsoid.

5.3 Forecasting network

The same FLANN network as discussed in Chapter 4 is used in our study. In the simulation, the bacteria foraging optimization (BFO) based algorithm is used for updating the weights of the network. A population of such bacteria is chosen to represent the initial

solutions of the model. Each bacterium updates its values using the BFO principle by suitably minimizing the mean squares error (MSE) as the cost function.
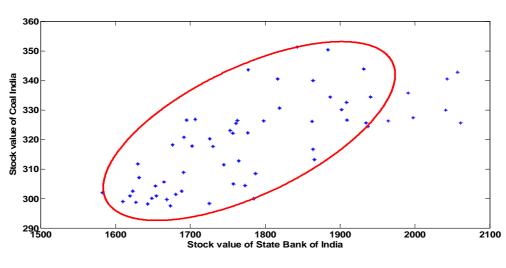
5.4 Simulation study

To evaluate the performance of the proposed algorithm real life data are collected and used. The algorithm described in the previous section is coded in MATLAB and runs on a PC with Intel Core 2 Duo 3.0 GHz with 4 GB RAM.
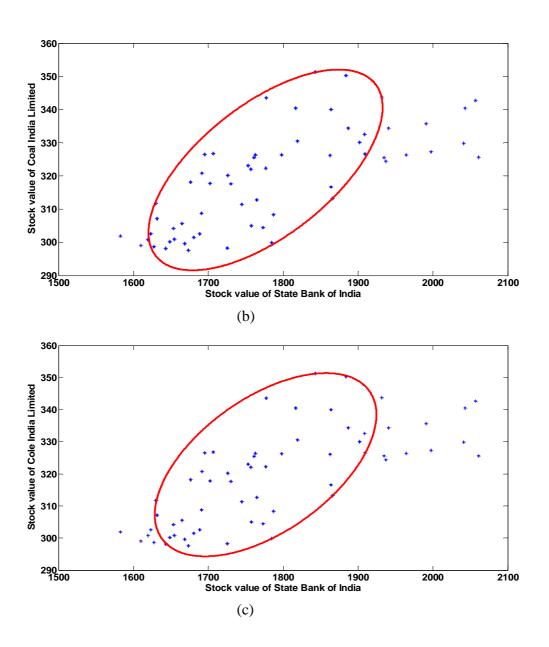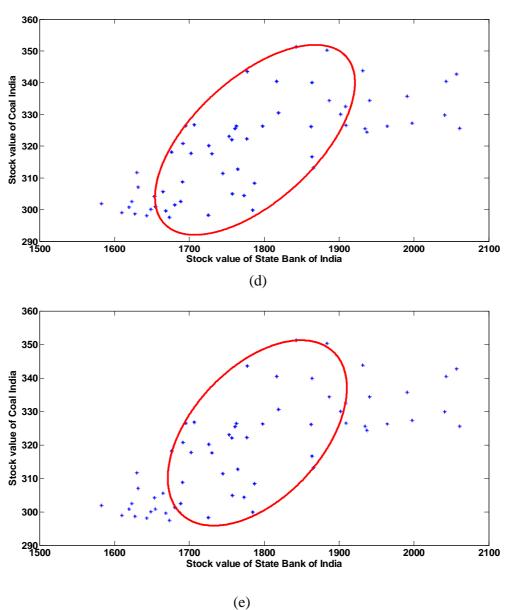
5.4.1 Data collection

The data for 31 stocks from Hang-Seng, 85 from DAX 100, 89 from FTSE 100, 98 from S&P 100 and 225 from Nikkei 225 stock indices are obtained from the website http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html [5.20]. The data of PORT-1 and PORT-5 correspond to weekly prices between March 1992 and September 1997. The weekly price of these stock can also be found from http://in.finance.yahoo.com/q/hp?s=%5EHSI [5.21]. Similarly the weekly closing price of the day of stocks for BSE is available in http://in.finance.yahoo.com/q/hp?s=%5EBSESN[5.22]. In the present study, the weekly stock values of Hang-Seng, DAX 100, FTSE 100, S&P 100 and Nikkei 225 between March 1992 and September 1997 were collected. A subset of 20 stocks from the 500 stocks that participated in Bombay Stock Exchange (BSE-500) index between December 2008 to January 2012 has been selected for the present study. The data collected for each one of these 20 stock indices consisted of the weekly closing price of the stock.

5.4.2 Construction of MVE from real life data

In the present study, it is assumed that some percent of the collected stock data of all the markets are contaminated with outliers. At first, the attempt is to neutralize the effect of outliers by using the MVE methodology. As explained in the previous section, the data without outliers will lie inside the ellipse and the data contaminated with outliers will remain outside it. This cannot be displayed in multidimensional space. To make the MVE method more clear, we have considered two stocks and explain the same in two dimensional space. Two stocks such as State Bank of India (SBI) and Coal India Limited (CIL) from BSE-500 stock is selected between period November-1, 2011 to January-31, 2012. The ellipsoid is found out by applying the MVE method and is shown in Fig.5.1.The x-axis represents the stock value for SBI and y-axis represents the stock value for CIL. It is observed that some points are present outside the ellipse. It implies that on some days the stock values changes abruptly from its normal variation. This heavy fluctuation of stock value is primarily due to some unforeseen situations such as sudden fall of oil price or political crisis etc. The stock values on such days may be considered as outliers.

(a)

153

(b)

(c)

(d)

(e)

(a)Assuming 10 % of collected data are contaminated by outliers.

(b)Assuming 20 % of collected data are contaminated by outliers.

(c)Assuming 30 % of collected data are contaminated by outliers.

(d)Assuming 40 % of collected data are contaminated by outliers.

(e)Assuming 50 % of collected data are contaminated by outliers.

Fig.5. 1 Minimum volume ellipsoid for SBI and CIL stock data from November to January 2012

The weight factors associated with such data are calculated. These data are suitably modified by multiplying the weighting factor obtained using (5.22).

5.4.3. The problem approach

(a) Using Markowitz mean-variance model

The MVE is applied to the weekly price of the stocks and accordingly they are modified. The new data set is used to calculate the corresponding modified weekly returns. The expected return is calculated by taking the mean of the modified weekly returns and accordingly the corresponding risk is found out.

(b) Using our proposed prediction based mean-variance (PBMV) model

In this case also the MVE is applied to the weekly price and it is suitably modified. Using the new set of data, the weekly return is calculated mathematically. Then, the FLANN forecasting network is used to predict the future weekly return by taking the modified weekly return as input parameters. The modified weekly return is not directly used as input rather some statistical information such as moving averages, mode and median of the input parameters are considered as the input to the network. Then, it is expanded using Chebyshev functional expansion to transform the input information to nonlinear form. Evolutionary computation selectively chooses functionally expanded variables for effective prediction. The weights of the FLANN model has been efficiently trained using BFO algorithm. In addition, the input features are also weighted suitably and the weight factors are also obtained using BFO.

(c) Constraint portfolio optimization using MOEAs

By applying the two models, the risk and return of individual assets are found out. This process is repeated for all the assets. After estimating the return and risk of individual assets, the portfolio optimization is carried out by using some efficient MOEAs. Two MOEAs (MOPSO, MOBFO) based on non-dominating sorting and two based on decomposition (P-MOEA/D, B-MOEA/D) have been applied. The constraint handling issue has also taken into consideration in the optimization process.

5.4.4 The simulation results

In previous chapter we observed, under identical condition, the MOBFO algorithm gives the best possible solutions among all MOEAs. Hence in this section we have applied MOBFO to Heng-Seng and BSE-500 stock indices assuming 10%,20%,30%,40%,50% of the stock data are contaminated by outliers.
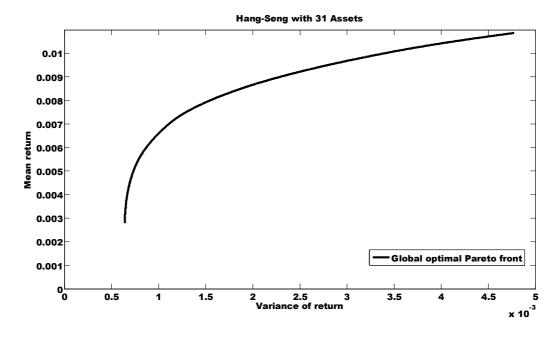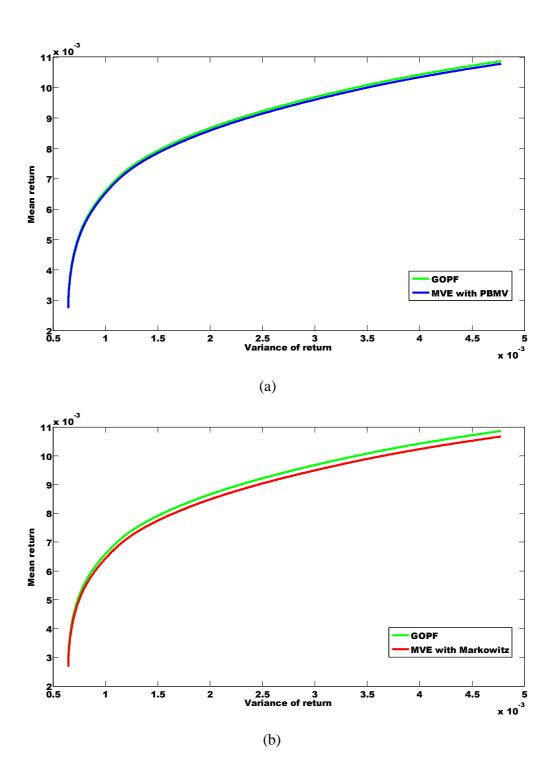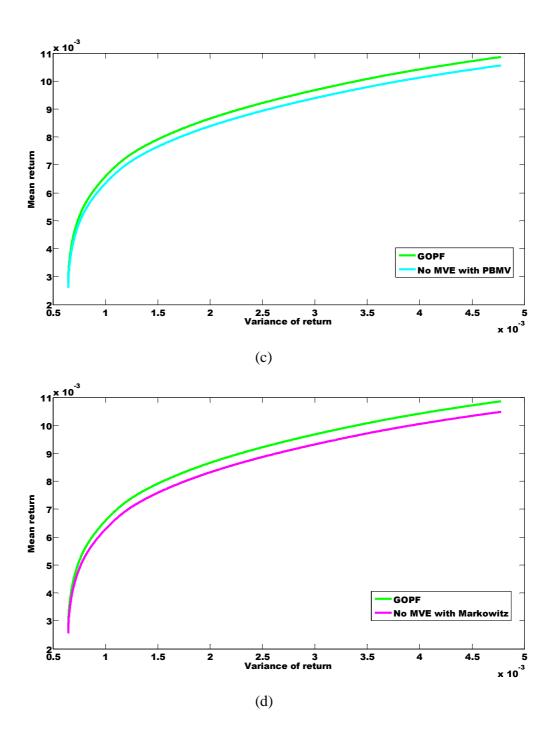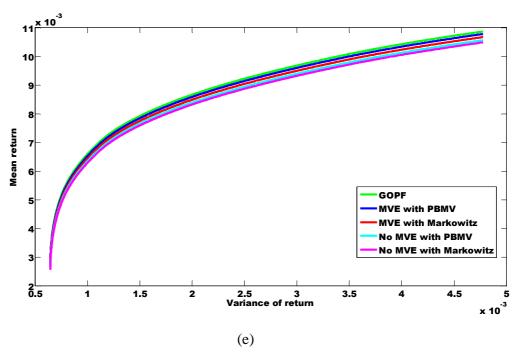

Fig.5.2. Global optimal Pareto front for Hang-Seng, stock indices

(a)

(b)

(c)

(d)

(e)

(a) applying MVE method and using the PBMV model

(b) applying MVE method and using Markowitz mean-variance model

(c) without applying MVE method and using the proposed PBMV model

(d) without applying MVE method and using Markowitz mean-variance model

(e) for all the four conditions

Fig.5.3. GOPF and Pareto front obtained by applying MOBFO to Hang-Seng stock assuming 20% of the data are contaminated with outliers.

The global optimal Pareto front (GOPF) corresponding to Hang-Seng stock   is depicted in Fig.5.2. From Fig.5.3, it is clear that the Pareto front obtained by MOBFO applying the MVE method and proposed PBMV model provide the best Pareto solution. The C metric is used to compare between this four different conditions as shown in Table-5.1.

Table-5.1.Comparison of results of C metric for MOBFO with different condition

	Without MVE and Markowitz	Without MVE and PBMV	With MVE and Markowitz	With MVE and PBMV
Without MVE and Markowitz	___	0.3988	0.3652	0.3211
Without MVE and PBMV	0.4523	0.4421	___	0.4012
With MVE and Markowitz	0.4672	___	0.4523	0.4242
With MVE and PBMV	0.5722	0.5421	0.5012	___

From the performance metric C, it is observed that MOBFO algorithm is giving better Pareto solution applying MVE and using PBMV model. The obtained results can be tested using six performance metrics and analyzing the Pareto front obtained. The statistical testing can also be performed for in depth analysis. The Pareto fronts obtained by assuming 20%, 30%, 40%, 50% of stock data contaminated by outliers are also shown in figure 5.4. It is seen that the results obtained are comparable to each other.
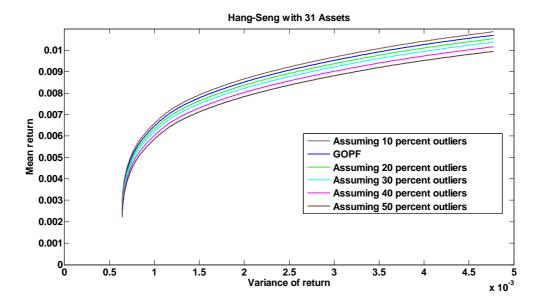

Fig.5.4.Pareto fronts obtained by applying MOBFO to Hang-Seng stock assuming 10%, 20%, 30%,40% and 50 % of the data contaminated with outliers.

Subsequently the MOBFO algorithm is applied to handle the cardinality constraint for DAX 100 stock applying MVE method and using PBMV model.
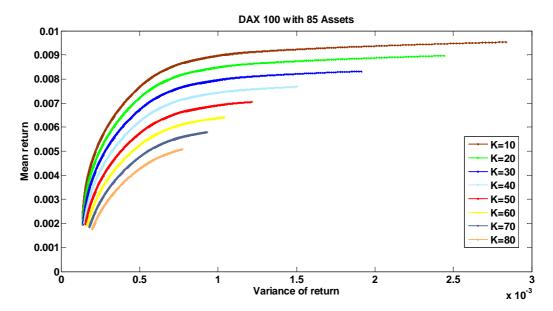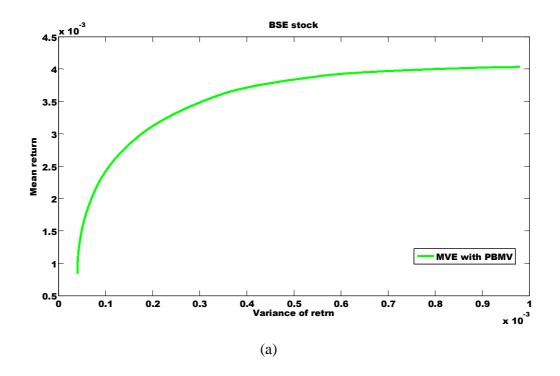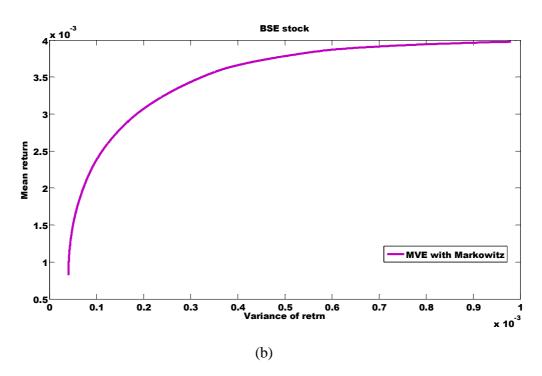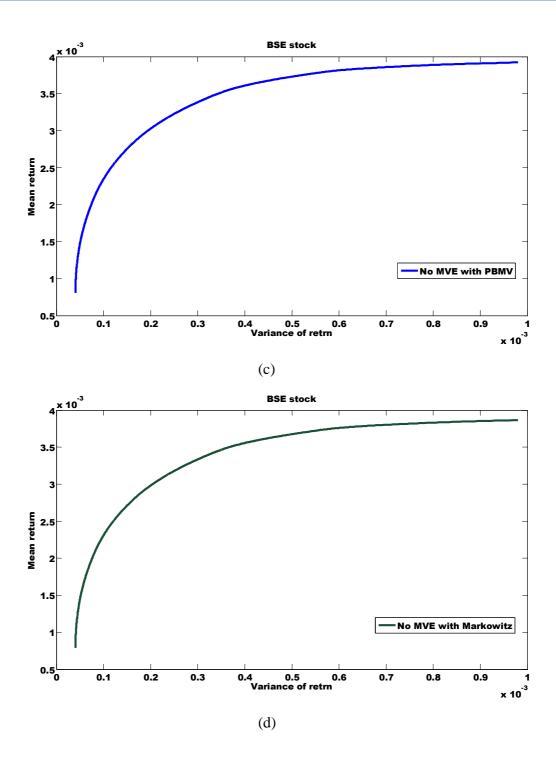

Fig.5.5.Pareto front for MOBFO for DAX 100 stock data by applying MVE method using PBMV models in the presence of cardinality.
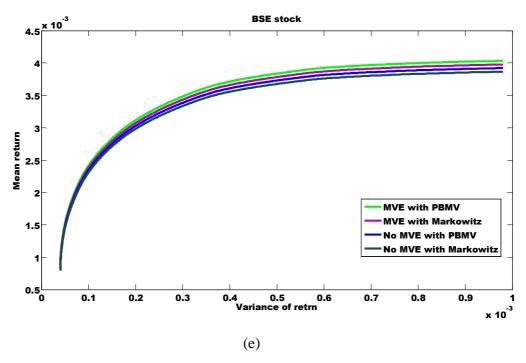
Thus, the MOBFO can handle cardinality constraint efficiently by applying this combination of MVE method and PBMV model.

The MOBFO algorithm is applied to 20 stocks of BSE where it is assumed that 10 percent of the stock is uncertain due to outliers. The Pareto front obtained with and without applying MVE method is shown in Figs.5.5.

(a)

(b)

(c)

(d)

(e)

(a) applying MVE method and using the PBMV model

(b) applying MVE method and using Markowitz mean-variance model

(c) without applying MVE method and using the proposed PBMV model

(d) without applying MVE method and using Markowitz mean-variance model

(e) for all the four conditions

Fig.5.6. The Pareto front obtained by applying MOBFO to BSE-500 stock

Similarly MOBFO can also be applied to other stock indices such as FTSE 100, S&P

100 and Nikkei 225 using both models. Similarly P-MOEA/D, B-MOEA/D, NS-MOPSO,

MOBFO can be applied to different markets by applying MVE and using PBMV model.

5.5 Conclusion

The Minimum volume ellipsoid (MVE) methodology is devised by using core set and Lagrange multipliers and is suitably applied to handle uncertainty present in the stock market data. The data with outliers are modified by multiplying appropriate weighting factor with the data. The FLANN network is chosen for predicting the desired parameters such as risk and return from the modified weighted data. Four MOEAs have been employed to obtain the final Pareto solution using this new estimated return and risk parameters. The experimental result reveals that the MOEAs are able to provide efficient Pareto solution in the presence of outliers in the stock data. In addition, the MOEAs provides better Pareto solution using proposed prediction based mean-variance (PBMV) model as compared to Markowitz mean-variance model.

It can be concluded that the proposed MVE method gives a quite satisfactory solution in the abrupt build-up of situations and exhibits good portfolio strategy. The implementation of the proposed model can also be done to a variety of benchmark data sets. The performance of proposed method can also be evaluated considering other real world constraints such as round-lot, turnover and trading.

Reference

[5.1] H. M. Markowitz, "Portfolio selection", Journal of Finance, vol. 7 no.1, pp. 77-91, 1952.

[5.2] T.J. Chang, S.C. Yang b, K.J. Chang, "Portfolio optimization problems in different risk measures using genetic algorithm", Expert Systems with Applications, vol. 36, issue 7, pp. 10529–10537, 2009.

[5.3] D. Lin, S. Wang, H. Yan, "A multiobjective genetic algorithm for portfolio selection problem", Proceedings of the ICOTA, Hong Kong, 15 –17 December, 2001.

[5.4] A.L. Soyster, "Convex programming with set-inclusive constraints and applications to inexact linear programming", Operations Research, vol. 21 no.5, pp.1154-1157, 1973.

[5.5] A. Ben-Tal, A. Nemirovski, Robust convex optimization, Mathematics of Operations Research,Vol. 23, pp.769–805, 1998.

[5.6] D. Goldfarb, G. Iyengar, "Robust portfolio selection problems", Mathematics of Operations Research, vol.28, no. 1, pp.1–38, 2003.

[5.7] D. Bertsimas, D. Pachamanova, "Robust multiperiod portfolio management in the presence of transaction costs", Computers & Operations Research, vol. 35, pp. 3–17, 2008.

[5.8] B.F. Moghadam, S.M. Seyedhosseini, "A particle swarm approach to solve vehicle routing problem with uncertain demand: a drug distribution case study", International Journal of Industrial Engineering Computations, pp. 55–64, 2010.

[5.9] E. Roghanian, A. Foroughi, "An empirical study of Iranian regional airports using robust data envelopment analysis", International Journal of Industrial Engineering Computations, vol.1,pp.65-72, 2010.

[5.10] M. Gharakhani, T. Taghipour, K.J. Farahani, "Robust multi-objective production planning", International Journal of Industrial Engineering Computations, pp.73-78, 2010.

[5.11] B. W. Silverman and D. M. Titterington, "Minimum covering ellipses". SIAM J. Sci. Statist. Comput., vol. 1, issue. 4, pp.401–409, 1980.

[5.12] Y. Zhang and L. Gao., "On numerical solution of the maximum volume ellipsoid problem", Technical report, Rice University, 2001.

[5.13] K. M. Anstreicher, "Improved complexity for maximum volume inscribed ellipsoids", SIAM Journal on Optimization, vol. 13, no. 2, 2002.

 [5.14] A. S. Nemirovskii, "On self–concordant convex–concave functions", Research Report 3/97, Optimization Laboratory Faculty of Industrial Engineering and Managament at Technion, Technion City, Haifa 32000, Israel, June 1997.

[5.15]Yu.E. Nesterov and A. S. Nemirovskii., "Interior Point Polynomial Methods in Convex Programming", SIAM Publications, Philadelphia, 1994.

[5.16] L.G. Khachiyan, "Rounding of polytopes in the real number model of computation", Mathematics of Operations Research, vol.21, no.2, pp. 307–320, 1996.

[5.17] Y. Zhang., "An interior-point algorithm for the maximum-volume ellipsoid problem", Technical Report TR98-15, Department of Computational and Applied Mathematics, Rice University, 1998.

[5.18] S. Boyd, L. Vandenberghe. Convex Optimization. Cambridge University Press, 2004.

[5.19] A. H. Sayed , Fundamentals of Adaptive Filtering, Wiley, NJ,2003.

[5.20].http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html

[5. 21]. http://in.finance.yahoo.com/q/hp?s=%5EHSI

[5.22] http://in.finance.yahoo.com/q/hp?s=%5EBSESN

[5.23] R. Marpnna, D. Martin and V. Yohai,"Roboust Stastics: Theory and methods", Wiley Series in Probability and Statics, 2006.

Chapter 6

Prediction Based Robust mean-variance Model for Constraint Portfolio Optimization

Chapter

6

In many situations, it is required to invest the money for future, but the relevant future data are not available. In addition, the present data are contaminated by outliers. Such complex problem needs an acceptable solution by involving robust prediction followed by efficient optimization method. For this robust prediction, the FLANN model trained with evolutionary computation is used as a predictor. The future return and risk of all assets have been predicted by FLANN using Markowitz model and prediction based mean-variance (PBMV) model. Then, constraint portfolio optimization is obtained using four efficient MOEAs. The experimental results reveal that, the proposed PBMV model in combination with MVE followed by efficient MOEAs give efficient portfolio strategy for future investment.

6.1 Introduction

The investment of funds in the presence of outliers is a very challenging and interesting problem. The handling of uncertainty has been reported by many researchers [6.1-6.8]. Tutuncu and Koenig [6.2] consider a box-type uncertainty structure for the mean and covariance matrix of the asset returns. They have solved the portfolio optimization problem by formulating it as a smooth saddle-point problem in this uncertain condition. Zhu and Fukushima [6.3] have demonstrated that the portfolio optimization problem can be formulated as linear or second-order cone programs by considering conditional value-at-risk (CVaR) for handling uncertainty. Huang *et al.* [6.4] have formulated the portfolio problem with uncertainty as a semi-definite program where only partial information on the exit time distribution function and the conditional distribution of portfolio return are available. De Miguel and Nogales [6.5] have proposed a novel approach for portfolio selection by minimizing certain robust estimators of portfolio risk. In their approach, robust estimation and portfolio optimization are performed by solving a single nonlinear program. Quaranta and Zaffaroni [6.6] studied a portfolio selection model in which the methodologies of robust optimization are used for the minimization of the conditional value at risk (CVaR) of assets. In the work of Seyed Jafar *et.al.* a new framework has been presented and the cardinality constrained portfolio problem is efficiently solved when all input parameters are subjected to uncertainty[6.7]. Bertsimas and Pachamanova [6.8] have studied the viability of different robust optimization approaches for multi-period portfolio selection.

However, in these robust optimization techniques the program dimension increases exponentially, as the number of assets present in the portfolio optimization increases.

Moreover, the problem becomes more challenging when many practical constraints are considered.

In many situations it is required to invest money in future where the future data are not available. In the recent times attention has been focused on future investment of the fund [6.9]. Kia-Hong Tee uses the n-degree lower partial moment (LPM) models and analyzes the effect of downside risk reduction on UK portfolio diversification and returns for managing funds in future.

In many situations, the future fund investment in the presence of outliers is important but difficult to solve. In the combined presence of these two conditions, the portfolio strategy becomes more challenging and is yet to be explored. In the present study, the first challenge of handling the outliers present in the input data is same as described in chapter 5. The minimum volume ellipsoid (MVE) which is formed by using core set and Lagrange multipliers, differentiate the data having outliers and without outliers [6.13], [6.18]. Then the weight factor associated with each uncertain datum is calculated by taking the appropriate parameters associated with the ellipsoid. The weight factor is designed in such a way that it does not change the data those are present inside the ellipsoid but suitably decreases the magnitudes of data which are present outside it. The weight factor is lowest for data far away from the center of the ellipsoid and vice-versa. Thereafter, the weight factors are multiplied with uncertain data to suitably modify it.

The second challenge of investing money in different assets requires a robust predictive algorithm. The FLANN which has been successfully applied in chapter 4 is used as the predictor. The literature survey reveals that the performance of FLANN is improved by providing some technical indicators of stock data instead of giving it directly [6.12].

Moreover evolutionary computation can be used to choose some of the expanded branch selectively to reduce the computational time by rejecting the branch having less contribution to the output [6.13]. In [6.13] it is shown that if the weight of each branch is updated by evolutionary computation it becomes less susceptible to local optima problem and also consumes less time to update the weight. In this chapter the FLANN is applied for prediction of risk and return of each asset using Markowitz and proposed PBMV models. The Pareto solutions of portfolio are found out by using four efficient MOEAs techniques.

6.2 Development of prediction based robust mean-variance model for constraint portfolio optimization

For the development of prediction based robust model, the minimum volume ellipsoid (MVE) method followed by FLANN using prediction based mean-variance (PBMV) model is applied.

6.2.1 Minimum volume ellipsoid

In this chapter also the same MVE approach is applied to mitigate the effect of outliers in the stock values.

6.2.2 Forecasting Model

A low complexity FLANN employing Chebyshev functional expansion as explained in Chapter 4 is used as a forecasting model. The forecasting potentiality of a network becomes efficient if fundamental analysis factors are used as inputs. The fundamental analysis is the study of economic, industry, and company conditions in an effort to determine the value of a company's stock. Ten technical indicators and five fundamental analysis factors are used as important parameters to study the future stock movement

efficiently. These ten indicators are explained in Section 6.2.3 followed by five fundamental analysis factors in section 6.2.4. Bacteria foraging optimization (BFO) algorithm is used for selecting some of the indicators and for updating the weights of the network. Each bacterium represents one weight of the forecasting model. A population of such bacteria represents the initial solutions of the model which are iteratively updated using the BFO principle by minimizing the mean squares error (MSE) as the cost function. The input to the network is nonlinearly expanded using Chebyshev functional expansions.

6.2.3 Technical indicators

The technical indicators have been used [6.13] as inputs to FLANN model to improve the performance of prediction. These technical indicators have been obtained from past stock market data. Technical indicators are important features to predict the future price levels, or the general price direction. A brief explanation of each indicator defined in [6.13] is provided in Table-6.1. These are:

(a) Simple Moving Average (SMA)

It is the simple average of the values by taking a window of the specified period.

(b) Exponential Moving Average (EMA)

It is also an average of the values in the specified period but it gives more weightage to recent values and thus it is more close to the actual values.

(c) Accumulation/Distribution Oscillator (ADO)

It measures money flow in the security. The ADO aims to measure the ratio of buying to selling by comparing price movements of a period to the volume of that period. Also it has been calculated for each day.

(d) Stochastic Oscillator (STO)

The stochastic Oscillator is a momentum indicator that shows the location of the current close relative to the high/low range over a set of number of periods. Closing levels which are consistently near the top of the range indicate accumulation (buying pressure) and those near the bottom of the range indicate distribution (selling pressure).

(e) On Balance Volume (OBV)

It is a momentum indicator that relates volume to price change.

(f) Williams %R (WILLIAMS)

It is a momentum indicator that measures overbought/oversold levels.

(g) Relative Strength Index (RSI)

It calculates the internal strength of the security.

(h) Price Rate of Change (PROC)

The PROC indicator displays the difference between the current price and a previous closing price for a given time period ago.

(i) Closing Price Acceleration (CPACC)

It is the acceleration of the closing prices during the given period.

(j) High Price Acceleration (HPACC)

It is the acceleration of the high prices in the given period.

Table 6.1. The list of technical indicators with their formulae used as inputs

Technical Indicators	Formula
Simple Moving Average (SMA)	$$\frac{1}{N}\sum_{i=1}^{N}x_i$$ N = No. of Days x_i = today's price
Exponential Moving Average (EMA)	$(P\times A)+(\text{Previous EMA}\times(1\text{-}A))$; A=2/(N+1) P – Current Price, A- Smoothing factor, N-Time Period
Accumulation/Distribution Oscillator (ADO)	$$\frac{(\text{C.P - L.P}) - (\text{ H.P - C.P})}{(\text{H.P - L.P})\times (\text{Period's Volume})}$$ C.P – Closing Price, H.P – Highest price, L.P – Lowest price
Stochastic Oscillator (STO)	$\%K = \dfrac{(\text{Today's Close - Lowest Low in K period})}{(\text{Highest High in K period - Lowest Low in K period})}\times100$ $\%D$ = SMA of $\%K$ for the Period.
On Balance Volume (OBV)	If Today's Close > Yesterday's Close OBV = Yesterday's OBV + Today's Volume If Today's Close < Yesterday's Close OBV = Yesterday's OBV - Today's Volume
WILLIAM's %R	$\%R = \dfrac{(\text{Highest High in n period - Today's Close})}{(\text{Highest High in n period - Lowest Low in n period})}\times100$
Relative Strength Index (RSI)	$\text{RSI} = 100 - \dfrac{100}{1 + (U/D)}$ U= total gain/n, D= total losses/n, n = number of RSI period
Price Rate Of Change (PROC)	$\dfrac{(\text{Today's Close - Close X-period ago})}{(\text{Close X-period ago})}\times100$
Closing Price Acceleration (CPACC.)	$\dfrac{(\text{ Close Price - Close Price N-period ago})}{(\text{Close Price N-period ago})}\times100$
High Price Acceleration (HPACC)	$\dfrac{(\text{ High Price - High Price N-period ago})}{(\text{High Price N-period ago})}\times100$

6.2.4 Fundamental analysis factors

In addition to technical indicators which depend on the past value of the data other features known as fundamental analysis factors are also used as inputs. These are generally macroeconomic parameters which affect the stock market. Five fundamental factors used in the study are crude oil prices, United States' GDP growth rate, corporate dividend rates, federal interest rates and commodity price index (CPI).

6.3 Simulation studies

In this chapter the algorithms are coded in MATLAB and were run on a PC with Intel Core 2 Duo 3.0 GHz with 4 GB RAM.

6.3.1 Data collection

The data for Hang-Seng and Nikkei-225 stock indices were obtained from OR-Library which is maintained by Prof. Beasley [6.14]. The data of PORT-1 and PORT-5 correspond to weekly prices between March 1992 and September 1997. The numbers of different assets for the above two benchmark indices are 31 and 225 respectively. The daily closing price, opening price, lowest value, and highest value on the day and the total volume of these stocks and weekly closing price are also available in [6.15]. Similarly these daily and weekly stock information for BSE have been collected from [6.16]. For the present study the daily and weekly value of 20 stocks from Heng-Seng, 20 from DAX 100, 20 from FTSE 100, 20 from S&P 100, 20 from Nikkei 225 and 20 stocks from the BSE-500 stocks between December 2008 to January 2012 have been collected.

6.3.2 The problem approach

(a) The procedure for MVE

We assume 50 percent of the collected stock data of all the markets are having outliers. The MVE approach discussed in Chapter 5 is applied to nullify the effect of uncertainty which modifies the unexpected data by multiplying it with appropriate weight factors. In Markowitz model the MVE method is applied to all the daily closing price, opening price, lowest value, highest value on the day and the total volume of stocks. But in the proposed PBMV model, the MVE is applied only to the daily closing prices of stocks.

(b) Using Markowitz mean-variance model

In this model the daily closing price, opening price, lowest value, highest value on the day and the total volume of stocks present with outliers are collected and are modified using the MVE. Ten technical and five fundamental indicators defined in Table-6.1 are calculated using the collected data. These indicators are employed as inputs to the FLANN forecasting model. The FLANN is used to predict the closing price of the stock for future time. Evolutionary computation technique is used to select some proper indicators for achieving effective prediction. The weights of the FLANN are also trained with the evolutionary computation based method. In the present simulation, the BFO algorithm is chosen to train the network parameters. The output of the FLANN structure provides future stock values. From this predicted closing price, the stock returns for a time horizon are computed. The return after a specified time is the mean of calculated returns. The individual risk of each stock and the risk between each pair of stocks are obtained from the variance and covariance of the return time series.

(c) Using proposed prediction based mean-variance (PBMV) model

In this model, the unexpected weekly closing stock values are modified using minimum volume ellipsoid (MVE) method. Then the weekly past returns are calculated from this modified weekly closing value of stock. The inputs used for the FLANN structure are financial variables such as the moving average, mode and median of the calculated past return, and the right combinations are selected using the BFO tool. The weights of the network are also trained by BFO. The output of the FLANN gives future returns. This process is repeated for all the assets to predict the corresponding returns after a fixed time. The individual risk of each stock and the risk between each pair of stocks are obtained from the covariance matrix of the time series of errors of prediction. The individual risk of each stock (variance) is found out by from the diagonal elements of the matrix.

(d) Constraint portfolio optimization using MOEAs

Using two different models the future risk and return of individual asset are found out. This process is repeated for all assets. After estimating the return and risk of all assets for a fixed time the portfolio optimization with some practical constraints are carried out by using NS-MOPSO, MOBFO, P-MOEA/D and B-MOEA/D multiobjective optimization algorithms.

6.3.3 Experimental results

In this section we have applied MOBFO to Heng-Seng and BSE-500 stock indices for future portfolio strategies. It is assumed that 10% of stock data are contaminated by outliers.
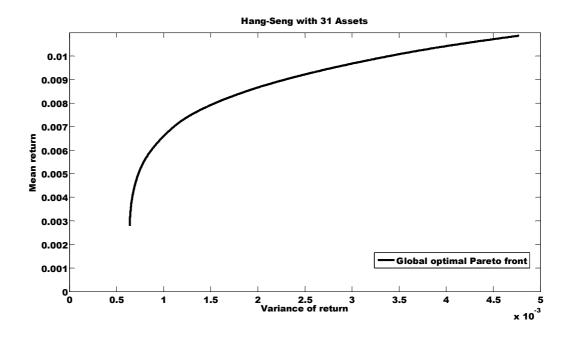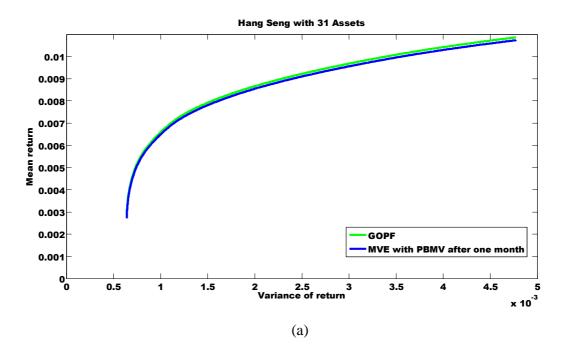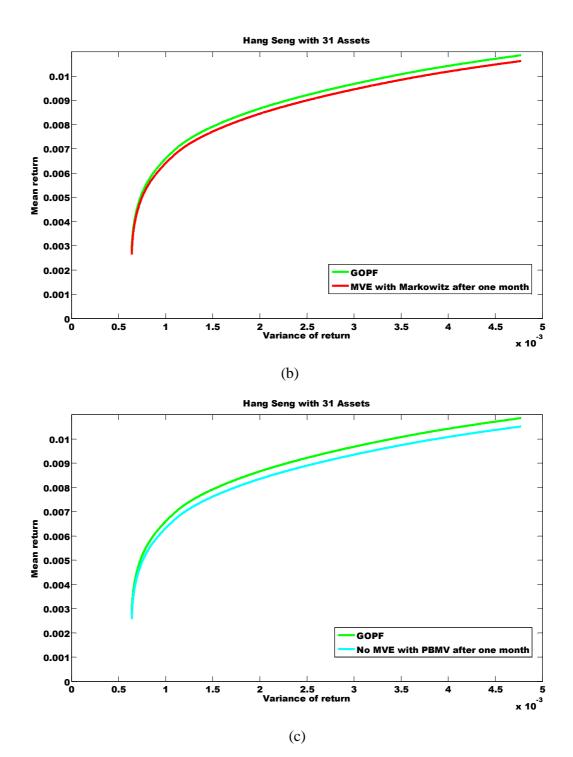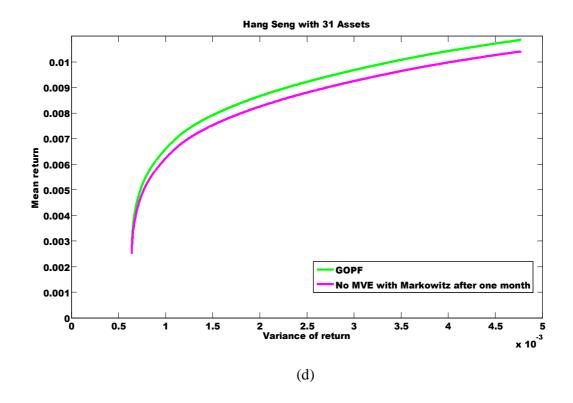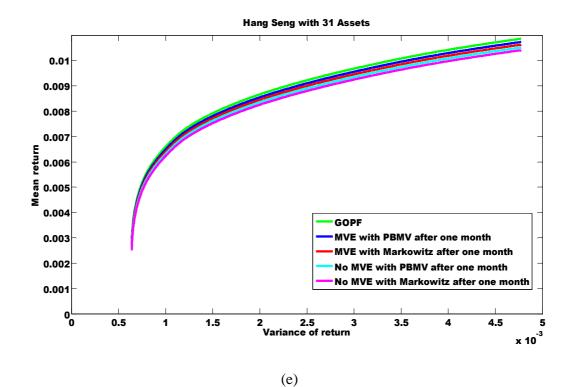
Fig.6.1. Global optimal Pareto front for Hang-Seng, stock indices

The global optimal Pareto front (GOPF) corresponding to Hang-Seng stock is depicted in Fig.6.1.

(a)

181

(b)

(c)

Hang Seng with 31 Assets

(d)

Hang Seng with 31 Assets

(e)

(a) applying MVE method and using the PBMV model

(b) applying MVE method and using Markowitz mean-variance model

(c) without applying MVE method and using the proposed PBMV model

(d) without applying MVE method and using Markowitz mean-variance model

(e) for all the four conditions

Fig.6.2. GOPF and Pareto front obtained by applying MOBFO to Hang-Seng stock after one month assuming 10 % of stock contaminated by outliers
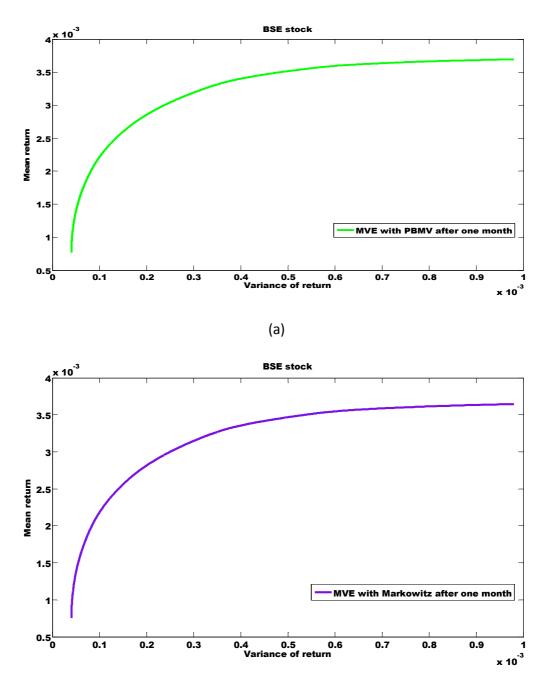
It is evident that the MOBFO applying MVE method and proposed PBMV model is providing better solutions in comparison to other, as its Pareto front is closer to the standard efficient frontier. Further, the performance of them is assessed using C metrics. The C metric is demonstrated in Table-6.2.
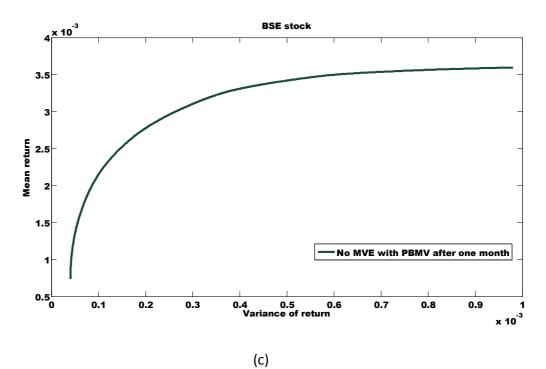
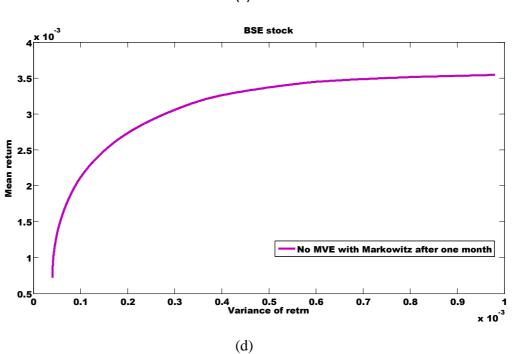Table-6.2. Comparison of results of C metric for MOBFO with different condition

|	Without MVE and Markowitz after one month	Without MVE and PBMV after one month	With MVE and Markowitz after one month	With MVE and PBMV after one month
Without MVE and Markowitz after one month	___	0.4132	0.3942	0.3423
Without MVE and PBMV after one month	0.4632	0.4321	___	0.4102
With MVE and Markowitz after one month	0.4732	___	0.4611	0.4321
With MVE and PBMV after one month	0.5911	0.5522	0.5213	___
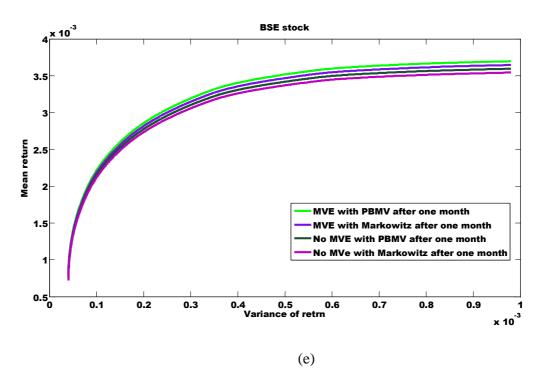
It clearly shows that most of the solutions obtained by MOBFO applying MVE and PBMV model dominate the solutions obtained by others. The obtained results can also be tested using six performance metrics. The statistical testing can also be performed for in depth analysis. Similarly, the MOBFO can also be applied to other stock indices such as DAX 100, FTSE 100, S&P 100 and Nikkei 225 using both models. Similarly P-MOEA/D, B-MOEA/D, NS-MOPSO can be applied to different markets by applying MVE and using PBMV model. These algorithms can also be used to handle cardinality constraint efficiently by applying this combination of MVE method and PBMV model.

The MOBFO algorithm is applied to 20 stocks of BSE where it is assumed that 10 percent of the stock is uncertain due to outliers and money has to invest after one month. The Pareto fronts obtained with and without applying MVE method are shown in Figs.6.3.

(a)

(b)

185

(c)

(d)

(e)

(a) applying MVE method and using the PBMV model after one month

(b) applying MVE method and using Markowitz mean-variance model after one month

(c) without applying MVE method and using the proposed PBMV model after one month

(d) without applying MVE method and using Markowitz mean-variance model after one month

(e) all the four conditions

Fig.6.3. The Pareto front obtained by applying MOBFO to 20 stocks (assets) from BSE-500 stock indices after one month assuming 10% of stock data are contaminated by outliers.

It is evident that the MOBFO algorithm, applying MVE method and proposed PBMV model is providing better solutions in comparison to others as it cover more risk-return area. It provides more option to the portfolio manager for investing money after one month.

6.4 Conclusion

The portfolio optimization issue for future time when the corresponding data are not available and the present available data are uncertain has been studied in this Chapter. A subset of 20 stocks from Heng-Seng and BSE-500 indices between December 2008 to January 2012 have been selected for obtaining portfolio strategy after one month, that is on February 2012. The effect of outliers in the stock data has been minimized using the MVE method. The MOBFO algorithms have been applied using both Markowitz mean-variance and prediction based mean-variance (PBMV) models. The proposed prediction based mean-variance (PBMV) portfolio optimization model in combination with minimum volume ellipsoid (MVE) method is observed to be effectively mitigating the effect of outliers for future investment. Experimental results demonstrate that the proposed PBMV portfolio optimization model outperforms the conventional Markowitz model for investing in future.

References

[6.1] S.J. Sadjadi, M. Gharakhani, E. Safari, "Robust optimization framework for cardinality constrained portfolio problem", Applied Soft Computing, vol. 12, issue. 1, pp. 91–99, 2012

[6.2] R.H. Tutuncu, M. Koenig, "Robust asset allocation, Annals of Operations Research", vol. 132, no. 1-4, pp. 157–187,2004.

[6.3] S. Zhu, M. Fukushima,"Worst-case conditional value-at-risk with application to robust portfolio management", Operations Research, vol. 57,no. 5, pp. 1155–1168, 2009.

[6.4] D. Huang, F.J. Fabozzi, M. Fukushima, "Robust portfolio selection with uncertain exit time using worst-case VaR strategy", Operations Research Letters, vol.35, pp.627–635, 2007.

[6.5] V. De Miguel, F.J. Nogales, "Portfolio selection with robust estimation", Operations Research, vol.57, no.3, pp.560–577, 2009.

[6.6] A.G. Quaranta, A. Zaffaroni, "Robust optimization of conditional value at risk and portfolio selection", Journal of Banking & Finance, vol.32, issue. 10, pp.2046–2056, 2008.

[6.7] Seyed Jafar Sadjadi, Mohsen Gharakhani, Ehram Safari, "Robust optimization framework for cardinality constrained portfolio problem", Applied Soft Computing,vol.12, pp- 91–99,2012.

[6.8] D. Bertsimas, D. Pachamanova, "Robust multiperiod portfolio management in the presence of transaction costs", Computers & Operations Research, vol. 35, pp. 3–17, 2008.

[6.9] Kai-Hong Tee,"The effect of downside risk reduction on UK equity portfolios included with managed futures funds", International Review of Financial Analysis, vol.18, pp-303–310,2009.

[6.10] K. M. Anstreicher,"Prediction based complexity for maximum volume inscribed ellipsoids", SIAM Journal on Optimization,vol.13, no.2, 2002.

[6.11] S. Boyd, L. Vandenberghe. Convex Optimization. Cambridge University Press, 2004.

[6.12] Ritanjali Majhi, G. Panda, G. Sahoo, "Development and performance evaluation of FLANN based model for forecasting of stock markets", Expert Systems with Applications, vol. 36, pp.6800-6808, 2009.

 [6.13] Ritanjali Majhi, G. Panda, G. Sahoo, "On the development of prediction based adaptive models for efficient prediction of stock indices using clonal-PSO (CPSO) and PSO techniques", Int. J. Business Forecasting and Marketing Intelligence, Vol. 1, No. 1, PP.50-67, 2008.

[6.14] http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html

[6.15] http://in.finance.yahoo.com/q/hp?s=%5EHSI

[6.16]http://in.finance.yahoo.com/q/hp?s=%5EBSESN

Chapter 7

Conclusion and Future Work

Chapter

7

In this chapter the overall contributions of the thesis are reported. The future research

problems are also outlined for further investigation on the same/ related topics.

7.1. Conclusion

The conclusion of the overall thesis is presented in this section and some of the major contributions achieved in the thesis are reported in the next section. Some future research problems related to the topics of the thesis and which may be attempted by interested readers are outlined in the last section.

Two novel multiobjective evolutionary algorithms (MOEAs) based on non-dominated sorting and two algorithms based on decomposition are proposed and suitably applied to portfolio optimization problem with budget, floor, ceiling and cardinality constraints by formulating it as a multiobjective optimization problem. On examining the performance metrics, it is observed that the proposed MOBFO approach is capable of identifying best possible Pareto solutions maintaining adequate diversity. The Pareto front obtained by MOBFO is closer to the standard efficient front covering more risk return area. The Sign test and Wilcoxon signed rank test are also performed to show the superiority of MOBFO over others. In terms of computational time, the P-MOEA/D is found to be the fastest among other such algorithms used in the thesis. All the four algorithms have been found to be potential candidates for solving constrained portfolio optimization problem. From the simulation results, it is evident that the investor does not have to invest money on all the available assets rather to invest in fewer assets i.e. approximately 10 percent of available assets to explore wide risk-return area. The portfolio manager has the option to make a tradeoff between risk and return for different cardinality constraints to decide on the portfolios according to the requirement.

A novel prediction based mean-variance (PVMV) model has been proposed and four MOEAs have been employed to solve the portfolio optimization problem. In the PBMV

model, the return is first predicted with a low complexity single layer neural network. The performance of four MOEAs in solving portfolio optimization problem using the proposed and Markowitz mean-variance models has been evaluated. From the simulation results it is observed that the proposed PBMV model is capable of identifying good Pareto solutions by maintaining adequate diversity. The comparison of results shows that the performance of PBMV is comparable to that of well known Markowitz mean-variance model.

In order to reduce the effects of uncertainty of the stock market data (outliers), the Minimum volume ellipsoid (MVE) methodology has been proposed. It has been observed through the experimental and theoretical studies that the MVE methodology is robust for handling outliers. It has been seen from the study that this method has effectively found out appropriate weight factors for all the data and those have been used to modify the contaminated data. The FLANN network has been used to predict risk and return for further processing. Experimental results reveal that the MOEAs provide good Pareto solution using this new predicted return and risk parameters. Moreover, the simulation results have shown that the MOBFO algorithm provides the best possible solutions among all MOEAs for uncertain market conditions. Furthermore, the MOBFO algorithm using PBMV model and MVE method has also been found to be robust in the presence of the cardinality constraint.

It is a challenging problem to find suitable portfolio strategy for investment of money for the future where the relevant future data are not available and the present data are uncertain due to the presence of outliers. To solve this problem the MVE methodology in combination with the PBMV model followed by FLANN based forecasting are chosen. Then the MOBFO algorithm is used to provide the best Pareto solutions.

7.2. Contribution Achieved

Some key contributions achieved in this thesis are listed below.

- Two novel MOEAs, based on non-dominated sorting such as nondominated sorting multiobjective particle swarm optimization (NS-MOPSO) algorithm and multiobjective bacteria foraging optimization (MOBFO) have been proposed to solve the constrained portfolio optimization problem by formulating it as a multiobjective minimization problem. Similarly two algorithms based on decomposition such as P-MOEA/D and B-MOEA/D have been also proposed and suitably applied to solve this problem by viewing it as a multiobjective maximization problem.

- Developed an prediction based mean-variance (PBMV) model incorporating prediction strategy as an useful alternative of Markowitz mean-variance model for solving constraint portfolio optimization problem.

- Developed multiobjective swarm intelligence based robust portfolio management method to neutralize the effect of outliers using minimum volume ellipsoid (MVE) based approach.

- Developed improved and robust swarm intelligence techniques for future investment of fund, with non availability of future data as well as uncertainty of the present data due to the presence of outliers.

7.3 Suggestions for future work

The work carried out in the present thesis can further be extended in many directions.

- To incorporate advanced local search operators into the proposed MOEAs algorithms which is expected to allow better exploration and exploitation of the search space.

- To investigate on the strengths and weaknesses of non-dominated sorting or decomposition based MOEAs. To develop new MOEAs based on any other algorithmic framework which may be better suited for portfolio optimization problem.

- To handle outliers in the financial time series, S-estimates, the minimum covariance determinate estimate and one-step reweighting method may be used as an useful alternative to minimum volume ellipsoid method dealt in the thesis.

- To evaluate the performance of proposed method considering other real world constraints like round-lot, turnover and trading.

- To test the performance of proposed MOEAs with other realistic data to validate its potentiality in addition to the benchmark problems.

- To apply the MOEAs to other financial applications such as asset allocation, risk management, option pricing etc.

  .

Sudhansu Kumar Mishra

Date of Birth:28th May.1979

Correspondence

PhD scholar, Department of Electronics and Communication Engineering,

National Institute of Technology,Rourkela,Odisha,India-769008

Ph:+91-9439536462 (M), +91-8986822135 (M), 06682-220493(Phone),

Email: sudhansu.nit@gmail.com, sudhansu_nit@yahoo.co.in

Qualification

- Ph.D. (continuing)

   National Institute of Technology, Rourkela, India-769008

- M.Tech.

   National Institute of Technology, Rourkela, India-769008 [2003]

- B.E (Electrical)

   Fakir Mohan University, Odisha, India [2001]

- +2 (Science)

   Council of Higher Secondary edution, Odisha, India [1996]

- 10th

   Board of Secondary Education, Odisha, India [1994]

Professional Experience

1. Assistant Professor, at Birla Institute of Technology, Mesra, India from Dec 2011 to till date.

   2. Sr. Lecturer, at C.V.R.C.E, Bhubaneswar,Odisha,India from May 2005 to June 2008.

3. Lecturer, at P.I.E.T, Rourkela, Odisha, India from August 2001 to July 2003.

Permanent Address

At: Ladarpali Chawk, PO: Attabira

Dist: Bargarh, PIN:768027, Odisha, India

Publications

- 05 journal Articles
- 15 Conference Paper