Expert Systems with Applications 38 (2011) 10161–10169

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem

Hanhong Zhu [a], Yi Wang [b], Kesheng Wang [a,*], Yun Chen [c]

[a] Department of Production and Quality Engineering, NTNU, N7491 Trondheim, Norway
[b] Nottingham Business School, NTU, Nottingham NG1 4BU, UK
[c] School of Public Economics and Adiministration, SHUFE, 200433 Shanghai, China

ARTICLE INFO

Keywords:
Expert system
Swarm Intelligence (SI)
Particle Swarm Optimization (PSO)
Portfolio management (PM)
Optimal portfolio
Sharp Ratio (SR)

ABSTRACT

One of the most studied problems in the financial investment expert system is the intractability of portfolios. The non-linear constrained portfolio optimization problem with multi-objective functions cannot be efficiently solved using traditionally approaches. This paper presents a meta-heuristic approach to portfolio optimization problem using Particle Swarm Optimization (PSO) technique. The model is tested on various restricted and unrestricted risky investment portfolios and a comparative study with Genetic Algorithms is implemented. The PSO model demonstrates high computational efficiency in constructing optimal risky portfolios. Preliminary results show that the approach is very promising and achieves results comparable or superior with the state of the art solvers.

1. Introduction

Portfolio management is one of the most studied topics in finance. The problem is concerned with managing the portfolio of assets that minimizes the risk objectives subjected to the constraint for guaranteeing a given level of returns. This paper deals with the mean–variance portfolio selection, which is formulated in a similar way as Markowitz did (Elton, Gruber, & Padberg 1976; Markowitz, 1952; Steinbach, 2001). Markowitz introduced the concepts of Modern Portfolio Theory (MPT). His theory has revolutionized the way people think about portfolio of assets, and has gained widespread acceptance as a practical tool for portfolio optimization. But in some cases, the characteristics of the problem, such as its size, real-world requirements (Campbell, Huisman, & Koedijk 2001; Gennotte, 1986; Louis, Jason, & Josef, 1999; Perold, 1984; Zhou & Li, 2000), very limited computation time, and limited precision in estimating instance parameters, may make analytical methods not particularly suitable for tackling large instances of the constrained mean–variance model. Therefore researchers and practitioners have to resort to heuristic techniques that are able to find high-quality solutions in a reasonable amount of time.

Due to the complexity and the instantaneity of the portfolio optimization model, applying meta-heuristic algorithms to portfolio selection and optimization is a good alternative to meet the challenge. Some remarkable studies have been presented to solve asset selection problem. Many meta-heuristic techniques (Chang, Meade, Beasley, & Sharaiha, 2000) have been applied in portfolio selection such as Genetic Algorithms, tabu search and simulated annealing for finding the cardinality constrained efficient frontier. Some hybrid techniques (Gaspero, Tollo, Roli, & Schaerf, 2007) have been applied in portfolio management such as local search and quadratic programming procedure. Preliminary results show that the approach is very promising and achieves results comparable or superior to the traditional solvers. Pareto Ant Colony Optimization (Doerner, Gutjahr, Hartl, Strauss, & Stummer, 2004) has been introduced as an especially effective meta-heuristic for solving the portfolio selection problem and compares its performance to other heuristic approaches (i.e., Pareto Simulated Annealing and the Non-Dominated Sorting Genetic Algorithm) by means of computational experiments with random instances. An artificial neural network model with the Particle Swarm Optimization algorithm (Giovanis, 2009) has been applied to portfolio management and shows the flexibility of hybrid models, such as the superiority in forecasting performance, in relation to the traditional econometric methodology, like Ordinary least square and ARCH-GARCH estimations. Fuzzy Analytic Hierarchy Process (AHP) (Tiryaki & Ahlatcioglu, 2009) has been combined with the portfolio selection problem to model the uncertain environments. A hybrid Genetic Algorithm approach (Jeurissen & van den Berg, 2005) has been investigated for tracking the Dutch AEX-index, it focused on building a tracking portfolio with minimal tracking error.

However, these approaches have some shortcomings in solving the portfolio selection problem. For example, fuzzy approach usu-

* Corresponding author. Tel.: +4773597119.
E-mail addresses: hanhong@stud.ntnu.no (H. Zhu), yi.wang@ntu.ac.uk (Y. Wang), Kesheng.wang@ntnu.no (K. Wang), chenyun@mail.shufe.edu.cn (Y. Chen).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2011.02.075

ally lacks learning ability (Chan, Wong, Tse, Cheung, & Tang, 2002); Artificial neural network approach has over-fitting problem and is often easy to trap into local minima (Casas, 2001); while as Genetic Algorithms (Alba & Troya, 1999) are applied to harder and bigger problems there is an increase in the time required to converge for finding adequate solutions.

In order to overcome these drawbacks, PSO model is introduced to solve the portfolio selection and optimization problem. PSO is a population based stochastic optimization technique developed in 1995 (Kennedy & Eberhart, 1995). The underlying biological metaphor for developing PSO algorithm is inspired by social behavior of bird flocking or fish schooling. PSO has become a popular optimization method as they often succeed in finding the best optimum by global search in contrast with most common optimization algorithms. In comparison with the dynamic programming, PSO allows the users to get the sub-optimal solution while dynamic programming cannot. It is very important for the portfolio selection and optimization problem.

There are very few studies on PSO, especially all most none of them deal with the performance comparison with other approaches for solving portfolio optimization problems. The main contribution of this study is to employ a PSO algorithm for portfolio selection and optimization in investment management. Asset allocation in the selected assets is optimized using a PSO based on Markowitz's theory. Using the PSO, an optimal portfolio can be determined. The rest of the paper is organized as follows. Section 2 describes models for portfolio optimization. In Section 3, the background of PSO and previous work are summarized. The PSO model for optimal portfolio is also discussed. In order to test the efficiency of the proposed PSO solver, a simulation and comparative study is performed in Section 4. Final conclusions and future research are drawn in Section 5.

2. Models for portfolio optimization (PO)

One of the fundamental principles of financial investment is diversification where investors diversify their investments into different types of assets. Portfolio diversification minimizes investors' exposure to risks, and maximizes returns on portfolios. It can be referred to as a multi-objective optimization problem.

There are many methods to solve the multi-objectives optimization problems. One basic method is to transfer the multi-objective optimization problems into a single-objective optimization problem. We can divide these methods into two different types. The first alternative is to select one important objective function as the objective function to optimize while the rest of objective functions are defined as constrained conditions. The second alternative is to construct only one evaluation function for optimization by weighting the multiple objective functions. The first method is defined by Markowitz mean–variance model (Markowitz, 1952).

2.1. Type 1: Markowitz mean–variance model

The first method is defined by Markowitz mean–variance model. In Markowitz mean–variance model, the security selection of risky portfolio construction is considered as one objective function and the mean return is defined as one of the constraints. This model is described as:

$$\mathrm{Min}\sum_{i=1}^{N}\sum_{j=1}^{N}w_i w_j \sigma_{ij} \quad (1)$$

$$\text{Subject to } \sum_{i=1}^{N}w_i r_i = \mathrm{R}^*, \quad (2)$$

$$\sum_{i=1}^{N}w_i = 1, \quad (3)$$

$$0 \leqslant w_i \leqslant 1 \quad i = 1,\ldots,N, \quad (4)$$

where N is the number of different assets, σ_{ij} is the covariance between returns of assets i and j, w_i is the weight of each stock in the portfolio, r_i is the mean return of stock i and R^* is the desired mean return of the portfolio.

2.2. Type 2: single objective function model

The second method is to construct only one evaluation function for modeling a portfolio optimization problem. Efficient Frontier and Sharpe Ratio models are described as the following:

2.2.1. Efficient Frontier

We can find the different objective function values by varying desired mean return R^*. The standard practice introduces a new risk aversion parameter $\lambda \in [0,1]$. With this new parameter λ, the model can be described as one objective function:

$$\mathrm{Min}\lambda\left[\sum_{i=1}^{N}\sum_{j=1}^{N}w_i w_j \sigma_{ij}\right] - (1-\lambda)\left[\sum_{i=1}^{N}w_i r_i\right] \quad (5)$$

$$\text{Subject to}\sum_{i=1}^{N}w_i = 1, \quad (6)$$

$$0 \leqslant w_i \leqslant 1 i = 1,\ldots,N. \quad (7)$$

When λ is zero, the model maximizes the mean return of the portfolio, regardless of the variance (risk). In contrast, when λ equals unity, the model minimizes the risk of the portfolio regardless of the mean return. So the sensitivity of the investor to the risk increases as λ increasing from zero to unity, while it decreases as λ approaches zero.

Each case with different λ value would have a different objective function value, which is composed of mean value and variance (risk). Tracing the mean return and variance intersections with different parameter λ, we can draw a continuous curve that is called an efficient frontier in the Markowitz theory (Markowitz, 1952). Since each point on an efficient frontier curve indicates an optimum, and this indicates the portfolio optimization problem is a multi-objective optimization problem. The introducing parameter λ makes the problem to be transfer into a single-objective function problem.

2.2.2. Sharpe Ratio model

Instead of focusing on the mean variance efficient frontier, we seek to optimize the portfolio Sharpe Ratio (SR) (Sharpe, 1966). The Sharpe Ratio combines the information from mean and variance of an asset. It is quite simple and it is a risk-adjusted measure of mean return, which is often used to evaluate the performance of a portfolio. It is described with the following equation:

$$SR = \frac{R_p - R_f}{StdDev(p)}, \quad (8)$$

where p is the portfolio, R_p is the mean return of the portfolio p, R_f is the test available rate of return of a risk-free security (i.e. the interest rate on a three-month U.S. Treasury bill). StdDev(p) is the standard deviation of R_p, in other words, it is a measure of risk of the portfolio. Adjusting the portfolio weights w_i, we can maximize the portfolio Sharpe Ratio in effect balancing the trade-off between maximizing the expected return and at the same time minimizing the risk. In this study, Sharpe Ratio is used in the PSO in order to find the most valuable portfolio with good stock combinations.

3. PSO for portfolio optimization

3.1. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a population based stochastic optimization technique inspired by social behavior of bird flocking. It belongs to Swarm Intelligence (SI), which originates from the study of natural creatures living in a group. Each individual possess little or no wisdom, but by interacting with each other or the surrounding environment, they can perform very complex tasks as a group.

PSO could be explained well in an imagined scenario: a group of birds are flying in an area to look for food, and there's only one piece of food in this area. Each bird in the group doesn't know the exact location of the food, but they are aware of the distance between the food and themselves. In this way, the easiest way to find the food is to follow the one who is closest to the food.

When it comes to the algorithm of PSO, it starts with the initialization of a population of random particles, each of which is associated with a position and a velocity. The velocities are adjusted according to the historical behavior of each particle and its neighbors while they fly through the search space. The positions are updated according the current position and the velocities at the next step. Therefore, the particles have a tendency to fly towards the better and better search area over the search process course.

In other words, a brief description of how the algorithm works is as follow: Initially, some particle is identified as the best particle in a neighborhood of particles, based on its fitness. All the particles are then accelerated in the direction of this particle, but also in direction of their own best solutions that they have discovered previously.

Occasionally the articles will overshoot their target, exploring the search space beyond the current best particles. All particles have the opportunity to discover better particles in route, in which case the other particles will change direction and head towards the new best particle. Since most functions have some continuity, chances are that a good solution will be surrounded by equally good, or better, solutions. By approaching the current best solution from different direction in search space, the chances are good that these neighboring solutions will be discovered by some of the particles.

The basic concept of PSO lies in accelerating each particle toward its *pbest* which was achieved so far by that particle, and the *gbest* which is the best value obtained so far by any particle in the neighborhood of that particle, with a random weighted acceleration at each time step.

Each particle tries to modify its position using the following information:

- The current positions ($\vec{X}(t)$),
- The current velocities ($\vec{V}(t)$),
- The distance between the *pbest* and the current position ($\vec{P}_i - \vec{X}(t)$),
- The distance between the *gbest* and the current position ($\vec{P}_G - \vec{X}(t)$).

3.2. Fitness function

Fitness function is a critical factor in the PSO method. Every particle in the PSO's population has a fitness value. A particle moves in solution space with respect to its previous position where it has met the best fitness value *pbest*, and the neighbor's previous position where neighbor has met the best fitness value *gbest*. In this paper, considering whether the portfolio is restricted or not, the Sharpe Ratio is used as a single objective function. This is defined as:

$$f_P = SR = \frac{\sum_{i=1}^{N} w_i r_i - R_f}{\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij}} \tag{9}$$

where f_p is the fitness value of particle p.

At every step, a particle's personal best position and the best neighbor in the swarm are updated if an improvement in any of the best fitness values is observed.

3.3. Particles movement

In the algorithm of PSO, each solution is represented by a particle in the search space. Each particle has its position, velocity, and fitness value. At each iteration, every particle moves towards its personal best position and towards the best particle of the swarm found so far. The particle movement is dependent on its current velocity and the velocity change is defined as:

$$\vec{v}_{ij}(t+1) = w\vec{v}_{ij}(t) + c_1 r_1[\vec{p}_{ij}(t) - \vec{x}_{ij}(t)] + c_2 r_2[\vec{p}_{gi}(t) - \vec{x}_{ij}(t)] \tag{10}$$

where index j is the dimension number of particle i, t is the iteration sequence, and c_1 and c_2 are positive constant parameters called acceleration coefficients. They are responsible for controlling the maximum step size, r_1 and r_2 are random numbers between $(0, 1)$. w is a constant, and $\vec{v}_{ij}(t+1)$ is particle i's velocity on the jth dimension at iteration t + 1. $\vec{v}_{ij}(t)$ is particle i's velocity on the jth dimension at iteration k. $\vec{x}_{ij}(t)$ is particle i's position on the jth dimension at iteration k.. $\vec{p}_{ij}(t)$ is the historical individual best position of the swarm. Finally, the new position of particle i, $\overrightarrow{x_{ij}}(t+1)$, is calculated by Eq. (11).

$$\vec{x}_{ij}(t+1) = \vec{x}_{ij}(t) + \vec{v}_{ij}(t+1) \tag{11}$$

Fig. 1 illustrates the updating process and the movement of a particle graphically while the flow chart of PSO algorithm is depicted in Fig. 2. The detailed algorithm can be found in the paper written by Bratton & Kennedy, 2007.

To improve the performance of PSO, the parameter can be adjusted. For example, the constant w can be replaced by Eq. (12), and also the constant c_1 and c_2 by Eqs. (13) and (14). However, in the experiment for a large scale portfolio (i.e. 49 stocks), the result didn't significantly be improved.

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter \tag{12}$$

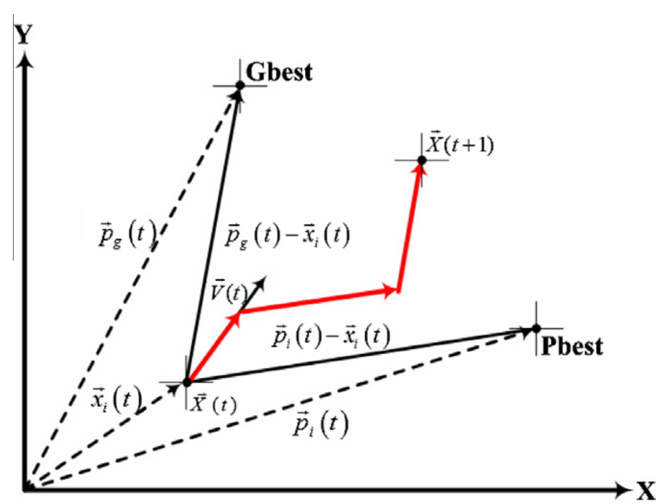


Fig. 1. Particle's updating process of PSO in 2D space.

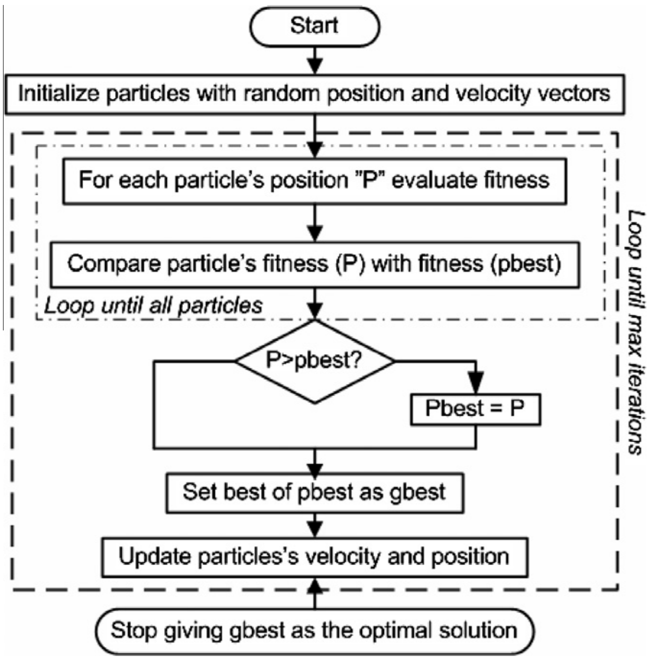


Fig. 2. The flowchart depicting the general algorithm of PSO.

$$c_1 = c_{1_{\max}} - \frac{c_{1_{\max}} - c_{1_{\min}}}{iter_{\max}} \times iter \tag{13}$$

$$c_2 = c_{2_{\max}} - \frac{c_{2_{\max}} - c_{2_{\min}}}{iter_{\max}} \times iter \tag{14}$$

3.4. Constraint satisfaction

There are two types of risky portfolios (Benninga, 2000): unrestricted and restricted. Unrestricted risky portfolios do not have constraints on the short selling of stocks. Investors can choose to sell a stock that the investor does not own based on the condition that the investor must buy it back after a period of time at a lower price hopefully. In other words, for unrestricted risky portfolios assets could have negative weights. As for restricted risky portfolios, they place constraints on the short selling of portfolios' underlying equities, and require that all underlying assets must have positive weights. Both unrestricted and restricted optimal risky portfolios must also satisfy another constraint, namely, the total weights of all assets must sum up to 1. The goal of an optimal risky portfolio is to find the optimal combination of all assets in order to achieve the maximum Sharpe Ratio.

(1) The restricted portfolio optimization problem for a risky portfolio with N assets is defined as:

$$MaxSR = Max \frac{\sum_{i=1}^{N} w_i r_i - R_f}{\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij}} \tag{15}$$

$$\text{Subject to} \sum_{i=1}^{N} w_i = 1, \tag{16}$$

$$0 \leqslant w_i \leqslant 1 i = 1, \ldots, N. \tag{17}$$

(2) The unrestricted portfolio optimization problem is defined as:

$$MaxSR = Max \frac{\sum_{i=1}^{N} w_i r_i - R_f}{\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij}} \tag{18}$$

$$\text{Subject to} \sum_{i=1}^{N} w_i = 1, \tag{19}$$

$$i = 1, \ldots, N. \tag{20}$$

As the number of assets in the risky portfolio increases, construction of an optimal risky portfolio becomes an increasingly high-dimensional optimization problem with a variety of constrains.

If the risky portfolio is restricted, then the *pbest* and *gbest* are evaluated by Eqs. (15)–(17). On the other hand, the *pbest* and *gbest* are evaluated using Eqs. (18)–(20). Whenever a particle flies to a new position in the search space, all the constraints on the portfolio must be satisfied in order to ensure a valid movement within the search space. When the terminating condition has been fulfilled, a particle with global optimum or near the optimum portfolio may be found.

4. Experiments and discussion

The PSO experiments for the portfolio optimization has been performed on three unrestricted risky portfolio cases, such as 8 stocks, 15 stocks and 49 stocks, and on three restricted risky portfolio cases with the same numbers of stocks. Table 1 shows the results of these 6 portfolios using three different approaches: PSO model, GA and VBA solver respectively. All stocks are selected from the Shanghai Stock Exchange 50 Index (the SSE 50 Index). Individual stock's historical daily returns are selected from 1 may 2009 to 3 April 2009. Unrestricted portfolios do not have constraints on short selling. In other words, the proportion of an asset in the portfolio could be negative or greater than 1.

Table 1
Six portfolios' results of PSO solver, GA solver and Excel solver.

Approach	Item	Portfolio					
		8 Stocks		15 Stocks		49 Stocks	
		Unrestricted (%)	Restricted (%)	Unrestricted (%)	Restricted (%)	Unrestricted (%)	Restricted (%)
PSO solver:	ER	1.14	0.72	15.43	0.84	152.19	0.95
	SD	4.22	2.90	30.91	2.76	111.04	4.30
	Sharpe Ratio	19.84	17.83	48.96	26.73	136.78	15.06
GA solver:	ER	0.60	0.53	0.53	0.57	1.86	0.22
	SD	4.17	2.63	6.71	2.47	11.53	2.73
	Sharpe Ratio	7.24	12.42	3.48	18.96	13.56	−2.99
VBA solver:	ER	1.03	0.76	1.81	0.87	3.02	0.91
	SD	3.78	3.17	4.24	2.94	2.89	3.92
	Sharpe Ratio	19.31	17.55	35.70	26.31	94.02	12.90
Risk free		0.03	0.02	0.03	0.01	0.03	0.04

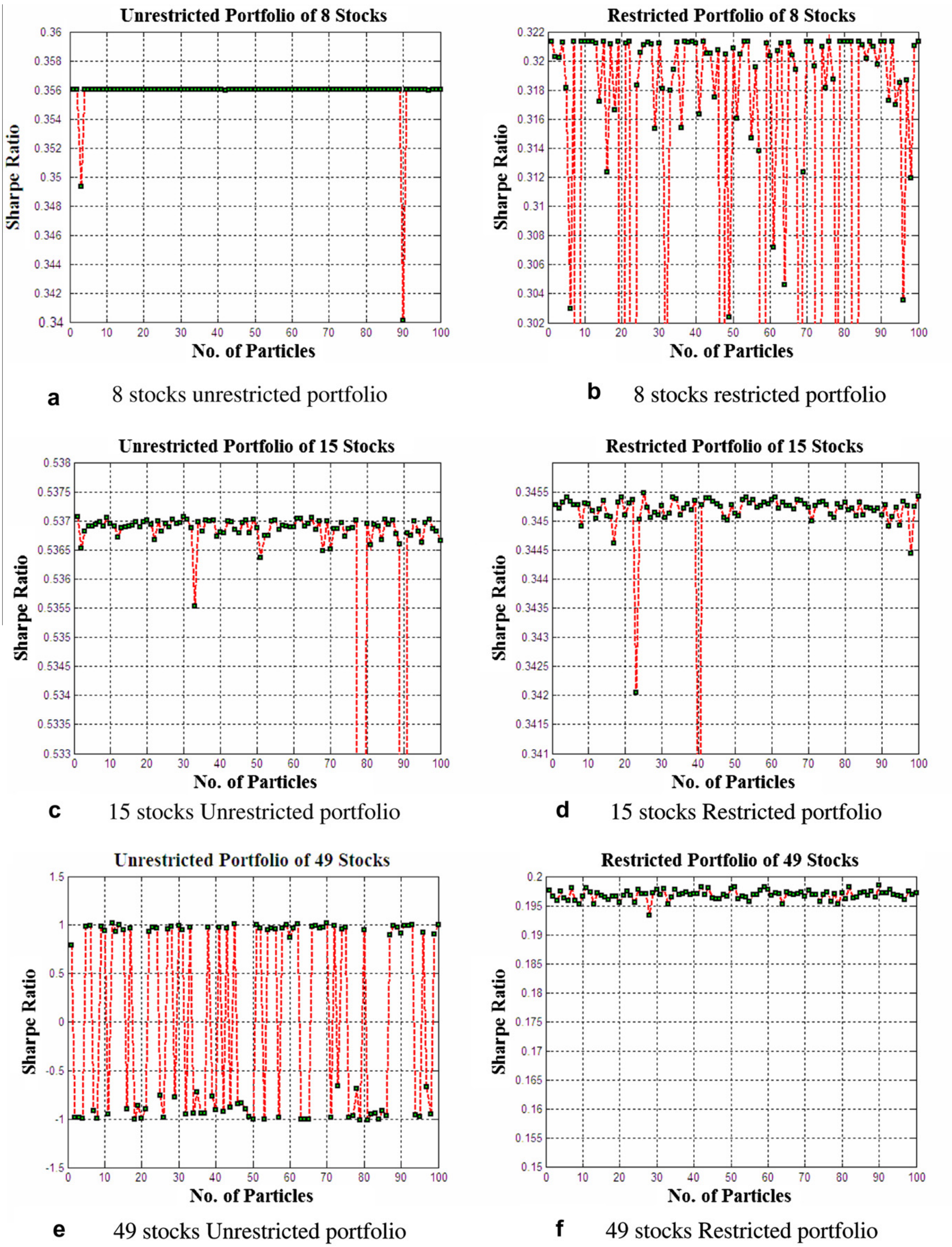
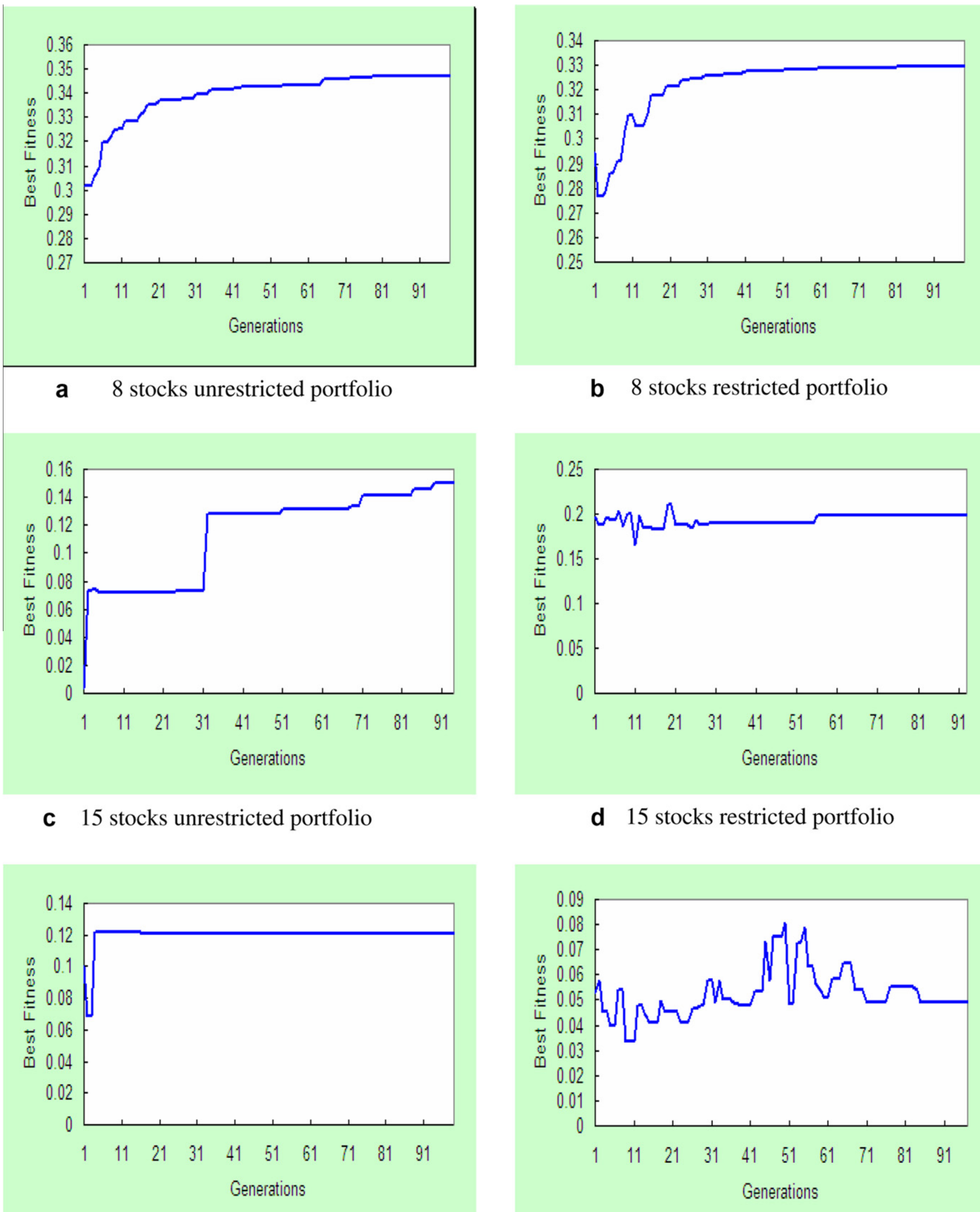


Fig. 3. PSO particle's updating process.

A GA is a stochastic optimization method based on the mechanisms of natural selection and evolution. In GAs, searches are performed based on a population of chromosomes representing solutions to the problem. A population starts from random values



a 8 stocks unrestricted portfolio

b 8 stocks restricted portfolio

c 15 stocks unrestricted portfolio

d 15 stocks restricted portfolio

e 49 stocks unrestricted portfolio

f 49 stocks restricted portfolio

Fig. 4. Evolutionary process in GeneHunter Wang, 2005.

and then evolves through succeeding generations. During each generation a new population is generated by propagating a good solution to replace a bad one and by combining or mutating existing solutions to construct new solutions. GAs have been theoreti-

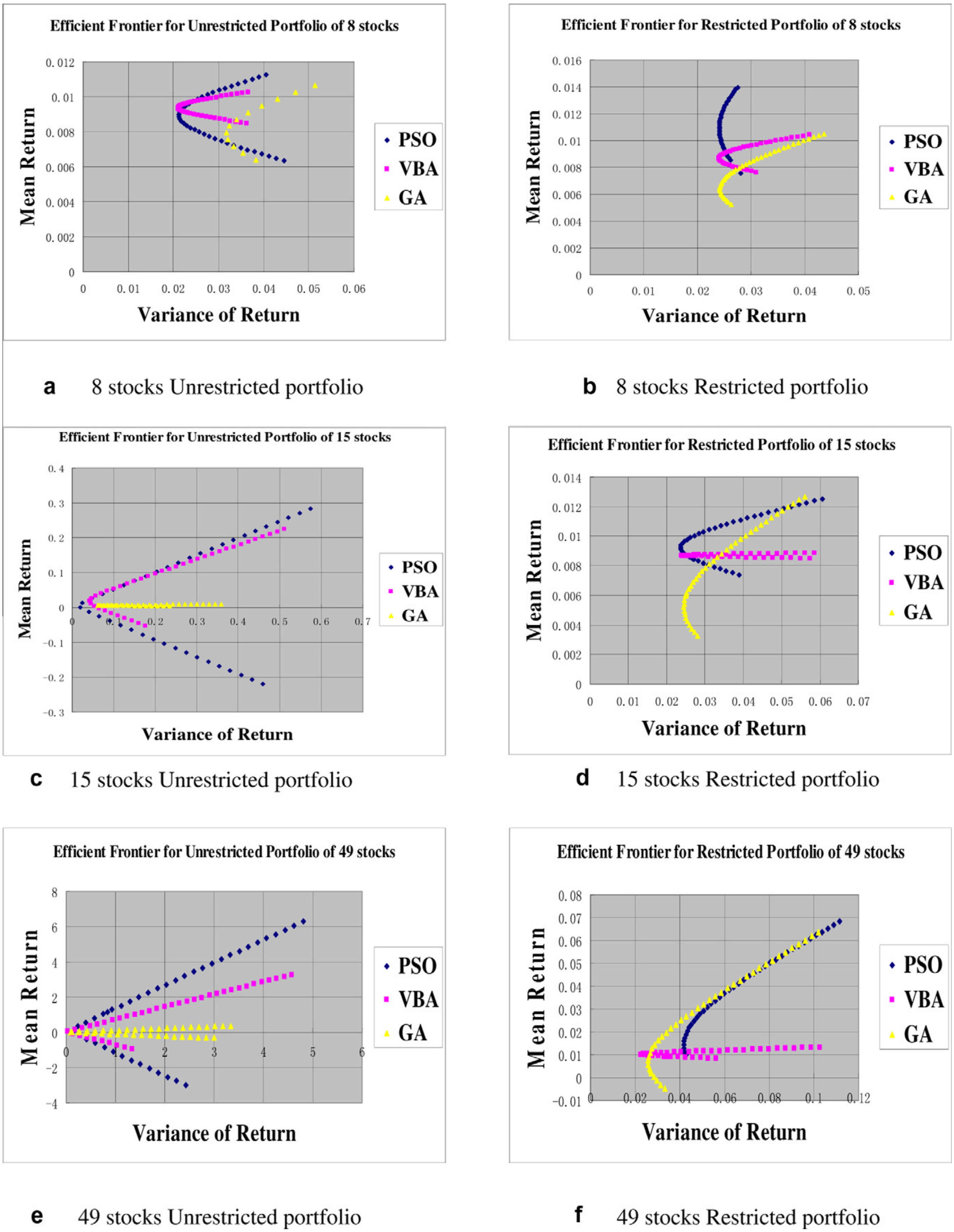


Fig. 5. The efficient frontier of the portfolio gotten from PSO, GA and VBA solver.

cally and empirically proven robust for identifying solutions to combinatorial optimization problems. GAs conduct an efficient parallel exploration of the search space, while only requiring minimum information on the objective function to be optimized. GAs also can be an alternative approach for portfolio optimization. In order to evaluate the performance of PSO model, we need to compare PSO with Genetic Algorithm solver.

In the experiments, the PSO Solver has been developed using Matlab as software development tool. The GA solver has been developed using the software development tool GeneHunter (Wang, 2005). Meanwhile, we also compare the result of them with the result of a traditional method using VBA (Visual Basic Application) solver.

The compositions of the optimal risky portfolios developed by PSO, GA and VBA for the 6 portfolios are shown in the Table 1. We can find that the values of Sharpe Ratio obtained by PSO are 19.84% (8 stocks with unrestricted), 17.83% (8 stocks with restricted), 48.96% (15 stocks with unrestricted), 26.73% (15 stocks with restricted), 136.78% (49 stocks with unrestricted), and 15.06% (49 stocks with restricted) respectively. These values are all better than the values obtained from the GA and VBA solver.

The updating process of the PSO solver for optimizing the 6 portfolios and with the termination condition 100 steps is showed in Fig. 3, where 100 particles are used and Sharpe Ration is selected as a fitness function. Comparing Fig. 3 (a) with (b), most of particles in unrestricted portfolio of 8 stocks converge to the best fitness value within 100 steps. But it does not happen in the restricted portfolio's PSO searching process. In the case of unrestricted and restricted portfolio of 15 stocks, seeing Fig. 3 (c) and (d), we can find that most of particles can convergence to the fitness value. However, when the number of the stocks get larger, for example, more than 49 stocks, Fig. (f) shows almost all particles in restricted portfolio could convergence to the fitness value, It is also not happen for unrestricted portfolio in Fig. (e). Regarding to the strategy to select number of stocks for optimization, it is suggested that for both restricted and unrestricted portfolio, we need to select 15 stocks in order to get the stability and efficiency of the PSO searching process.

In GA solver, the following parameters are set up for these 6 portfolios: Population size = 100; chromosome length = 32-bit; crossover rate = 0.9; mutation rate = 0.01; generation gap = 0.98; and termination condition = 100 generations. The GA evolutionary process is shown in Fig. 4. From these figures, we can indicate two factors as the following: (1). the convergence of the unrestricted portfolios' evolutionary process seems to be better than the restricted portfolios because portfolios' fitness function of unrestricted portfolio has fewer constraints; (2) the tendency of convergence will be slower when the numbers of assets increase. Therefore the suggested numbers of assets could be around 15, which is also corresponding to the suggestion from PSO solver mentioned above.

The termination condition is very important for the finding the optimal risky portfolio, because we have to make balance between efficiency and precision. According to many testing, in the PSO solution, the termination condition is 100 steps for the portfolios of 8 stocks, 200 steps for the portfolios of 15 stocks, and 1000 steps for the portfolios of 49 stocks. In the GA solution, the termination condition is 100 generations for the portfolios of 8 stocks, 300 generations for the portfolios of 15 stocks, and 2000 generations for the portfolios of 49 stocks.

Taking the set of 6 optimal portfolios obtained by PSO, GA and VBA solvers, their efficient frontier are traced out in Fig. 5. These curves show that the efficient frontiers obtained by PSO solver are almost above the others comparing with GA and VBA. It means that we can get higher mean return under the same or lower risk using the POS solver. However, there is an exception case in the Fig. 5(d) and we can see when the variance is greater than 0.05, the efficient frontier obtained by GA is better than PSO and VBA solvers. Comparing GA with VBA, we also note that when the number of stocks is 8, the VBA solver is better than GA. But for 49 stocks, the GA is significantly better than VBA solver.

Based on the experimental results on unrestricted and restricted portfolios, we could draw a conclusion that the performance of PSO approach is better than both the GA and the traditional VBA solver generally. PSO solver clearly shows the good efficiency and effectiveness of solving high-dimensional constrained optimization problems.

The experimental data is from ''http://finance.yahoo.com/9/hp?s=000016.ss''. The data has been selected from SSE 50 index, and the time period is from 1 May 2009 to 3 April 2009. The Risk Free has been selected subjectively. ER is ''Expected Return'', and SD is ''Standard Deviation''.

5. Conclusion

A fundamental principle of financial investments is diversification where investors diversify their investments into different types of assets. Portfolio diversification minimizes investors' exposure to risks, and maximizes returns on portfolios. The paper focuses on solving the portfolio optimization problem in finance investment management. A meta-heuristic Particle Swarm Optimization method has been developed to optimize investment portfolios, where the objective functions and constraints are based on both the Markowitz model and the Sharp Ratio model.

The PSO algorithm bears similarity to other biologically inspired optimizing algorithms. Like the GA, it is population-based, it is typically initialized with a population (swarm) of random encodings of solutions, and search proceeds by updating these encodings over a series of generations (iterations). Unlike the GA, PSO has no explicit selection process as all particles persist over time. Instead a memory in the form of *gbest/pbest* is substituted for selection.

In order to make a valid comparison with other methods, different test problems were solved and the results obtained when compared with the results of Genetic Algorithms (GA), Visual Basic for Applications (VBA) demonstrated the superiority of the PSO algorithm.

The key learning mechanisms in the PSO algorithm are driven by a metaphor of social behavior that good solutions uncovered by one member of a population are observed and copied by other members of the population. Of course, these learning mechanisms abound in business and other social settings. Good business strategies, good product designs, and good theories stimulate imitation and subsequent adaptation. Particle swarm algorithms will be a successful optimization tool in a variety of applications, and has clear potential for application to financial modeling.

Future research may be conducted to further investigate the application of some derived models or hybrid models of PSO to other investment strategy problems, for example tracking the index and so on. Another further investigation may be put on methods for improving the efficiency of the PSO solver for large portfolios in investment management.

References

Alba, E., & Troya, J. M. (1999). A survey of parallel distributed genetic algorithms. *Complexity, 4*(4), 31–52.

Benninga, S. (2000). *Financial modeling* (Second ed.). Massachusetts: The MIT Press.

Bratton, D., Kennedy, J. (2007). *Defining a Standard for Particle Swarm Optimization, in Swarm Intelligence Symposium*, 2007. SIS 2007. IEEE, 2007, pp. 120–127.

Campbell, R., Huisman, R., & Koedijk, K. (2001). Optimal portfolio selection in a value-at-Risk framework. *Journal of Banking & Finance, 25*, 1789–1804.

Casas, C.A. (2001). Tactical Asset Allocation: An Artificial Neural Network Based Model, *Proceedings of International Joint Conference on Neural Networks*, pp. 1811–1816.

Chan, M. C., Wong, C. C., Tse, W., Cheung, B., Tang, G. (2002). *Artificial intelligence in portfolio management*, H. Yin et al. (Eds.): IDEAL 2002, Lecture Notes in Computer Science, Vol. 2412, pp. 403-409.

Chang, T. J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations research, 27*, 1271–1302.

Doerner, K., Gutjahr, W., Hartl, R., Strauss, C., & Stummer, C. (2004). Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals of Operations Research, 131*, 79–99.

Elton, E. J., Gruber, M. J., & Padberg, M. W. (1976). Simple criteria for optimal portfolio selection. *The Journal of Finance, 31*, 1341–1357.

Gaspero, L.D., Tollo, G., Roli, A., Schaerf, A. (2007). Hybrid metaheuristics for portfolio selection problems, *MIC 2007–Metaheuristics International Conference*. Montreal, June 2007.

Gennotte, G. (1986). Optimal portfolio choice under incomplete information. *The Journal of Finance, 41*, 733–749.

Giovanis, E. (2009). *The Arbitrage Pricing Theory and the Capital Asset Pricing Models and Artificial Neural Networks Modeling with Particle Swarm Optimization (PSO)*, SSRN eLibrary (http://ssrn.com/abstract=1351249).

Jeurissen, R., van den Berg, J. (2005). *Index tracking using a hybrid genetic algorithm*. Computational Intelligence Methods and Applications, 2005 ICSC Congress.

Kennedy, J., Eberhart, R. (1995). Particle swarm optimization, *IEEE International Conference on Neural Networks – Conference Proceedings*, Vol. 4, pp. 1942–1948.

Louis, K. C. C., Jason, K., & Josef, L. (1999). On portfolio optimization: Forecasting covariances and choosing the risk model. *The Review of Financial Studies, 12*, 937–974.

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance, 7*, 77–91.

Perold, A. F. (1984). Large-Scale portfolio optimization. *Management Science, 30*, 1143–1160.

Sharpe, W. F. (1966). Mutual fund performance. *The Journal of Business, 39*, 119–138.

Steinbach, M. C. (2001). Markowitz Revisited: Mean–variance models in financial portfolio analysis. *SIAM Review, 43*, 31–85.

Tiryaki, F., & Ahlatcioglu, B. (2009). Fuzzy portfolio selection using fuzzy analytic hierarchy process. *Information Science, 179*, 53–69.

Wang, K. (2005). *Applied computational intelligence in intelligent manufacturing systems*. Advanced Knowledge International.

Zhou, X. Y., & Li, D. (2000). Continuous-time mean–variance portfolio selection: A stochastic LQ framework. *Applied Mathematics and Optimization, 42*, 19–33.