Symbiotic Multi-swarm PSO for Portfolio Optimization

Ben Niu[1], Bing Xue[1], Li Li[1], and Yujuan Chai[2]

[1] College of Management, Shenzhen University,
Shenzhen, Guangdong 518060, China
[2] Faculty of Science, McMaster University,
Hamilton, Ontario L8S4L8, Canada
drniuben@gmail.com

Abstract. This paper presents a novel symbiotic multi-swarm particle swarm optimization (SMPSO) based on our previous proposed multi-swarm cooperative particle swarm optimization. In SMPSO, the population is divided into several identical sub-swarms and a center communication strategy is used to transfer the information among all the sub-swarms. The information sharing among all the sub-swarms can help the proposed algorithm avoid be trapped into local minima as well as improve its convergence rate. SMPSO is then applied to portfolio optimization problem. To demonstrate the efficiency of the proposed SMPSO algorithm, an improved Markowitz portfolio optimization model including two of the most important limitations are adopted. Experimental results show that SMPSO is promising for this class of problems.

Keywords: Symbiotic PSO, particle swarm, portfolio optimization**.**

1 Introduction

Portfolio Optimization (PO), also known as mean-variance optimization (MVO), is risk management tool which allows you to construct optimal portfolios considering the trade-off between market risk and expected returns.

PO problem is NP-hard and non-linear with many local optima. Mathematical programming methods have been applied to this problem for a long time [1, 2, 3]. Nowadays, a number of different heuristic algorithms have been proposed for solving this problem, including genetic algorithms (GA) [4, 5], simulated annealing [6], neural networks [7] and others [8, 9, 10].

However most of the PO models used in those pioneer works may often be considered too basic, as it ignores many of the constrains, such as the transaction fee and whether short sale is permitted, and the upper and the lower bounds of proportion of each asset in the portfolio. In this work, we use a modified PO model considering the transaction costs and no short sales. The main motivation of this study is to employ an improved multi-swarm cooperative PSO (MCPSO) for the modified PO model.

MCPSO was firstly proposed by B. Niu in 2005[11], which is inspired by the phenomenon of symbiosis in natural ecosystems, where many species have developed cooperative interactions with other species to improve their survival. MCPSO has been successfully applied in many problems, including function optimization [11],

D.-S. Huang et al. (Eds.): ICIC 2009, LNAI 5755, pp. 776–784, 2009.

neural networks training[12], fuzzy modeling designing[13] etc. In this paper we will apply an improved MCPSO, i.e. symbiotic multi-swarm particle swarm optimization (SMPSO) to find efficient portfolio by solving the PO model.

The rest of the paper is organized as follows. Section 2 gives a review of PSO and a description of the proposed algorithm SMPSO. Section 3 describes portfolio optimization model. Section 4 gives the detailed experimental studies. Finally, conclusions are drawn in Section 5.

2 PSO and SMPSO

2.1 Particle Swarm Optimization (PSO)

The basic PSO is a population based optimization tool, where the system is initialized with a population of random solutions and the algorithm searches for optima by updating generations. In PSO, the potential solutions, called particles, fly in a *D*-dimension search space with a velocity which is dynamically adjusted according to its own experience and that of its neighbors.

The position of the *i*th particle is represented as $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, where $x_{id} \in [l_d, u_d]$, $d \in [1, D]$, l_d, u_d are the lower and upper bounds for the *d*th dimension, respectively. The rate of velocity for particle i is represented as $\vec{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, is clamped to a maximum velocity vector $\vec{v}_{\max}$, which is specified by the user. The best previous position of the *i*th particle is recorded and represented as $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$, which is also called $pbest$. The index of the best particle among all the particles in the population is represented by the symbol g, and p_g is called $gbest$. At each iteration step, the particles are manipulated according to the following equations:

$$v_{id} = wv_{id} + R_1 c_1 (P_{id} - x_{id}) + R_2 c_2 (p_{gd} - x_{id}), \tag{1}$$

$$x_{id} = x_{id} + v_{id}. \tag{2}$$

Where w is inertia weight; c_1 and c_2 are acceleration constants; and R_1, R_2 are random vectors with components uniformly distributed in [0, 1]. For Eq. (1), the portion of the adjustment to the velocity influenced by the individual's own $pbest$ position is considered as the cognition component, and the portion influenced by $gbest$ is the social component. After the velocity is updated, the new position of the *i*th particle in its *d*th dimension is recomputed. This process is repeated for each dimension of the *i*th particle and for all the particles in the swarm.

2.2 Symbiotic Multi-swarm Particle Swarm Optimizer (SMPSO)

In our previous proposed MCPSO algorithm, the population is divided into several sub-swarms, in which some sub-swarms are master swarms and the other sub-swarms are slave swarms. Both the master and slave swarms have different properties. The master swarms update particles information according to the slave swarms and their own. While the slave swarms update the particles information only based on their own

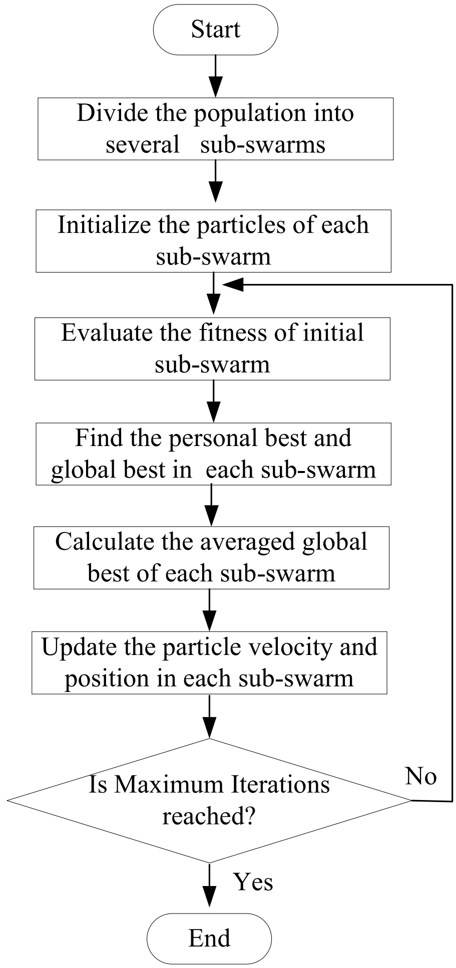

Fig. 1. Flow chart of SMPSO

information. It should be noted that there is no information exchange between slave swarms which will slow down the convergence rate. The detailed introduction of MCPSO can be referred to [11].

To deal with this issue, the population in SMPSO consists of several sub-swarms with the same properties, i.e. they are both identical sub-swarms. Each sub-swarm can supply many new promising particles to other sub-swarm as the evolution proceeds. Each sub-swarm updates the particle states based on the best position discovered so far by all the particles both in the other sub-swarms and its own. The interactions between the other sub-swarms and its own influence the balance between exploration and exploitation and maintain a suitable diversity in the population, even when it is approaching the global solution, thus reducing the risk of converging to local sub-optima.

Table 1. Pseudocode for SMPSO algorithm

```
Algorithm SMPSO
Begin
   Randomize positions and velocities of N × P particles in search space. Divide
   whole population into N species with P particles randomly;
   Evaluate the fitness value of each particle
   Repeat
      Do in parallel
         Swarm n , 1 ≤ n ≤ N
      End Do in parallel
      Barrier synchronization  //wait for all processes to finish
      Select the center particle and determinate its position according to Eq.(4)
      Evolve each sub-swarm   //Update the velocity and position using Eq. (3) and
      (2), respectively
      Evaluate the fitness value of each particle
   Until a terminate-condition is met
   End
```

The search information can be transformed among sub-swarms by a center communication mechanism that uses a center particle whose position is averaged by the sub-swarms to guide the flight of particles in all the sub-swarms. During the flight each particle of the sub-swarm adjusts its trajectory according to its own experience, the experience of its neighbors, and the experience of the particles in other sub-swarms, making use of the best previous position encountered by itself, its neighbors and the center particle position. In this way, the search information can be transformed between sub-swarms which can accelerate the convergence rate.

In SMPSO, we use a population of $N{\times}P$ individuals, or in symbiosis terminology, an ecosystem of $N{\times}P$ organisms. The whole population is divided into N species to modeling symbiosis in the context of the evolving ecosystems (for convenience, each species has the same population size P). As in nature, the species are separated breeding populations and evolve parallel, while interact with one another within each generation and have a symbiotic relationship.

To realize this mechanism, we propose a modification to the original PSO velocity update equation. In each generation, particle i in species n will evolve according to the following equations:

$$v_i^n(t+1) = wv_i^n(t) + R_1c_1(p_i^n - x_i^n(t)) + R_2c_2(p_g^n - x_i^n(t)) + R_3c_3(p_c^n - x_i^n(t)), \tag{3}$$

where p_i^n and p_g^n are the best previous solution achieved so far by particle i and the species n, respectively. R_3 is a random value between 0 and 1. c_3 is acceleration constant; P_c^n represents the center position of the global best particle in all the sub-swarms. After N sub-swarms update their positions and best performed particle is found, a center particle is updated according to the following formula:

$$P_C^n(t+1) = \frac{1}{N}\sum_{i=1}^{N} p_g^n(t)\,,$$
$$n = 1, 2, \cdots N\,, \tag{4}$$
$$i = 1, 2, \cdots P\,.$$

Unlike other particles, the center particle has no velocity, but it is involved in all operations the same as the ordinary particle, such as fitness evaluation, competition for the best particle, except for the velocity calculation. The flow chart SMPSO is shown in Fig.1. and pseudocode for the SMPSO is listed in Table 1.

3 Portfolio Optimization Problem

The portfolio optimization problem is one of the most important issues in asset management, which deals with how to form a satisfying portfolio. Modern portfolio analysis started from pioneering research work of Markowitz (1952) [14]. The original portfolio optimization model is usually called mean–variance model, firstly proposed by Markowitz. In this paper, we use an improved mean–variance model considering the transaction costs and no short sales. It assumes an investor allocates the wealth among *n* assets. Some notations are introduced as follows:

r_i : The yield of the i asset, $i = 1, \ldots \ldots, n$;
$R = (R_1 \ldots \ldots, R_n)$S : $R_i = E(r_i)$ denoting the expected yield;
$\sigma_{ij} = \mathrm{cov}(r_i, r_j)$: the covariance of r_i and r_j ;
$\mathrm{X} = (x_1 \ldots \ldots, x_n)$: x_i is proportionment of the i asset that investor want to invest;
$k = (k_1 \ldots \ldots, k_n)$: k_i is the transaction fee of the i asset;
λ : The risk factor distributing in [0, 1]. Larger λ represents investor love risk more.

Based on these defined variables, the function $f(x)$ and $g(x)$ denotes the revenue and risk in the portfolio optimization problem can be obtained as following:

$$f(x) = \sum_{i=1}^{n} R_i x_i - \sum_{i=1}^{n} k_i x_i\,, \tag{5}$$

$$g(x) = \sum_{i=1}^{n}\sum_{i=1}^{n} \sigma_{ij} x_i x_j\,. \tag{6}$$

The improved portfolio optimization model can be formulated as:

$$\min F(x) = \min\{\lambda g(x) - (1-\lambda)\, f(x)\}$$
$$\begin{cases} \sum_{i=1}^{n} x_i = 1; \\ 0 < x_i. \end{cases} \tag{7}$$

Where $0 < x_i$ means that the short sale is not permitted.

When we use SMPSO to solve the model, there is an n demension search space denoting n kinds of sassets, and the position of the particle $X = (x_1 \dots \dots, x_n)$ presents the proportionment of every assesst. The position of the particle with the minimum fitness value is the best selection of portfolio optimization.

4 Illustrative Examples

In order to test the effectiveness of SMPSO for portfolio optimization, we use the data of five assets as the sample that can be referred to [15]. k_i is set as 0.075%. Different risk preference is considering, where three value of risk factors λ (0.2, 0.5, 0.8) identifying the different kind of inverstors is used.

In applying PSO to the above model, w_{max}, w_{min}, c_1, c_2 are set to be 0.9, 0.4, 2.0, 2.0, respectively. For SMPSO, $c_1 = c_2 = 1.367$ and $c_3 = 2$ is used, w_{max}, w_{min} is set the same as those defined in PSO. The max iterations of the two methods are set to be 200. A total of 50 runs are performed.

Numerical results with different λ obtained by the standard PSO and the SMPSO are showed in the Table 2-3. Figures 2-4 present the mean relative performance using different λ generated by PSO and SMPSO.

The max value, the min value, the standard deviation and the mean value are summarized in Table 2-3. It is clear that for almost of all the different risk preferences, SMPSO owns smaller standard deviation and mean value, which demonstrated it outperforms PSO in terms of result robustness and solution quality.

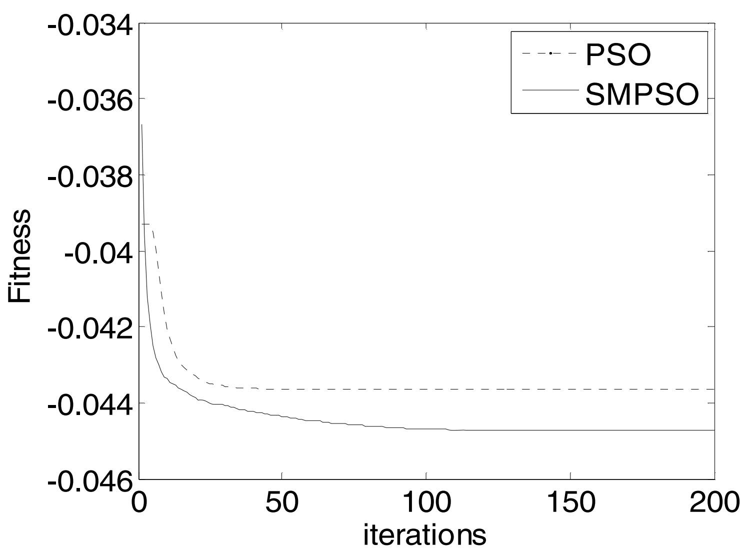

Fig. 2. Mean relative performance using $\lambda = 0.2$

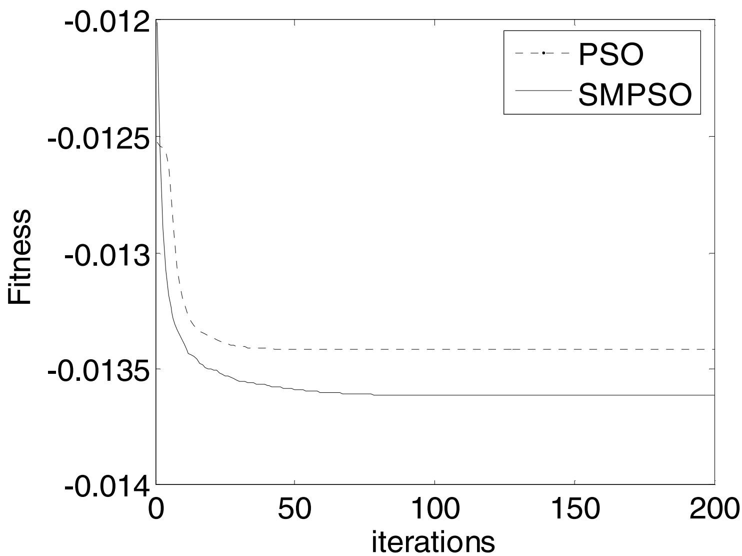

Fig. 3. Mean relative performance using $\lambda = 0.5$

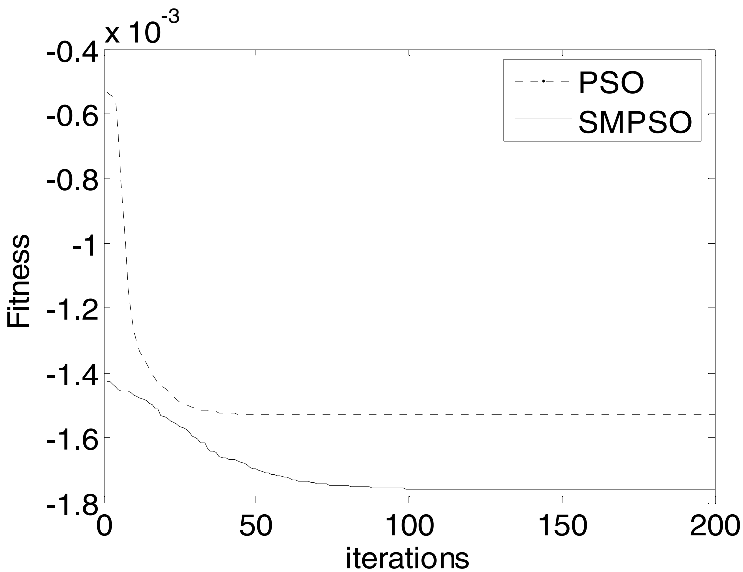

Fig. 4. Mean relative performance using $\lambda = 0.8$

From Figures 2-4, it is obviously found that SMPSO has quicker convergence rate in the different situations compared with PSO. Furthermore, its convergence process is much steadier than that of PSO.

Table 2. Numerical results with different λ

		Worst	Best	Mean	Std
$\lambda = 0.2$	PSO	-4.07894e-002	-4.46873e-002	-4.36409e-002	1.07146e-003
	SMPSO	-4.47326e-002	-4.47331e-002	-4.47331e-002	7.68445e-008
$\lambda = 0.5$	PSO	-1.29264e-002	-1.36157e-002	-1.34150e-002	1.617577e-004
	SMPSO	-1.36159e-002	-1.36159e-002	-1.36159e-002	4.32891e-012
$\lambda = 0.8$	PSO	-1.01019e-003	-1.76112e-003	-1.52787e-003	2.17589e-004
	SMPSO	-1.76128e-003	-1.76129e-003	-1.76128e-003	3.46453e-010

Table 3. Numerical results with different λ

		x_1	x_2	x_3	x_4	x_5
$\lambda = 0.2$	PSO	3.5878e-006	1.7489e-005	3.4526e-001	1.8246e-011	6.5472e-001
	SMPSO	2.2211e-008	1.2714e-007	2.9494e-001	6.1913e-008	7.0506e-001
$\lambda = 0.5$	PSO	5.4377e-013	1.3446e-001	7.9098e-001	4.4508e-002	3.0046e-002
	SMPSO	7.5996e-013	1.3851e-001	7.8621e-001	4.2837e-002	3.2448e-002
$\lambda = 0.8$	PSO	3.1966e-011	6.3375e-001	9.2079e-002	2.7417e-001	3.7942e-013
	SMPSO	1.3609e-010	6.3285e-001	9.1060e-002	2.7608e-001	1.5641e-009

All the results presented in the tables and figures can proved that the SMPSO could be a more effective way for the investors to solve the portfolio optimizations problems.

5 Conclusions

In this paper, we proposed a new variant of original PSO, i.e. symbiotic multi-swarm PSO that is inspired by the phenomenon of symbiosis in natural ecosystems. SMPSO is based on a multiple swarms scheme, in which the whole population is divided into several sub-swarms. The particles in each sub-swarm are enhanced by the experience of its own and the other sub-swarms. By introducing the center communication mechanism the search information can be transferred among sub-swarms, that help accelerate the convergence rate and avoid the particles be trapped into local minima.

We also use the improved Markowitz model considering two real-world constraints to test our proposed algorithm. The preliminary experimental results suggest that SMPSO have superior features, both in high quality of the solution and robustness of the results. Our proposed portfolio model and SMPSO are applicable and reliable in real markets with large number of stocks.

Acknowledgment

This work is supported by Shenzhen-Hong Kong Innovative Circle project (Grant no.SG200810220137A) and Project 801-000021 supported by SZU R/D Fund.

References

1. Young, M.R.: A Minimax Portfolio Selection Rule with Linear Programming Solution. Management Science 44, 673–683 (1998)
2. Arenas, M., Bilbao, A., Rodriguez Uria, M.V.: A Fuzzy Goal Programming Approach to Portfolio Selection. European Journal of Operational Research 133, 287–297 (2001)
3. Ballestero, E., Romero, C.: Portfolio Selection: A Compromise Programming Solution. Journal of the Operational Research Society 47, 1377–1386 (1996)
4. Oh, K.J., Kim, T.Y., Min, S.: Using Genetic Algorithm to Support Portfolio Optimization for Index Fund Management. Expert Systems with Applications 28, 371–379 (2005)
5. Yang, X.: Improving Portfolio Efficiency: A Genetic Algorithm Approach. Computational Economics 28, 1–14 (2006)
6. Crama, Y., Schyns, M.: Simulated Annealing for Complex Portfolio Selection Problems. European Journal of Operational Research 150, 546–571 (2003)
7. Fernandez, A., Gomez, S.: Portfolio Selection Using Neural Networks. Computers & Operations Research 34, 1177–1191 (2007)
8. Derigs, U., Nickel, N.H.: On a Local-search Heuristic for a Class of Tracking Error Minimization Problems in Portfolio Management. Annals of Operations Research 131, 45–77 (2004)
9. Derigs, U., Nickel, N.H.: Meta-heuristic Based Decision Support for Portfolio Optimization with a Case Study on Tracking Error Minimization in Passive Portfolio Management. OR Spectrum 25, 345–378 (2003)
10. Schlottmann, F., Seese, D.: A Hybrid Heuristic Approach to Discrete Multi-Objective Optimization of Credit Portfolios. Computational Statistics & Data Analysis 47, 373–399 (2004)
11. Niu, B., Zhu, Y.L., He, X.X., Wu, H.: MCPSO: A Multi-Swarm Cooperative Particle Swarm Optimizer. Applied Mathematics and Computation 185, 1050–1062 (2007)
12. Niu, B., Zhu, Y.-l., He, X.-X.: A Multi-Population Cooperative Particle Swarm Optimizer for Neural Network Training. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3971, pp. 570–576. Springer, Heidelberg (2006)
13. Niu, B., Zhu, Y.L., He, X.X., Shen, H.: A Multi-swarm Optimizer Based Fuzzy Modeling Approach for Dynamic Systems Processing. Neurocomputing 71, 1436–1448 (2008)
14. Markowitz, H.W.: Foundations of Portfolio Theory. Journal of Finance 46, 469–477 (1991)
15. Yang, K.Y., Wang, X.F.: Solving the Multi-solution Portfolio Selection Model Based on the GA (Chinese). Journal of ShanDong finance college 6, 60–63 (2003)