Nonlinear Analysis: Real World Applications 10 (2009) 2396–2406

Contents lists available at ScienceDirect

Nonlinear Analysis: Real World Applications

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

Particle swarm optimization approach to portfolio optimization

Tunchan Cura *

Istanbul University, Faculty of Business Administration, Quantitative Methods, Turkey

ARTICLE INFO

Article history:
Received 26 February 2008
Accepted 24 April 2008

Keywords:
Particle swarm optimization
Portfolio optimization
Efficient frontier

ABSTRACT

The survey of the relevant literature showed that there have been many studies for portfolio optimization problem and that the number of studies which have investigated the optimum portfolio using heuristic techniques is quite high. But almost none of these studies deals with particle swarm optimization (PSO) approach. This study presents a heuristic approach to portfolio optimization problem using PSO technique. The test data set is the weekly prices from March 1992 to September 1997 from the following indices: Hang Seng in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA and Nikkei in Japan. This study uses the cardinality constrained mean-variance model. Thus, the portfolio optimization model is a mixed quadratic and integer programming problem for which efficient algorithms do not exist. The results of this study are compared with those of the genetic algorithms, simulated annealing and tabu search approaches. The purpose of this paper is to apply PSO technique to the portfolio optimization problem. The results show that particle swarm optimization approach is successful in portfolio optimization.

1. Introduction

The particle swarm optimization (PSO) approach is a heuristic technique introduced comparatively recently by Kennedy and Eberhart [1]. There are very few studies on PSO in the literature, and almost none of them deals with portfolio optimization (PO). This study presents a new approach to PO using PSO.

PO consists of the portfolio selection problem in which we want to find the optimum way of investing a particular amount of money in a given set of securities or assets [5]. Although the task of yielding minimum risk and maximum return looks simple, there is more than one way of establishing an optimum portfolio. Markowitz [2,6] formulated the fundamental theorem of a mean–variance portfolio framework, which explains the trade-off between mean and variance, representing expected returns and risk of a portfolio, respectively. An advanced model was introduced by Konno and Yamazaki [3] in which a mean-absolute deviation (MAD) model and absolute deviation are utilized as a measure of risk. However, it was insensitive to some extremes, which could be the source of serious error, contrary to the suggestion that the MAD model is suitable under all circumstances [7]. As Mansini and Sprenza stated [8], most of the portfolio selection models assume a perfect fractionability of the investments; however, securities are negotiated as multiples of a minimum transaction lot in the real world, and they suggested a mixed integer programming model with minimum lot constraint for portfolio selection.

Some researchers have investigated the multi-period PO case, in which investors invest continuously rather than at intervals or only once. Celikyurt and Ozekici [15] accomplished this, assuming that there are some economic, social, political and other factors affecting the asset returns. They formed their stochastic market with respect to these factors, and they used a Markov chain approach in their study.

This study basically employs the Markowitz mean–variance model. However, the standard model does not contain any cardinality or bounding constraints, which restrict the number of assets and, the upper and the lower bounds of proportion

* Corresponding address: Istanbul University, Faculty of Business Administration, Quantitative Methods, 34157 Istanbul, Turkey.
E-mail address: tunchan@istanbul.edu.tr.

1468-1218/$ – see front matter
doi:10.1016/j.nonrwa.2008.04.023

of each asset in the portfolio, respectively. Chang et al. [9] and Fernandez and Gomez [5] used a modified Markowitz model called a "cardinality constrained mean–variance (CCMV) model". Here, the CCMV model is used and is solved by a PSO approach.

There are some reports of solving the PO problem using heuristic methods. These methods consist of genetic algorithms (GA) [9,6,10], tabu search (TS) [9], simulated annealing (SA) [9,11,12], neural networks [5] and others [13,8,14]. The results of this study are compared with those of the GA, SA and TS approaches [9]. The test data set is the weekly prices from March 1992 to September 1997 from the following indices: Hang Seng in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA and Nikkei in Japan. The number of different assets for each of the test problems is 31, 85, 89, 98, and 225, respectively.

2. CCMV model for PO

This study uses the CCMV model [9] and [5], which is derived from the well-known Markowitz standard model, which is:

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j \sigma_{ij} \tag{1}$$

$$\text{subject to } \sum_{i=1}^{N} x_i \mu_i = R^*, \tag{2}$$

$$\sum_{i=1}^{N} x_i = 1, \tag{3}$$

$$0 \le x_i \le 1. \quad i = 1, \ldots, N \tag{4}$$

where N is the number of different assets, σ_{ij} is the covariance between returns of assets i and j, x_i is the proportion of asset i in the portfolio, μ_i is the mean return of asset i and R^* is the desired mean return of the portfolio.

In order to observe the different objective function values for varying R^* values, standard practice introduces a risk aversion parameter $\lambda \in [0, 1]$. With this new parameter, the model can be described as:

$$\min \lambda \left[\sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j \sigma_{ij}\right] - (1-\lambda)\left[\sum_{i=1}^{N} x_i \mu_i\right] \tag{5}$$

$$\text{subject to } \sum_{i=1}^{N} x_i = 1, \tag{6}$$

$$0 \le x_i \le 1 \quad i = 1, \ldots, N. \tag{7}$$

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, we can say that the sensitivity of the investor to the risk increases as λ approaches 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 return and variance. Tracing the mean return and variance intersections, we draw a continuous curve that is called an efficient frontier in the Markowitz theory [2]. Since every point on an efficient frontier curve indicates an optimum, the PO problem is a multi-objective optimization problem. So, a definition must be adopted for the concept of optimal solution. This study used the Pareto optimality definition, which questions whether a feasible solution of the PO problem will be an optimal solution (or non-dominated solution) if there is no other feasible solution improving one objective without making the other worse [5]. For the problem defined in Eqs. (5)–(7), the efficient frontier is a curve that gives the best trade-off between mean return and risk. Fig. 1 shows such a curve corresponding to the smallest benchmark problem (Hang Seng) described in Section 4. This efficient frontier has been computed taking 2000 different λ values; that is, there have been 2000 distinct objective function values for the resulting solutions. Thus, each of the solutions corresponds to a point in the efficient frontier. This curve was called a standard efficient frontier by Fernandez and Gomez [5].

Some additional variables have to be included in the standard model in order to describe the CCMV model. As mentioned above, there are two constraints in the CCMV model in addition to those of the original model. The first one is to restrict K, the number of assets in the portfolio. If the decision variable $z_i \in \{0, 1\}$ is 1, asset i will be included in the portfolio, otherwise it will not be. The second constraint is that an included asset's proportion is within the lower and the upper bounds, ε_i and δ_i, respectively. Thus, the CCMV model is:

$$\min \lambda \left[\sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j \sigma_{ij}\right] - (1-\lambda)\left[\sum_{i=1}^{N} x_i \mu_i\right] \tag{8}$$

$$\text{subject to } \sum_{i=1}^{N} x_i = 1, \tag{9}$$

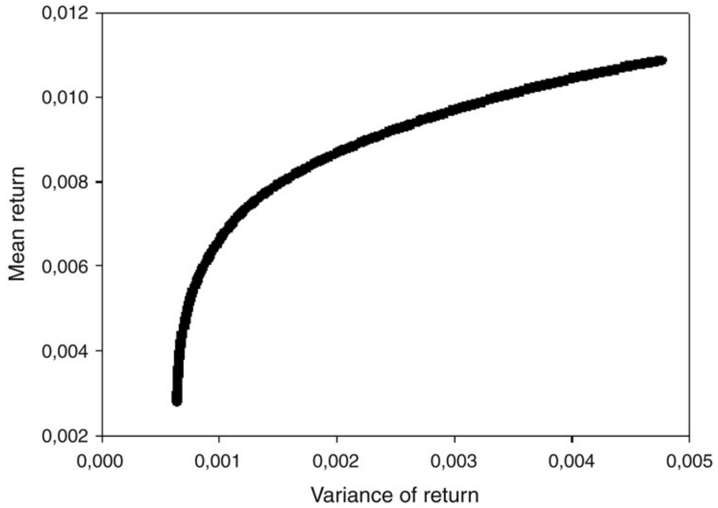

Fig. 1. Standard efficient frontier corresponding to Hang Seng benchmark problem.

$$\sum_{i=1}^{N} z_i = K, \tag{10}$$

$$\varepsilon_i z_i \le x_i \le \delta_i z_i, \quad i = 1, \ldots, N \tag{11}$$

$$z_i \in \{0, 1\} \quad i = 1, \ldots, N. \tag{12}$$

In the presence of cardinality and bounding constraints, the resulting efficient frontier, which was called a general efficient frontier by Fernandez and Gomez [5], can be different from that obtained with the standard model.

3. PSO approach for optimization of the CCMV model

The formulation in Eqs. (8)–(12) is a mixed quadratic and integer programming problem for which efficient algorithms do not exist [5]. Thus, this study, for solving a PO problem, introduces a PSO heuristic method, which is one of the latest evolutionary optimization methods and is based on the metaphor of social interaction and communication such as bird flocking and fish schooling. The swarm in PSO consists of a population and each member of the population is called a particle, which represents a portfolio in this study.

This study follows the "gbest neighborhood topology" described by Kennedy et al. [4], according to which, each particle remembers its best previous position and the best previous position visited by any particle in the whole swarm. In other words, a particle moves towards its best previous position and towards the best particle.

We can consider that there should be N dimensions, each representing an asset, for each particle. Indeed, this consideration will organize the swarm formation in this study with two modifications: First, each particle includes proportion variables denoted by x_{pi} ($p = 1, \ldots, P$, where P is the number of particles in the swarm); Second, each particle includes decision variables denoted by z_{pi}. Thus, the number of dimensions that a particle owns will be $2 \times N$.

3.1. Fitness function

Kennedy and Eberhart [1] suggested a fitness value associated with each particle. Thus, a particle moves in solution space with respect to its previous position where it has met the best fitness value, and the neighbor's previous position where the neighbor has met the best fitness value. In this study, the fitness function is defined as:

$$f_p = \lambda \left[\sum_{i=1}^{N} \sum_{j=1}^{N} z_{pi} x_{pi} z_{pj} x_{pj} \sigma_{ij} \right] - (1 - \lambda) \left[\sum_{i=1}^{N} z_{pi} x_{pi} \mu_i \right] \tag{13}$$

where f_p is the fitness value of particle p.

At each one of the iterations, 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.

3.2. Moving a particle

We have mentioned that a particle moves towards its personal best position and towards the best particle of the swarm at each one of the iterations. Indeed, this movement depends on its current velocity, which is defined as:

$$vz_{pi}^{t+1} = vz_{pi}^{t} + \omega_1 \times (Gz_{bi} - z_{pi}^{t}) + \omega_2 \times (Gz_{pi} - z_{pi}^{t}) \tag{14}$$

$$vx_{pi}^{t+1} = \begin{cases} vx_{pi}^{t} + \omega_1 \times (Gx_{bi} - x_{pi}^{t}) + \omega_2 \times (Gx_{pi} - x_{pi}^{t}) & \text{if } z_{pi}^{t+1} = 1, \\ vx_{pi}^{t} & \text{otherwise,} \end{cases} \tag{15}$$

where both ω_1 and ω_2 denote uniform random numbers between 0 and 2, t and b denote the iteration number and the best particle in the swarm respectively, vx_{pi}^{t} denotes the velocity of particle p on dimension x_i, and vz_{pi}^{t} denotes the velocity of particle p on dimension z_i. As seen in Eq. (15), vx_{pi}^{t+1} will be updated if asset i is selected by particle (or portfolio) p at iteration $t+1$, which means $z_{pi}^{t+1} = 1$, and z_{pi}^{t+1} is described in Eq. (16). Gx_{pi} denotes the best previous position of particle p on dimension x_i, and Gz_{pi} denotes the best previous position of particle p on dimension z_i. Thus, particle p moves at iteration $t+1$ as follows:

$$z_{pi}^{t+1} = round\left(\frac{1}{1+e^{-\varsigma}} - \alpha\right) \tag{16}$$

$$x_{pi}^{t+1} = \begin{cases} x_{pi}^{t} + vx_{pi}^{t+1} & \text{if } z_{pi}^{t+1} = 1, \\ x_{pi}^{t} & \text{otherwise} \end{cases} \tag{17}$$

where $\varsigma = z_{pi}^{t} + vz_{pi}^{t+1}$ and α is set to 0.06. For a given particle, if the velocity on dimension z_i^t is zero, this particle will not move in that dimension at iteration $t+1$. Suppose $vz_{pi}^{t+1} = 0$ and $z_{pi}^{t} = 0$, hence $1/(1+e^0) = 0.5$ and $round(0.5) = 1$, which means that particle p will move in dimension $z_i(z_{pi}^{t+1} = 1)$ at iteration $t+1$. In order to avoid such an unwanted move, we can use α as seen in Eq. (16).

3.3. Constraint satisfaction

As we discussed above, each particle would have been repositioned in $2 \times N$ dimensional search space at the end of any iteration. We know that particles represent candidate solutions, and each particle must be feasible and satisfy Eqs. (9)–(11). Usually, the constraints appear in the fitness (energy) function with some penalty weights. Inspired by the similar approach taken by [9], this study employs the arrangement algorithm shown in Fig. 2 in order to guarantee that any repositioned particle is feasible.

To explain this representation further, suppose particle p has distinct $K_p^* = \sum_{i=1}^{N} z_{pi}$ assets, and Q is the set of assets which are held by p. If $K_p^* < K$, then some assets must be added to Q; if $K_p^* > K$, then some assets must be removed from Q until $K_p^* = K$. Let us consider the case where $K_p^* < K$. We need to decide which of the remaining assets is to be added. This study suggests exploiting one of the two arrangements with equal probabilities. That is, if a random number between 0 and 1 is less than 0.5, we select an asset i at random, such that $i \notin Q$, then we add i to Q. Otherwise, we select the maximum c-valued asset i such that $i \notin Q$ and then add i to Q.

$$\theta_i = 1 + (1-\lambda)\mu_i \quad i = 1, \ldots, N \tag{18}$$

$$\rho_i = 1 + \lambda \frac{\sum_{j=1}^{N} \sigma_{ij}}{N} \quad i = 1, \ldots, N \tag{19}$$

$$\Omega = -1 \times \min(0, \theta_1, \ldots, \theta_N) \tag{20}$$

$$\Psi = -1 \times \min(0, \rho_1, \ldots, \rho_N) \tag{21}$$

$$c_i = \frac{\theta_i + \Omega}{\rho_i + \psi} \quad i = 1, \ldots, N. \tag{22}$$

For a given asset, c-value basically gives the proportion between mean return and mean risk with respect to aversion parameter. Thus, this value may give an idea about which asset may be added to or removed from Q. Eqs. (20) and (21) are used to avoid miscomputation of c_i in the extraordinary case(s) of $(1-\lambda)\mu_i < -1$ and/or $\lambda \frac{\sum_{j=1}^{N} \sigma_{ij}}{N} < -1$.

In the case of $K_p^* > K$, we need to decide which of the assets is to be removed. This study suggests one of the two arrangements with equal probabilities, which are (1) randomly selecting an asset in Q then removing it or (2) selecting the minimum c-valued asset in Q and then eliminating it.

As mentioned above, the x_i dimensions of a particle (a candidate solution) give the proportions. Due to Eq. (9), the sum of x_i dimensions where $i \in Q$ must be equal to 1. Let χ be the current sum of x_i. If we reposition $x_{pi} = x_{pi}/\chi$ for all $i \in Q$, then Eq. (9) will be satisfied.

Arrange (p)
p : current particle (that is solution) and consists of: Q, the set of K_p^* distinct assets in the current particle; x_{pi}, the current proportion value for asset i; and z_{pi}, the current decision value for asset i.
Begin
While ($K_p^* < K$)
  If random(0, 1) < 0.5 **then**
    i = **randomly select an asset such that** $i \notin Q$
  Else
    i = **select the maximum** c**-valued asset such that** $i \notin Q$
  End if
  $\psi = \omega_1(Gx_{bi} - x_{pi}) + \omega_2(Gx_{pi} - x_{pi}) + x_{pi}$
  If $vx_{pi} + \psi \geq 0$ **then**
    $z_{pi} = 1$
    $Q = Q \cup [i]$
    $K_p^* = K_p^* + 1$
  End if
End while
While ($K_p^* > K$)
  If random(0, 1) < 0.5 **then**
    i = **randomly select an asset such that** $i \in Q$
  Else
    i = **select the minimum** c**-valued asset such that** $i \in Q$
  End if
  $z_{pi} = 0$
  $vx_{pi} = vx_{pi}^{beforemove}$ // vx_{pi} *stays unchanged*
  $x_{pi} = x_{pi}^{beforemove}$ // x_{pi} *stays unchanged*
  $Q = Q - [i]$
  $K_p^* = K_p^* - 1$
End while
While (true)
  $\chi = \sum_{i \in Q} x_{pi}$
  $x_{pi} = x_{pi} / \chi, \quad \forall i \in Q$
  $\eta = \sum_{i \in Q} \max(0, x_{pi} - \delta_i)$
  $\phi = \sum_{i \in Q} \max(0, \varepsilon_i - x_{pi})$
  If $\eta = 0$ **and** $\Phi = 0$ **then**
    Exit arrangement algorithm
  $t_i = \max(0, \delta_i - x_{pi}) \ \forall i \in Q$
  $\delta^* = \sum_{i \in Q} t_i$
  $e_i = \max(0, x_{pi} - \varepsilon_i) \ \forall i \in Q$
  $\varepsilon^* = \sum_{i \in Q} \varepsilon_i$
  For $i = 1$ **to** N
    If $z_{pi} = 1$ **then**
      If $t_i > 0$ **then**
        $x_{pi} = x_{pi} + (t_i / \delta^*) \times \eta$
      Else
        $x_{pi} = \delta_i$
      End if
      If $e_i > 0$ **then**
        $x_{pi} = x_{pi} - (e_i / \varepsilon^*) \times \Phi$
      Else
        $x_{pi} = \varepsilon_i$
      End if
    End if
  End for
End while
End

Fig. 2. Arrangement algorithm for particle p.

According to Eq. (11), we need to satisfy $\varepsilon_i \leq x_{pi} \leq \delta_i$ for all $i \in Q$ as well. Let $t_i = \delta_i - x_{pi}$ and δ^* be the sum of t_i, where $i \in Q$ and $t_i > 0$. Let $e_i = x_{pi} - \varepsilon_i$ and ε^* be the sum of e_i, where $i \in Q$ and $e_i > 0$. Let η be the sum of $(-1 \times t_i)$ where iEQ and $t_i < 0$, let $O(\Phi)$ be the sum of $(-1 \times e_i)$ where $i \in Q$, and $e_i < 0$. If particle p exceeds the upper bound or goes below the lower bound on any dimension then it will be arranged as follows:

$$x_{pi} = \begin{cases} x_{pi} + \dfrac{t_i}{\delta^*}\eta & \text{if } t_i > 0 \\ \delta_i & \text{if } t_i < 0 \\ x_{pi} - \dfrac{e_i}{\varepsilon^*}\phi & \text{if } e_i > 0 \\ \varepsilon_i & \text{otherwise} \end{cases} \qquad \forall i \in Q. \tag{23}$$

3.4. PSO heuristic

Bringing together all that we have discussed until now, the PSO heuristic used in this study is shown in Fig. 3.

4. Computational experiments

In this section, we present the results obtained when searching the general efficient frontier that provides the solution of the problem formulated in Eqs. (8)–(12). The PSO approach of this study has been compared to three other approaches, GA, TS and SA [9]. The test data, which have been used elsewhere ([5] and [9]), were obtained from http://people.brunel.ac.uk/%7Emastjjb/jeb/orlib/portinfo.html. These data correspond to weekly prices between March

Table 1
The experimental results of four heuristics

Index	Assets		GA	TS	SA	PSO
Hang Seng	31	Mean Euclidian distance	0.0040	0.0040	0.0040	0.0049
		Contribution percentage (%)	64.7059	19.6078	5.8824	9.8039
		Variance of return error (%)	1.6441	1.6578	1.6628	2.2421
		Mean return error (%)	0.6072	0.6107	0.6238	0.7427
		Time (s)	18	9	10	34
DAX 100	85	Mean Euclidian distance	0.0076	0.0082	0.0078	0.0090
		Contribution percentage (%)	31.3725	19.6078	19.6078	29.4118
		Variance of return error (%)	7.2180	9.0309	8.5485	6.8588
		Mean return error (%)	1.2791	1.9078	1.2817	1.5885
		Time (s)	99	42	52	179
FTSE 100	89	Mean Euclidian distance	0.0020	0.0021	0.0021	0.0022
		Contribution percentage (%)	45.0980	15.6863	11.7647	27.4510
		Variance of return error (%)	2.8660	4.0123	3.8205	3.0596
		Mean return error (%)	0.3277	0.3298	0.3304	0.3640
		Time (s)	106	42	55	190
S&P 100	98	Mean Euclidian distance	0.0041	0.0041	0.0041	0.0052
		Contribution percentage (%)	27.4510	17.6471	27.4510	27.4510
		Variance of return error (%)	3.4802	5.7139	5.4247	3.9136
		Mean return error (%)	1.2258	0.7125	0.8416	1.4040
		Time (s)	126	51	66	214
Nikkei	225	Mean Euclidian distance	0.0093	0.0010	0.0010	0.0019
		Contribution percentage (%)	43.1373	23.5294	21.5686	11.7647
		Variance of return error (%)	1.2056	1.2431	1.2017	2.4274
		Mean return error (%)	5.3266	0.4207	0.4126	0.7997
		Time (s)	742	234	286	919

```
Begin
λ = 0
While (λ ≤ 1)
  P = 20⌊√N⌋
  Randomly initialize all particles in the
    swarm
  Arrange(p)        p = 1, …, P
  Compute f_p       p = 1, …, P  //see Eq.(13)
  G_p = copy of p and  f_Gp = f_p  p = 1, …, P
  Find particle b in the swarm such
          that f_b is the minimum
  γ = f_b
  for counter = 1 to ⌊1000N / P⌋
    For p = 1 to P
      For i = 1 to N
        Move particle p on dimension z_i
                    //see Eq.(14) and (16)
        If z_pi = 1 then
          vx_pi^beforemove = vx_pi
          x_pi^beforemove = x_pi
          Compute the velocity vx_pi
                    //see Eq.(15)
          If vx_pi + x_pi ≥ 0 then
            move particle p on
              dimension x_i // see Eq.(17)
          Else
            z_pi = 0
            x_pi = 0
          End if
        End if
        Arrange(p)
        Compute f_p  // see Eq.(13)
        If f_p < f_Gp then
          G_p = copy of p and f_Gp = f_p
        End if
        If f_Gp < γ then
          γ = f_Gp
          b = p    //the best neighbor
        End if
      End for
    End for
  End for
  λ = λ + Δλ
End while
End
```

Fig. 3. PSO heuristic for PO.

1992 and September 1997 from the indices: Hang Seng in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA and Nikkei 225 in Japan. For each set of test data, the number, N, of different assets is 31, 85, 89, 98 and 225, respectively.

All the results have been computed using the values $K = 10$, $\varepsilon_i = 0.01$ $(i = 1, \dots, N)$ and $\delta_i = 1$ $(i = 1, \dots, N)$ for the problem formulation, and $\Delta\lambda = 0.02$ for the implementation of the algorithms. Since $\Delta\lambda = 0.02$, the number of different λ values, denoted by ξ, is 51. The algorithms used the same test data and were run on the same Pentium M 2.13 GHz computer with 1 GB RAM. Each of the four heuristics has evaluated $1000N$ portfolios without counting the initializations.

Taking the sets of Pareto optimal portfolios obtained with each heuristic, we trace out their *heuristic efficient frontier*. This study compared the standard efficient frontiers and the corresponding heuristic efficient frontiers. For comparison of the

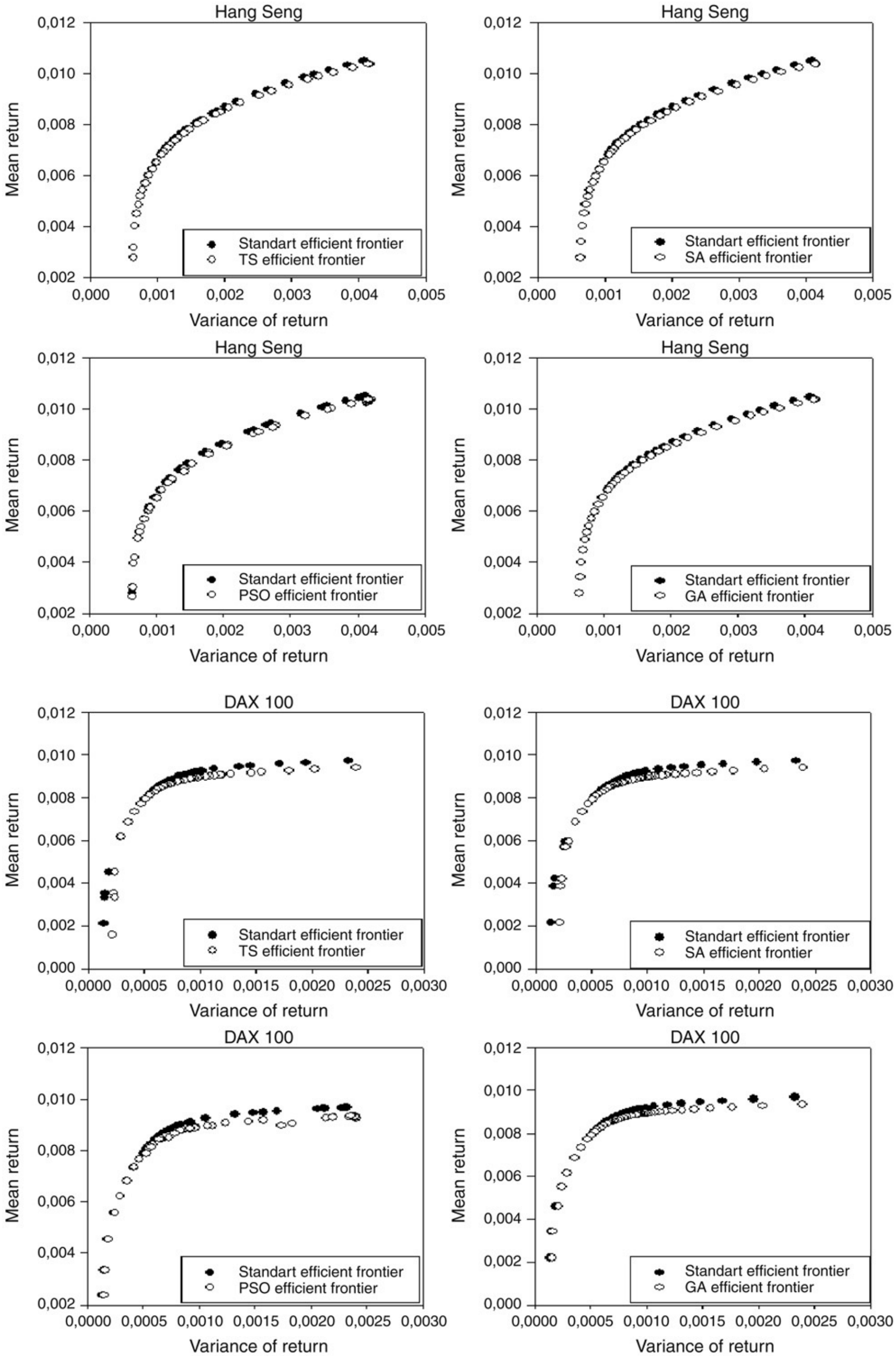

Fig. 4. Heuristic efficient frontiers for Hang Seng and DAX 100.

standard efficient frontier and the heuristic efficient frontier, this study has used mean Euclidian distance, contribution percentage, variance of return error, mean return error and execution time (in units of seconds). Table 1 shows the comparative results, and Figs. 4–6 show the comparison of efficient frontiers.

Let the pair (v_i^s, r_i^s) $(i = 1, \ldots, 2000)$ represent the variance and mean return of the point in the standard efficient frontier, and let the pair (v_j^h, r_j^h) $(j = 1, \ldots, \xi)$ represent the variance and mean return of the point in the heuristic efficient frontier.

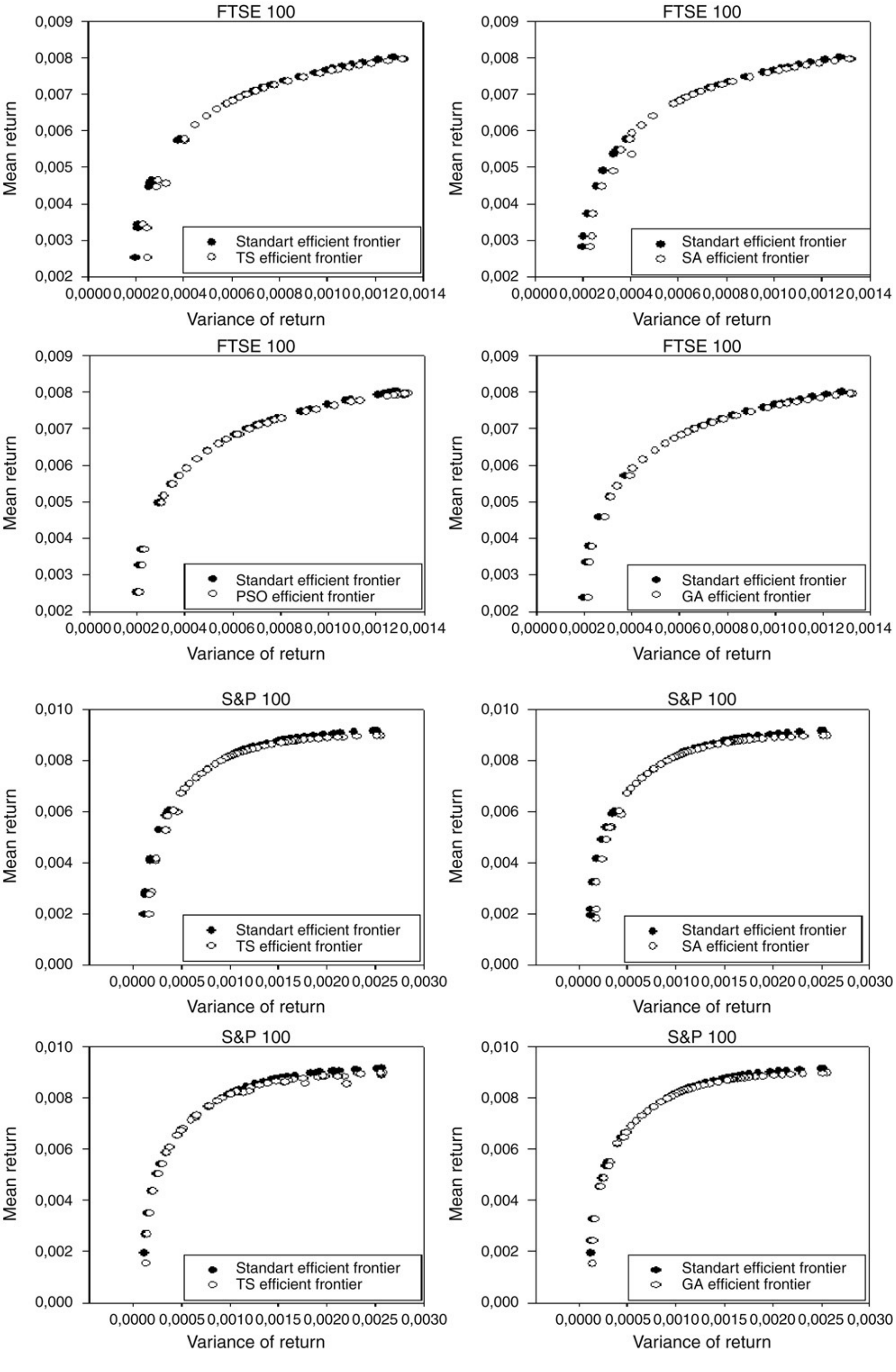

Fig. 5. Heuristic efficient frontiers for FTSE 100 and S&P 100.

Thus, let $\left(v^s_{i_j}, r^s_{i_j}\right)$ be the closest standard point to the heuristic point (v^h_j, r^h_j), where i_j is defined as:

$$i_j = \underset{i=1,\ldots,2000}{\arg\min} \left(\sqrt{\left(v^s_i - v^h_j\right)^2 + \left(r^s_i - r^h_j\right)^2} \right) \quad j = 1, \ldots, \xi. \tag{24}$$

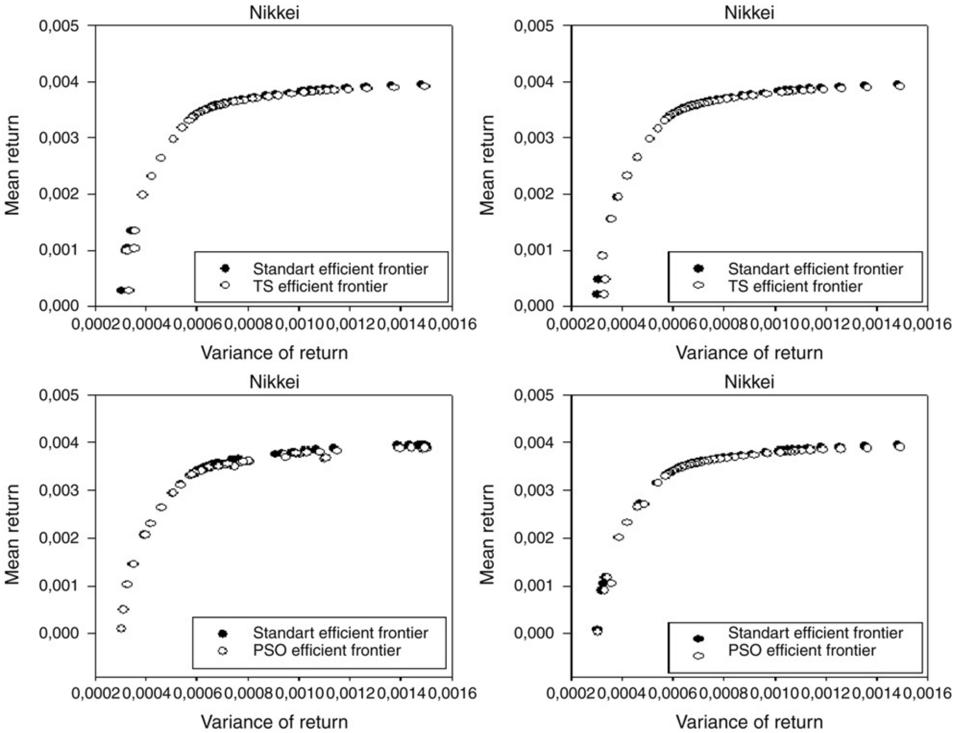

Fig. 6. Heuristic efficient frontiers for Nikkei.

Thus, we have defined mean Euclidian distance as:

$$\left(\sum_{j=1}^{\xi}\sqrt{\left(v_{i_j}^{s}-v_j^{h}\right)^2+\left(r_{i_j}^{s}-r_j^{h}\right)^2}\right)\Big/\xi.$$

Inspired by the analysis described by Fernandez and Gomez [5], we have merged the heuristic efficient frontiers into one (see Fig. 7), and we have removed the dominated solutions from it. Thus, the contribution percentage measure, for any given result of the heuristic algorithms, gives us the proportion of the surviving points to the entire merged heuristic efficient frontier. Note that, the neural network approach [5] gives similar solutions to those of this study. In other words, it finds better solutions when dealing with portfolios that demand low risk of investment policies as well. However, the neural network approach cannot find better solutions for the Hang Seng benchmark problem.

The other two measures, which are variance of return error and mean return error, have been defined as:

$$\left(\sum_{j=1}^{\xi}100\left|v_{i_j}^{s}-v_j^{h}\right|/v_j^{h}\right)\times\frac{1}{\xi}$$

and

$$\left(\sum_{j=1}^{\xi}100\left|r_{i_j}^{s}-r_j^{h}\right|/r_j^{h}\right)\times\frac{1}{\xi}$$

respectively.

5. Conclusion

This study was focused on solving the portfolio selection problem and tracing out its efficient frontier. A Markowitz-based cardinality constrained mean–variance model that includes cardinality and bounding constraints was used to develop a particle swarm optimization-based heuristic method. The results were compared to those obtained from heuristic methods based on (1) genetic algorithms, (2) tabu search and (3) simulated annealing.

The experimental results have shown that none of the four heuristics has clearly outperformed the others in all kinds of investment policies. However, Fig. 7 shows that, when dealing with problem instances that demand portfolios with a low risk of investment, the particle swarm optimization model gives better solutions than the other heuristic methods.

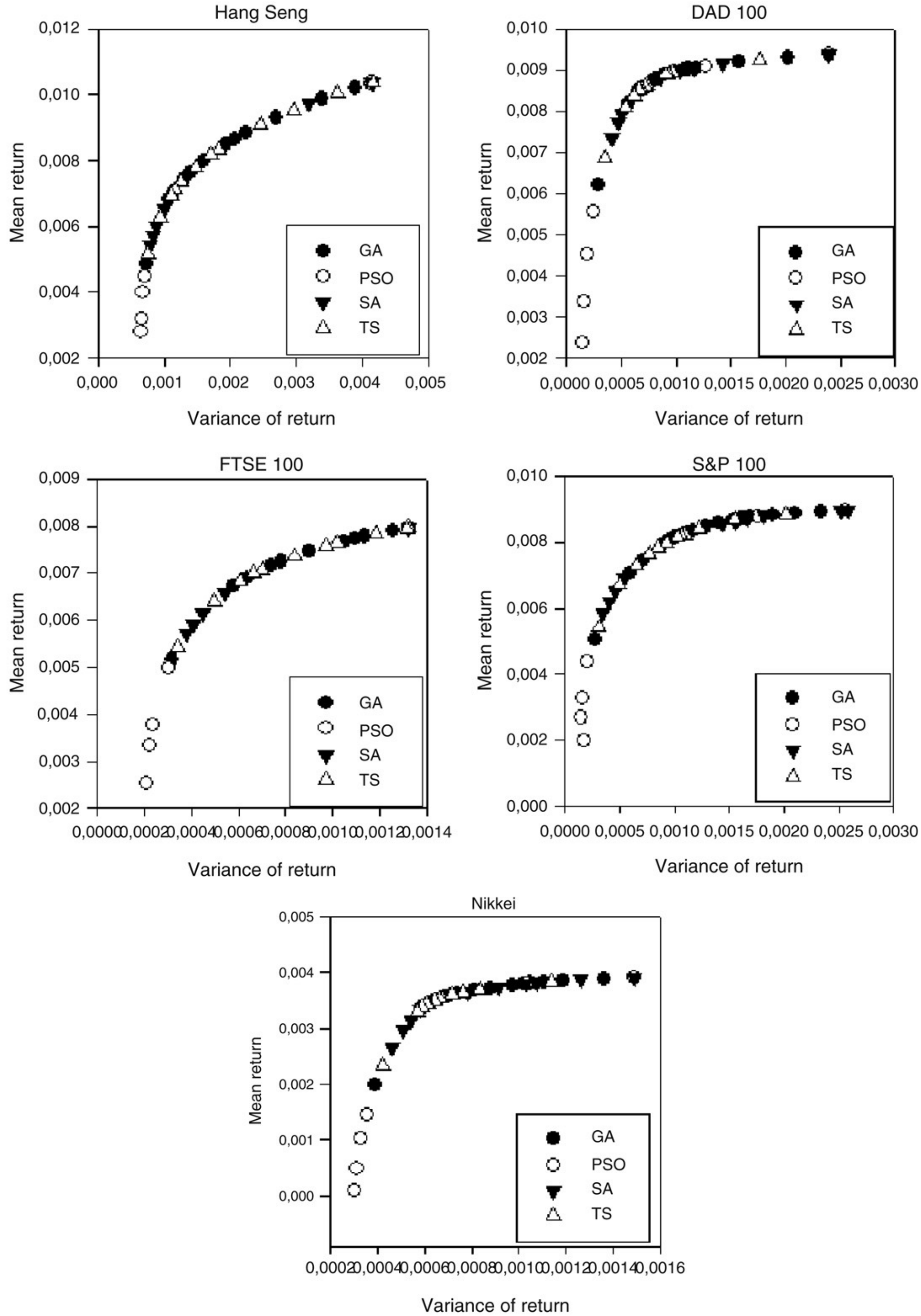

Fig. 7. Contributions to the merged efficient frontiers.

References

[1] J. Kennedy, R.C. Eberhart, Particle Swarm Optimization, in: Proc. IEEE Int'l Conf. on Neural Networks, vol. IV, 1995, pp. 1942–1948.
[2] H.M. Markowitz, Portfolio selection, Journal of Finance (1952) 77–91.

[3] H. Konno, H. Yamazaki, Mean-absolute deviation portfolio in optimization model and its application to Tokyo stock market, Management Science 37 (1991) 519–531.
[4] J. Kennedy, R.C. Eberhart, Y. Shi, Swarm Intelligence, Morgan Kaufmann, San Mateo, CA, SA, 2001.
[5] A. Fernandez, S. Gomez, Portfolio selection using neural networks, Computers & Operations Research 34 (2007) 1177–1191.
[6] K.J. Oh, T.Y. Kim, S. Min, Using genetic algorithm to support portfolio optimization for index fund management, Expert Systems with Applications 28 (2005) 371–379.
[7] M. Liu, Y. Gao, An algorithm for portfolio selection in a frictional market, Applied Mathematics and Computation 182 (2006) 1629–1638.
[8] R. Mansini, M.G. Speranza, Heuristic algorithms for the portfolio selection problem with minimum transaction lots, European Journal of Operational Research 114 (1999) 219–233.
[9] T.J. Chang, N. Meade, J.E. Beasley, Y.M. Sharaiha, Heuristics for cardinality constrained portfolio optimization, Computers & Operations Research 27 (2000) 1271–1302.
[10] X. Yang, Improving portfolio efficiency: A genetic algorithm approach, Computational Economics 28 (2006) 1–14.
[11] Y. Crama, M. Schyns, Simulated annealing for complex portfolio selection problems, European Journal of Operational Research 150 (2003) 546–571.
[12] U. Derigs, N.H. Nickel, On a local-search heuristic for a class of tracking error minimization problems in portfolio management, Annals of Operations Research 131 (2004) 45–77.
[13] U. Derigs, N.H. Nickel, Meta-heuristic based decision support for portfolio optimization with a case study on tracking error minimization in passive portfolio management, OR Spectrum 25 (2003) 345–378.
[14] F. Schlottmann, D. Seese, A hybrid heuristic approach to discrete multi-objective optimization of credit portfolios, Computational Statistics & Data Analysis 47 (2004) 373–399.
[15] U. Celikyurt, S. Ozekici, Multiperiod portfolio optimization models in stochastic markets using the mean-variance approach, European Journal of Operational Research 179 (2007) 186–202.