Neurocomputing 98 (2012) 90–100

Contents lists available at SciVerse ScienceDirect

Neurocomputing

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

Bacterial foraging based approaches to portfolio optimization with liquidity risk

Ben Niu [a,b,c,*], Yan Fan [a], Han Xiao [a], Bing Xue [d]

[a] College of Management, Shenzhen University, Shenzhen 518060, China
[b] e-Business Technology Institute, The University of Hong Kong, Hong Kong, China
[c] Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031, China
[d] Evolutionary Computation Research Group, Victoria University of Wellington, Wellington, New Zealand

ARTICLE INFO

Available online 6 June 2012

Keywords:
Bacterial foraging optimization
Genetic algorithms
Particle swarm optimization
Portfolio optimization
Liquidity risk

ABSTRACT

This paper proposes a bacterial foraging based approach for portfolio optimization problem. We develop an improved portfolio optimization model by introducing the endogenous and exogenous liquidity risk and the corresponding indexes are designed to measure the endogenous/exogenous liquidity risk, respectively. Bacterial foraging optimization (BFO) is employed to find the optimal set of portfolio weights in the improved Mean-Variance model. BFO-LDC which is a modified BFO with linear deceasing chemotaxis step is proposed to further improve the performance of BFO. With four benchmark functions, BFO-LDC is proved to have better performance than the original BFO. And then comparisons of the results produced by BFO, BFO-LDC, particle swarm optimization (PSO), and genetic algorithms (GAs) for the proposed portfolio optimization model are presented. Simulation results show that BFOs can obtain both near optimal and the practically feasible solutions to the liquidity risk portfolio optimization problem. In addition, BFO-LDC outperforms BFO in most cases.

1. Introduction

Portfolio optimization (PO) is an investment problem which tries to maximize the expected return by selecting a proper combination of various securities among a large number of them in financial industry [1]. PO is a NP-hard and non-linear problem with many local optima. A lot of work has been done which attempted to solve this problem by a variety of techniques, such as cutting planes, interior point methods, decomposition etc, but exact solution methods failed to solve large-scale instances of the problem. The advent of evolutionary computation (EC) inspired as a new technique for optimal selection of portfolio assets. A number of different evolutionary computation approaches have been proposed to solve this problem, including genetic algorithms [2], simulated annealing [3], neural networks [4] and others [5–7].

Recently, bacterial foraging optimization (BFO) has emerged as a powerful technique for optimization problems [8–10]. It has been successfully applied to solve many real world problems like harmonic estimation [11], transmission loss reduction [12], active power filter for load compensation [13], power network [14], load forecasting [15], and stock market prediction [16]. In BFO, each bacterium updates its position using chemotaxis, swarming, reproduction, elimination, and dispersal. Among them the chemotaxis procedure is a key step in BFO.

In [17], we focused on investigating the ability of BFO to achieve high-quality solutions to general PO problem and also a new and more complicated PO problem, which was under liquidity risk environment. The liquidity risk is one of the most important adjustable parameters in PO problem. Bangia et al. [18] proposed a novel portfolio optimization model considering both the liquidity risk and the market risk by VAR, which named BDSS model and Heude–Wynendaele model. Consigi [19] studied the mean-VAR model in the case of fat tailed distribution. Berkowitz [20] applied the VAR to measure the bank liquidity risk. Anil et al. [21] constructed a new liquidity risk model with implications for market risk, but they ignored the endogenous liquidity risk.

In [17], we considered the difference between Chinese securities market (a centralized auction system) and foreign securities market (a market-maker system). An improved model using VAR to measure both market and liquidity risk was proposed, and three evolutionary computation techniques (GA, PSO and BFO) were applied to solve the new model with complex constrains. The comparative results illustrated that the improved PO model and the performance of BFO was relatively efficient. However, the improved PO model is a non-linear and complex optimization problem, the results produced by all the three approaches (GA, PSO and BFO) are not optimal solutions.

* Corresponding author at: College of Management, Shenzhen University, Shenzhen 518060, China. Tel.: +86 755 2653 4489; fax: +86 755 2653 5450.
E-mail addresses: drniuben@gmail.com, drniuben@hotmail.com, niuben@hku.hk (B. Niu).

0925-2312/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2011.05.048

To solve this mixed-integer non-linear programming (NP-hard) more efficiently, we propose a modified BFO with the chemotaxis step varying dynamically as linear functions of iterations in this paper. In original BFO the chemotaxis step length is set as a constant value. There is not any mechanism to keep the balance of global search and local search. This restricts BFO to be applied to complex optimization problems. Our proposed method employed a linear decreasing chemotaxis step strategy, which allowed each bacterium kept a good balance between exploration and exploitation by decreasing its run-length unit linearly.

In order to demonstrate the performance of the proposed algorithm, BFO-LDC is applied into the improved portfolio optimization model and compared the test results with those of the original BFO, GA and PSO.

The rest of the paper is organized as follows. Section 2 gives a review of BFO and a description of the proposed algorithm BFO-LDC. In Section 3, it will be shown that BFO-LDC outperforms BFO on four benchmark functions. Section 4 describes the improved portfolio optimization model and a detailed design algorithm of BFO approaches for liquidity risk portfolio optimization. In Section 5, experimental settings and experimental results are given. Finally, Section 6 concludes the paper.

2. BFO and BFO-LDC

2.1. Bacterial foraging optimization (BFO)

Based on the biology and physics underlying the foraging behavior of *Escherichia coli*, Liu and Passino [9] exploited a variety of bacterial swarming and social foraging behaviors, discussed how the control system on the *E. coli* dictated and how foraging should proceed. In the bacterial foraging process, four motile behaviors (chemotaxis, swarming, reproduction, and elimination and dispersal) are mimicked.

(1) Chemotaxis: This process simulates the movement of an *E. coli* cell through swimming and tumbling via flagella. Biologically an *E. coli* bacterium can move in two different ways. It can swim for a period of time in the same direction or it may tumble, and alternate between these two models of operation for a run lifetime. Supposed $\theta^i(j,k,l)$ represents the ith bacterium at jth chemotactic kth reproductive and lth elimination and dispersal step. $C(i)$ is the size of the step taken in the random direction specified by the tumble (run length unit). Δ indicates a vector in the random direction whose elements lie in $[-1, 1]$. Then in computation chemotaxis the movement of the bacterium may be represented by

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{1}$$

(2) Swarming: *E. coli* cells can cooperatively self-organize into highly structured colonies with elevated environmental adaptability using an intricate communication mechanism (e.g., quorum-sensing, chemotactic signaling and plasmid exchange). Generally speaking, the cells provide an attraction signal to each other so they swarm together. The mathematical representation for swarming can be represented by

$$J_{CC}(\theta, p(j,kl)) = \sum_{i=1}^{n} J_{CC}^{i}(\theta, \theta^i(j,k,l))$$
$$= \sum_{i=1}^{n}\left[h_{repelent}\exp\left(-w_{repelent}\sum_{m=1}^{P}(\theta_m - \theta_m^i)^2\right)\right]$$

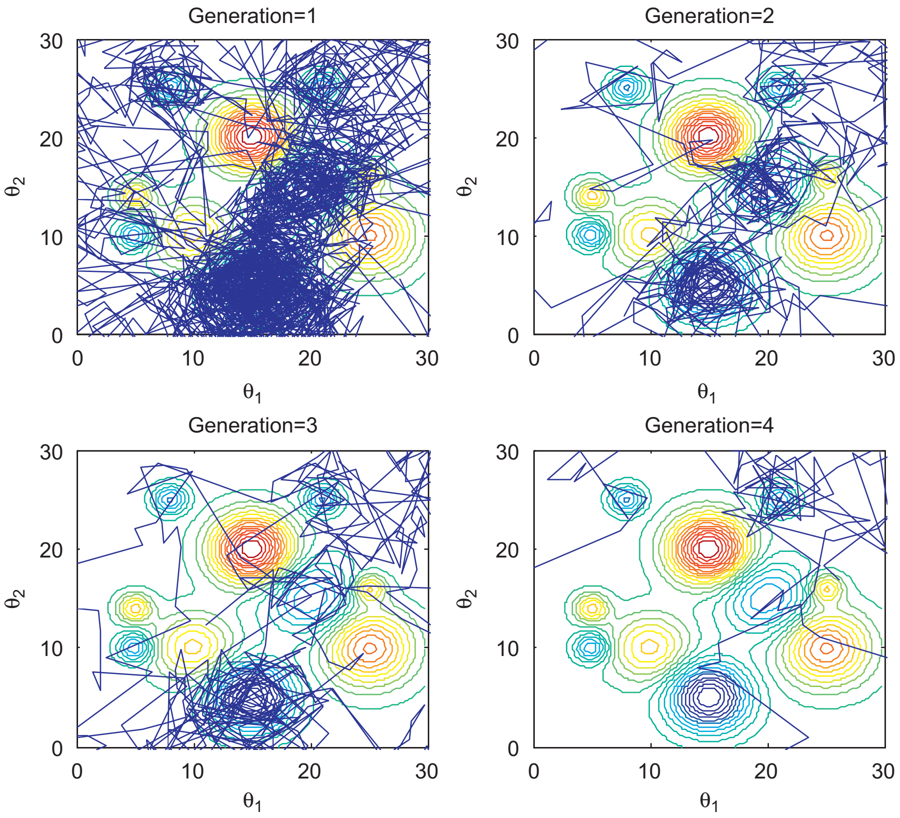

Fig. 1. Bacterial trajectories with $C = 1.5$.

$$+\sum_{i=1}^{n}\left[-d_{attract}\exp\left(-w_{attract}\sum_{m=1}^{P}(\theta_m-\theta_m^i)^2\right)\right] \tag{2}$$

where $J_{cc}(\theta,P(j,k,l))$ is the cost function value to e added to be added to the actual cost function to be minimized to present a time varying cost function, S is the total number of bacteria, P is the number of parameters to be optimized which are presented in each bacterium, and $d_{attract}$, $w_{attract}$, $h_{replent}$, and $w_{replent}$ are different coefficients that should be chosen properly.

(3) Reproduction: The most unhealthy bacteria die and each of the other healthier bacteria each splits into two bacteria, which replace the locations. This makes the population of bacteria constant.

(4) Elimination and dispersal: It is possible that in local environment, the lives of a population of bacteria change either gradually (e.g., via consumption of nutrients) or suddenly due to some other influences. Events can occur such that all the bacteria in a region are killed or a group disperse and emerge in a new part of the environment. They have the effect of possibly destroying the chemotactic progress, but they also have the effect of assisting in chemotaxis, since dispersal may place bacteria near the good food sources. From a broad perspective, elimination and dispersal are parts of the population-level long-distance motile behavior.

2.2. BFO with linear deceasing chemotaxis step (BFO-LDC)

To illustrate the importance of chemotaxis step length, a two-dimensional fitness function was adopted from Passino [8] as the benchmark function and parameters were set to be the same. A comparison between the bacterial motion trajectories using different chemotaxis step lengths was presented in Figs. 1–3, respectively. From the figures, it can be concluded that when chemotaxis step length C was too large ($C=1.5$) bacteria failed to locate the global optimum by swimming without stop (Fig. 1). While C was too small ($C=0.01$), it took a long time for the swarm to find the global optimum (Fig. 2). An expected result with faster searching time and global optimal orientation can be obtained when intermediate chemotaxis step length ($C=0.1$) was responsible (Fig. 3).

In the original BFO, the chemotaxis step length C is a constant. So, it is hard to maintain a balance between global and local search ability, and this influences the accuracy and speed of the search.

We proposed a scheme to modulate the chemotaxis step size with a view to improve its convergence behavior without imposing additional requirements in terms of numbers of evaluations. The step length was adjusted during reproduction and elimination to shorten the time of approaching the global optimum at the beginning, and to improve the accuracy in the end.

In our proposed method, a linearly decreasing chemotaxis step length is used over iterations. It starts with a large value C_{max} and linearly decreases to C_{min} at the maximal number of iterations. The mathematical representation of this strategy is shown as following:

$$C(i,k,l)=C_{min}+\frac{l-j}{j}(C_{max}-C_{min}) \tag{3}$$

where $C(i,k,l)$ is the chemotaxis step length of ith bacterium at the kth $(0<k<K)$ reproduction loop in the lth $(0<l<L)$ elimination-dispersal event. J is the maximal number of iterations (chemotaxis step), j is the current number of iterations. With $C_{min}=C_{max}$, the system becomes a special case of fixed chemotaxis step length, as the original proposed BFO algorithm. From hereafter, this BFO algorithm will be referred to as BFO-LDC. Pseudocode for the BFO-LDC algorithm is listed in Table 1.

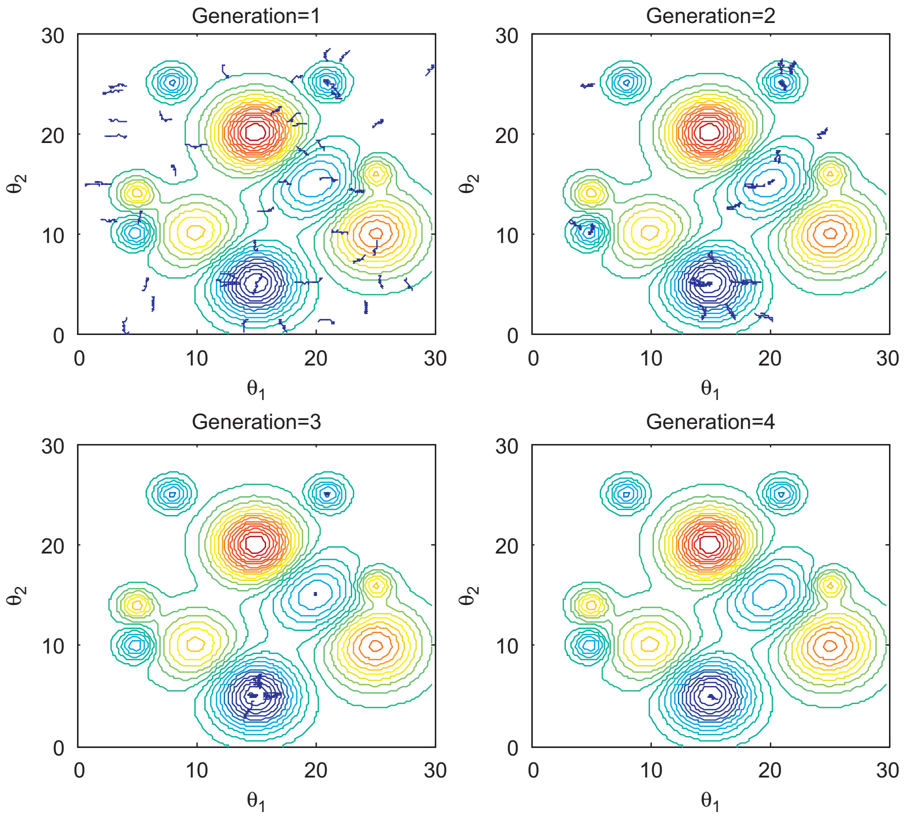

Fig. 2. Bacterial trajectories with $C=0.01$.

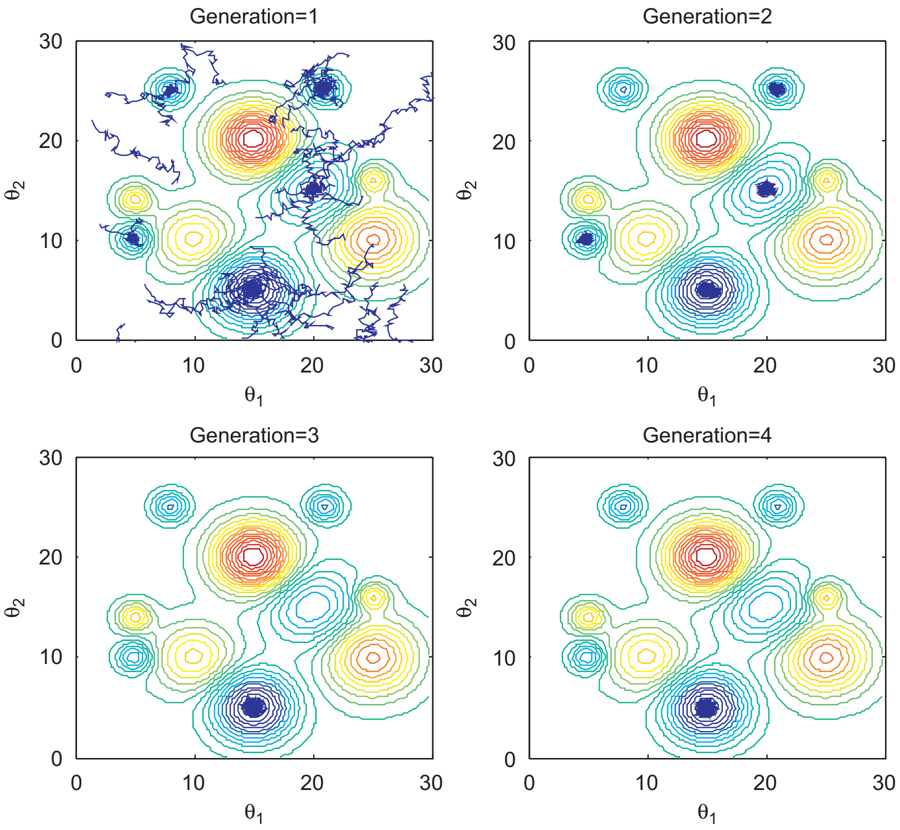

Fig. 3. Bacterial trajectories with $C=0.1$.

Table 1
Pseudocode for the BFO-LDC algorithm.

```
FOR (l = 1:L)
  FOR (k = 1:K)
    FOR (j = 1:J)
      FOR each bacterium i
        Tumble: Generate a random vector Δ(i) ∈R^p with each
element $\Delta_m(i)$, m = 1, 2, ..., D a random number on [−1, 1]
        Run: Let $\theta^i(h+1,k,l)=\theta^i(j,k,l)+C(i,k,l)\Delta(i)/\sqrt{\Delta^T\Delta}$
          $C(i,k,l)=C_{\min}+\frac{J-j}{j}(C_{\max}-C_{\min})$
          Swim: Let m = 0 (counter for swim length)
            While (m < N_s) m = m + 1.
            If J(i,j,k,l) < J_last, then
            J_last = J(i, j+1, k, l);
            $\theta^i(j+1,k,l)=\theta^i(j,k,l)+C(i)\Delta(i)/\sqrt{\Delta^T\Delta}$
            Calculate the new J(i, j+1, k, l) using $\theta^i(j+1,k,l)$
              Else
              let m=N_s.
            END
          END
        END
      END
    END
END
```

3. Benchmark functions

A set of well-known benchmark functions that are extensively used to compared both BFO-type and non-BFO-type bio-heuristic algorithm, were used to evaluate the performance, both in terms of solution quality and convergence rate, of the proposed algorithm. Our test suite includes 4 well-known benchmark functions, which present different difficulties to the algorithms to be evaluated. These benchmark functions can be grouped into unimodal functions (f_1, f_2) and multimodal functions (f_3, f_4). For each of these functions, the goal is to find the global minimizer. The following functions were used:

1. Sphere function

$$f_1(x)=\sum_{i=1}^{n}x_i^2 \tag{4}$$

2. Rosenbrock function

$$f_2(x)=\sum_{i=1}^{n}100\times(x_{i+1}-x_i^2)^2+(1-x_i)^2 \tag{5}$$

3. Rastrigin function

$$f_3(x)=\sum_{i=1}^{n}100\times(x_{i+1}-x_i^2)^2+(1-x_i)^2 \tag{6}$$

4. Griewank function

$$f_4(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1 \tag{7}$$

Table 2 lists the dimension of each function, their global minimizer, the ranges of their search space and iterations. An asymmetrical initialization procedure was used in this paper following the work reported in [22], in which the population was initialized only in a portion of the search space. This is to prevent a centre-seeking optimizer from "accidentally" finding the global optimum.

Table 2
Parameter settings for benchmark functions.

Function	Dim	Minimum value	Range of search	Iterations
f_1	15	0	$[-100, 100]^n$	1000
f_2	15	0	$[-100, 100]^n$	10,000
f_3	15	0	$[-5.12, 5.12]^n$	10,000
f_4	15	0	$[-600, 600]^n$	10,000

Table 3
Parameter settings for BFO-LDC.

f	N_c	N_{re}	N_{ed}	$C_{\max}$	$C_{\min}$	N_s
f_1	1000	5	2	0.2	0.01	4
f_2	1000	5	2	0.2	0.01	4
f_3	1000	5	2	0.1	0.01	4
f_4	1000	5	2	0.6	0.001	4

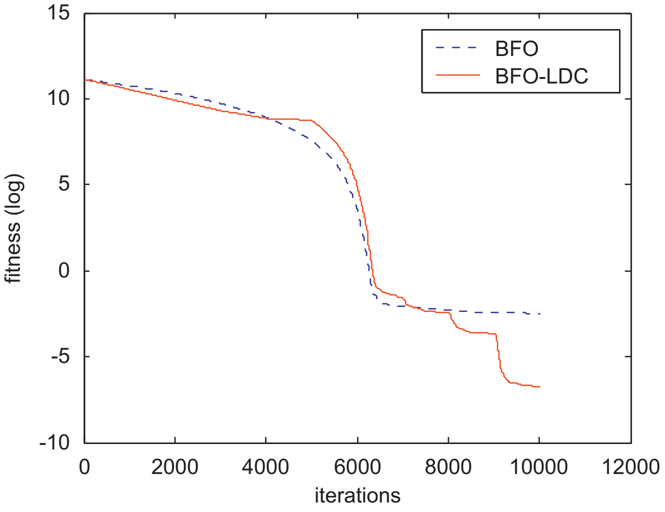

Fig. 4. Convergence curve of sphere function.

The performance of BFO-LDC was compared with standard BFO. In BFO, for all the benchmark functions,$N_c = 1000$, $N_{re} = 5$, $N_{ed} = 2$, and $N_s = 4$. The dimension size and the maximum number of iterations are set to 15 and 10,000 for all functions. The chemotaxis step length is critical for the convergence behavior of BFO. A suitable value for the chemotaxis step length usually provides a balance between global and local exploration abilities and consequently results in a better optimum solution. In BFO-LDC, a linearly decreasing chemotaxis step length is used which started at $C_{\max}$ and ends at $C_{\min}$, and their values are shown in Table 3. The results reported in this section were averaged over 10 simulations.

Figs. 4–7 show the comparison of the convergence curves of basic BFO and BFO-LDC during 10,000 generations for f_1, f_2, f_3 and f_4, respectively. From these figures, BFO-LDC kept on optimizing towards a better fitness, whereas the BFO stagnated and flattened out with no further improvement. The results indicate superiority in terms of speed of convergence for our proposed BFO-LDC algorithm for the four classical test functions, without sacrificing accuracy, especially the most complex optimization problem function f_4.

The stimulation results for the proposed BFO-LDC algorithm and basic BFO for four benchmark problems are shown in Tables 4–7. For each function, our improved BFO algorithm obtains better results with the comparison of basic BFO.

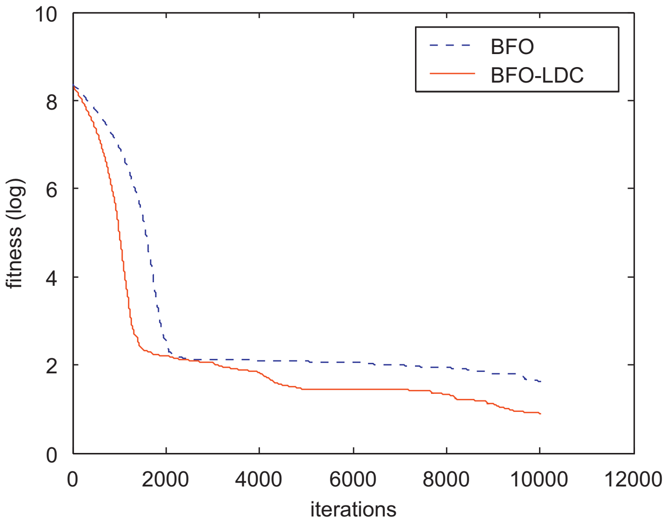

Fig. 5. Convergence curve of Rosenbrock function.

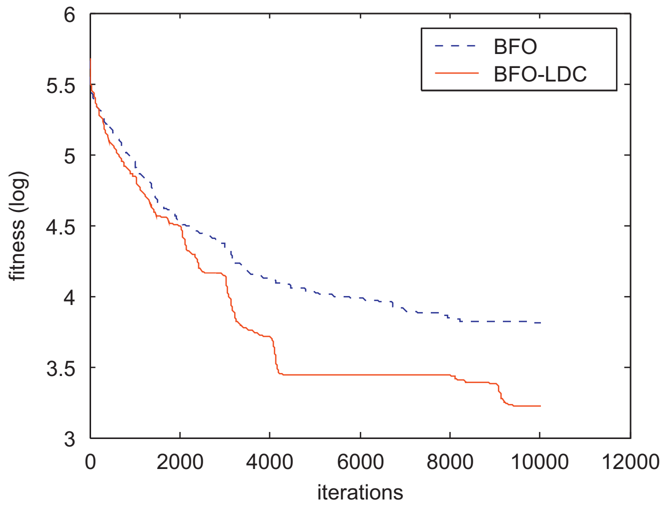

Fig. 6. Convergence curve of Rastrigin function.

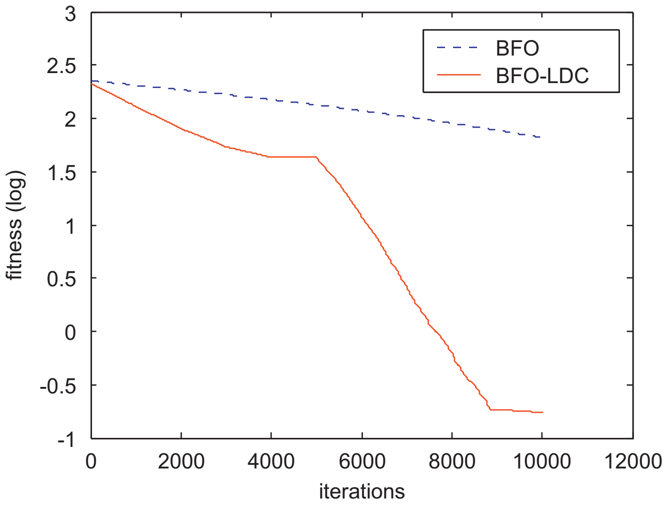

Fig. 7. Convergence curve of Griewank function.

Table 4
Numerical results for sphere function.

Algorithms	Best	Worst	Mean	Std
BFO	0.0637	0.1005	0.0800	0.0122
BFO-LDC	9.4411e−004	0.0014	0.0012	1.3828e−004

Table 5
Numerical results for Rosenbrock function.

Algorithms	Best	Worst	Mean	Std
BFO	22.2595	107.3795	42.9600	29.8036
BFO-LDC	5.0536	11.4833	8.0572	2.2018

Table 6
Numerical results for Rastrigin's function.

Algorithms	Best	Worst	Mean	Std
BFO	37.1416	49.9942	45.1861	3.4579
BFO-LDC	18.4889	31.1916	25.0785	4.6219

Table 7
Numerical results for Griewank function.

Algorithms	Best	Worst	Mean	Std
BFO	45.6331	99.1808	65.9662	17.8595
BFO-LDC	0.0080	1.4884	0.1731	0.4624

4. Liquidity risk portfolio optimization based on BFO approaches

4.1. Liquidity risk portfolio optimization

The liquidity is the vitality of stock market. It is regarded as an important symbol of the maturity also a significant quality indicator of the stock market. The market provides sufficient liquidity to help investors to change their assets to cash. But it is not liquid all the time. It has friction and there must be liquidity risk. Modern portfolio analysis started from the work of Markowitz [23] who proposed the original Mean-Variance model, then Alexander and Baptista [24] established a new model named mean-VAR to measure portfolio problem, using VAR instead of the variance. However, the conditional portfolio problem only considered the market risk and the liquidity risk was not involved. Ignoring liquidity risk or lack of liquidity risk management awareness will cause huge economic losses. For example, some of the most famous victims are Long Term Capital Management (LTCM), British Baring bankruptcy, and "Shan Yi" securities firm in Japan collapsed. So we proposed a new model with considering the liquidity risk.

Liquidity is another important feature of the securities as well as the price volatility. It is generally believed a price balance ability when a lot of trading in securities. According to their source, liquidity risk can be divided into two categories: exogenous liquidity risk and endogenous liquidity risk.

(1) Exogenous liquidity risk is a result of market characteristics. It is common to all market players and unaffected by the actions of any participant. Market determines the exogenous liquidity risk, which produces the same impact to each market participant and is not subject to the influence of individual traders' behavior. Heavy trading volume, small and stable bid-ask spreads, stable and high levels of quote depth, which make exogenous liquidity good. So the risk is small.

(2) Endogenous liquidity risk, in contrast, is specific to one's position in the market, varies across market participants, and the exposure of any single participant is affected by his/her actions. It is mainly driven by the size of the position: the larger the size, the greater the endogenous liquidity risk.

Usually, liquidity risk was measured from four aspects: the width, depth, speed and flexibility. Traditionally, BDSS model (Bangia et al. [18]) focused on quantifying exogenous liquidity risk rather than endogenous liquidity risk, and it used the following index to measure the exogenous risk:

$$ECL = \tfrac{1}{2}P_t(\overline{s}+\alpha\overline{\delta}) \tag{8}$$

$\overline{s}$ is the average *relative spread* (where relative spread, a normalizing device which allows for easy comparison across different instruments, is defined as form: $\overline{s}=1/k\sum_{i=1}^{k}(ask_i-bid_i)/(ask_i+bid_i)/\ 2)$. $\overline{\delta}$ is the volatility of relative spread. α is the scaling factor such that we achieve roughly a 99% probability coverage. P_t is today's mid-price for the asset or instrument.

But, usually the intraday data is not easy to get, and the data is not important. Furthermore, Chinese securities market is different from that of the foreign ones. It is an auction system, and so we make these indexes to measure the liquidity risk: max daily price, min price, close price and the turnover rate.

$$V = \frac{p_h-p_l}{p_c}\frac{1}{to} \tag{9}$$

p_h, p_l, and p_c are the max, min and close daily price, respectively, to is the relation turnover rate, V is only used to measure the exogenous risk. It does not contain the trading volume and size. So the above index cannot measure the endogenous risk. We use the daily volume to weight the max, min and close price

$$P_i = \sum_{n-1}^{3} p_i\ vol(n)/[vol(1)+vol(2)+vol(3)] \quad i=h,l,c \tag{10}$$

$vol(n)$ is the daily volume $n=1,\ 2,\ 3$, P_t is the weighted price i = max, min, *close*.

Using the daily volume weighted price to replace the original price, it takes the endogenous and exogenous liquidity risk into account. Liquidity risk index formula is as follows:

$$ECL = \frac{1}{2}P_t(\overline{WAV}+\alpha'\delta') \tag{11}$$

$\overline{WAV}$ is the average relative price, and $WAV=(P_h-P_l/P_c)(1/to)$.

In this paper, we assume that an investor allocates his/her wealth among n assets. Some notations are introduced as follows:

x_i is the proportion of the money used in the $i$thasset, and $\sum_{i=1}^{n}x_i=1$.
$x_i \geq 0$ means there is no short sales.
r_i is the yield of the ith asset.
$\sigma_{ij} = \text{cov}(r_i,\ r_j)$ is the covariance of r_i and r_j.
R_x means the expected rate of revenue of the portfolio.
λ is the risk-averse factor, which distributes in [0,1]. Smaller λ represents the investor could bear larger risk. We assume that the c is 99%, so the $\Phi^{-1}(c)=2.33$. Our improved portfolio optimization model can be formulated as

$$\begin{aligned}\min\quad F(x) &= \lambda[\Phi^{-1}(c)\delta_x - R_x + 1/2(\overline{WAV}+\alpha'\delta')] - (1-\lambda)\sum_{i=1}^{n}x_i r_i \\ &= \lambda\left[2.33\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}x_i x_j\sigma_{ij}} - \sum_{i=1}^{n}x_i r_i + 1/2\left(\overline{WAV}+2.33\sum_{i-1}^{n}\sum_{j=1}^{n}x_i x_j\sigma'_{ij}\right)\right]\end{aligned}$$

Table 8
The encoding of a bacterium.

θ_1, θ_2, …, θ_n	$F(x)$
The position of a bacterium in every dimension	Fitness function

$$-(1-\lambda)\sum_{i=1}^{n}x_i r_i$$

$$\text{s.t.}\begin{cases}\sum_{i=1}^{n}x_i=1\\ x_i\geq 0\end{cases} \tag{12}$$

4.2. BFO approaches for liquidity risk portfolio optimization

4.2.1. Encoding scheme

The way to encode a potential solution into a bacterium is a key issue in BFO based methods. We encode a potential solution of the proposed PO model as an n-dimensional vector, where each variable represents the holdings of asset i in the portfolio. We use the real-number encoding method to construct their indexes. In our solution, the bacterium location θ consists of sequence of real number, presenting the proportion of each assest x_i. Each dimension of the θ presents the proportion of a given assest, as shown in Table 8.

The encoding sheme for the bacteria colony P is showed in the following (the size is the population size of the bacterial colony):

$$P=\begin{bmatrix}\theta_{11} & \theta_{12} & \theta_{13} & \theta_{14} & \cdots & F(x_1)\\ \theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} & \cdots & F(x_2)\\ \theta_{31} & \theta_{32} & \theta_{33} & \theta_{34} & \cdots & F(x_4)\\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots\\ \theta_{size1} & \theta_{size2} & \theta_{size3} & \theta_{size4} & \cdots & F(x_{szie})\end{bmatrix} \tag{13}$$

4.2.2. Fitness function design

A fitness function must be used to evaluate the fitness value of each bacterium within the population of each generation. Bacterium with good position should get more opportunities to be selected as a parent, whereas poor ones may be eliminated. Within the context of the proposed PO model, the fitness function used is the expected shared surplus as described in Eq. (8). We also add a penalty term to the objective function for each violated constraint: the larger the violation of the constraint, the larger the increase in the value of the objective function. A portfolio which is unacceptable for the investor must be penalized enough to be rejected by the minimization process. The PO problems have two constraints: $\sum_{i=1}^{n}x_i=0$ and $x_i\geq 0$. It used a penalty term for considering the two constraints. The final fitness function is formed as follows:

$$\text{fitness}=F(x)+k_1f_1(x)+k_2f_2(x) \tag{14}$$

where $f_1(x)=\max(0,\quad \sum_{i=1}^{n}x_i-1-\text{Errgoal})$, Errgoal represents the infinitesimal (it is set to 1e−10 in our experiment). $f_2(x)=\max(-x_i,0)$ and $k_1=k_2=10^4$, $F(x)$ is the original fitness function (see Eq. (12)).

4.2.3. Initialization of parameters

This includes two sets of parameters: the parameters involved in the fitness function, and the parameters of BFOs including the population initialization. The initial population of the BFO or BFO-LDC is created as follows:

$$P=rand(p,s)\times(ub-lb)-lb \tag{15}$$

where p is the population size, S is the dimension of each bacterium that represents the number of the assets. ub and lb are upper and lower limits, respectively.

The parameters involved in the fitness function will be given in the experimental study, and the initialization of BFOs is now taken up:

(1) The population size $p=50$.
(2) The values of ub and lb are set to 1 and 0, respectively.
(3) The swimming length $N_s=4$.
(4) The number of iterations in a chemotactic loop j is set to 80.
(5) The number of reproduction steps K is set to 5.
(6) The number of elimination and dispersal events L is set to 2.
(7) The probability of elimination/dispersal p_{ed} is set to 0.25.
(8) Cell-to-cell signaling is not used. Therefore, the values of $d_{attract}$,$w_{attract}$,$h_{repelent}$, and $w_{repelent}$ are irrelevant.
(9) Two additional parameters of BFO-LDC: $C_{\max}$ and $C_{\min}$ are set to 0.2 and 0.01, respectively.
(10) The initial location of each bacterium, which is a function of several parameters, $f(p,s,K,L,J)$ is specified by a random number in the range [0,1].

5. Illustrative examples

5.1. Experimental settings

We choose four assets as the sample: PuFa Bank (600000), JiangXi Copper Industry (600362), ShangHai Automotive Industry (600124) and China Petrochemical Corporation (600028), which are from different industries, different places. The basic data about the assets were taken from January 1st in 2009 to December 30th in 2009, and we got the interrelated index value needed in the experiment based on them. We considered the different kinds of the investors, and four different risk factors λ are use to identify the different kinds inverstors. The relation number is set as follows:

$$\lambda=(0.15,0.4,0.6,0.85)$$
$$r=(0.003496,0.006156,0.007198,0.002933)$$

$$\sigma=\begin{matrix}[0.000821, & 0.00061, & 0.000352, & 0.000398,\\ 0.00061, & 0.001828, & 0.000597, & 0.000513,\\ 0.000352, & 0.000597, & 0.001174, & 0.000282,\\ 0.000398, & 0.000512, & 0.000282, & 0.000582]\end{matrix}$$

$$\sigma'=\begin{matrix}[0.000126, & 0.000125, & 0.000088, & -0.0000142,\\ 0.0000125, & 0.0000149, & 0.000056, & 0.000042,\\ 0.000088, & 0.0000556, & 0.001203, & 0.0007658,\\ -0.0000142, & 0.000042, & 0.0000765, & 0.00713]\end{matrix}$$

$$\overline{WAV}=(0.0355,\quad 0.00956,\quad 0.0831,\quad 0.10047)$$

For comparison purposes, four swarm based algorithms were used, which are PSO, GA, BFO and BFO-LDC. The parameters for BFOs are presented in the previous section, so here we only addressed the parameters setting of the other two algorithms, PSO and GA.

In PSO, the inertia weight is decreased linearly from 0.9 in the first iteration to 0.4 in the last iteration, and the acceleration constants $c_1=c_2=2$. In GA, the selection method is roulette wheel and the crossover method is one-point crossover. The crossover rate is set to 0.6 and mutation rate is set to 0.001. For fair comparison, the population size and maximum iterations are set as 80 and 800 for all the algorithms. A total of 15 runs for each experimental setting are performed. Because the values of all the data are small, all the results are multiplied by 100 in the following section.

5.2. Experimental results

Numerical results with different λ obtained by the standard PSO, GA and BFOs are showed in Tables 9–12. The final portfolio selection results are also listed in those tables. Figs. 4–7 present the mean relative performance with different λ generated by the four algorithms.

According to the tables and the figures, we can find that:

(1) The fitness value grows up according to the increase of the risk-averse factor λ, and this trend consists with the structure of the fitness function. From the data in the tables, it can be seen the profit reduces along with the rise of λ, so does the risk. When $\lambda = 0.15$, the investor can bear the highest risk, the value of the profit rate and the risk rate in Table 9 are the largest ones. The opposite results are produced when $\lambda = 0.85$.

(2) With the different λ, the proportion of the four assets is different.When λ is bigger, asset 3 posses smaller proportion, and asset 1 and asset 4 occupy higher proportions. The results show that high risk can be with the more profit, so the market can compensate for the liquidity risk, from which we know that liquidity risk must be considered when investing stocks.

Table 9
Numerical results with different λ.

Numerical results	$\lambda = 0.15$			
	BFO	PSO	GA	BFO-LDC
x_1	0.1792	0.1845	0.1051	0.1829
x_2	0.2529	0.1636	0.1168	0.3878
x_3	0.3511	0.4848	0.5516	0.2066
x_4	0.2169	0.1671	0.2265	0.2227
Profit	0.0053	0.0056	0.0057	0.0053
Risk	0.1255	0.1286	0.1420	0.1250
Max	1.0003	1.5104	1.7837	0.9677
Min	0.9782	1.1627	1.0530	0.9664
Mean	0.9846	1.3206	1.4431	0.9669
Std	0.0065	0.0086	0.2258	3.6000e−04

Table 10
Numerical results with different λ.

Numerical results	$\lambda = 0.4$			
	BFO	PSO	GA	BFO-LDC
x_1	0.4055	0.4157	0.3096	0.4557
x_2	0.2361	0.2468	0.2129	0.2068
x_3	0.2654	0.2409	0.3279	0.2409
x_4	0.0930	0.0966	0.1496	0.0966
Profit	0.0050	0.0050	0.0051	0.4878
Risk	0.1046	0.1036	0.1146	10.3488
Max	3.4871	5.2052	5.2609	3.3939
Min	3.4044	3.9536	3.5029	3.3928
Mean	3.4294	4.3355	4.5746	3.3932
Std	0.0246	0.1113	0.4870	3.4700e−04

Table 11
Numerical results with different λ.

Numerical results	$\lambda = 0.6$			
	BFO	PSO	GA	BFO-LDC
x_1	0.4714	0.5922	0.4411	0.5407
x_2	0.2654	0.2595	0.3556	0.2753
x_3	0.2133	0.1088	0.1004	0.1614
x_4	0.0499	0.0395	0.1029	0.0226
Profit	0.0049	0.0046	0.0047	0.0048
Risk	0.0983	0.0939	0.0992	0.0946
Max	5.3957	7.0581	9.0874	5.3234
Min	5.3326	5.8601	5.6425	5.3223
Mean	5.354	6.4033	6.9243	5.3226
Std	0.0183	0.1494	0.973	3.1600e-04

Table 12
Numerical results with different λ.

Numerical results	$\lambda = 0.85$			
	BFO	PSO	GA	BFO-LDC
x_1	0.7510	0.7544	0.6366	0.7503
x_2	0.2130	0.2034	0.3051	0.2400
x_3	0.0343	0.0040	0.0032	0.0007
x_4	0.0017	0.0202	0.0551	0.0090
Profit	0.0041	0.0040	0.0043	0.0041
Risk	0.0929	0.0916	0.0935	0.0928
Max	7.8803	10.0926	11.3107	7.7322
Min	7.7438	7.9064	8.0570	7.7314
Mean	7.7807	9.1335	9.8877	7.7317
Std	0.0380	0.4691	0.7987	2.0100e-04

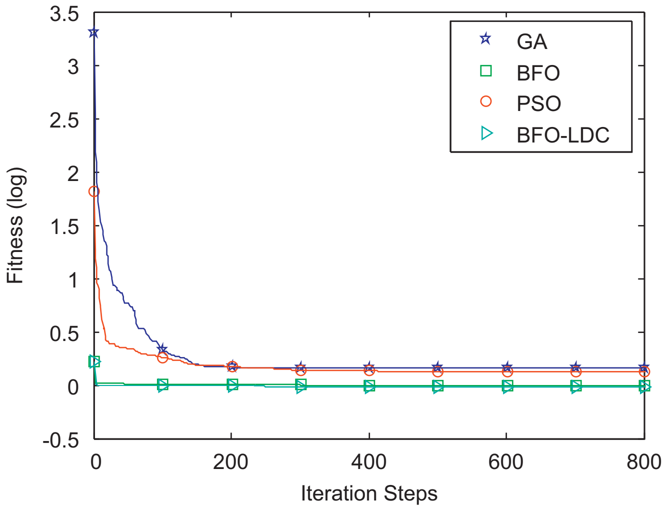

Fig. 8. $\lambda = 0.15$.

(3) From the data of the maximum value, minimum value and standard deviations in the tables and Figs. 8–11, it is clear that the results generated by BFO-LDC are the most stable (the smallest standard deviations) and most precise ones (the smallest mean fitness value). At the same time, it can be concluded that the results obtained by BFO-LDC are better than that of GA, PSO and BFO, and this explains the improvement of BFO is effective.

(4) Comparing the convergence graphs presented in Figs. 8–11, among these three algorithms, BFO-LDC and BFO are both superior to GA and PSO for all the test cases. BFO-LDC is highly competitive with BFO and usually produces better performance.

Added to the basic statistical tests (mean, average and standard deviation) presented in the above tables, analysis of variance (ANOVA) test was also carried out to validate the efficacy of all the four algorithms. The graphical analyses of four different λ are done through box plot, which are shown in Figs. 12–15. The box plot can provide an excellent visual summary of many important aspects of a distribution. The box stretches from the lower hinge (defined as the 25th percentile) to the upper hinge (the 75th percentile) and therefore contains the middle half of the scores in the distribution. The median is shown as a line across the box. Therefore, one-fourth of the distribution is between this line and

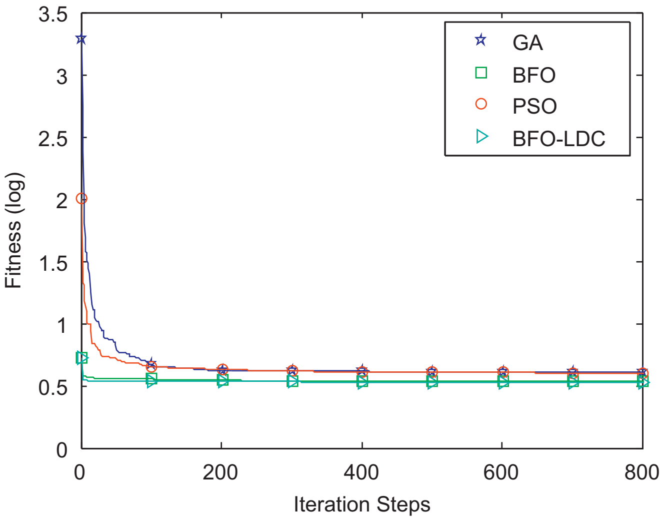

Fig. 9. $\lambda = 0.4$.

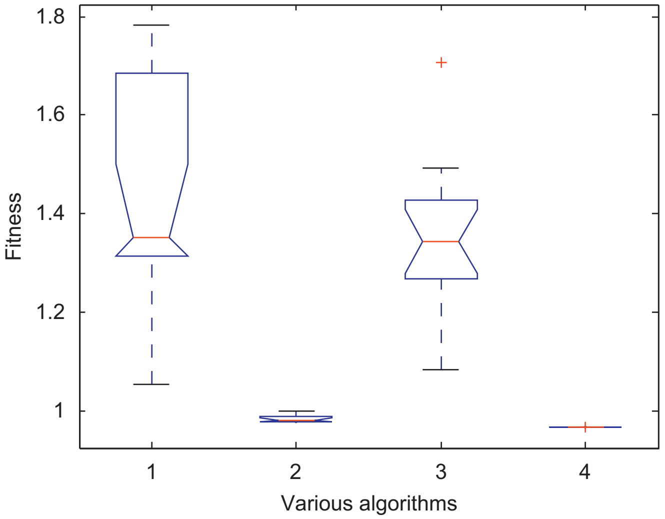

Fig. 12. $\lambda = 0.15$.

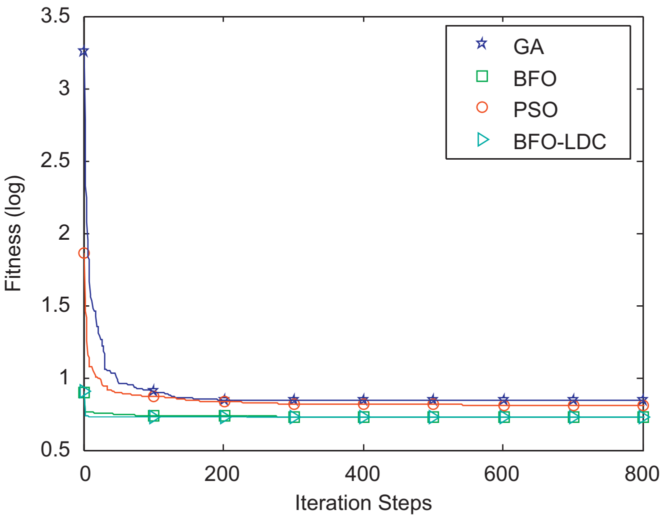

Fig. 10. $\lambda = 0.6$.

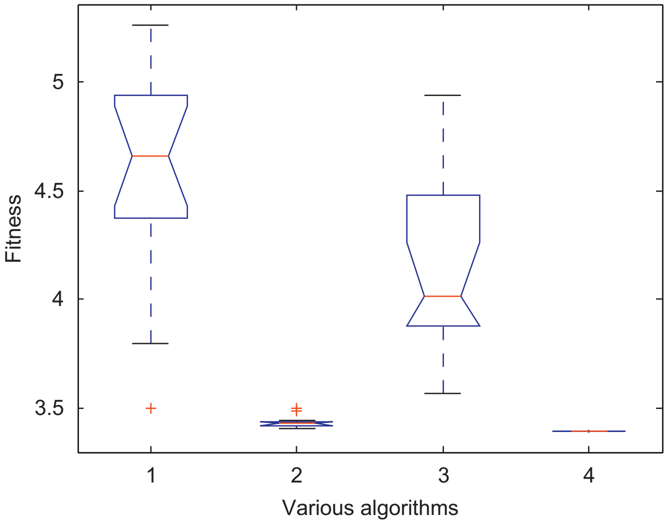

Fig. 13. $\lambda = 0.4$.

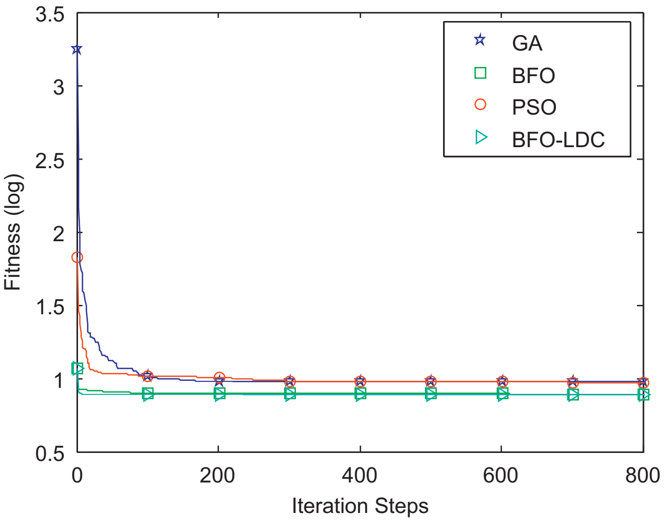

Fig. 11. $\lambda = 0.85$.

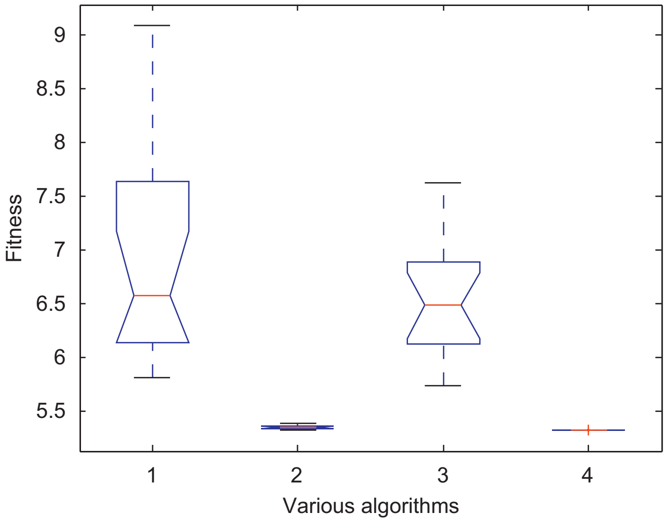

Fig. 14. $\lambda = 0.6$.

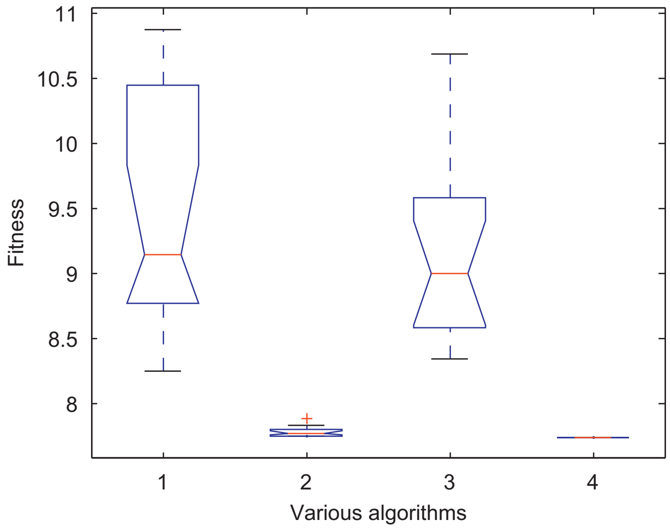

Fig. 15. $\lambda = 0.85$.

the top of the box and one-fourth of the distribution is between this line and the bottom of the box.

Figs. 12–15 present the graphical performance representation of all algorithms in 15 runs. From this box plot representation, it is clearly visible and proved that the BFO-LDC provided better results for the four test cases than that of GA, PSO and BFO. BFO yielded a slight worse result than that of BFO-LDC, while it gave a much better result than that obtained by GA and PSO.

6. Conclusions

In this paper, we focused on solving the portfolio optimization problem with liquidity risk using BFO based methods. Instead of using standard Mean-Variance model, we proposed a new model using VAR measuring both market and liquidity risk. The improved portfolio model is a non-linear and complex optimization problem which is much harder to be solved by conventional techniques.

We employ a relatively new swarm intelligence based method, BFO to solve this model. In additional, a linear decreasing chemotaxis step strategy is included in original BFO and thus an improved BFO algorithm BFO-LDC is proposed that is applied in the same problem. The obtained results indicate that the high convergence rate and accuracy shown in the experimental results illustrate the high performance of proposed BFO. Generally, BFO-LDC outperforms BFO. (Generally, BFO-LDC obtains better results than BFO.)

Our future works include to improve the performance BFO based on some biological mechanisms, to insert more real-world constraints into PO model and to study other risk measures.

Acknowledgment

This work is supported by National Natural Science Foundation of China (Grant no. 71001072), China Postdoctoral Science Foundation (Grant no. 20100480705), Science and Technology Project of Shenzhen (Grant no. JC201005280492), The Natural Science Foundation of Guangdong Province (Grant no. 9451806001002294), 863 Project (Grant no. 008AA04A105), and Project 801-000021 supported by SZU R/D Fund.

References

[1] F.D. Freitas, A.F. De Souza, A.R. De Almeida, Prediction-based portfolio optimization model using neural networks, Neruocomputing 72 (2009) 31–60.

[2] T.J. Chang, N. Meade, J.E. Beasley, Y.M. Sharaiha, Heuristics for cardinality constrained portfolio optimization, Comput. Oper. Res. 27 (2000) 1271–1302.

[3] Y. Crama, M. Schyns, Simulated annealing for complex portfolio selection problems, Eur. J. Oper. Res. 150 (2003) 546–571.

[4] A. Fernandez, S. Gomez, Portfolio selection using neural networks, Comput. Oper. Res. 34 (2007) 1177–1191.

[5] U. Derigs, N.H. Nickel, On a local-search heuristic for a class of tracking error minimization problems in portfolio management, Ann. Oper. Res. 131 (2004) 45–77.

[6] 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.

[7] F. Schlottmann, D. Seese, A hybrid heuristic approach to discrete multi-objective optimization of credit portfolios, Comput. Stat. Data Anal. 47 (2004) 373–399.

[8] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Syst. Mag. 22 (2002) 52–67.

[9] Y. Liu, K.M. Passino, Biomimicry of social foraging bacteria for distributed optimization: models, principles, and emergent behaviors, J. Optim. Theory Appl. 115 (2002) 603–628.

[10] B. Niu, Y.L. Zhu, X.X. He, H. She, Q.H. Wu, A. Lifecycle, Model for simulating bacterial evolution, Neurocomputing 72 (2008) 142–148.

[11] S. Mishra, A hybrid least square fuzzy bacterial foraging strategy for harmonic estimation, IEEE Trans. Evol. Comput. 9 (2005) 61–73.

[12] M. Tripathy, S. Mishra, L.L. Lai, Q.P. Zhang, Transmission loss reduction based on FACTS and bacteria foraging algorithm, Lect. Notes Comput. Sci. 4193 (2006) 222–231.

[13] S. Mishra, C.N. Bhende, Bacterial foraging technique based optimized active power filter for load compensation, IEEE Trans. Power Delivery 22 (2007) 457–465.

[14] M. Tripathy, S. Mishra, Bacteria foraging based to optimize both real power loss and voltage stability limit, IEEE Trans. Power Syst. 22 (2007) 240–248.

[15] M. Ulagammai, P. Venkatesh, P.S. Kannan, N.P. Padhy, Application of bacteria foraging technique trained and artificial and wavelet neural networks in load forecasting, Neurocomputing 70 (2007) 2659–2667.

[16] R. Majhi, G. Panda, G. Sahoo, P. Dash, D. Das, Stock market prediction of S&P 500 and DJIA using bacterial foraging optimization technique, in: Proceedings of the IEEE Congress on Evolutionary Computation, Singapore, IEEE Service Center, 2007, pp. 2569–2575.

[17] B. Niu, H. Xiao, L.J. Tan, Y. Fan, J.J. Rao, Liquidity risk portfolio optimization using swarm intelligence, Commun. Comput. Inf. Sci. 93 (2010) 551–558.

[18] A. Bangia, F.X. Diebold, T. Schuerman, Modeling liquidity risk with implications for traditional market risk measurement and management, J. Banking Finance 26 (1999) 445–474.

[19] G. Consigli, Estimation and mean-VAR portfolio selection in markets subject to financial instability, J. Banking Finance 26 (2006) 1355–1382.

[20] J. Berkowitz, Incorporating liquidity risk into value-at-risk models, J. Deriv. 27 (2000) 67–89.

[21] B. Anil, D. Francis, S. John, Modeling Liquidity Risk-with Implications for Traditional Market Risk Measurement and Management, Kluwer Academic Publishers, vol. 12, 1999, pp. 68–73.

[22] P.J. Angeline, Evolutionary optimization versus particle swarm optimization: philosophy and performance differences, Lect. Notes Comput. Sci. 1447 (1998) 601–610.

[23] H.W. Markowitz, Foundations of portfolio theory, J. Finance 46 (1991) 469–477.

[24] G. Alexander, J. Baptista, Economic implications of using a mean-VAR model for portfolio selection—a comparison with Mean-Variance analysis, J. Econ. Dyn. Control 26 (2004) 1159–1193.

Ben Niu received the B.S. degree in mechanical engineering from Hefei Union University, Hefei, China, in 2001, the M.S. degree in enterprise information management from Anhui Agriculture University, Hefei, China, in 2004, and the Ph.D. degree in mechanical & electrical engineering from Shenyang Institute of Automation of the Chinese Academy of Sciences, Shenyang, China, in 2008. From 2008 to the present, he has been with the Department of Management Science of Shenzhen University. His main fields of research are swarm intelligence, bio-inspired computing, multiobjective optimization and their applications on RFID system, business intelligence, and portfolio optimization.

Yan Fan obtained her B.S. degree in information management and information system from the College of Management, Tianjin Polytechnic University, China, in July 2008. She is currently pursuing her M.S. degree in management science and engineering in College of Management, Shenzhen University. Her research interests include bacterial foraging optimization and its application on portfolio optimization and RFID network planning.

Xiao Han obtained his combined bachelor's degree in management science and finance, from Finance and Economics University of Jiangxi, China, in July 2008. Currently he is a master student in College of Management, Shenzhen University. His research focus is computational intelligence, especially particle swarm optimization and bacterial foraging optimization.

Bing Xue obtained the B.S. degree in information management and information system from Henan University of Finance and Economics, Zhengzhou, China, in 2007 and the M.S. degree in management science and engineering from Shenzhen University, Shenzhen, China, in 2010. She is currently working toward her Ph.D. degree in school of engineering and computer science in Victoria University of Wellington, Wellington, New Zealand. Her current research interest focus on swarm intelligence algorithms such as bacterial foraging optimization and particle swarm optimization, both for the portfolio selection problems.