International Journal of Knowledge and Language Processing
Volume 6, Number 4, **2015**
www.ijklp.org
KLP International ⓒ2015 ISSN 2191-2734
pp.20–36

A novel personal-best information guided particle swarm optimization

Song Huang[1*], Na Tian[1,2], Yan Wang[1], Zhicheng Ji[1]
[1]Institute of Electrical Automation
Jiangnan University
Wuxi 214122, China

[2]Institute of Educational Informatization
Jiangnan University
Wuxi 214122, China
huangson2@163.com, tianna@jiangnan.edu.cn, wangyan_sytu @126.com, zcji@jiangnan.edu.cn

Received December 2015; Accepted December 2015

ABSTRACT. *Suffering from getting trapped in local optimum, researchers have developed numerous versions of particle swarm optimization (PSO) to prevent premature convergence. A personal-best information guided particle swarm optimization is proposed in this paper. Firstly, the global version and the local version of personal-best terms are defined to balance the exploration and exploitation and the modified cognition component is formed with a chosen strategy of the two terms. And then the original cognition component is replaced by the modified cognition component. The performance of the proposed algorithm is evaluated on ten benchmark functions and compares with some well-known PSO variants. Simulation results indicate that the proposed algorithm has statistically superior performance than other PSO variants.*

Keywords: swarm intelligence; particle swarm optimization; premature stagnation; personal-best

[*] Corresponding Author.

1. **Introduction.** Particle swarm optimization (PSO) is a stochastic, population-based optimization technique introduced by Kennedy and Eberhart in 1995 [1-2] and has numerous applications in data clustering problem [3-5], vehicles scheduling problem [6-9], economic and emission dispatch problem [10, 11] and so on. Das and Pattnaik et al. applied particle swarm optimization (PSO) to train Artificial Neural Network (the number of layers, input and hidden neurons, the type of transfer functions etc.) for the channel equalization [12]. Salim and Nabag et al. presented a study of particle swarm optimization (PSO) to identify the offline parameters of the Nexa 1.2 kW proton exchange membrane fuel cell (PEMFC) system [13]. After the optimal modeling strategy of solid oxide fuel cell (SOFC) is designed, Jiang and Wang et al. developed a simple and efficient barebone particle swarm optimization (BPSO) algorithm to determine the parameters of SOFC. For improve the performance, a hybrid learning strategy is proposed for BPSO and used to the SOFC model [14].

But in the complicated system, the applications of particle swarm optimization in these problems often trapped in local optima. On the purpose of efficiently avoiding premature stagnation, researchers developed many novel techniques to modify particle swarm optimization. In recent years, PSO with different topology strategies and new techniques has received increasing attention. Time-adaptive topology [16], increasing topology connectivity [18] and scale-free topology [19] have different optimization process and were employed into the PSO [16, 18, 19]. New techniques from other fields, such as competitive and cooperative [23] and orthogonal design [24], were developed in particle swarm optimization to overcome its weaknesses. Social learning mechanism from the society behavior was also introduced into particle swarm optimization (SL-PSO) by Cheng and Jin [7]. Lim and Isa also developed a two learning phases (teaching and peer-learning phases) PSO (BTPLPSO) [10] to obtain better solution.

In this paper, a novel personal-best information guided PSO algorithm is proposed. In this strategy, personal-best positions guided term with weighted sum of all particles' personal-best positions is adopted. Then the global version and local version of this term is defined. After that, the modified cognition component is formed by the two terms. Finally, the original cognition component in velocity update process is replaced by modified cognition component in the basic PSO and LPSO algorithm. The global version term and the local version term have different abilities between the exploration and exploitation and both used to guide particles. Therefore, to make use of information from personal-best positions can achieve a good tradeoff between the exploration and exploitation and effectively improves the PSO performance.

The remainder of this paper is organized as follows: section 2 introduces the detail of PSO algorithm. Section 3 describes the implementation of particle swarm optimization using mean personal-best information guided strategy strategy. The simulation results and analysis of the proposed algorithm is shown in section 4. A conclusion is finally given in section 5.

2. **Particle swarm optimization.**

2.1. **Velocity and position update.** Particle swarm optimization can search the optimal solution for the specific issue by updating the particle velocity and position. Suppose n particles and m dimensions. Then the position vector and velocity vector of particle i can be donated as $v_i = (v_{i1}, v_{i2}, ..., v_{im})$ and $x_i = (x_{i1}, x_{i2}, ..., x_{im})$. The personal-best position of particle i is indicated as $p_{ibest} = (p_{ibest,1}, p_{ibest,2}, ..., p_{ibest,m})$. The global-best position is indicated as $g_{best} = (g_{best,1}, g_{best,2}, ..., g_{best,m})$. Each particle updates the velocity v_{ij}^{t+1} according to the current velocity and the distance from p_{ibest} and g_{best}. The velocity and position is manipulated according to the followings:

$$v_{ij}^{t+1} = \omega v_{ij}^{t} + c_1 r_1 (p_{ibest,j}^{t} - x_{ij}^{t}) + c_2 r_2 (g_{best,j}^{t} - x_{ij}^{t}) \tag{1}$$

$$x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \tag{2}$$

where ω is inertia weight. c_1, c_2 are the cognitive and social component acceleration coefficients. t is the current iteration number. r_1, r_2 are random value in [0 1]. Inertia weight ω is a vital parameter which extremely affects the performance of PSO. Linearly decreasing inertia weight is a typical option for solving wide varieties of problems and is manipulated by the following equation:

$$\omega = \omega_{max} - (\omega_{max} - \omega_{min}) \frac{t}{Iter_{max}} \tag{3}$$

where ω_{min} and ω_{max} is the minimum and maximum value. $Iter_{max}$ is the maximum iteration number. The advised value of ω_{min} and ω_{max} are 0.9 and 0.4 [25]. The stop criterion is that a satisfactory solution is found or the maximum iteration number is met.

3. **Personal-best information guided particle swarm optimization (PIGPSO).**

3.1. **Theory of algorithm.** The personal-best particles carry useful information for the optimization problem and influence the decision where the particles will make to move. Making use of the information is able to correct some particles' error decision. Therefore, the personal-best information guided strategy may have more opportunities to correct the misguided particle. In next section, particle swarm optimization using personal-best information guided strategy will be presented in details.

3.2. **The detail of personal-best information guided strategy.**

Step 1: Find the best and the worst fitness of personal-best positions.

Among all the personal-best positions, the best and the worst fitness should be firstly found:

$$f_{min} = \min\{f_{p_{ibest}}\}\ i = 1, 2, ..., n; \tag{4}$$

$$f_{max} = \max\{f_{p_{ibest}}\}\ i = 1, 2, ..., n; \tag{5}$$

where $f_{p_{ibest}}$ is the personal-best fitness of particle i; f_{min}, f_{max} represents the best and the worst fitness among all the personal-best positions. n represents the total number of particles.

Step 2: Normalize the personal-best fitness.

In this step, the work is to normalize the personal-best fitness according to the following equations:

$$p_i = \frac{f_{\max} - f_{p_{ibest}}}{f_{\max} - f_{\min}} \tag{6}$$

p_i is the normalize the personal-best fitness of the particle i.

Step 3: Calculate the proportion of the personal-best fitness of the particle i. Donate the proportion as r_i and r_i should be calculated by the following pseudo-code.

Algorithm 2 Calculate r_i pseudo-code

begin
    for $i=1$ to n
        if $f_{\max} == f_{\min}$
            $r_i = 1/n$;
        else
            $r_i = p_i / \sum_{i=1}^{n} p_i$
        end
    end
end

Step 4: All personal-best positions guided term $a_{i,PIG}$.

In order to obtain the term $a_{i,PIG}$, the global version $a_{i,PIG}^{g}$ and local version $a_{i,PIG}^{l}$ of personal-best positions guided term should be calculated. By weighted sum of all personal-best positions, the global version $a_{i,PIG}^{g}$ is calculated:

$$a_{i,PIG}^{g} = \sum_{i=1}^{n} r_i (p_{ibest} - x_i) \tag{7}$$

where $a_{i,PIG}^{g}$ represents the global version of personal-best positions guided term of the particle i. By using parts of personal-best positions, the local version of personal-best positions guided term $a_{i,PIG}^{l}$ should be calculated by the following pseudo-code. Set the control parameter of local version as K. Then the topology size of local version is $2K+1$.

Algorithm 3 The pseudo-code of $a_{i,PIG}^{l}$

begin
    Initialize the local size K.
    Sort the personal-best fitness of all particles.
    for $i=1$ to n
        $f_{\min} = \min\{f_{p_{kbest}}\} \quad k = (i-K):(i+K)$;
        $f_{\max} = \max\{f_{p_{kbest}}\} \quad k = (i-K):(i+K)$;

$p_i = \dfrac{f_{\max} - f_{p_{ibest}}}{f_{\max} - f_{\min}}$

if $f_{\max} == f_{\min}$

$r_i = 1/(2*K+1);$

$a_{i,PIG}^{l} = \sum_{i=1}^{n} r_i (p_{ibest} - x_i^t)$

else

if $i - K < 0$

$r_i = p_i / \sum_{j=1}^{2*K+1} p_j$

$a_{i,PIG}^{l} = \sum_{j=1}^{2*K+1} r_j (p_{jbest} - x_j^t)$

elseif $i - K > 0 \parallel i + K < n$

$r_i = p_i / \sum_{j=i-K}^{i+K} p_j$

$a_{i,PIG}^{l} = \sum_{j=i-K}^{i+K} r_j (p_{jbest} - x_j^t)$

elseif $i + K > n$

$r_i = p_i / \sum_{j=i-2*K}^{n} p_j$

$a_{i,PIG}^{l} = \sum_{j=i-2*K}^{n} r_j (p_{jbest} - x_j^t)$

end

end

end

end

The global version term $a_{i,PIG}^{g}$ and the local version term $a_{i,PIG}^{l}$ of the particle i have different abilities between the exploration and exploitation. Therefore, the two terms are both used to guide some particles and after the two terms are acquired, personal-best information guided strategy $a_{i,PIG}$ should be calculated by the following pseudo-code.

Algorithm 4 The calculation of $a_{i,PIG}$ pseudo-code

begin

for $i = 1$ to n

if $rand < 0.5$

$a_{i,PIG} = a_{i,PIG}^{l}$

else

$a_{i,PIG} = a_{i,PIG}^{g}$

```
            end
      end
end
```

Step 5: Modify velocity update equation.

For basic PSO and LPSO algorithm, the original cognition component in velocity update equation will be replace by mean personal-best information guided term $a_{i,PIG}$. Therefore, PIG-PSO algorithm and PIG-LPSO algorithm are formed. The modified velocity update equation is as follows:

$$v_{ij}^{t+1} = \omega v_{ij}^{t} + rand \bullet a_{ij,PIG}^{t} + c \bullet rand \bullet (g_{best,j}^{t} - x_{ij}^{t}) \quad (8)$$

The flowchart of PIG-PSO algorithm is shown in Fig.1.

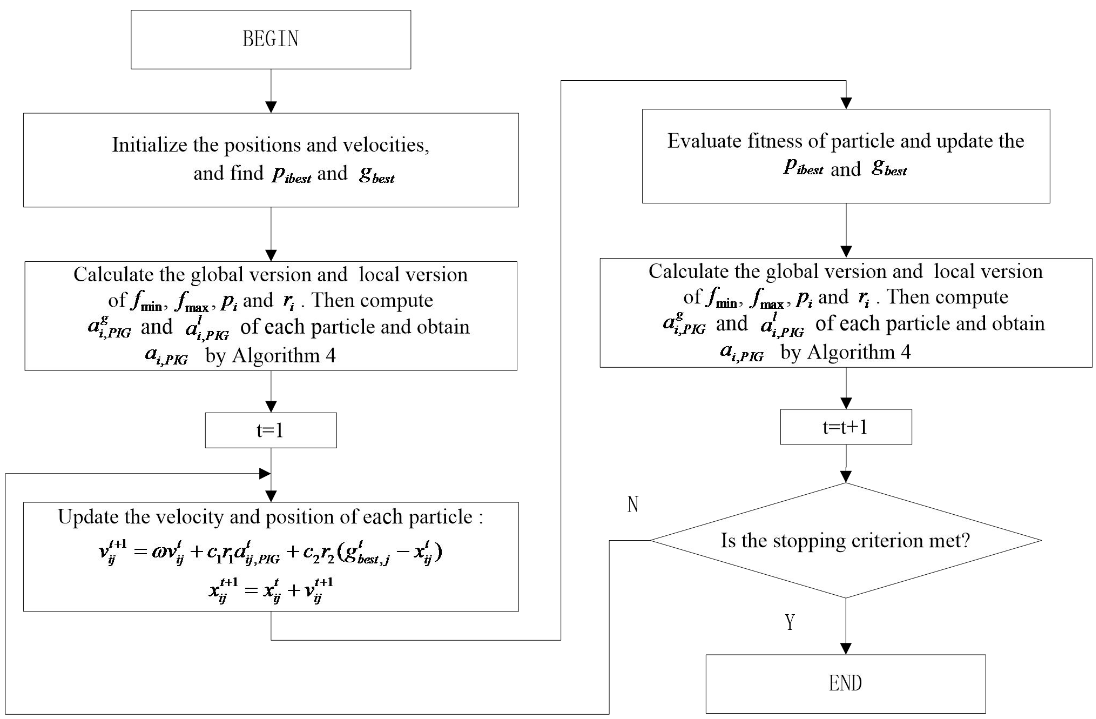


FIGURE.1 THE FLOWCHART OF PIG-PSO ALGORITHM-

4. **Experiments and results.**

4.1. **Test benchmark functions.** In this section, ten well-known benchmark functions [26-29] are used to test the performance of the proposed algorithms. All benchmark functions have one single global optimum. The expressions are shown as follows and the dimensions, admissible range of the variables and the optimal value of the benchmark functions are summarized in Table 1.

1. Sphere Model (unimodal function)

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

2. Schewefel's Problem 1.2 (unimodal function)

$$f_2(x)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i}x_j\right)^2$$

3. Schewefel's Problem 2.21 (unimodal function)

$$f_3(x)=\max_i\{|x_i|,1\le i\le n\}$$

4. Generalized Griewank Function (multimodal function)

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

5. Rotated Rastrign Function (multimodal function)

$$f_5(x)=\sum_{i=1}^{n}\left[y_i^2-10\cos(2\pi y_i)+10\right],y=M\times x$$

6. Rotated Rosenbrock Function (multimodal function)

$$f_6(x)=\sum_{i=1}^{n-1}\left[100\left(y_i^2-y_{i+1}\right)^2+\left(y_i-1\right)^2\right],y=M\times x$$

7. Rotated Elliptic Function (unimodal function)

$$f_7(x)=\sum_{i=1}^{n}\left(10^6\right)^{(i-1)/(n-1)}y_i^2,y=M\times x$$

8. Shifted Schewefel's Problem 2.21 (unimodal function)

$$f_8(x)=\max_i\left\{|y_i|,1\le i\le n\right\}+fbias_8,y=x-o$$

$$where\quad fbias_8=-450.$$

9. Shifted Rotated Ackley's Function (multimodal function)

$$f_9(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}z_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi z_i\right)+20+e+fbias_9$$

$$where\quad fbias_9=-140,z=\left(x-o\right)\times M'$$

10. Shifted Rotated Weierstrass Function (multimodal function)

$$f_{10}(x)=\sum_{i=1}^{n}\left(\sum_{K=0}^{k\max}\left[a^k\cos\left(2\pi b^k\left(z_i+0.5\right)\right)\right]\right)-n\sum_{K=0}^{k\max}\left[a^k\cos\left(2\pi b^k\times 0.5\right)\right]+fbias_{10}$$

$$where\quad a=0.5,b=3,k\max=20,fbias_{10}=90,z=\left(x-o\right)\times M'$$

TABLE 1. SPECIFICATION OF THE 20 BENCHMARK FUNCTIONS

Function	D	Search range	$f_{\min}$
$f_1(x)$	20/50	$[-100 , 100]^D$	0
$f_2(x)$	20/50	$[-100, 100]^D$	0
$f_3(x)$	20/50	$[-100, 100]^D$	0
$f_4(x)$	20/50	$[-600 ,600]^D$	0
$f_5(x)$	20/50	$[-5.12, 5.12]^D$	0
$f_6(x)$	20/50	$[-100, 100]^D$	0
$f_7(x)$	20/50	$[-1.28, 1.28]^D$	0
$f_8(x)$	20/50	$[-100, 100]^D$	-450
$f_9(x)$	20/50	$[-32,32]^D$	-140
$f_{10}(x)$	20/50	$[-0.5 ,0.5]^D$	90

4.2. **Experimental settings.**

4.2.1. **Comparison of PIG-PSO and PIG-LPSO with basic PSO and LPSO.** For the basic PSO, LPSO, PIG-PSO and PIG-LPSO algorithms in the experiment, several parameters are set the same. The maximum iteration number sets as 5000. Population size sets as 30. c_1 and c_2 are set to 2, and c in PIG-PSO and PIG-LPSO algorithms are also set to 2. The inertia weight ω in PSO and PIG-PSO algorithms is 0.7, and is linearly decreased from 0.9 to 0.4 in LPSO and PIG-LPSO algorithms. The experiment is conducted on MATLAB 2011a and tests on ten benchmark functions with 20, 50 dimensions. We did 50 independent trials to avoid stochastic error and the results of average best fitness (ABF), median best fitness (MBF) and standard deviation (SD) with 20, 50 dimensions are shown in Table 2, 4. The rank, average rank and final rank of average best fitness are also shown in Table 2, 4.

Wilcoxon's rank sum test is method to determine two algorithms are statistically different or not and returns three value, $p-value$, $h-value$ and $zval$. Significance level is set to 0.05, which indicates that the hypothesis is accepted with 95% certainty. If $p-value$ is larger than 0.05, then $h-value$ is equal to zero, which indicates that two results are not statistically different. Otherwise, the value of $h-value$ is equal to 1 or -1 which indicates they are statistically different. $h-value=1$ indicates that our algorithm is better than the compared algorithm in statistics. The statistics results of the Wilcoxon's rank sum test are shown in Table 3, 5, respectively. In Table 3, 5, the rows of '1','0','-1' give the numbers of that $h-value$ is equal to 1, 0 or -1 [30].

TABLE 2. MINIMIZATION RESULTS OF TWENTY BENCHMARK FUNCTIONS (ITERATION = 5000 AND D= 20)

function		PIG-PSO	PSO	PIG-LPSO	LPSO
$f_1(x)$	ABF	3.680e-058	5.012e-012	**2.205e-064**	4.940e-052
	MBF	1.424e-063	2.996e-013	**1.804e-072**	1.335e-053
	SD	2.154e-057	2.343e-011	**1.190e-063**	2.077e-051
	Rank	2	4	1	3
$f_2(x)$	ABF	**1.000e+002**	4.381e+003	4.333e+002	6.500e+003
	MBF	**3.283e-014**	5.014e+003	6.010e-011	5.000e+003
	SD	**7.071e+002**	4.105e+003	3.064e+003	4.917e+003
	Rank	1	3	2	4
$f_3(x)$	ABF	**2.917e-014**	2.468e+000	3.627e-010	6.574e-004
	MBF	**1.248e-015**	2.347e+000	4.033e-011	4.143e-004
	SD	**1.098e-013**	9.388e-001	9.371e-010	7.809e-004
	Rank	1	4	2	3
$f_4(x)$	ABF	4.851e-002	4.776e-002	**2.727e-002**	3.346e-002
	MBF	3.074e-002	3.444e-002	**2.337e-002**	2.706e-002
	SD	8.949e-002	3.456e-002	**2.048e-002**	2.941e-002
	Rank	4	3	1	2
$f_5(x)$	ABF	7.497e+001	1.093e+002	**5.795e+001**	6.928e+001
	MBF	6.765e+001	1.124e+002	**4.825e+001**	5.720e+001
	SD	3.570e+001	**2.270e+001**	4.321e+001	3.430e+001
	Rank	3	4	1	2
$f_6(x)$	ABF	**3.332e+005**	4.591e+007	7.726e+005	6.107e+008
	MBF	**1.058e+002**	4.511e+003	2.865e+004	2.261e+008

	SD	**2.319e+006**	2.047e+008	2.986e+006	1.158e+009
	Rank	1	3	2	4
$f_7(x)$	ABF	**1.106e+003**	7.549e+003	1.122e+003	1.163e+004
	MBF	**4.343e+002**	5.938e+003	7.721e+002	6.614e+003
	SD	2.144e+003	6.376e+003	**9.775e+002**	1.498e+004
	Rank	1	3	2	4
$f_8(x)$	ABF	**-4.410e+002**	-4.184e+002	-4.220e+002	-4.062e+002
	MBF	**-4.500e+002**	-4.158e+002	-4.187e+002	-4.118e+002
	SD	**1.509e+001**	1.815e+001	2.450e+001	1.618e+001
	Rank	1	3	2	4
$f_9(x)$	ABF	**-1.193e+002**	-1.191e+002	-1.192e+002	-1.190e+002
	MBF	**-1.193e+002**	-1.191e+002	-1.191e+002	-1.191e+002
	SD	**6.278e-002**	8.289e-002	7.696e-002	1.021e-001
	Rank	1	3	2	4
$f_{10}(x)$	ABF	1.016e+002	1.037e+002	**1.005e+002**	1.048e+002
	MBF	1.013e+002	1.035e+002	**1.008e+002**	1.053e+002
	SD	2.925e+000	2.440e+000	**2.343e+000**	2.496e+000
	Rank	2	3	1	4
Avg.		1.7	3.3	**1.6**	3.4
Final		2	3	**1**	4

TABLE 3. WILCOXON'S RANK SUM TEST RESULTS OF TWENTY BENCHMARK FUNCTIONS (ITERATION = 5000 AND D= 20)

function		PIG-PSO and PSO	PIG-LPSO and LPSO
$f_1(x)$	p-Value	7.066e-018	7.066e-018
	h-Value	1	1
	zval	-8.613e+000	-8.613e+000
$f_2(x)$	p-Value	2.071e-017	1.699e-016
	h-Value	1	1
	zval	-8.489e+000	-8.241e+000
$f_3(x)$	p-Value	7.066e-018	7.066e-018
	h-Value	1	1
	zval	-8.613e+000	-8.613e+000
$f_4(x)$	p-Value	8.796e-002	6.391e-001
	h-Value	0	0
	zval	-1.706e+000	-4.688e-001
$f_5(x)$	p-Value	5.049e-008	1.538e-002
	h-Value	1	1
	zval	-5.449e+000	-2.423e+000
$f_6(x)$	p-Value	9.293e-010	5.008e-008
	h-Value	1	1
	zval	-6.121e+000	-5.451e+000
$f_7(x)$	p-Value	1.924e-014	7.678e-015
	h-Value	1	1

	zval	-7.655e+000	-7.772e+000
$f_8(x)$	p-Value	2.560e-010	8.134e-004
	h-Value	1	1
	zval	-6.323e+000	-3.348e+000
$f_9(x)$	p-Value	3.692e-005	4.748e-002
	h-Value	-1	-1
	zval	4.125e+000	1.981e+000
$f_{10}(x)$	p-Value	1.604e-004	9.432e-013
	h-Value	1	1
	zval	-3.774e+000	-7.138e+000
1		**8**	**8**
0		1	1
-1		1	1

TABLE 4. MINIMIZATION RESULTS OF TWENTY BENCHMARK FUNCTIONS (ITERATION = 5000 AND D= 50)

function		PIG-PSO	PSO	PIG-LPSO	LPSO
$f_1(x)$	ABF	2.000e+002	3.462e+003	**1.542e-004**	4.600e+003
	MBF	2.383e-010	7.423e+001	**5.424e-012**	1.741e-011
	SD	1.414e+003	5.572e+003	**1.089e-003**	6.764e+003
	Rank	2	3	1	4
$f_2(x)$	ABF	**5.978e+003**	6.478e+004	1.993e+004	6.432e+004
	MBF	**4.586e+003**	6.386e+004	1.888e+004	6.043e+004
	SD	**6.983e+003**	1.538e+004	1.094e+004	2.014e+004
	Rank	1	4	2	3
$f_3(x)$	ABF	**4.912e-001**	3.786e+001	1.688e+000	2.375e+001
	MBF	**4.030e-001**	3.846e+001	1.546e+000	2.411e+001
	SD	**3.060e-001**	3.491e+000	7.986e-001	3.131e+000
	Rank	1	3	2	4
$f_4(x)$	ABF	**5.311e-001**	4.488e+001	3.758e+000	1.448e+001
	MBF	1.177e-001	2.489e+000	1.076e-001	**7.396e-003**
	SD	**1.742e+000**	5.809e+001	1.787e+001	3.349e+001
	Rank	1	4	2	3
$f_5(x)$	ABF	**3.461e+002**	6.175e+002	3.824e+002	4.547e+002
	MBF	**3.427e+002**	6.111e+002	3.930e+002	4.076e+002
	SD	1.028e+002	**8.302e+001**	8.321e+001	1.454e+002
	Rank	1	4	2	3
$f_6(x)$	ABF	**3.325e+007**	1.007e+010	1.220e+008	2.576e+010
	MBF	**2.447e+004**	6.397e+009	1.904e+006	1.667e+010
	SD	**1.654e+008**	1.128e+010	6.045e+008	2.469e+010
	Rank	**1**	3	2	4
$f_7(x)$	ABF	**7.436e+003**	7.194e+004	2.494e+004	1.228e+005
	MBF	**6.151e+003**	6.774e+004	1.356e+004	8.964e+004
	SD	**4.542e+003**	3.334e+004	3.406e+004	1.030e+005

	Rank	1	3	2	4
$f_8(x)$	ABF	**-4.454e+002**	-3.740e+002	-4.248e+002	-3.650e+002
	MBF	**-4.495e+002**	-3.803e+002	-4.433e+002	-3.630e+002
	SD	**1.444e+001**	1.912e+001	3.025e+001	2.107e+001
	Rank	1	3	2	4
$f_9(x)$	ABF	**-1.189e+002**	-1.188e+002	-1.188e+002	-1.188e+002
	MBF	-1.188e+002	-1.188e+002	-1.188e+002	-1.188e+002
	SD	**3.878e-002**	4.623e-002	4.266e-002	6.143e-002
	Rank	1	2	2	2
$f_{10}(x)$	ABF	1.309e+002	1.422e+002	**1.299e+002**	1.426e+002
	MBF	1.315e+002	1.419e+002	**1.297e+002**	1.428e+002
	SD	5.608e+000	4.562e+000	5.356e+000	**4.265e+000**
	Rank	2	3	1	4
Avg.		**1.2**	3.2	1.8	3.5
Final		**1**	3	2	4

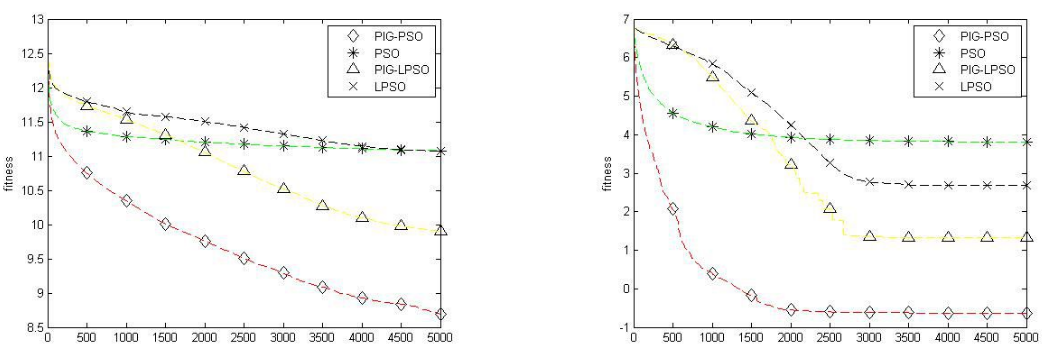
TABLE 5. WILCOXON'S RANK SUM TEST RESULTS OF TWENTY BENCHMARK FUNCTIONS (ITERATION = 5000 AND D= 50)

function		PIG-PSO and PSO	PIG-LPSO and LPSO
$f_1(x)$	p-Value	5.638e-017	3.742e-003
	h-Value	1	1
	zval	-8.372e+000	-2.899e+000
$f_2(x)$	p-Value	7.066e-018	7.550e-017
	h-Value	1	1
	zval	-8.613e+000	-8.338e+000
$f_3(x)$	p-Value	7.066e-018	7.066e-018
	h-Value	1	1
	zval	-8.613e+000	-8.613e+000
$f_4(x)$	p-Value	1.241e-015	6.852e-006
	h-Value	1	-1
	zval	-8.000e+000	4.498e+000
$f_5(x)$	p-Value	7.123e-017	2.483e-002
	h-Value	1	1
	zval	-8.344e+000	-2.243e+000
$f_6(x)$	p-Value	2.071e-017	9.540e-018
	h-Value	1	1
	zval	-8.489e+000	-8.579e+000
$f_7(x)$	p-Value	7.066e-018	3.257e-013
	h-Value	1	1
	zval	-8.613e+000	-7.283e+000
$f_8(x)$	p-Value	4.250e-017	3.566e-013
	h-Value	1	1
	zval	-8.405e+000	-7.271e+000
$f_9(x)$	p-Value	1.944e-004	1.221e-002

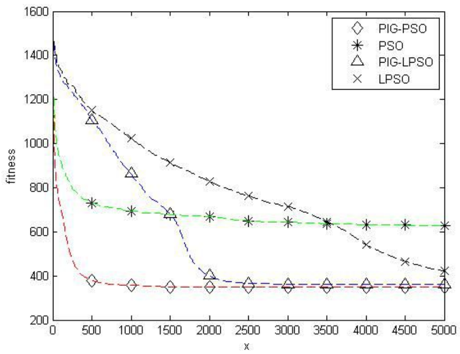
	h-Value	-1	-1
	zval	3.726e+000	2.505e+000
$f_{10}(x)$	p-Value	2.259e-014	5.037e-016
	h-Value	1	1
	zval	-7.634e+000	-8.110e+000
1		**8**	**8**
0		0	0
-1		1	2

From the rank of 20 dimensions, the numbers of first-rank for PIG-PSO algorithm and PIG-LPSO algorithm are six and four. From the rank of 50 dimensions, the numbers of first-rank for PIG-PSO algorithm and PIG-LPSO algorithm are eight and two. Neither of PSO algorithm and LPSO algorithm obtains the first-rank. It is obviously that PIG-PSO algorithm and PIG-LPSO algorithm obtain more number of the first-rank than that of LPSO algorithm and PSO algorithm. From the average rank and final rank in Tables 2, 4, PIG-PSO algorithm and PIG-LPSO algorithm all ranks the first and second for 20 dimensions and 50 dimensions. Therefore, for most of benchmark function, the quality of solutions of PIG-LPSO algorithm and PIG-PSO algorithm are higher than that of LPSO algorithm and PSO algorithm. From Wilcoxon's rank sum test, the number of $h - value = 1$ for the PIG-LPSO and PIG-PSO algorithm is significantly larger than that for the PIG-LPSO and PIG-PSO algorithm. It is proved that the PIG-LPSO and PIG-PSO algorithm perform better than the PSO and LPSO algorithm with 95% certainty. Therefore, mean personal-best information guided strategy is an effective strategy to improve the PSO performance.

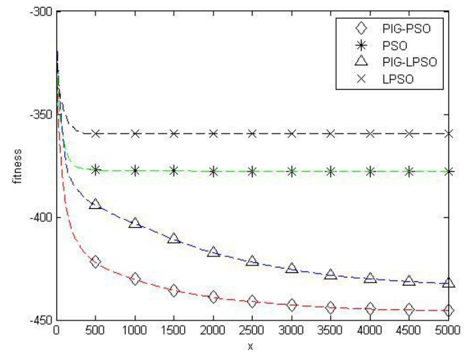
The evolutions of average fitness on these five functions $f_2(x)$, $f_4(x)$, $f_5(x)$, $f_8(x)$, $f_{10}(x)$ for 20 dimensions are shown in Fig.2(a-e). Here, $f_2(x)$ is a unimodal function, $f_4(x)$ is a multimodal function, $f_5(x)$ is a rotated function $f_8(x)$ is a shifted function and $f_{10}(x)$ is a unimodal function, of which the minimum is not zero. Specifically, vertical axis is the logarithm of average fitness. In these figures, it's easy to see that PIG-LPSO algorithm and PIG-PSO algorithm have a better convergence rate and high accuracy of the solution.



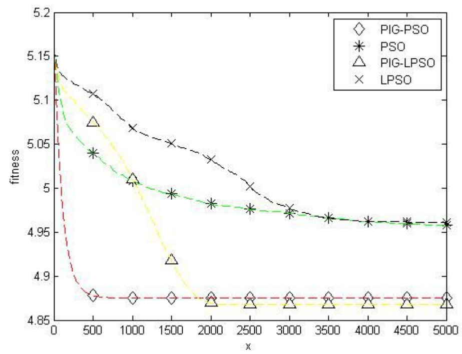
(a) (b)



(c)



(d)



(e)

FIGURE.2 CONVERGENCE PERFORMANCE OF PIG-PSO, PIG-LPSO, PSO AND LPSO ON $f_2(x)$ $f_4(x)$ $f_5(x)$ $f_8(x)$ $f_{10}(x)$ (50 DIMENSIONS)

4.2.2. **Comparisons of PIG-LPSO with other PSO algorithms.** In this section, some recent algorithms are used to evaluate the efficiency of PIG-LPSO algorithm. The compared PSO variants (APSO [27], FLPSO-QIW [28], FlexiPSO [29], FPSO [30], FIPSO [31], OLPSO-L [32], HPSO-TVAC [33], and RPPSO [34]) and the detail of its parameters sittings are shown in Table 6. Four benchmark functions, which are Sphere, Rosenbrock, Griewank and Ackley. For PIG-LPSO algorithm, population size sets as 30 and function dimension sets as 50. The maximum iteration sets as 1×10^4. The experiment of PIG-LPSO will independently run 30 times. Other parameters in PIG-LPSO are set to the same as that in section 4.2.1. The computational results of PIG-LPSO and the results of the compared algorithms from the paper [31] are shown in Table 7. The average best fitness and standard deviation are also presented in Table 7. For comparison, the rank, average rank and final rank of average best fitness for each benchmark function are shown in Table 7.

TABLE 6. SOME WELL-KNOWN PSO VARIANTS

algorithms	Topology	parameters sittings
APSO	Fully connected	$\omega: 0.9-0.4,\ c_1+c_2:\ [3.0, 4.0],\ \delta=[0.05,\ 0.1],\ \sigma_{\max}=1.0,\ \sigma_{\min}=0.1$
FLPSO-QIW	Comprehensive learning	$\omega: 0.9-0.4,\ c_1: 2-1.5,\ c_2:\ 1\text{-}1.5,\ m=1,\ P_i=[0.1,\ 1],\ K_1=0.1,\ K_2=0.001,\ \sigma_1=1,\ \sigma_2=0$
FlexiPSO	Fully connected and local ring	$\omega: 0.5-0.0,\ c_1, c_2, c_3:\ [0.0, 2.0],\ \varepsilon=0.1,\ \alpha=0.01\%$
FPSO	Decreasing	$\chi=0.729,\ \sum c_i=4.1$

FIPSO	Local URing	$\chi = 0.729,\ \sum c_i = 4.1$
OLPSO-L	Orthogonal learning	$\omega: 0.9-0.4,\ c = 2.0\ G = 5$
HPSO-TVAC	Fully connected	$\omega: 0.9-0.4,\ c_1:\ 2.5\text{-}0.5\ c_2 = 0.5-2.5$
RPPSO	Random	$\omega: 0.9-0.4,\ c_{large} = 6,\ c_{small} = 3$

TABLE 7. EXPERIMENT RESULTS FOR PIG-LPSO AND EIGHT PSO VARIANTS ON TEN FUNCTIONS. (ITERATION = 10000 AND D= 50 AND N=30)

function		APSO	FLPSO-QIW	FlexiPSO	FPSO	FIPSO	OLPSO-L	HPSO-TVAC	RPPSO	PIG-LPSO
Sphere	best	2.50E-01	2.90E-81	1.78E-04	7.02E+01	2.96E-01	4.86E-33	1.09E-05	1.28E-02	1.99E-90
	SD	1.81E-01	5.97E-81	5.23E-05	6.98E+01	8.06E-01	5.15E-33	3.69E-06	2.98E-02	3.12E-90
	Rank	7	2	5	9	8	3	4	6	**1**
Rosenbrock	best	4.62E+01	4.22E+01	4.48E+01	5.68E+01	4.77E+01	4.30E+01	4.60E+01	4.76E+01	3.74E+01
	SD	1.53E+00	2.39E-01	1.04E+00	7.08E+00	8.44E-01	3.18E+00	5.70E-01	4.30E-01	1.25E+01
	Rank	6	2	4	7	9	3	5	8	**1**
Griewank	best	1.70E-01	5.75E-04	8.34E-03	1.86E+00	1.93E-01	0.00E+00	3.86E-03	7.08E-03	1.77E-15
	SD	8.21E-02	2.21E-03	9.48E-03	9.28E-01	3.47E-01	0.00E+00	6.55E-03	1.85E-02	6.77E-14
	Rank	7	3	6	9	8	**1**	4	5	2
Ackley	best	6.60E-02	3.43E-14	3.55E-03	1.80E+00	1.70E-01	5.09E-15	1.57E-03	7.47E-01	1.31E-14
	SD	2.57E-02	1.07E-14	5.36E-04	1.10E+00	3.38E-01	1.79E-15	1.99E-04	9.17E-01	3.21E-15
	Rank	6	3	5	9	7	**1**	4	8	2
Avg. rank		6.5	2.5	5	8.5	8	2	4.25	6.75	1.5
Final rank		6	3	5	9	8	2	4	7	1

From the rank for each benchmark function, it can be seen that PIG-LPSO performs best on Sphere, Rosenbrock functions and OLPSO-L performs best on Griewank and Ackley. However, PIG-LPSO performs the second best on Griewank and Ackley. From average rank and final rank, we can see that PIG-LPSO ranks first among nine PSO variants. Therefore, statistics analysis indicates the proposed algorithm have better performance than the other eight PSO variants on these functions. In general, PSO algorithm with all personal-best positions guided strategy has a high quality of the solutions and fast convergence rate.

5. **Conclusion.** In this paper, a novel personal-best information guided PSO algorithm has been introduced to enhance the performance. Firstly, all personal-best positions are used to form a cognition component term with weighted sum of all particles' personal-best fitness. Then the global version and local version of this term is defined and the modified cognition component is formed with a chosen strategy of the two terms. Finally, the original cognition component is replaced by modified cognition component in the basic PSO and

LPSO algorithm. Ten benchmark functions with 20 and 50 dimensions have been employed to evaluate the strategy. Experimental results show that personal-best information guided strategy is an effective strategy to improve PSO's performance. Compared with several PSO variants in the literature, the PIG-LPSO algorithm also performs best among these PSO variants. In conclusion, this strategy enhances the PSO's performance and is an available optimization method.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (Project No: 61572238), by the National High-tech Research and Development Projects of China (Project No: 2014AA041505).

REFERENCE

[1] Eberhart R C, Kennedy J. A new optimizer using particle swarm theory[C]. Proceedings of the sixth international symposium on micro machine and human science. 1995, 1: 39-43.

[2] Ebenhart R. Kennedy. Particle Swarm Optimization[C]. Proceeding IEEE Inter Conference on Neural Networks, Perth, Australia, Piscat-away. 1995, 4: 1942-1948.

[3] Rana S, Jasola S, Kumar R. A boundary restricted adaptive particle swarm optimization for data clustering[J]. International Journal of Machine Learning and Cybernetics, 2013, 4(4): 391-400.

[4] Jiang B, Wang N. Cooperative bare-bone particle swarm optimization for data clustering[J]. Soft Computing, 2014, 18(6): 1079-1091.

[5] Jiang B, Wang N, Wang L. Particle swarm optimization with age-group topology for multimodal functions and data clustering[J]. Communications in Nonlinear Science and Numerical Simulation, 2013, 18(11): 3134-3145.

[6] Goksal F P, Karaoglan I, Altiparmak F. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery[J]. Computers & Industrial Engineering, 2013, 65(1): 39-53.

[7] Gong Y J, Zhang J, Liu O, et al. Optimizing the vehicle routing problem with time windows: a discrete particle swarm optimization approach[J]. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 2012, 42(2): 254-267.

[8] Belmecheri F, Prins C, Yalaoui F, et al. Particle swarm optimization algorithm for a vehicle routing problem with heterogeneous fleet, mixed backhauls, and time windows[J]. Journal of intelligent manufacturing, 2013, 24(4): 775-789.

[9] Marinakis Y, Iordanidou G R, Marinaki M. Particle swarm optimization for the vehicle routing problem with stochastic demands[J]. Applied Soft Computing, 2013, 13(4): 1693-1704.

[10] Jiang S, Ji Z, Wang Y. A novel gravitational acceleration enhanced particle swarm optimization algorithm for wind–thermal economic emission dispatch problem considering wind power availability[J]. International Journal of Electrical Power & Energy Systems, 2015, 73: 1035-1050.

[11] Jadoun V K, Gupta N, Niazi K R, et al. Modulated particle swarm optimization for economic emission dispatch[J]. International Journal of Electrical Power & Energy Systems, 2015, 73: 80-88.

[12] Das G, Pattnaik P K, Padhy S K. Artificial neural network trained by particle swarm optimization for

non-linear channel equalization[J]. Expert Systems with Applications, 2014, 41(7): 3491-3496.

[13] Salim R, Nabag M, Noura H, et al. The parameter identification of the Nexa 1.2 kW PEMFC's model using particle swarm optimization[J]. Renewable Energy, 2015, 82: 26-34.

[14] Jiang B, Wang N, Wang L. Parameter identification for solid oxide fuel cells using cooperative barebone particle swarm optimization with hybrid learning[J]. International Journal of Hydrogen Energy, 2014, 39(1): 532-542.

[15] Bonyadi M R, Li X, Michalewicz Z. A hybrid particle swarm with a time-adaptive topology for constrained optimization[J]. Swarm and Evolutionary Computation, 2014, 18: 22-37.

[16] Lim W H, Isa N A M. Particle swarm optimization with increasing topology connectivity[J]. Engineering Applications of Artificial Intelligence, 2014, 27: 80-102.

[17] Liu C, Du W B, Wang W X. Particle Swarm Optimization with Scale-Free Interactions[J]. PLoS One, 2014, 9(5).

[18] Li Y, Zhan Z H, Lin S, et al. Competitive and cooperative particle swarm optimization with information sharing mechanism for global optimization problems[J]. Information Sciences, 2015, 293: 370-382.

[19] Qin Q, Cheng S, Zhang Q, et al. Multiple strategies based orthogonal design particle swarm optimizer for numerical optimization[J]. Computers & Operations Research, 2015, 60: 91-110.

[20] Cheng R, Jin Y. A social learning particle swarm optimization algorithm for scalable optimization[J]. Information Sciences, 2015, 291: 43-60.

[21] Lim W H, Isa N A M. Teaching and peer-learning particle swarm optimization[J]. Applied Soft Computing, 2014, 18: 39-58.

[22] Deep K, Thakur M. A new crossover operator for real coded genetic algorithms[J]. Applied Mathematics and Computation, 2007, 188(1): 895-911.

[23] Liang J J, Qin A K, Suganthan P N, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. Evolutionary Computation, IEEE Transactions on, 2006, 10(3): 281-295.

[24] Suganthan P N, Hansen N, Liang J J, et al. Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization[J]. 2005.

[25] Yao X, Liu Y, Lin G. Evolutionary programming made faster[J]. Evolutionary Computation, IEEE Transactions on, 1999, 3(2): 82-102.

[26] Beheshti Z, Shamsuddin S M H, Hasan S. MPSO: median-oriented particle swarm optimization[J]. Applied Mathematics and Computation, 2013, 219(11): 5817-5836.

[27] Zhan Z H, Zhang J, Li Y, et al. Adaptive particle swarm optimization[J]. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2009, 39(6): 1362-1381.

[28] Tang Y, Wang Z, Fang J. Feedback learning particle swarm optimization[J]. Applied Soft Computing, 2011, 11(8): 4713-4725.

[29] Kathrada M. The flexi-PSO: Towards a more flexible particle swarm optimizer[J]. Opsearch, 2009, 46(1): 52-68.

[30] De Oca M A M, Stützle T, Birattari M, et al. Frankenstein's PSO: a composite particle swarm optimization algorithm[J]. Evolutionary Computation, IEEE Transactions on, 2009, 13(5): 1120-1132.

[31] Mendes R, Kennedy J, Neves J. The fully informed particle swarm: simpler, maybe better[J]. Evolutionary Computation, IEEE Transactions on, 2004, 8(3): 204-210.

[32] Zhan Z H, Zhang J, Li Y, et al. Orthogonal learning particle swarm optimization[J]. Evolutionary

Computation, IEEE Transactions on, 2011, 15(6): 832-847.

[33] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. Evolutionary Computation, IEEE Transactions on, 2004, 8(3): 240-255.

[34] Zhou D, Gao X, Liu G, et al. Randomization in particle swarm optimization for global search ability[J]. Expert Systems with Applications, 2011, 38(12): 15356-15364.