Applied Mathematics & Information Sciences
*An International Journal*

© 2013 NSP
Natural Sciences Publishing Cor.

# A new particle filter based on organizational adjustment particle swarm optimization

*Zhimin Chen*[1], *Yuming Bo*[1], *Panlong Wu*[1], *Weijun Zhou*[1]

[1]School of Automation, Nanjing University of Science and Technology, Nanjing, 210094, china

Received: Jun 8, 2012; Revised Jul. 13, 2012; Accepted Jul. 20, 2012
Published online: 1 January 2013

**Abstract:** A dynamic organizational adjustment particle swarm optimization-based particle filter algorithm (OAPSO-PF) is presented in this paper in order to solve the problem of low precision and complicated calculation of particle filter based on particle swarm optimization algorithm(PSO-PF). Through the mutual competition and collaboration among organizations, this algorithm allows the particles to adapt to the environment better and thus reach the goal of global optimization, accordingly enhancing the quality of particles and avoiding particle degradation by the adoption of optimal particle retention. Finally different models are used for simulation experiment and the results indicate that this new algorithm improves the operation speed and precision, it is applicable to practical engineering.

**Keywords:** particle filter; particle swarm optimization; organizational adjustment

## 1. Introduction

In the piratical engineering application, there is always flicker noise[1]. Flicker noise is characterized in distinct Gaussian feature, while most of the ordinary filter algorithms are within the framework of Kalman filter theory which will show relatively huge errors in system state and variance estimation giving large flicker noise, influencing the effect of radar tracking. Particle filter (PF) [2] is a statistic filtering based on Monte Carlo Method and Recursive Bayesian estimation. It is widely applied to positioning and navigation of non-Gaussian noise and non-linear system, fault diagnosis, target tracking and mode identification field [3] as its state function and observation function has no non-linear hypothesis. Nevertheless, PF may confront with the problem of weight degradation [4] which if solved by resampling method may result unavoidable particle impoverishment[5-6].

Particle swarm optimization-based PF (PSO-PF) is a typical representative of Intelligent Optimization Algorithm which introduces PSO algorithm into PF[7]. In PSO-PF, the sample distribution is inclined to move to the area with higher posterior probability [8]. PSO-PF improves the particle degradation of PF and is easier for actualization. Unfortunately PSO-PF is a process of iterative optimization which will prolong the calculation time because of the high iterative frequency [9]. Moreover, PSO-PF may be easily trapped into local optimization, influencing the precision and stability of practical engineering application [10].

Proceeding from the perspective of organization, particles are used to form different organizations in this paper and X is put forward by fully making use of the individual collaboration feature and self-earning of organizations. A new particle filter algorithm based on dynamic organizational adjustment particle swarm optimization. The experimental results prove that OAPSO-PF improves the efficiency of particle filter.

## 2. Particle filter

Particle filter is an approximate calculation of Bayes estimation based on sampling theory. Combining Monte Carlo Method and Bayesian Theory together, particle filter follows the basic thought that to find a group of random sample for approximation of posterior probability density, replace the infinitesimal calculus in light of posterior probability density function by sample mean value, and thus acquire the minimal estimate of variance [11].

* Corresponding author: e-mail: chenzhimin@188.com

© 2013 NSP
Natural Sciences Publishing Cor.

Assuming the nonlinear dynamic process is expressed as follows:

$$x_k = f(x_{k-1}, v_{k-1}) \quad (1)$$

$$y_k = h(x_k, n_k) \quad (2)$$

If the initial probability density of the state is known as $p(x_0|y_0) = p(x_0)$, then the state predictive equation is:

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1} \quad (3)$$

and the state renewal equation is:

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})} \quad (4)$$

Where

$$p(y_k|y_{1:k-1}) = \int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k \quad (5)$$

Supposing there is a known and easily-sampling importance function $q(x_{0:k}|y_{1:k})$, which is rewrote as

$$q(x_{0:k}|y_{1:k}) = q(x_0)\prod_{j=1}^{k} q(x_j|x_{0:j-1}, y_{1:j}) \quad (6)$$

then the weight formula is

$$w_k = \frac{p(y_{1:k}|x_{0:k})p(x_{0:k})}{q(x_k|x_{0:k-1}, y_{1:k})q(x_{0:k-1}, y_{1:k})} = w_{k-1}\frac{p(y_k|x_k)p(x_k|x_{k-1})}{q(x_k|x_{0:k-1}, y_{1:k})} \quad (7)$$

To sample $N$sample points $\{x_{k-1}^i\}_{i=1}^N$ from$p(x_{k-1}|y_{1:k-1})$, then the probability density is:

$$p(x_{k-1}|y_{1:k-1}) = \sum_{i=1}^{N} w_{k-1}^i \delta(x_{k-1} - x_{k-1}^i) \quad (8)$$

and the renewal probability density is:

$$w_k^i = w_{k-1}^i \frac{p(y_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, y_k)} \quad (9)$$

## 3. Algorithms in this paper

### *3.1. principle of OAPSO*

In economics, $C$oase[12] defined that "transaction cost" is for explanation of the size and formation of organization, $W$ilcoxreferring to $C$oasethe method of reducing "transaction cost", two models are combined to bring forward the organization learning model [13], based on which the literature[14] comes up with the organization evolution algorithm. Being enlightened by this, this paper proceeds from the perspective of organization and put forward with the organization adjustment particle swarm optimization which is then fused with particle filter.

In this article, individual is donated by real-value vector, taking into account only the minimum, therefore the individual fitness$F$itnessis defined as$f(x)$. Organization is the set of several individuals that are the members of the organization, wherein these individuals with the best fitness is the leader of the organization. If there is more than one member in an organization with the same optimal values, then the leader can be randomly selected from them. The fitness of the leader is that of the organization, and the intersection of different organizations is empty. Here are 4 types of organizational adjustment operators in the algorithm presented.

(1) Splitting operator. In this operator, the prerequisite for organization$org$ splitting is as follows:

$$|org| > Max \quad (10)$$

where $|org|$ denotes the number of member in $org$, $Max(< N_0)$ the allowable maximal number of organizational member, $N_0$ the number of all organizational member in initialization, $Max$ and $N_0$ pre-set parameters. If a parent organization $org_i$ meets the splitting prerequisite, then it will split into two progeny organizations $org_{c1}$and $org_{c2}$ by the following method. $org_i|/3 \sim 2|org_i|/3$ members are selected randomly from $org_i$ so as to form progeny organization $org_{c1}$, and the rest members from progeny organization $org_{c2}$. Then from the current population the organization $org_i$ is deleted, while $org_{c1}$ and $org_{c2}$ are added to the next-generation evolutionary population.

(2)Annexation operator. Assuming two parent organizations of $org_{i1} = \{x^1, x^2, \cdots, x^{m1}\}$ and $org_{i2} = \{y^1, y^2, \cdots, y^{m2}\}$, $m_1$ and $m_2$ are the number of members in organization $org_{i1}$ and $org_{i2}$, and $Fitness(org_{i1}) \geqslant Fitness(org_{i2})$, then a self-attached organization $org_c = \{z^1, z^2, \cdots z^{m1+m2}\}$ can be generated by annexing $org_{i2}$through $org_{i1}$, wherein $z^i = x^i, i = 1, 2, \cdots, m_1$. If $U_{j+m1}(0,1) < AS$, $z^{j+m1}(j = 1, 2, \cdots, m_2)$, then it will be generated by annexation strategy 1, or otherwise by annexation strategy 2. Here the subscript in $U_{j+m1}(0,1)$ denotes each random number generated by$j+m1$, $AS \in (0,1)$ is a pre-set parameter. Two annexation strategies are presented as follows.

Assuming the leader of $org_{i1}$ is $(a_1, a_2, \cdots, a_n)$, and the new individual is $r_j = (r_{j,1}, r_{j,2}, \cdots, r_{j,n})$, $j = 1, 2, \cdots, m_2$. Then in annexation strategy 1, $r_j$ is generated by the following equation:

$$r_{j,k} = \begin{cases} x_k, & d_{j,k} < x_k \\ \bar{x}_k, & d_{j,k} > x_k \\ d_{j,k}, & \text{otherwize} \end{cases} \quad (11)$$

in which, $d_{j,k} = a_k + U_k(0,1) \times (a_k - y_k^j)$, $k = 1, 2, \cdots, n$. In annexation strategy 1, $r_j$ is generated by the following equation:

$$r_{j,k} = \begin{cases} x_k + \beta_k \times (\bar{x}_k - x_k), & U_k(0,1) < 1/n \\ a_k, & \text{otherwize} \end{cases} \quad (12)$$

© 2013 NSP
Natural Sciences Publishing Cor.

where$k=1,2,\cdots,n$, $\beta_k=U(0,1)$. After $r_j$ is calculated, it is further verified by the following equation$z^{j+m1}$:

$$z^{j+m1}=\begin{cases} r_j, & Fitness(r_j)\geqslant Fitness(y^j)\\ r_j, & Fitness(r_j)\geqslant Fitness(y^j)\text{and}\{U_j(0,1)< \\ & \exp(Fitness(r_j)-Fitness(y^j))\}\\ y^j, & \text{otherwize}\end{cases} \quad (13)$$

In fact, annexation strategy 1 is a heuristic crossover operator, while annexation strategy 2 is mutation operator. From equation (13), it can be observed that when the fitness of $r_j$ exceeds $y^j$, $r_j$ enters $org_c$ for enhancement of organization fitness; when the fitness of $r_j$ is better than $y^j$, $r_j$ may enter $org_c$ at some probability, that is, the closer the fitness of $r_j$ to $y^j$, the larger the probability will be, and this conduces to protection of population diversity. Finally, $org_{i1}$ and $org_{i2}$ are deleted from the population, and $org_c$ is added to the next-generation population.

(3) Cooperation operator. Given two parent organizations of $org_{i1}=\{x^1,x^2,\cdots,x^{m1}\}$ and $org_{i2}=\{y^1,y^2,\cdots,y^{m2}\}$, if $U(0,1)<CS$, then the two progeny organizations $org_{c1}$ and $org_{c2}$ will be generated by cooperation strategy 1, or otherwise by strategy 2. Here $CS\in(0,1)$ is a pre-set parameter, and the two cooperation strategies are as follows:

Let $(a_1,a_2,\cdots,a_n)$ be the leader of $org_{i1}$, $(b_1,b_2,\cdots,b_n)$ the leader of $org_{i2}$, and two new individuals generated from cooperation $u=(u_1,u_2,\cdots,u_n)$ and $l=(l_1,l_2,\cdots,l_n)$, then in cooperation strategy 1, $u$ and $l$ can be generated by the following equation:

$$\begin{cases} u_k=\beta_k\times a_k+(1-\beta_k)\times b_k\\ l_k=(1-\beta_k)\times a_k+\beta_k\times b_k\end{cases} \quad (14)$$

where, $k=1,2,\cdots,n$, $\beta=U(0,1)$.

In cooperation strategy 2, $u$ and $l$ are generated by

$$\begin{cases} u=(a_1,a_2,\cdots,a_{i_1-1},b_{i_1+1},\cdots,b_{i2},a_{i_2+1},a_{i_2+2},\cdots,a_n)\\ l=(b_1,b_2,\cdots,b_{i_1-1},a_{i_1+1},\cdots,a_{i2},b_{i_2+1},b_{i_2+2},\cdots,b_n)\end{cases} \quad (15)$$

after $u$ and $l$ generated, $org_{c1}$ and $org_{c2}$ are respectively confirmed as:

$$org_{c1}=\begin{cases}\{x^1,x^2,\cdots,x^{i-1},u,x^{i+1},\cdots,x^{m1}\},\\ \qquad \exists x_i\in org_{i1}, Fitness(x^i)<Fitness(u)\\ org_{i1}, \qquad \text{otherwize}\end{cases} \quad (16)$$

$$org_{c2}=\begin{cases}\{y^1,y^2,\cdots,y^{j-1},l,y^{j+1},\cdots,y^{m2}\},\\ \qquad \exists y_i\in org_{i2}, Fitness(y^j)<Fitness(l)\\ org_{i2}, \qquad \text{otherwize}\end{cases} \quad (17)$$

Finally $org_{i1}$ and $org_{i2}$ can be deleted from the current population, and $org_{c1}$ and $org_{c2}$ are added to the next-generation population. In fact, Cooperation Strategy 1 is arithmetic crossover, and Strategy 2 is discrete crossover.

(4) Speed update operator. Let $h^g(t)=(h_1^g,h_2^g,\cdots,h_n^g)^T$ denote the position experienced by the optimal leader in all organizations, $h^i(t)=(h_1^i,h_2^i,\cdots,h_n^i)^T$ denotes the best position experienced by the leaders in $i$th organization. Then particles in $i$the organization update their speeds and positions in $t$th generation by the following approach:

$$\begin{cases} v_j^i(t+1)=w(t)v_j^i(t)+c_1r_1(h_j^i(t)-x_j^i(t))+\\ c_2r_2(h_j^g(t)-x_j^i(t))\\ x_j^i(t+1)=x_j^i(t)+v_j^i(t+1)\end{cases} \quad (18)$$

Here inertia weight is employed in order to update the speed and position vectors. When the inertia weight is relatively large, the algorithm is capable of exploring a greater searching space. In the final stage, the smaller inertia weight will perform delicate detection of the searching space.

### 3.2. analysis on convergence of OAPSO

If an individual component $x^i$ is denoted by $L$ bit binary string, then it means to quantify the interval $[x,\bar{x}]$ into $2^L$ discrete values and precision is $\varepsilon=(\bar{x}-x)/2^L$, so the convergence of the algorithm can be directly analyzed by real coding. Given the required precision$\varepsilon$, then the searching space $S$ can be regarded as discrete space with the size as follows:

$$|S|=\prod_{i=1}^{n}[\frac{\bar{x_i},x_i}{\varepsilon}] \quad (19)$$

For each element $x\in S$, its fitness is $Fitness(x)$. Let $F=\{Fitness(x)|x\in S\}$, apparently $|F|\leqslant|S|$, which is equivalent to $F=\{F_1,F_2,\cdots,F_{|F|}\}$ where $F_1>F_2>\cdots>F_{|F|}$. Based on different $Fitness(x)$, $S$ can be classified into several nonvoid subset $\{S_i\}$, in which $S_i=\{x|x\in S$ and $Fitness(x)=F_i\}$, $i=1,2,\cdots,|F|$, then

$$\bigcup_{i=1}^{|F|}S_i=S \quad (20)$$

$$\sum_{i=1}^{|F|}|S_i|=|S| \quad (21)$$

$$S_i\cap S_j=\emptyset,\forall i\neq j \quad (22)$$

$$S_i\neq\emptyset,\forall i\in\{1,2,\cdots,|F|\} \quad (23)$$

$F_1$ is the global optimal solution $F^*$, whereas the set $S_1$ contains all individuals whose fitness equals to $F^*$.

In this algorithm, the number of organization varies with the frequency of adjustment, while the total number of individuals remains the same, e.g., the population $q=\{x^1.x^2,\cdots,x^{N0}\}$. Let $Q$ denote the set of all populations, then the number of such populations is:

$$|Q|=\left(\begin{vmatrix} S+N_0-1\\ N_0\end{vmatrix}\right) \quad (24)$$

To measure the strengths and weaknesses of such population, the fitness of $q$ is defined as:

$$Fitness(q)=\max\{Fitness(x^i)|i=1,2,\cdots,N_0\} \quad (25)$$

© 2013 NSP
Natural Sciences Publishing Cor.

therefore $F_{|F|} \leqslant Fitness(q) \leqslant F_1$, $\forall q \in Q$.

$Q$ is divided into nonvoid subset $\{Q_i\}$

$$Q_i = \{q | q \in Q and Fitness(q) = F_i\}, \quad i = 1,2,\cdots,|F| \tag{26}$$

Obviously,

$$\bigcup_{i=1}^{|F|} Q_i = Q \tag{27}$$

$$Q_i \neq \emptyset, \forall i \in \{1,2,\cdots,|F|\} \tag{28}$$

and set $Q_i$ contains all populations with fitness.

Let $q_{ij}$ denotes the $j$th population in $Q_i$, $i = 1, 2, \cdots, |F|$, $j = 1,2,\cdots,|Q|$. With the function of organization evolution operator, population state transfer from $q_{ij}$ to $q_{kl}$ can be expressed as $q_{ij} \rightarrow q_{kl}$. Let $pr_{ij,kl}$ the probability of $q_{ij}$ transferring to $q_{kl}$, $pr_{ij,k}$ the probability of $q_{ij}$ transferring to any of the population in $Q_k$, and $pr_{i,k}$ the probability of any population in $Q_i$ transferring to any of the population in $Q_k$, then apparently

$$pr_{ij,k} = \sum_{l=1}^{|Q_k|} pr_{ij,kl}, \sum_{k=1}^{|F|} pr_{ij,k} = 1, pr_{i,k} \geqslant pr_{ij,k} \tag{29}$$

The global convergence of this optimization method is is proved below:

**Definition1.** An evolutionary algorithm converges to the global optimum solution, iff

$$\lim_{t\to\infty} pr\{Fitness(q(t)) = F^*\} = 1 \tag{30}$$

wher$pr$ denotes probability, $q(t)$ the $t$the generation population.

**Theorem 1.** When the organization adjust the particle swarm, $\forall_{i,k} \in \{1,2,\cdots,|F|\}$, then

$$\begin{cases} pr_{i,k} > 0, & k \leqslant i \\ pr_{i,k} = 0, & k > i \end{cases} \tag{31}$$

**Demonstration:** $\forall q_{ij} \in Q_i \Rightarrow \exists x^l \in q_{ij}$, $Fitness(x^l) = F_i$; $j = 1,2,\cdots,|Q_i|$. Given the population $q_{ij}$ transferred to $q_{kl}$ under the function of organization evolutionary operator. Splitting operator does not generate any new individuals, but just transfers some particles to another organization, so it does not take any impact on the population fitness. Besides, in the annexation and cooperation operators, the parent leaders with greater adaptability in two parental organizations are duplicated to the progeny organizations. Finally, in the speed update operator, particles are moving toward the current optimal value with a specific wight. Since the retaining of optimal individual is used in the process of speed update operator, if the particles after speed update precedes the current optimal individual, then the new particles will be replaced by the current-generation particles; otherwise the current -generation particles should be retained. Because $x^l$ is definitely a leader in population $q_{ij}$ with fitness no less than other leaders, then

$$x^l \in q_{kl} \Rightarrow Fitness(q_{kl}) \geqslant Fitness(q_{ij}) \Rightarrow k \leqslant i$$
$$\Rightarrow pr_{i,k} = 0, \forall k > i \tag{32}$$

Given individual $x^{'}$ with $Fitness(x^{'}) = F_k$, $k \leqslant i$and $x^{'}$ has $n_1$ components $\{x_1, x_2, \cdots, x_{n1}\}$, wherein each component is different from those corresponding to the positions by $x^1$. $\exists org \in q_{ij}$ and the leader of $org$ is $leader = x^1$. Supposing $|org| \leqslant Max$, then the probability of executing annexation operator on $org$ is $pr_1 = (1 - |org|/N_0)/2 > 0$. In annexation operator strategy 2, for each component, if $U(0,1) < 1/n$, then the component will be randomly reelected from $[\bar{x_i}, x_i]$, thus the probability of generating $x^{'}$ by $x^1$ is

$$pr_2 = (1 - \frac{1}{n})^{(n-n_1)} \cdot \prod_{i=1}^{n_1} \frac{\varepsilon}{n(\bar{x_i} - x_i)} > 0 \tag{33}$$

After the effect of evolutionary operator, the probability of transferring from $q_{ij}$ to any population in $Q_k$ is

$$pr_{ij} > pr_1 \times (1 - AS) \times pr_2 > 0 \tag{34}$$

Therefore, $pr_{i,k} \geqslant pr_{ij,k} > 0, \forall k \leqslant i$.

**Theorems2.** Organizational evolutionary algorithm feature global convergence.

**Demonstration:** Elitism strategy is employed here. From Theorem 1, $pr_{i,k} > 0, k \leqslant i$ can be inferred that the probability transferring from one state to another $pr > 0$, e.g., Hence the probability of failure in finding the optimal solution for the probability is

$$\lim_{n\to\infty} (1-c)^n = 0 \tag{35}$$

In other words, this optimized algorithm finds the optimal solution with probability 1, and thus features global convergence.

### 3.3. process of OAPSO-PF

(1) When k=0, take$N$particles as samples from importance function at the initial time. The fitness function are expressed in equation(36)

$$Fitness = \exp[-\frac{1}{2R_k}(z_{New} - z_{\mathrm{Pr}ed})] \tag{36}$$

Meanwhile, Assume these $N$ samples as$N$ initial organizations.

(2) calculate the importance value:

$$w_k^i = w_{k-1}^i p(z_k | x_{k-1}^i) = w_{k-1}^i p(z_k | x_k^i) \tag{37}$$

(3) if the termination conditions are satisfied, go to step (6), Otherwise execute the next step.

© 2013 NSP
Natural Sciences Publishing Cor.

(4) If each organization meets the splitting condition, then execute the splitting operator.

(5) Randomly select two parental organization $org_{i1}$ and$org_{i2}$, then randomly select annexation or cooperation operators that are applied to the two parental organization.

(6) Update the speed and position of particles, calculate the speed $v_k^{im}$ of particle$x_k^{im}$ after $m$th iteration through equation (38). By using equation (39), the particle$x_k^{im}$, under the effect of speed $v_k^{im+1}$, reach the position $x_k^{im+1}$ of next iteration:

$$v_k^{im+1}(t+1) = w(t)v_k^{im+1}(t) + c_1 r_1(pb_{k-1}^i - x_{k-1}^{im}) + c_2 r_2(pg_k - x_{k-1}^{im}) \quad (38)$$

$$x_{k+1}^{im+1} = x_k^{im} + v_k^{im+1} \quad (39)$$

(7) Compare their fitness, update $pb$and$pg$:

$$pb_k^i = \begin{cases} pb_k^i, & Fitnessx_g) < Fitness(pb_k^i) \\ x_g, & Fitnessx_g) > Fitness(pb_k^i) \end{cases} \quad (40)$$

$$pg_k \in \{x_k^1, x_k^2, x_k^3, \cdots x_k^N | Fitness(x)\} = \max\{Fitness(x_k^1), Fitness(x_k^2), \cdots Fitness(x_k^N)\} \quad (41)$$

(8) When the optimal value of particle complies with the initially-set threshold value$\varepsilon$or algorithm reached maximum iteration times $\lambda$, optimization should be stopped. Else jump to step(3).

(9) Calculate the importance weight of the particles after optimization and perform normalization:

$$w_k^i = w_k^i / \sum_{i=1}^{N} w_k^i \quad (42)$$

(10) State output:

$$\tilde{x} = \sum_{i=1}^{N} w_k^i x_k^i \quad (43)$$

## 4. Simulation experiment

### *4.1. Univariate nonstationary growth model*

Choosing a univariate nonstationary growth model (UNGM) [15], let$w(t)$ and $v(t)$ are zero-mean Gaussian noise, the process and measurement model of the simulated objects are given as follows:

$$x(t) = 0.5x(t-1) + \frac{25x(t-1)}{1+[x(t-1)]^2} + 8\cos[1.2(t-1)] + w(t) \quad (44)$$

$$z(t) = \frac{x(t)^2}{20} + v(t) \quad (45)$$

By using three algorithm, state estimation and tracking of this non-linear system are performed.

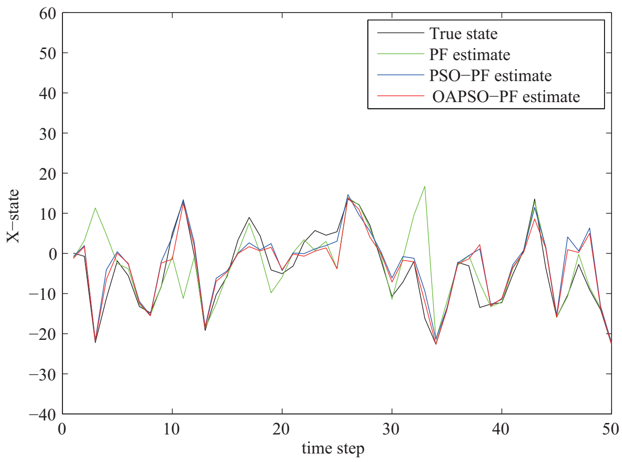


**Figure 1** state estimation of different algorithm

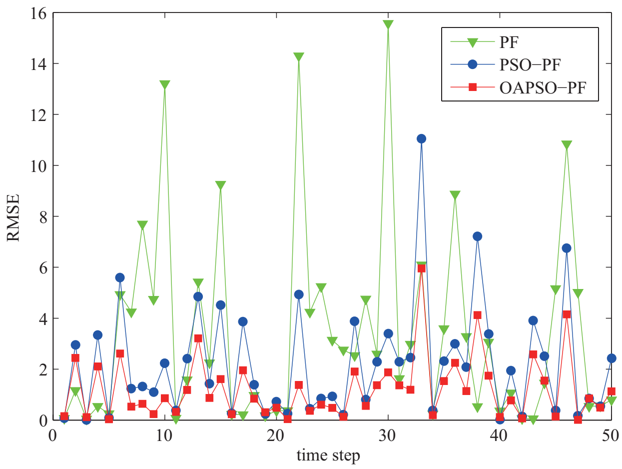


**Figure 2** RMSE of different algorithm

**Table 1** comparison of simulation by UNGM model in different parameters environment ($Q = 10$)

| Parameters | Algorithm | RMSE | Time |
|---|---|---|---|
| $\varepsilon = 0.05$ $\lambda = 100$ | PSO-PF | 1.9625 | 0.3672 |
| | OAPSO-PF | 1.3728 | 0.3387 |
| $\varepsilon = 0.05$ $\lambda = 200$ | PSO-PF | 1.7637 | 0.5303 |
| | OAPSO-PF | 1.3201 | 0.3809 |
| $\varepsilon = 0.075$ $\lambda = 100$ | PSO-PF | 1.678 | 0.5136 |
| | OAPSO-PF | 1.170 | 0.4885 |
| $\varepsilon = 0.075$ $\lambda = 200$ | PSO-PF | 1.4942 | 0.7054 |
| | OAPSO-PF | 1.0852 | 0.5963 |

Giving the process noise variance$Q = 10$, measurement noise variance $R$ =1, and particle number$N = 100$, and initially-set threshold value is$\varepsilon$and maximum iteration times is$\lambda$, simulation is conducted.

© 2013 NSP
Natural Sciences Publishing Cor.

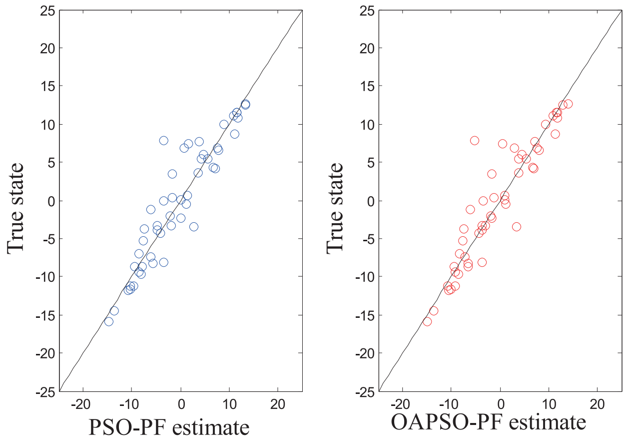


**Figure 3** comparison of correlation

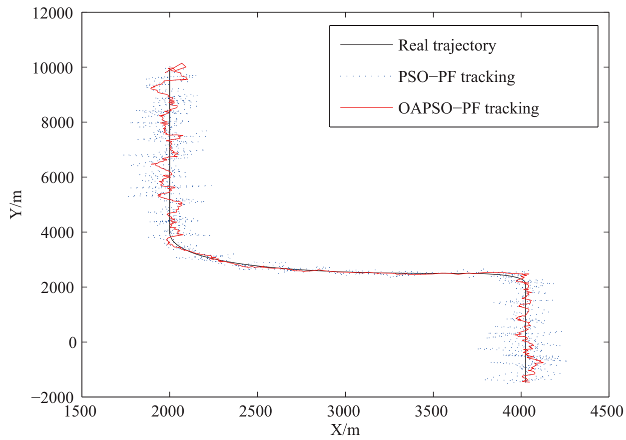


**Figure 4** different algorithm's performance in tracking

From the experimental result, it is apparent that giving different noise intensities, the errors of PSO and OAPSO-PF are obviously lower that the standard PF as the particle swarm optimization is an iterative optimizing process, which effectively increases the effectiveness of particles. OAPSO-PF can achieve the best precision, correlation, and reduces calculation time compared to PSO-PFIn addition, it can be observed from can be seen from Table 1 that the change of maximum number of iteration will influence greatly the changes of OAPSO-PF, which also indicates that OAPSO-PF increases the success of optimization and reduces the probability of reaching the maximum iterations.

### *4.2. target tracking model*

Given that the radar is tracking a target moving on a flat surface and in uniform linear motion along $y$ axis at $t = 0 \rightarrow 400$th second, during which the speed is$-15m/s$ and the starting point of target is (2000m, 1000m). At $t = 400 \rightarrow 600$ the target turns slowly at$90°$ along $x$ axis, wherein the accelerated speed is $u_x = u_y = 0.075m/s^2$ and the speed decreases to 0 after the turning. The slow turning since $t = 610s$ at $90°$ showed accelerated speed $u_x = u_y = 0.3m/s^2$, and ended at $660s$ where the accelerated speed decreased to 0. The radar scanning cycle is $T = 2s$.

**Table 2** comparison of simulation parameters

| Algorithm | Absolute value of error in X direction | Absolute value of error in Y direction | Time |
|---|---|---|---|
| PSO-PF | 32.7382 | 49.6251 | 4.6572 |
| OAPSO-PF | 24.7261 | 35.4621 | 4.2108 |

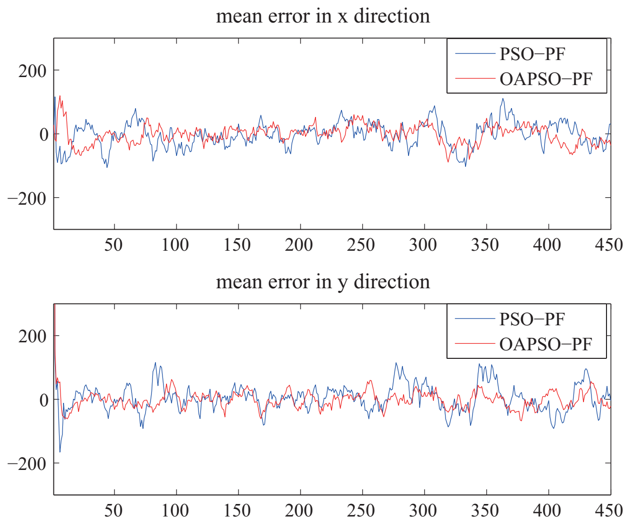


**Figure 5** mean curve of filter error

The simulation results show that errors in target tracking at the turning are increasing, while PSO-PF displays more evident errors. The tracking trace fluctuates a lot, while the error of $OAPSO - PF$ is smaller. At the same time, a batter tracking trace fitness and a shorter calculation time are proved, conducing to accurate and high-efficient tracking of maneuvering target turning continuously.

## 5. Conclusion

Standard PSO-PF only uses the memory feature of particles in particle swarm optimization, whereas other features are neglected. The algorithm presented here makes full use of coordination and self-learning features of the individuals in an organization to guide the particles to evolve con-

© 2013 NSP
Natural Sciences Publishing Cor.

stantly, finally reaching the goal of global optimization. At last through experiment the performance of the algorithm is analyzed and the results may suggest that this algorithm features stable performance and high successful rate, accordingly helping to enhance the efficiency of particle filtering.

## Acknowledgements

This paper was supported by the National Nature Science Foundation of China (No. 61104196), the National defense pre-research fund of China (No. 40405020201), the Specialized Research Fund for the Doctoral Program of Higher Education of China (No. 20113219110027).

## References

[1] Sunny Chauhan and Neeraj Dhiman, Common Theorems on Fuzzy Metric Spaces Relation, Math. Sci. Lett. 1 (2012) 89-96.

[2] Wang Xue-guang, Chen Shu-hong, An Segmentation Algorithm Based on Two- Method, Inf. Sci. Lett. 1 (2012) 77-83.

[3] Zhang G, Cheng Y, Yang F, Pan Q. Particle filter based on PSO. *Intelligent Computation Technology and Automation, Hunan*, (2008) 121-124.

[4] Jing W, Zhao H, Song C H, et al. A optimized particle filter based on PSO algorithm. *International Conference on Future BioMedical Information Engineering, Sanya, China,* (2009)122-125.

[5] Yihua Yu, Xuanyuan Zheng. Particle filter with ant colony optimization for frequency offset estimation in OFDM systems with unknown noise distribution. *Signal Processing.* **91**(2011) 1339-1342.

[6] Sanjeev M, Maskell S. *A* tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing,* **50**(2002)174-188.

[7] Chen Shu, Wu Guo Qing and Ye Jun Min, A Domain Oriented Approach for Network Software Requirement Modeling, Inf. Sci. Lett. **1** (2012) 21-39.

[8] Zhao, Jing, Li, Zhiyuan. Particle filter based on Particle Swarm Optimization resampling for vision tracking. *Expert Systems with Applications.* **37**(2010) 8910-8914.

[9] Chunlin Ji, Yangyang Zhang, Mengmeng Tong, et al. Particle Filter with Swarm Move for Optimization. *Lecture Notes in Computer Science.* (2008)909-918.

[10] Ying Li, Bendu Bai, Yanning Zhang. Improved particle swarm optimization algorithm for fuzzy multi-class SVM. *Journal of Systems Engineering and Electronics.21*(2010) 509-513.

[11] Xin Liu, Mahesan Niranjan. State and parameter estimation of the heat shock response system using Kalman and particle filters. *Bioinformatics.* **28**(2012)1501-1507.

[12] Coase R H. The firm, the market and the law. *Chicago, USA: University of Chicago Press,* (1988).

[13] Wilcox J R. Organization learning within a learning classifier System. *MS Dissertation. Illonois, USA: university of Illinois. Department of Computer science,* (1995).

[14] Liu Jing, Zhong Weicai, Liu Fang, et al. An organizational evolutionary algorithm for constrained and unconstrained optimization problems. *Chinese journal of computers,* **27**(2004) 157-167.

---

**Zhimin Chen** received Master's degree from Nanjing University of Science and Technology in 2010. He is currently a PhD candidate in Nanjing University of Science and Technology. His research interests are in the areas of Mathematics, intelligent optimization, network architecture and information systems.

**Yuming Bo** received the Master's degree from Nanjing University of Science and Technology in 1987, and the PhD degree in automation from Nanjing University of Science and Technology in 1996. He is currently a professor, PhD supervisor and president of school of automation in Nanjing University of Science and Technology. His research interests are in the areas of Control theory and control applications, information processing, intelligent optimization algorithm and information systems.

**Panlong Wu** received the Master's degree from Nanjing University of Science and Technology in 2004, and the PhD degree in automation from Nanjing University of Science and Technology in 2008. He is currently a professor and supervisor of master students in Nanjing University of Science and Technology. His research interests are in the areas of information processing, applied mathematics and information systems.

© 2013 NSP
Natural Sciences Publishing Cor.

**Weijun Zhou** received the Master's degree from Nanjing University of Science and Technology in 2005, and the PhD degree in automation from Nanjing University of Science and Technology in 2008. He is currently an associate professor in Nanjing University of Science and Technology. His research interests are in the areas of Control theory and control applications, information processing, intelligent optimization algorithm and applied mathematics

© 2013 NSP
Natural Sciences Publishing Cor.