Front. Comput. Sci. China 2009, 3(1): 31–37
DOI 10.1007/s11704-009-0008-4
RESEARCH ARTICLE

Monitoring of particle swarm optimization

Yuhui SHI(✉)[1], Russ EBERHART[2]

1 Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China
2 Department of Electrical and Computer Engineering, Indiana University-Purdue University Indianapolis, Indianapolis, IN 46202, USA

Abstract In this paper, several diversity measurements will be discussed and defined. As in other evolutionary algorithms, first the population position diversity will be discussed followed by the discussion and definition of population velocity diversity which is different from that in other evolutionary algorithms since only PSO has the velocity parameter. Furthermore, a diversity measurement called cognitive diversity is discussed and defined, which can reveal clustering information about where the current population of particles intends to move towards. The diversity of the current population of particles and the cognitive diversity together tell what the convergence/divergence stage the current population of particles is at and which stage it moves towards.

Keywords particle swarm optimization, population diversity, cognitive diversity

1 Introduction

Particle swarm optimization (PSO) was invented by Russ Eberhart and James Kennedy in 1995 through simplifying a social simulation model which was originally developed to simulate the process of birds seeking food [1,2]. The PSO algorithm is a population-based evolutionary algorithm. Like other evolutionary algorithms, each individual (called particle in PSO) in the population represents a candidate solution to the problem to be solved. Unlike other evolutionary algorithms, each individual/particle has a velocity parameter associated with it in addition to its position parameter in the solution space, which is the only parameter that an individual in other evolutionary algorithms has [3–5]. Each particle "flies" through the solution space with a velocity which is dynamically changed according to its own flying experience and its companion's flying experience. It is this velocity changing rule through which all the particles communicate and share information among themselves. Furthermore, it is this sharing and communicating mechanism that enables particles to fly towards better and better search areas while at the same time to risk to be stuck into local minima [6,7].

The search process or flying trajectories of particles are complicated and nonlinear. To search for good enough solutions, especially for the multi-modal optimization problems, the search process needs to have the ability to converge at some time while diverging at other times in order to have the ability to find good enough solutions and to be able to avoid to be stuck in un-wanted local minima. Therefore, it is critical to have a capability to monitor the search process of PSO in order to first understand the PSO search process and then design a better algorithm or even have possibilities to control the search process later.

A straightforward approach to measure the diversity of PSO is to use the standard deviation of the fitness values of all the population particles [8,9]. Population fitness values are attributes of the PSO behaviors and not the PSO particles themselves directly. Therefore, this kind of diversity measurement is simple but it is an indirect measurement of the population diversity. In Ref. [10], the diversity of PSO has been looked at from different perspectives. Each particle in a PSO has an n-dimensional velocity associated with it in addition to its position as in other evolutionary algorithms. Therefore, diversities with regards to particles' positions and velocities have been discussed in Ref. [10] instead of only the position diversity as in other evolutionary algorithms. Furthermore, for velocity diversity, both velocity speed diversity and velocity directional diversity were discussed since each velocity has two attributes, its speed and its direction. The velocity speed tells how fast a particle is

Received August 11, 2008; accepted December 1, 2008

E-mail: Yuhui.Shi@xjtlu.edu.cn

flying and the velocity direction tells where a particle is flying towards.

In this paper, several diversity measurements will be discussed to monitor the search process. The basic PSO algorithm will be reviewed in Section 2. In Section 3, diversity measurements will be discussed, followed by discussions and conclusions in Sections 4 and 5, respectively.

2 Particle swarm optimization algorithm

The original PSO algorithm is very simple in concept and easy in implementation. It can be formulated as [6,7]

$$v_{id}(t+1) = w_{id}(t)v_{id}(t) + c_1 rand()(p_{id}(t) - x_{id}(t)) + c_2 Rand()(p_{gd}(t) - x_{id}(t)), \quad (1a)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1). \quad (1b)$$

where c_1 and c_2 are positive constants, and *rand*() and *Rand*() are two random functions in the range [0,1] and are different for all dimensions and all particles; $x_{id}(t)$ represents the dth position value of the ith particle at time step t; $p_{id}(t)$ represents the dth position value of the best previous position (the position giving the best fitness value) of the ith particle at the time step t; The symbol g represents the index of the best particle among all the particles in its neighborhood; $v_{id}(t)$ represents the rate of the dth position value change (velocity) for particle i at time step t; $w_{id}(t)$ is the inertia weight for the dth element of particle i at time step t. Usually, all the $w_{id}(t)$ will have the same value for simplicity, but the inertia weight can be dynamically adjusted according to the current and historical performance of the particles, which will improve the PSO's performance since the search process of a PSO algorithm is nonlinear and complicated [9]. A simple and straightforward approach is to linearly decrease inertia weigh over the course of PSO [6]. Other PSO parameters can be fixed and/or even can be dynamically changed to affect the search process in the hope of having a more diverse or better performed PSO particles [11,12].

Equation (1) is the equation governing the flying trajectory of particles. Eq. (1a) tells how the velocity is to be changed. In order not to violate the physical law, the velocity can not be changed abruptly and shall be changed from the current velocity, which is reflected by the first part of the Eq. (1a) as a "flying" particle's momentum. The other two parts of the Eq. (1a) reflect the learning and collaboration capability of a particle. The second part reflects a particle's self-learning capability or self-cognition, that is, a particle learns from its own flying experience. The third part reflects particle's collaboration capability, that is, a particle learns from "flying" experience of its neighboring particles. The position of a "flying" particle is adjusted according to the Eq. (1b) [3,6,7].

There are two most commonly used versions of PSOs, global version and local version. In a global version PSO, a single and unique *gbest* is shared by all particles in the whole population. In a local version PSO, each particle in the population may have different *gbest* which is the best performed particle within the particle's own neighborhood. In both global and local version PSO, particles fly through the search space with dynamically changed velocities according to the Eq. (1a). The neighborhood of each particle is generally defined as its topologically nearest particles at each side instead of Euclidean neighborhood. The global version PSO can be considered as a special case of a local version PSO if the whole population is considered as each particle's neighborhood. It has been claimed that the global version PSO converges fast, but with potential to converge to the local minimum, while the local version PSO might have more chances to find better solutions slowly [13,14].

The process for asynchronously implementing the *global* version of PSO is as follows [4,5]:

1) Randomly initialize the position and velocity values for a population of particles on n dimensions in the solution space.
2) For each particle, evaluate its performance according to its fitness (or evaluation) function in n variables.
3) If it is the first generation, it is copied as *pbest*, otherwise compare particle's fitness value with particle's *pbest*. If current fitness value is better than *pbest*, then set *pbest* value equal to the current fitness value, and the *pbest* position equal to the current position of the particle in n-dimensional solution space.
4) Compare fitness value with the population's overall previous best. If current value is better than *gbest*, then set the current particle to be the *gbest*.
5) Update the velocity and position of the particle according to Eq. (1).

The above steps loop through cycle by cycle until usually a good enough solution has been found or a pre-set maximum number of generations have been reached.

PSO is very good at finding good enough solutions for a large range of problems, such as constrained optimization problems [15], multi-objective optimization problems [16], etc.

For the purpose of easiness of discussion below and without loss of generality and clarity, the time step t is omitted in the following discussion and the positions and velocities of particles in a population at time step t can be mathematically represented in vector format as follows, where m is the population size of the swarm, and n is the number of dimensions (variables) for each particle.

$$x_i = \{x_{i1}, x_{i2}, \ldots, x_{in}\}, i = 1, 2, \ldots, m \qquad (2a)$$

$$v_i = \{v_{i1}, v_{i2}, \ldots, v_{in}\}, i = 1, 2, \ldots, m \qquad (2b)$$

$$p_i = \{p_{i1}, p_{i2}, \ldots, p_{in}\}, i = 1, 2, \ldots, m \qquad (3)$$

The population of particles' positions and velocities can also be represented in matrix forms as follows:

$$X = (x_1, x_2, \ldots, x_m)^{\mathrm{T}} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}. \qquad (4)$$

$$V = (v_1, v_2, \ldots, v_m)^{\mathrm{T}} = \begin{pmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & & & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mn} \end{pmatrix}. \qquad (5)$$

The same applies to *pbest* as below.

$$P = (p_1, p_2, \ldots, p_m)^{\mathrm{T}} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & & & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mn} \end{pmatrix}. \qquad (6)$$

3 Population diversity

As mentioned in Section 1, a PSO search process is complicated and nonlinear. Generally, a good PSO algorithm should have a balanced exploration and exploitation capability. When the particles are away from good enough solutions and are diverse enough, the PSO should have more exploitation capability than exploration capability, that is, the PSO should be more in the converging process; when the particles are clustered and are away from good enough solutions, the PSO should have more exploration capability than exploitation capability, that is, the PSO shall be more in the diverging process. Intuitively, a good way to monitor the degree of convergence/divergence of PSO search process is to calculate or evaluate its diversity from time to time.

For PSO, unlike other evolutionary algorithms, each particle (or individual) has a velocity associated with it in addition to its position; therefore for PSO, not only the particles' position diversity needs to be studied, but also particles' velocity diversity. In this section, population diversity measurements are discussed.

3.1 Position diversity

Several methods have been provided to measure the particle swarm's population position diversity and the velocity diversity, such as element-wise population position diversity, Euclidean distance-based population position diversity. For the population position diversity, in Ref. [10], the dimension-wise diversities were studied and defined so that the particles' diversity/performance can be monitored dimension-by-dimension since it is not uncommon that in PSO, particles are in convergence stage with regards to some dimensions while in divergence stage for some other dimensions. For detail on position diversity, please refer to Ref. [10].

3.2 Velocity diversity

In Ref. [10], for the population velocity diversity, both the directional and speed diversities of the particles' velocities in a population are discussed. For the population velocity speed diversity, dimension-wise diversity is defined while for the population velocity directional diversity, it is calculated by measuring the angular difference between each particle's normalized velocity's direction and the average normalized velocity's direction.

One concern with the population velocity directional diversity calculation using the angles between the particles' velocities and the average velocity is that all the angles are not at the same plane. Therefore, two velocities having the same angles to the average velocity does not mean the two have the same relations with the average velocity.

Here we propose to use the same dimension-wise method to calculate the population velocity directional diversity as in Ref. [10].

First, the velocity vector for each particle is normalized to have a unit length, therefore all the velocity vectors have the same length while the velocity's directional information remains the same.

$$v_{ij}^{nor} = \frac{v_{ij}}{\sqrt{\sum_{j=1}^{n} v_{ij}^2}}. \qquad (7)$$

For particles in a population, they will have different velocity directional behaviors for different dimensions. Directional differences for some dimensions among particles will be small while for others they maybe much larger. Even for some dimensions, they may end up having the same dimensional directional behavior. Therefore, it is helpful and/or useful to know the dimension-by-dimension velocity directional diversity,

$$\bar{v}_j^{nor} = \frac{1}{m}\sum_{i=1}^{m} v_{ij}^{nor}. \quad (8)$$

$$D_j = \frac{1}{m}\sqrt{\sum_{i=1}^{m}(v_{ij}^{nor} - \bar{v}_j^{nor})^2}. \quad (9)$$

In Eq. (8), the normalized average velocity value for each dimension is calculated, while the dimension-wise velocity directional diversity is calculated by Eq. (9). For an n-dimensional problem, we end up to have a n dimension-wise velocity directional diversity vector $(D_1, D_2, \ldots, D_n)$.

The velocity directional diversity as a whole can be calculated by combining the n dimensional velocity directional diversities. Several straightforward ways are discussed below.

3.2.1 Weighted summation of velocity directional diversities

The n dimension-wise velocity directional diversities can be weighted and then summed to be the velocity directional diversity as a whole as shown below:

$$D = \sum_{i=1}^{n} w_i D_i. \quad (10)$$

When all the weights are equal, Eq. (10) can be re-written as

$$D = w\sum_{i=1}^{n} D_i. \quad (11)$$

When w equals to 1, it is the summation of the directional diversities; when w equals to $1/n$, it is the average of the directional diversities.

3.2.2 Weighted maximization of velocity directional diversities

The n dimension-wise velocity directional diversities can be weighted and then the maximum weighted directional diversity is considered to be the velocity directional diversity as a whole as shown below:

$$D = \max\{w_i D_i\}. \quad (12)$$

When all the dimension-wise directional diversities are weighted the same and have the same weight 1, then the velocity directional diversity as a whole is equal to the maximal dimension-wise directional diversity.

3.2.3 Weighted sum-square of velocity directional diversities

The velocity directional diversity as a whole can also be the weighted sum-square of all dimension-wise velocity directional diversities as shown below:

$$D = \sqrt{\sum_{i=1}^{n} w_i D_i^2}. \quad (13)$$

When all weights are equal to 1, it is the Euclidean distance of the dimension-wise velocity directional diversity vector.

3.3 Cognitive diversity

Unlike other evolutionary algorithms, PSO has a memory of historical experience, that is, PSO has n different $pbest$, with one for each particle. These n different $pbest$ govern where particles fly toward. These n $pbest$ are updated generation-by-generation and therefore reveal some dynamic information about where the target search areas tend to be generation over generation. Therefore, the diversity of these n $pbest$ provides us rich information about particles' dynamic target search areas. In this section, measurements for the diversity of n $pbest$, called cognitive diversity, are discussed.

3.3.1 Element-wise cognitive diversity

A straightforward approach is to calculate the difference among all elements of all n *pbest* as shown below:

$$\bar{p} = \frac{1}{n\times m}\sum_{i=1}^{m}\sum_{j=1}^{n} p_{ij}. \quad (14)$$

$$D_E^c = \frac{1}{n\times m}\sum_{i=1}^{m}\sum_{j=1}^{n}[p_{ij} - \bar{p}]^2. \quad (15)$$

In Eq. (14), the average value of all elements of all n *pbest* is first calculated and Eq. (15) defines the cognitive diversity as the average squared difference between all elements of all n *pbest* and the average value calculated in Eq. (14)

3.3.2 Euclidean distance-based cognitive diversity

In this approach, the Euclidean distance between each pair of particle's *pbest* is first calculated, then the distance is normalized within the dynamic range of particle's position $[a, b]$, finally the Euclidean distance-based cognitive diversity is defined as the average of all the normalized distances as shown in Eqs. (16)–(18) below:

$$d^c(p_i, p_j) = \|p_i - p_j\|. \quad (16)$$

$$\tilde{d}^c(p_i, p_j) = \frac{d^c(p_i, p_j)}{\|a - b\|}. \quad (17)$$

$$D_{ED}^c = \frac{2}{n(n-1)}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\tilde{d}(p_i, p_j). \quad (18)$$

3.3.3 Dimension-wise cognitive diversity

It is not uncommon that for some dimensions, n $pbest$ have very similar dimensional values while other dimensions have very different dimensional values, therefore, it is also critical to look at the cognitive diversity dimension-by-dimension. The dimension-wise cognitive diversity can be calculated using the Eq. (19) shown below:

$$\bar{p}_j = \frac{1}{m}\sum_{i=1}^{m} p_{ij},$$
$$D_j^c = \frac{1}{m}\sum_{i=1}^{m} [p_{ij} - \bar{p}_j]^2. \quad (19)$$

In Eq. (19), the average value is first calculated dimension-by dimension. Then for each dimension, the average difference between the element value of that dimension and the average value for all particles is defined as the dimensional cognitive diversity. Therefore, we have an n-dimensional cognitive diversity vector $(D_1^c, D_2^c, \ldots, D_n^c)$. Based on this n-dimensional cognitive diversities, there are several ways to measure the dimension-wise cognitive diversity as a whole.

3.3.3.1 Weighted summation cognitive diversity

The dimension-wise cognitive diversity can be defined as a weighted summation of n dimensional cognitive diversities as shown below:

$$D_{WS}^c = \sum_{j=1}^{n} w_j D_j^c. \quad (20)$$

where w_j is a positive value (weight) less than or equal to 1. If all the dimensions are treated equally, we have $w_j = 1/n$ for all values of j. Then Eq. (20) can be rewritten as Eq. (21):

$$D_{WSew}^c = \frac{1}{n}\sum_{j=1}^{n} D_j^c. \quad (21)$$

3.3.3.2 Weighted maximization cognitive diversity

The maximal value of n weighted dimensional cognitive diversities can be defined as the dimension-wise cognitive diversity as a whole as shown below:

$$D_{WM}^c = \max\{w_j D_j^c\}, \qquad j = 1, 2, \ldots, n \quad (22)$$

where w_j is the weight for the D_j^c dimensional cognitive diversity for the jth dimension.

3.3.3.3 Cognitive diversity vector length

The dimension-wise cognitive diversity can also be defined as the weighted Euclidean length of the dimensional cognitive diversity vector as shown in Eq. (23):

$$D_L^c = \sqrt{\sum_{j=1}^{n} w_j D_j^{c^2}}, \qquad j = 1, 2, \ldots, n \quad (23)$$

when all weights are the same and equals to 1, it is the Euclidean distance of the dimensional cognitive diversity vector.

3.3.3.4 Normalized dimension-wise cognitive diversities

Another method to calculate the cognitive diversities analogous to Eqs. (19)–(23) is to normalize the *pbest* position values with regards to the dynamic range $[a, b]$ before calculating the cognitive diversities. The ith particle's *pbest* position value can be normalized as

$$\hat{p}_{ij} = \frac{p_{ij}}{|b_j - a_j|}, \qquad a_j \leqslant p_{ij} \leqslant b_j;$$
$$i = 1, 2, \ldots, m; \qquad j = 1, 2, \ldots, n \quad (24)$$

The n normalized dimensional cognitive diversity can then be calculated according to Eq. (25):

$$\bar{p}_j = \frac{1}{m}\sum_{i=1}^{m} \hat{p}_{ij},$$
$$D_{jN}^c = \frac{1}{m}\sum_{i=1}^{m} [\hat{p}_{ij} - \bar{p}_j]^2. \quad (25)$$

The normalized dimension-wise cognitive diversity as a whole can then be calculated by using the weighted summation, weighted maximization, or diversity vector length approach as defined in Eqs. (20)–(23).

4 Discussion

The diversity of particles is a good measurement to use for monitoring the search process of PSO, and a good metric to use for revealing when the particles have risk to be stuck in local minima. The PSO algorithm has a memory of previous best performance for each particle in addition to a population of particles. Therefore, there are two kinds of diversities, one for current particles (population diversity) while the other for *pbest* (cognitive diversity). The diversities of current particles, whether it is position diversity, velocity speed diversity, velocity directional diversity, or a combination of them, tells how diverse the current particles' positions or velocities are. The diversities of *pbest*, the cognitive diversities, on the other hand, reveal how diverse the target areas where the current particles intend to fly toward generation-over-generation. The higher the cognitive diversity is, the more diverse target areas that particles will move toward are, that is, the population of particles has more potential to be more diverse.

With different combinations of PSO's population diversity and cognitive diversity, the PSO will have different transitions between exploration (or global search) and exploitation (or local search). Table 1 lists all the possible combinations.

Table 1 Combinations of PSO population diversity and cognitive diversity

	High population diversity	Low population diversity
High cognitive diversity	global search → global search	local search → global search
Low cognitive diversity	global search → local search	local search → local search

4.1 High population diversity vs. High cognitive diversity

When PSO is under this combination, the current population of particles has high diversity, that is, the PSO is more in the exploration stage. Since the cognitive diversity is also high, the particles have more opportunity to fly toward high diverse target areas. The particles will keep their exploration capability (or global search capability).

4.2 High population diversity vs. Low cognitive diversity

Under this combination, the population of particles with high diversity flies toward low diverse target areas. The particle population changes its search capability from exploration capability (or global search capability) to exploitation capability (or local search capability). The particles are in converging stage.

4.3 Low population diversity vs. High cognitive diversity

Under this combination, the population of particles with low diversity flies toward high diverse target areas. The particles have the tendency to become more and more diverse, and the particle population changes its search capability from exploitation capability to exploration capability. The particles are expanding. This is helpful when the particles need to jump out of local minima. The particles are in diverging stage.

4.4 Low population diversity vs. Low cognitive diversity

Under this combination, the population of particles with low diversity flies toward low diverse target areas. The particles keep their exploitation capability and keep on their fine search.

These four combinations can describe the search process of PSOs. A PSO with good search capability should have the capability to change among the four combinations. An effective search process shall be a process that is repeatedly changing between converging process and diverging process.

5 Conclusions

Monitoring the search process of a PSO is very useful for understanding how the PSO algorithm can find good enough solutions for a problem. One big problem for applying PSO and other evolutionary algorithms is that it can get stuck in a local minimum before it finds a good enough solution for a problem. In order to avoid or reduce the risk of getting stuck in a local minima, some mutation-like operations have been added to the PSO. When the PSO is on the verge of being stuck in an un-wanted local minima, mutation operation is added to move particles away from the un-wanted local minima. Judging whether the PSO is on the verge of getting stuck in an un-wanted local minima is very critical here. A useful metric for revealing this kind of information is the diversity of particles. Like in the other evolutionary algorithms, in this paper, first the position diversity was introduced and defined. Unlike the other evolutionary algorithms, PSO also has diversity parameter in addition to position parameter. Therefore, velocity diversity was discussed and defined. For velocity diversity, both velocity speed diversity and velocity directional diversity were discussed and defined. For PSO, it also has a memory of its best performed position for each particle in its history. Therefore, a diversity called cognitive diversity was also discussed and defined. The population diversity including position diversity and velocity diversity reveals clustering information about the current population of particles, while the cognitive diversity reveals clustering information about where the current population of particles intend to move towards. The two diversities together reveal the information about the transition of particles from one kind of convergence/divergence stage to another.

The measurement of diversity is only the first step, but it is an important step. By understanding the search process of PSO better and better, we can then start to manage diversity. For example, we can use it to design a better PSO in which the search process can be controlled to have optimal diversity values over the running course of the PSO. Or simply, it can at least provide us some guidelines of when and how to apply mutation-like operations in PSO. In order to manage diversity, we first need to be able to measure it. In this paper, we have discussed several approaches for measuring diversity.

References

1. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science. Piscataway: IEEE Service Center, 1995, 39–43

2. Kennedy J, Eberhart R. Particle swarm optimization. In: Procedings of IEEE International Conference on Neural Networks (ICNN), 1995, IV: 1942–1948
3. Eberhart R, Shi Y H. Comparison between genetic algorithms and particle swarm optimization. In: Porto V W, Saravanan N, Waagen D, Eiben A E, eds. Evolutionary Programming VII: Proceedings of 7th Annual Conference on Evolutionary Programming. Berlin: Springer-Verlag, 1998, 611–616
4. Eberhart R, Shi Y H. Computational Intelligence: Concepts to Implementations. Morgan Kaufmann Publishers, 2007
5. Kennedy J, Eberhart R, Shi Y H. Swarm Intelligence. Morgan Kaufmann Publishers, 2001
6. Shi Y H, Eberhart R. Parameter selection in particle swarm optimization. In: Proceedings of the 1998 Annual Conference on Evolutionary Computation, 1998, 591–600
7. Shi Y H, Eberhart R. A modified particle swarm optimizer. In: Proceedings of the 1998 IEEE International Conference on Evolutionary Computation. Piscataway: IEEE Press, 1998, 69–73
8. Shi Y H, Eberhart R, Chen Y B. Implementation of evolutionary fuzzy system. IEEE Transactions on Fuzzy Systems, 1999, 7(2): 109–119
9. Shi Y H, Eberhart R. Fuzzy adaptive particle swarm optimization, In: Proceedings of the 2001 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2001, 101–106
10. Shi Y H, Eberhart R. Population diversity of particle swarm optimization. In: Proceedings of the 2008 Congress on Evolutionary Computation, 2008, 1063–1067
11. Fan H Y, Shi Y H. Study on Vmax of particle swarm optimization. In: Proceedings of the Workshop on Particle Swarm Optimization. Indianapolis: Purdue School of Engineering and Technology, IUPUI. April, 2001
12. Ratnaweera A, Halgamuge S, Watson H. Self-organizing hierarchical particle swarm optimizer with time varying accelerating Coefficients. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 240–255
13. Kennedy J, Mendes R. Population structure and particle swarm performance. In: Proceedings of the 2002 Congress on Evolutionary Computation. Honolulu, 2002
14. Mendes R, Kennedy J, Neves J. The fully informed particle swarm: simpler, maybe better. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 204–210
15. Parsopoulos K E, Vrahatis M N. Particle swarm optimization method for constrained optimization problems. In: Sincak P, et al, eds. Intelligent Technologies – Theory and Application, 2002, 214–220
16. Reyes-Sierra M, Coello Coello C A. Multi-objective particle swarm optimizers: a survey of the state-of-the-art. International Journal of Computational Intelligence Research, 2006, 2(3): 287–308