International Journal of Computational Intelligence Systems, Vol. 7, No. 6 (December 2014), 1022-1043

A new joint spectral radius analysis of random PSO algorithm

Jun Liu[1,2], **Hongbin Ma**[2,3]*, **Xuemei Ren**[2,3], **Tianyun Shi**[1], **Ping Li**[1]

[1] *Institute of Computing Technologies, China Academy of Railway Sciences, Beijing, China*
[2] *School of Automation, Beijing Institute of Technology, Beijing, China*
[3] *National Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing, China*

Received 8 June 2013

Accepted 24 December 2013

Abstract

The existing stability analysis of particle swarm optimization (PSO) algorithm is chiefly concluded by the assumption of constant transfer matrix or time-varying random transfer matrix. Firstly, one counterexample is provided to show that the existing convergence analysis is not possibly valid for PSO system involving random variables. Secondly, the joint spectral radius, mainly calculated by the maximum eigenvalue of the product of all asymmetric random transfer matrices, is introduced to analyze and discuss convergence condition and convergence rate from numerical viewpoint with the aid of Monte Carlo method. Numerical results show that there is one major discrepancy between some preview convergence results and our corresponding results, helping us to deeply understand the tradeoff between exploration ability and exploitation ability as well as providing certain guideline for parameter selection.

Keywords: Particle swarm optimization, Convergence analysis, The joint spectral radius, Monte Carlo method

1. Introduction

Particle swarm optimization (PSO), firstly developed by Kennedy and Eberhart in 1995 [1][2], is essentially one typical stochastic global optimization method. Its original idea is mainly inspired by searching for the food by swarm animals, such as birds flocking, fish schooling, to name a few. In the original or improved PSO algorithm, two concepts including the velocity and the position are introduced to update each particle in the whole evolutionary process. By updating the velocity and the position of each particle in a simple way, its computational complexity, as well as the computational time, is relatively low compared with other evolutionary methods, leading to the concise and efficient code of PSO algorithm. Therefore, it has been widely and successfully applied to many practical systems in the realistic world and its main task is to search for optimal combinatorial parameters to minimize or maximize the related objective function of the scientific and engineering problems.

Roughly speaking, the most existing work on PSO algorithm may be mainly classified into the following categories: the essential and fundamental theory, multi-objective optimization, locating and tracking the target in the uncertain and dynamic environment, and its practical applications, etc. Firstly, it is of great significance to investigate the convergence condition [3], the trajectory of single particle [4], the topology of all particles [5], parameter selection [6], and the improved PSO algorithm

*To whom all correspondences should be addressed. Email: {DrJLBit,mathmhb}@163.com

[7][8][9][10][11][12], etc. And the corresponding results are helpful for understanding its essence and improving the efficiency of finding the suboptimal or global optimum. Secondly, to optimize multi-objective functions in parallel, PSO algorithm is mainly to search for the Pareto dominance and the Pareto-optimal front [13] in the solution space by maintaining the diversity of the swarm. Thirdly, the standard PSO algorithm is widely utilized in the uncertain and dynamical environment [14][15][16][17] to locate and track the targets [18][19][20][21][22]. Last but not least, it also has been successfully applied in the fields of electronic power system [23], neural network [24][25], electromagnetic [26], PID controller [27] and game theory [28], etc.

As mentioned above, convergence analysis of PSO algorithm plays a significant role on parameter selection, the trajectory of each particle and the tradeoff between exploration ability and exploitation ability, etc. The existing theoretical analysis is mainly conducted for the simplified PSO algorithm, regarding random variables as constant parameters, however, the standard PSO algorithm in fact involves random variables. Ozcan and Mohan [29][30] discuss the trajectory of one dimensional particle and multiple dimensional particles in the absence of random variables. Trelea [31] gives the convergence analysis from the perspective of spectral radius, and analyzes the convergence condition and convergence speed in the presence of the deterministic transfer matrix, helping us to understand the mechanism of PSO algorithm at one step. Clerc and Kennedy [3] also discuss the stability of the simplified PSO algorithm by a large number of complicated equations, and introduce the constriction factor method as well as the important parameters. Van den Bergh and Engelbrecht [4] present the fully detailed convergence analysis on the simplified PSO algorithm and discuss the trajectory of single particle under different parameters. Kadirkamanathan [32] regards PSO system as the nonlinear feedback system and discusses the stability of PSO algorithm including random variables by using Lyapunov function from the viewpoint of state space. Considering the expectation and variance of the position, Jiang and Luo [33] present the convergence condition and parameter selection under the fixed previous best positions of each particle and all particles. Poli [34] and Milan [35] consider the stability of the standard PSO algorithm under random variables and conclude convergence conditions from the perspective of mean and variance. Luis Fernandez and Esperanza Garcia [36] consider PSO algorithm as the stochastic damped mass-spring system, and analyze first and second order trajectories and the role of crucial parameters. More importantly, it also discusses the corresponding spectral radius, the mean, the covariance and the derivatives of those trajectories.

According to the aforementioned convergence analysis, several main contributions of this paper are highlighted as follows:

1. We highlight that the existing convergence conditions of the simplified PSO algorithm are not valid for the standard PSO algorithm, furthermore, one constructed counterexample demonstrates that the parameters, which strictly subject to the previous convergence conditions, finally give rise to divergence behavior of all particles.

2. The joint spectral radius, which is firstly introduced and defined in this paper to discuss the stability of the standard PSO algorithm, measures convergence rate of all particles and describes the mathematical tradeoff between exploration ability and exploitation ability. More importantly, it roughly subjects to normal distribution under large independent runs according to the corresponding numerical results.

3. Because of asymmetric transfer matrix involving random variables in PSO system, it is very challenging to calculate the final product matrix and the joint spectral radius, which determines the stability of the standard PSO algorithm and the convergence rate. To handle with this problem, Monte Carlo method is utilized to analyze convergence condition and discuss parameter selection.

The rest of this paper is organized as follows. Section 2 provides a brief description of the standard

PSO algorithm including the inertia weight method and the constriction factor method. To demonstrate that the existing convergence conditions are not valid for the standard PSO algorithm, convergence analysis of the simplified PSO algorithm, together with one counterexample, is briefly given in Section 3. In order to discuss convergence conditions and measure convergence rate, the joint spectral radius is firstly defined and calculated in Section 4. As for two typical PSO methods, the mean and variance spectral radii under different parameters are calculated with the aid of Monte Carlo method in Section 5. To demonstrate the discrepancy between the existing convergence rates and those of our results, the mean spectral radius is analyzed and calculated under several benchmark functions in Section 6. Section 7 summarizes several interesting remarks and concludes the future work.

2. The Random Particle Swarm Optimization

To clearly illustrate the random PSO algorithm, it is firstly necessary to introduce several notations and definitions in the standard PSO algorithm. In an n-dimensional solution space Ω, let the position and the velocity of ith particle be $X_i = (x_{i1}, x_{i2}, \cdots, x_{in})$ and $V_i = (v_{i1}, v_{i2}, \cdots, v_{in})$. Let $P_i = (p_{i1}, p_{i2}, \cdots, p_{in})$ and $G = (g_1, g_2, \cdots, g_n)$ be the personal best previous position of ith particle and all particles, respectively. Cognitive factor c_1 and social factor c_2, generally setting to 2.0, are non-negative constant real parameters. Notice that two random uniform distributed numbers including r_{1j} and r_{2j} are generated for the jth dimension of each particle at each step, therefore, the system of random PSO algorithm is the second order time-varying dynamical system involving random variables.

Several main steps of the standard PSO algorithm [37][38] can be briefly summarized as:

Step 1: Initialize the parameters including the particle's position X, the particle's velocity V, the number of all particles N, the dimension of solution space D, the range of inertia weight and acceleration coefficients, etc.

Step 2: Calculate the fitness $F(X_i(t))$ and modify the best previous position $P_i(t+1)$ of each particle by

$$P_i(t+1) = \begin{cases} X_i(t) & \text{if } F(X_i(t)) < F(P_i(t)) \\ P_i(t) & \text{if } F(X_i(t)) \geqslant F(P_i(t)) \end{cases} \quad (1)$$

where t denotes the number of current generation.

Step 3: Update the previous best position G of all particles by

$$G(t+1) = \arg\min\{F(P_1(t)), F(P_2(t)), \cdots, F(P_n(t))\}. \quad (2)$$

Step 4: Updating the velocity and the position of each particle, which can reflect the idea of searching for the food by the birds or the fish, is the core model of the standard PSO algorithm, typically including the inertia weight method and the constriction factor method.

Firstly, the velocity and the position of each particle in the inertia weight method are described by

$$V_{ij}(t+1) = \omega V_{ij}(t) + c_1 r_{1j}(P_i(t) - X_{ij}(t)) + c_2 r_{2j}(G(t) - X_{ij}(t)) \quad (3)$$

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \quad (4)$$

where the inertia weight ω denotes the momentum coefficient of previous velocity. Large ω in the early stage encourages the powerful exploration ability to locate the adjacent region of suboptimal or global optimum, while small ω in the latter stage promotes the powerful exploitation ability to precisely locate the optimum. In addition, i and j denote the index of the particle and the dimension in this particle, respectively. Moreover, the first part in (3) represents the momentum of previous velocity, while the second part so-called cognitive component represents the personal thinking of each particle, and the third part so-called social component represents the cooperation among all particles.

Secondly, the velocity and the position of each particle in the constriction factor method are defined as

$$V_{ij}(t+1) = \chi(V_{ij}(t) + c_1 r_{1j}(P_i(t) - X_{ij}(t)) + c_2 r_{2j}(G(t) - X_{ij}(t))) \quad (5)$$

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \quad (6)$$

where the constriction factor χ is generally set to $\frac{2}{|2-\varphi-\sqrt{\varphi^2-4\varphi}|}$ with $\varphi = c_1 + c_2$.

Actually, the inertia weight method and the constriction factor method have been considered as one mathematical model, only having the different inertia weight and acceleration coefficients.

Step 5: Go to Step 2 until the stopping condition is achieved.

According to core idea of PSO algorithm and its main steps, the programming code of PSO algorithm is easy to implement and its computational time is generally less than that of genetic algorithm. In the standard PSO algorithm, it is crucial to properly select the inertia weight ω and acceleration coefficients c_1 and c_2, since those parameters deeply control and determine behaviors of the particles and the tradeoff between exploration ability and exploitation ability, etc.

3. Convergence Analysis and One Counterexample

The purpose of this section is to introduce a brief description of the existing convergence analysis in the presence of the constant transfer matrix and calculate the corresponding spectral radius. In addition, we also provide one counterexample with respect to the existing convergence condition. Exactly speaking, those crucial parameters, which strictly yield to the previous convergence condition, finally lead to divergence behavior of the particles.

According to main steps of PSO algorithm, convergence analysis is chiefly related to (3) and (4). In order to simplify those equations, we assume

$$Q(t) = \frac{\phi_1(t)P(t) + \phi_2(t)G(t)}{\phi(t)} \tag{7}$$

where $\phi_1(t) = c_1 r_{1j}$, $\phi_2(t) = c_2 r_{2j}$ and $\phi(t) = \phi_1(t) + \phi_2(t)$.

According to (7), the equations (3) and (4) can be simplified as

$$v(t+1) = \omega(t)v(t) + \phi(t)y(t) \tag{8}$$

$$y(t+1) = -\omega(t)v(t) + (1-\phi(t))y(t) \tag{9}$$

where $y(t)$ is equal to $Q(t) - X(t)$ and $v(t)$ is for short $V(t)_{ij}$.

Therefore, (8) and (9) from the perspective of the matrix can be rewritten as

$$Y(t+1) = M(t)Y(t) \tag{10}$$

where

$$Y(t+1) = \begin{bmatrix} v(t+1) \\ y(t+1) \end{bmatrix}, M(t) = \begin{bmatrix} \omega(t) & \phi(t) \\ -\omega(t) & 1-\phi(t) \end{bmatrix}. \tag{11}$$

Furthermore, the personal best position $P(t)$ and the global best position $G(t)$ are generally assumed to be constant parameters. With respect to the simplified PSO algorithm, the time-varying transfer matrix $M(t)$ can be thought of as the constant transfer matrix M, therefore, the convergence condition is only determined by the eigenvalues of matrix M. Essentially, the simplified PSO algorithm mainly neglects random variables in the transfer matrix.

For completeness, the main task of next subsection is to briefly introduce the previous convergence analysis result for the simplified PSO algorithm.

3.1. *Previous Convergence Analysis of PSO Algorithm*

According to (10), the real and complex eigenvalues of the constant transfer matrix M are

$$\lambda_1 = \frac{1+\omega-\phi-\sqrt{\Delta}}{2} \tag{12}$$

$$\lambda_2 = \frac{1+\omega-\phi+\sqrt{\Delta}}{2} \tag{13}$$

where the discriminant Δ is equal to $(\phi-\omega-1)^2 - 4\omega$.

When the parameters including inertia weight and acceleration coefficients are chosen such that $\triangle \geqslant 0$ and $|\lambda_{1,2}| < 1$, the constant transfer matrix M has two real eigenvalues whose absolute magnitude is strictly less than 1, which means that the particles have convergence behavior or probably converge into one attractor or the suboptimal optimum. The corresponding conditions can be concluded by

$$\begin{cases} \omega < \phi + 1 - 2\sqrt{\phi} \\ \omega > \frac{\phi}{2} - 1. \end{cases} \tag{14}$$

Hence, the area of the parameters including ω and ϕ can be plotted as the area **F1** in Fig. 1.

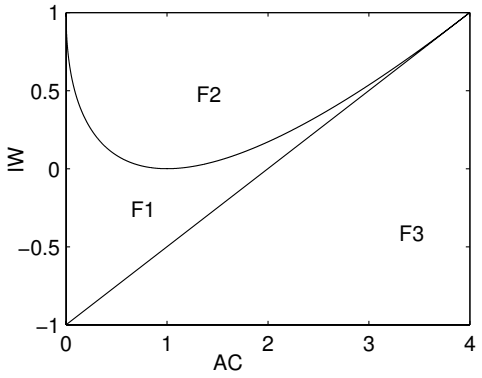


Figure 1: The previous convergence region of the simplified PSO algorithm, where *IW* and *AC* denote inertia weight ω and ϕ, respectively.

When the parameters are chosen such that $\triangle < 0$ and $|\lambda_{1,2}| < 1$, matrix M has two complex eigenvalues whose absolute magnitude is also strictly less than 1, and the particles also have the oscillation convergence behavior. The corresponding conditions can be expressed by

$$\phi + 1 - 2\sqrt{\phi} < \omega < 1. \tag{15}$$

Therefore, in this case, the area of the parameters including ω and ϕ can be also shown as the area **F2** in Fig. 1.

According to (14) and (15), the previous convergence conditions of the simplified PSO algorithm are

$$\begin{cases} \omega < 1 \\ \omega > \frac{\phi}{2} - 1 \\ \phi > 0. \end{cases} \tag{16}$$

As depicted in Fig. 1, the previous convergence region of PSO algorithm is also plotted in [4][31], where the parameters in **F1** and **F2** can lead to real and complex eigenvalues subjecting to $|\lambda_{1,2}| < 1$, respectively. Furthermore, the parameters in **F3** can lead to real eigenvalues satisfying $|\lambda_1| > 1, |\lambda_2| < 1$ or $|\lambda_1| < 1, |\lambda_2| > 1$. The areas outside the region $0 \leqslant \omega \leqslant 1, 0 \leqslant \phi \leqslant 4$ are not in our consideration since PSO algorithm does not generally set those parameters outside the rectangle region.

3.2. Previous Spectral Radius Analysis of PSO Algorithm

Definition 1 *Spectral radius of matrix M is defined as the largest modulus of the eigenvalues including λ_1 and λ_2, that is,*

$$\rho(M) = \max\{|\lambda| : \exists\, \boldsymbol{x} \neq \boldsymbol{0}, M\boldsymbol{x} = \lambda \boldsymbol{x}\} \tag{17}$$

where ρ denotes spectral radius of the transfer constant matrix M. In addition, λ and $\boldsymbol{x}$ denote the eigenvalue and its eigenvector, respectively.

As for the simplified PSO algorithm, if the parameters subject to the conditions of $0.5\phi - 1 < \omega < \phi + 1 - 2\sqrt{\phi}$ and $\omega < 1$, spectral radius $\rho(M)$ in the convergence region **F1** equals to λ_2.

If the parameters yield to the conditions of $\omega > \phi + 1 - 2\sqrt{\phi}$ and $\omega < 1$, both eigenvalues are two conjugate non-real complex numbers. According to (12) and (13), those spectral radii can be rewritten as

$$\lambda_{1,2} = \frac{1 + \omega - \phi \pm \sqrt{4\omega - (\phi - \omega - 1)^2 i}}{2} \tag{18}$$

therefore

$$|\lambda_1|^2 = |\lambda_2|^2 = \frac{(1 + \omega - \phi)^2 + 4\omega - (\phi - \omega - 1)^2}{4} = \omega \tag{19}$$

then

$$|\lambda_1| = |\lambda_2| = \sqrt{\omega} \tag{20}$$

which implies that

$$\rho(M) = \sqrt{w}. \tag{21}$$

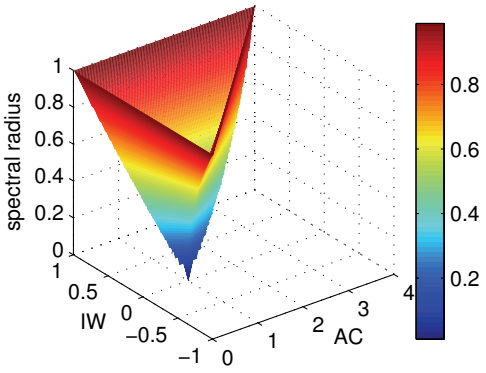


Figure 2: Spectral radius of the simplified PSO algorithm.

As mentioned above, we can obviously conclude that the spectral radius $\rho(M)$ equals to λ_2 when the eigenvalues are real numbers, while the spectral radius $\rho(M)$ is equal to $\sqrt{\omega}$ when λ_1 and λ_2 are complex numbers. From the viewpoint of the graph, spectral radii in the three dimensional space are plotted in Fig. 2.

According to the previous convergence result, there is an interesting phenomenon in the case of convergence rate. Specifically speaking, convergence rate of the particles is generally determined by inertia weight ω and acceleration coefficients c_1 and c_2 in the standard PSO algorithm. However, according to (20) and (21), convergence rate of the particles is only dependent on inertia weight, since spectral radius of matrix M finally equals to $\sqrt{\omega}$ under the condition of $\Delta < 0$, therefore, the above-mentioned analysis should be deeply analyzed and discussed in the next subsections.

3.3. *One Counterexample of Previous Convergence Result*

The main task of this subsection is to introduce one counterexample which demonstrates that the previous convergence conditions do not coincide with the standard PSO method according to simulation numerical results. Specifically speaking, those crucial parameters, yielding to the previous convergence conditions (16), can give rise to divergence behavior of the particles.

According to convergence conditions of the simplified PSO algorithm, the parameters, which can be selected to ω=0.91, c_1=1.9 and c_2=1.9, make the particles convergent according to (16). However, those parameters finally give rise to divergence behavior in the standard PSO algorithm since the elements of the accumulated matrix in PSO system, denoting the product of all previous random matrices, become larger and larger in the evolutionary process, such as Fig. 3.

Remark 1 *According to the behaviors of the particles in this counterexample, we may conclude that the previous convergence conditions possibly do not fit for the standard PSO algorithm, whose transfer matrix is time-varying and consists of random variables.*

4. The Joint Spectral Radius

In order to deeply analyze the essence of PSO algorithm, the joint spectral radius is firstly defined and introduced to analyze convergence conditions and measure convergence rate of the standard PSO algorithm. Since the transfer matrix $M(t)$ is one time-varying random matrix, convergence conditions, corresponding to its convergence rate, are only determined by the product of matrices $M(t)$ ($1 \leqslant t \leqslant \infty$), therefore, the equation (10) can be further explicitly rewritten as

$$Y(t+1) = M(t)M(t-1)\cdots M(1)M(0)Y(0). \quad (22)$$

Without loss of generality, one dimension of all particles is merely considered to be analyzed, and

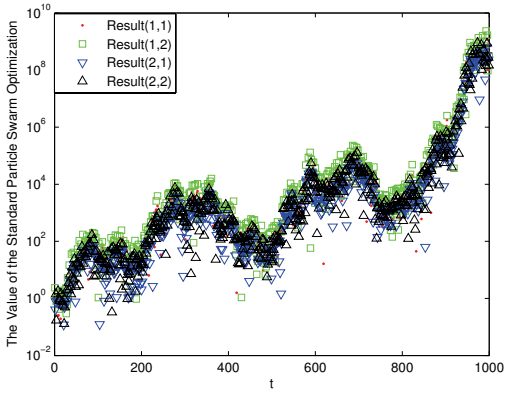


Figure 3: The elements in the accumulated matrix in the whole evolutionary process, where the constant inertia weight is set to 0.91 and acceleration coefficients are selected to be 1.90. Additionally, $Result(i, j)$ denotes the (i, j) element in the accumulated matrix.

the convergence condition of each particle should be concluded as follows

$$M(t)M(t-1)\cdots M(1)M(0) \to 0 \quad \text{as} \quad t \to \infty. \tag{23}$$

If the product of (23) approaches to 0, the particle can converge into the adjacent region of the local or global optimum. From the perspective of spectral radius, the equation (23) can be rewritten by

$$\rho \left(\begin{bmatrix} \omega(t) & \phi(t) \\ -\omega(t) & 1-\phi(t) \end{bmatrix} \begin{bmatrix} \omega(t-1) & \phi(t-1) \\ -\omega(t-1) & 1-\phi(t-1) \end{bmatrix} \cdots \begin{bmatrix} \omega(1) & \phi(1) \\ -\omega(1) & 1-\phi(1) \end{bmatrix} \begin{bmatrix} \omega(0) & \phi(0) \\ -\omega(0) & 1-\phi(0) \end{bmatrix} \right) \longrightarrow 0 \tag{24}$$

as $t \to \infty$.

However, it is very hard to calculate the product of matrices $M(t)$ $(1 \leqslant t \leqslant \infty)$ and its eigenvalues from the theoretical viewpoint. Firstly, transfer matrices $M(t)$ $(t \geqslant 1)$ are asymmetric matrices. Secondly, every transfer matrix $M(t)$ is composed of random variables r_1 and r_2 subjecting to uniform distribution. In order to overcome those aforementioned problem, Monte Carlo method is adopted to compute two eigenvalues of the product of all transfer matrices involving random matrices.

To measure convergence rate and analyze convergence condition of the standard PSO algorithm, the joint spectral radius is introduced to conclude some interesting and important results with the aid of Monte Carlo method.

Definition 2 *The joint spectral radius ϱ is defined as the geometric spectral radius of the product of the whole random transfer matrices, and it can be mathematically expressed by*

$$\varrho = \rho^{\frac{1}{t}}(M(t)M(t-1)\cdots M(1)M(0)) \tag{25}$$

where t denotes the number of generations in the whole search process.

According to the results of N independent times, the mean spectral radius, which possibly denotes the convergence rate of the standard PSO system or the measurement of the tradeoff between exploration ability and exploitation ability, can be calculated by

$$Mean(\varrho) = \frac{1}{N}\sum_{i=1}^{N}\varrho_i. \tag{26}$$

The variance spectral radius, which denotes the measurement of the uncertainty of the joint spectral radius, can be also expressed by

$$Var(\varrho) = \frac{1}{N-1}\sum_{i=1}^{N}(\varrho_i - Mean(\varrho))^2. \tag{27}$$

Remark 2 *Naturally, the random factor in PSO algorithm plays a great role on convergence analysis and convergence conditions. The previous work also considers random variables r_1 and r_2, which*

may not reflect the realistic evolutionary process. For example, the random transfer matrix in [34] is changed into the constant matrix by expectation operator from the perspective of mean and variance, calculating and concluding convergence conditions under random factors. However, as for the standard PSO algorithm, it firstly needs to compute the product of all random transfer matrices, and then calculates the distribution of their eigenvalues, and finally gives the mean and variance of their eigenvalues. In addition, the operator of calculating spectral radius is essentially one nonlinear operator, that is to say,

$$E(\rho^{\frac{1}{t}}(M(t)M(t-1)M(t-2)\cdots M(1)M(0))) \neq \rho^{\frac{1}{t}}(E(M(t))E(M(t-1))\cdots E(M(1))E(M(0))) \tag{28}$$

5. The Joint Spectral Radius of Two Typical PSO Methods

The main purpose of this section, with the aid of Monte Carlo method, is to analyze and discuss the relationship between the joint spectral radius and crucial parameters including inertia weight and acceleration coefficients. Exactly speaking, with respect to the standard PSO algorithm including the constriction factor method and the inertia weight method, this section is to mainly deeply discuss each relationship among ω, c_1, c_2 and ϱ, $Mean(\varrho)$, $Var(\varrho)$.

5.1. The Constriction Factor Method

Regarding the constriction factor method, the constriction factor χ is generally set to 0.729 and acceleration coefficients c_1 and c_2 are set to 2.0 according to [3], however, the time-varying transfer matrix is composed of the random variables.

5.1.1. Calculation of ϱ in the Constriction Factor Method

One simple example is to introduce the method of calculating ϱ to illustrate the computation process. Taking one whole evolutionary process for example, the transfer matrix in the constriction factor method can be expressed by

$$M(t) = \begin{bmatrix} 0.729 & \phi(t) \\ -0.729 & 1-\phi(t) \end{bmatrix}. \tag{29}$$

Suppose that the number of generations is set to 1000 in the whole evolutionary process, the product M_p of all transfer matrices involving random variables can be expressed as

$$M_p = M(1000)M(999)\cdots M(1) \tag{30}$$

that is to say, the final product M_p is

$$M_p = \begin{bmatrix} 0.729 & 1.417 \\ -0.729 & -0.417 \end{bmatrix}\cdots\begin{bmatrix} 0.729 & 1.097 \\ -0.729 & -0.097 \end{bmatrix} = \begin{bmatrix} 0.980 & -0.592 \\ -0.162 & 0.098 \end{bmatrix}\times 10^{-39}. \tag{31}$$

Two eigenvalues of the product M_p of all random transfer matrices are

$$\lambda_1 = 0, \ \ \lambda_2 = 1.082\times 10^{-39}. \tag{32}$$

According to (25), the joint spectral radius ϱ is

$$\varrho = 0.914. \tag{33}$$

5.1.2. Distribution of the Joint Spectral Radius

Due to the nature of time-varying random matrix of $M(t)$, it is very hard to give explicit mathematical results for the joint spectral radius, it is of importance to analyze and discuss the distribution of the joint spectral radius via Monte Carlo method. As depicted in Fig. 4, the comparison on the mean spectral radius and the variance spectral radius under different numbers of generations are plotted to analyze the interesting results. It can be concluded from Fig. 4 that all mean spectral radii, which similarly subject to the normal distribution, are nearly equal to 0.786 corresponding to different variance. It highlights that the mean spectral radius is mainly determined by the constriction factor χ, c_1 and c_2, while the variance spectral radius is mainly dependent on the number of generations. Naturally speaking, large and small number of generations can give

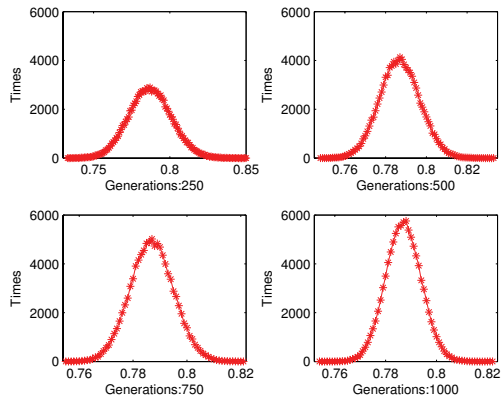


Figure 4: The distribution of the joint spectral radius under different numbers of generations.

rise to small and large variance of spectral radius, respectively.

In order to further investigate the relationship between the number of generations and the joint spectral radius, the mean and variance spectral radii are compared and summarized according to Fig. 5(a) and Fig. 5(b), respectively. When the number of generations changes from 100 to 1000, the mean spectral radius is nearly equal to 0.783 and the variance spectral radius quickly decreases. As can be seen from Fig. 5(a), small number of generations results in the unstable mean spectral radius while large number of generations leads to the relative stable value. In order to measure the instability under different number of generations, the corresponding variance spectral radius is plotted in Fig. 5(b).

5.1.3. Relationship of Joint Spectral Radius and Parameters when $c_1 + c_2 = 4$

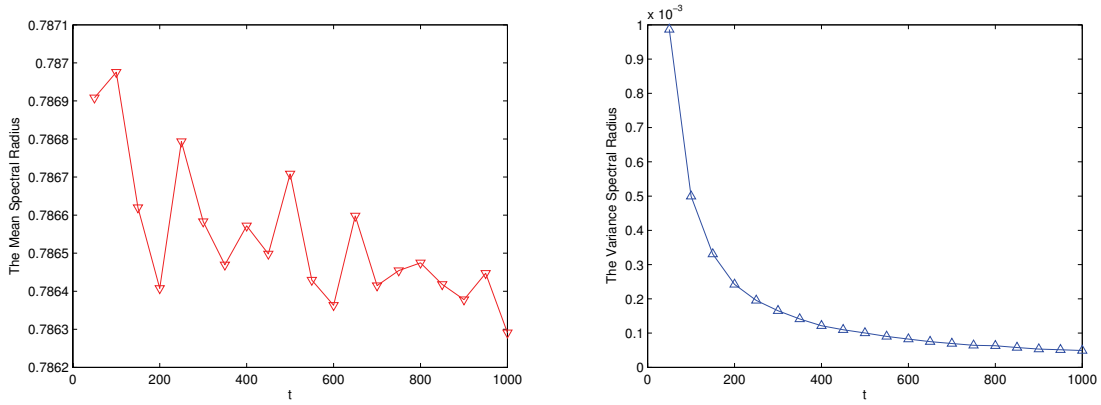
The joint spectral radius not only mainly depends on the number of generations, but also is greatly determined by acceleration coefficients including c_1 and c_2. In order to study strong relationship between spectral radius and acceleration coefficients under the condition of $c_1 + c_2 = 4.0$, the mean spectral radii are listed and compared in Fig. 6(a). Moreover, the variance spectral radius denoting to the uncertainty of the joint spectral radius is shown in Fig. 6(b).

On the basis of Fig. 6(a) and Fig. 6(b), several interesting results and remarks, which are helpful for parameter selection and understanding the essence of the standard PSO algorithm, can be concluded as follows:

1. According to the previous convergence analysis to the simplified PSO algorithm, spectral radius should be only dependent on constriction factor χ when the eigenvalues are complex numbers, in other words, acceleration coefficients c_1 and c_2 should not play a role on spectral radius when the discriminant Δ is strictly smaller than 0. However, according to Fig. 6(a), acceleration coefficients indeed have a great effect on the joint spectral radius. For example, the mean spectral radius under the parameters (χ=0.1, c_1=2, c_2=2) is larger than that of the parameters (χ=0.1, c_1 = 1, c_2=3).

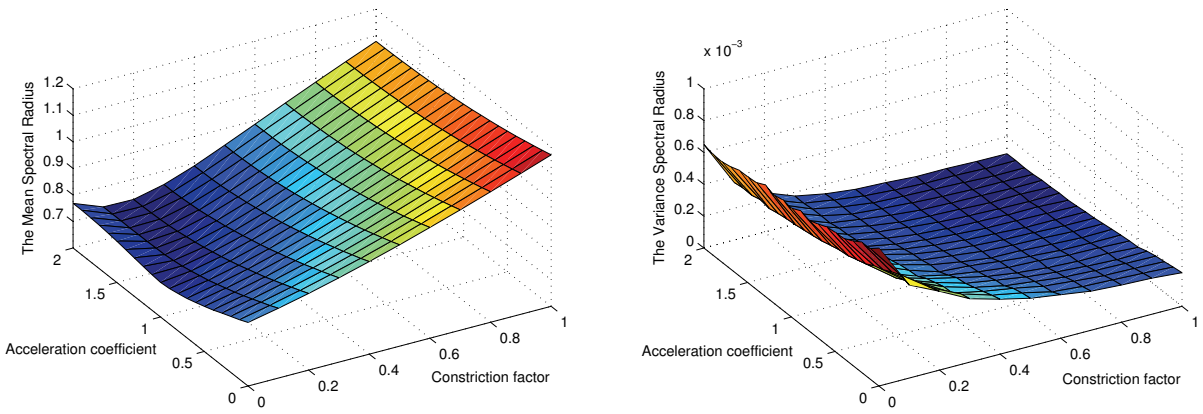
2. In terms of Fig. 6(a), those results can explain why the particles have the low convergence speed at the end of search stage. That is to say, all particles do not easily converge into the suboptimal or global solution under small constriction factor χ. Small χ in the latter search stage can result in large ratio of divergence behavior according to (16) and large spectral radius, such as χ=0.01, $c_1 = c_2 = 2$.

3. In the realistic applications, it needs to quickly converge into the suboptimal solution. With respect to the constriction factor method, it



(a) $Mean(\varrho)$ (b) $Var(\varrho)$

Figure 5: The joint spectral radius and the number of generations from 100 to 1000.



(a) $Mean(\varrho)$ (b) $Var(\varrho)$

Figure 6: The joint spectral radius and other parameters ($c_1 + c_2 = 4.0$) in the constriction factor method.

can be obviously seen from Fig. 6(a) that the parameters (χ=0.2, c_1=2, c_2=2) can lead to the smallest spectral radius and achieve the quickest convergence rate of all particles.

5.2. The Inertia Weight Method

The inertia weight method has been successfully and widely utilized in the realistic optimization applications and can be mathematically described by (3) and (4), however, the essence of this method is hardly studied because of inertia weight $\omega(t)$, random variables r_1 and r_2 in transfer matrices M(t) ($t \geqslant 1$). The purpose of this subsection is to discuss convergence rate of the inertia weight method, convergence conditions and the relationship between the joint spectral radius and crucial parameters, etc. In the general case, the inertia weight ω in the whole search process often linearly decreases from 0.9 to 0.4 and acceleration coefficients c_1 and c_2 are set to 2.0, that is to say, the transfer matrix M(t) ($t \geqslant 1$) consists of time-varying parameters $\omega(t)$ and $\phi(t)$.

Firstly, it is also to introduce and calculate the detailed process of calculating ϱ in the inertia weight method, and the transfer matrix can be expressed by

$$M(t) = \begin{bmatrix} \omega(t) & \phi(t) \\ -\omega(t) & 1-\phi(t) \end{bmatrix} \tag{34}$$

where $\omega(t)$ generally decreases from 0.9 to 0.4 and $\phi(t)$ is one random variable in the range of [0, 4]. Therefore, the final product M_p, according to (30), can be calculated by

$$\begin{aligned} M_p &= \begin{bmatrix} 0.900 & 1.304 \\ -0.900 & -0.304 \end{bmatrix} \cdots \begin{bmatrix} 0.400 & 1.083 \\ -0.400 & -0.083 \end{bmatrix} \\ &= \begin{bmatrix} -0.310 & -0.036 \\ 0.126 & 0.015 \end{bmatrix} \times 10^{-51}. \end{aligned} \tag{35}$$

The corresponding eigenvalues of final product matrix are

$$\lambda_1 = -0.295 \times 10^{-51}, \quad \lambda_2 = -3.709 \times 10^{-68} \tag{36}$$

and its joint spectral radius in the inertia weight method is

$$\varrho = 0.8881. \tag{37}$$

Secondly, the joint spectral radius not only measures the convergence rate of the particles, but also controls divergence behavior or convergence behavior of the particles. Due to random variables in transfer matrix, it is necessary to analyze the distribution of the joint spectral radius according to the law of large numbers, and its corresponding distribution is depicted in Fig. 7. It can be clearly seen from Fig. 7 that the joint spectral radii are similarly yielded to the normal distribution. The mean spectral radius is approximately equal to 0.9099 when the inertia weight ω linearly changes from 0.9 to 0.4 and acceleration coefficients including c_1 and c_2 are usually set to 2.0.

Remark 3 *According to Fig. 4 and Fig. 7, the joint spectral radius in the standard PSO algorithm is likely yielded to the normal distribution, whose center point, together with the corresponding variance, is closely related to inertia weight ω, c_1 and c_2.*

In order to systematically and comprehensively analyze the relationship among crucial parameters, $Mean(\varrho)$ and $Var(\varrho)$, the mean spectral radii are listed under different ranges of inertia weight, where the inertia weight ω linearly decreases from the initial weight ω_s to the final weight ω_e. According to (14), the initial and final inertia weights also play a great role on the center of the joint spectral radius.

From the perspective of the graph, the mean spectral radii, which can denote convergence rate of the whole swarm, are depicted in Fig. 8. According to the detailed spectral radius in Fig. 8, several results can be concluded as follows:

1. On the basis of the mean spectral radii, the border condition between convergence behavior and divergence behavior is closely related to the ranges of [1.0 0.7], [0.9 0.8], [0.8 0.9] and [0.7 1.0] in Fig. 8. Roughly speaking, if the initial weight ω_s and the final weight ω_e do not yield to (38), the particles may have divergence behavior in the evolutionary process. If ω_s and ω_e subject to (38), the particles may have convergence behavior in the final evolutionary process. The border condition between both behaviors can be roughly described as

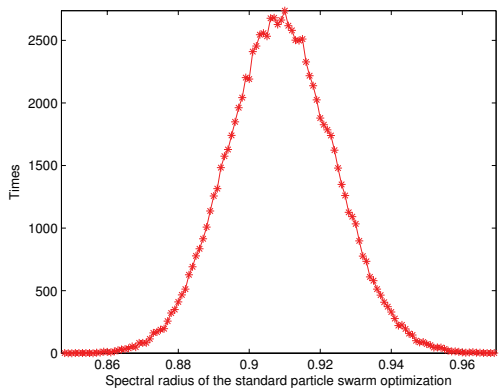


Figure 7: The distribution of the joint spectral radius in the inertia weight method, where the number of independent runs is 100000 and the number of generations in each run is 500 times.

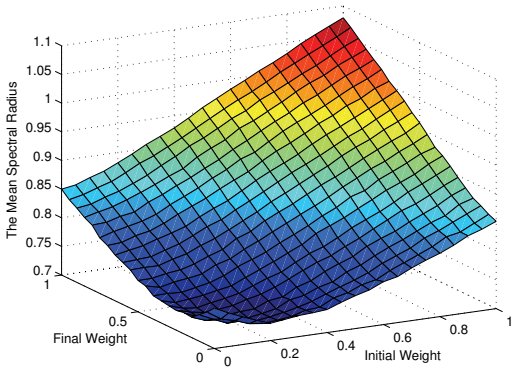


Figure 8: The relationship between $Mean(\varrho)$ and the range from ω_s to ω_e, where c_1, setting to the identical value c_2, is equal to 2.

$$\begin{cases} \omega_s + \omega_e < 1.7 \\ \omega_s > 0 \\ \omega_e > 0. \end{cases} \tag{38}$$

2. It can be also summarized from Fig. 8 that the parameters, which fulfill the condition of (14) and the previous convergence conditions, finally lead to divergence behavior, therefore, there is also one of contradictive examples between the previous convergence conditions and numerical results.

3. According to Fig. 8, the initial weight ω_s and the final weight ω_e are set to very small values, resulting in relatively large spectral radius and low convergence rate. It can be clearly concluded that this phenomenon also explains why the particles do not easily converge into the suboptimal or global optimum in the latter search process.

In order to provide the guideline for parameter setting and the tradeoff between exploration ability and exploitation ability at the initialization stage, the joint spectral radius is closely related to the initial weight ω_s and the final weight ω_e in the presence of $c_1 = c_2$, and the corresponding mathematical model can be described by

$$\begin{aligned} \varrho = & -0.1933\omega_s^3 - 0.192\omega_e^3 - 0.1763\omega_s^2\omega_e - 0.1774\omega_s\omega_e^2 \\ & + 0.4683\omega_s^2 + 0.4665\omega_e^2 + 0.4701\omega_s\omega_e - 0.1751\omega_s \\ & - 0.1749\omega_e + 0.7487. \end{aligned} \tag{39}$$

5.3. Further Discussions

The main goal of this subsection is to discuss two general cases to show the importance of crucial control parameters. In the first case, acceleration coefficients ($c_1 = c_2$) change from 0.0 to 2.0. In the second case, the sum of c_1 and c_2 is equal to 4.0. Additionally, the number of generations in per run is set to 1000 times and the number of runs is set to 5000, respectively. In two cases, we are mainly concerned with the relationship between the mean and variance spectral radii and acceleration coefficients.

Firstly, the inertia weight ω linearly decreases from 0.9 to 0.4 and acceleration coefficient c_1 equaling to c_2 is from 0.0 to 2.0. The mean spectral radius is depicted in Fig. 9.

According to Fig. 9, if the acceleration coefficients (c_1=c_2) increase from 0.0 to 2.0, we can obtain that

1. When acceleration coefficient c_1 setting to c_2 is equal to 0, the mean spectral radius is 1, leading to divergence behavior of all particles.

2. When c_1 is equal to 0.2, the mean spectral radius gets the smallest value which is near to 0.8, leading to the quickest convergence rate. And acceleration coefficients including c_1 and c_2 should not be selected to small value, such as 0.0~0.2.

Secondly, acceleration coefficient c_1 in another typical case increases from 0.0 to 2.0 yielding to the condition of c_1+c_2=4.0 and inertia weight ω linearly decreases from 0.9 to 0.4. The mean spectral radii is shown in Fig. 10.

As can be seen from Fig. 10, we may conclude that

1. Small c_1 and large c_2 probably lead to a large ratio of divergence behavior and finally result in the divergence behavior. Essentially, acceleration coefficients play a great role on the attractor of PSO system.

2. Under the condition of c_1+c_2=4, acceleration coefficient c_1, which equals to 2.0, leads to the smallest spectral radius and the largest convergence rate.

3. According to the variance spectral radius, the divergence ratio under small acceleration coefficient is larger than that of large acceleration coefficient, therefore, small acceleration coefficient gives rise to large variance probably due to large divergence ratio.

6. Simulation

In order to show and demonstrate the major difference between previous results and our results, sev-

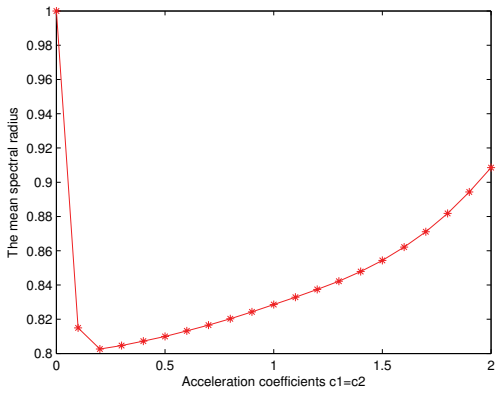


Figure 9: $Mean(\varrho)$ under c_1=c_2.

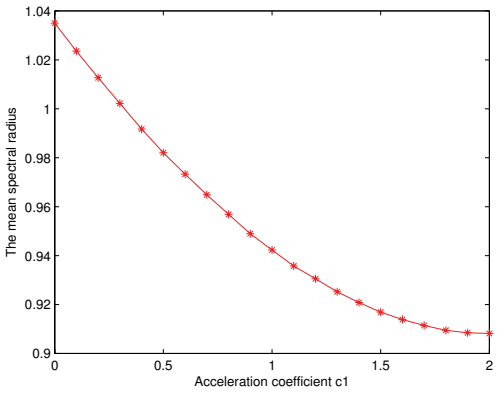


Figure 10: $Mean(\varrho)$ under c_1+c_2 = 4.0

eral typical benchmark functions, i.e. Sphere function, Griewank function, Rastrigin function Rosenbrock function, Schaffer's f6 function and two typical multi-objectives functions, which have different typical fitness landscapes and may serve as the examples of a wide range of optimization problems, are testified in the simulations for the purpose of comparing convergence rate of all particles and convergence conditions of the standard PSO algorithm. These results may give some intuitive hints to understand the role of those parameters and the convergence behavior.

6.1. Benchmark Functions

Sphere function is the simplest objective function in the benchmark optimization functions. It can represent the simple objective function with no local optima and it can be explicitly expressed by

$$f_1(\mathbf{X}) = \sum_{i=1}^{n} x_i^2 \tag{40}$$

where $|\mathbf{V}_{ij}| \leqslant 100$, $|\mathbf{X}_{ij}| \leqslant 100$. The particle's initialization range is $(50,100)^n$. The global optimum point of Sphere function is $(0,0,\cdots,0)^n$.

Griewank function is one of the most complicated optimization functions because of many local optima around the global optimum. It can represent the practical complex system and its formula is

$$f_2(\mathbf{X}) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}}) + 1 \tag{41}$$

where $|\mathbf{V}_{ij}| \leqslant 600$, $|\mathbf{X}_{ij}| \leqslant 600$. The particle's initialization range is $(300,600)^n$. The global optimum of Griewank function is $(0,0,\cdots,0)^n$.

Rastrigin function is a challenging optimization function. It is very hard to search the global optimum because of many local optima in the solution space. The formula of Rastrigin function is

$$f_3(\mathbf{X}) = \sum_{i=1}^{n}(x_i - 10\cos(2\pi x_i) + 10) \tag{42}$$

where $|\mathbf{V}_{ij}| \leqslant 5.12$, $|\mathbf{X}_{ij}| \leqslant 5.12$. The particle's initialization range is set to $(2.56,5.12)^n$. The global optimum of Rastrigin function is $(0,0,\cdots,0)^n$.

Rosenbrock function is also a challenging optimization function. Through it only has few local optima, the solution space around the global optimum is very flat, therefore, it is hard to converge around the global optimum. The formula of Rosenbrock function is

$$f_4(\mathbf{X}) = \sum_{i=1}^{n-1}(100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2). \tag{43}$$

where $|\mathbf{V}_{ij}| \leqslant 30$, $|\mathbf{X}_{ij}| \leqslant 30$. The initialization range of the particles is $(15,30)^n$. The global optimum of Rosenbrock function is $(1,1,\cdots,1)^n$.

Schaffer's f6 function has many local optima around the global optimum in the whole solution space. Once getting into the local optimum, the particles hardly get out of local optima. The formula of Schaffer's f6 function is

$$f_5(x,y) = 0.5 + \frac{(\sin\sqrt{x^2+y^2}) - 0.5}{(1+0.001(x^2+y^2))^2} \tag{44}$$

where $|x| \leqslant 100$, $|y| \leqslant 100$. The initialization range of the particles is (50,100). The global optimum of Schaffer's f6 function is (0,0).

In addition, one multi-objectives function by Deb ($0.1 \leqslant x_1, x_2 \leqslant 1$) can be mathematically described by

$$\min f_1(x_1, x_2) = x_1 \tag{45}$$

$$\min f_2(x_1, x_2) = \frac{g(x_2)}{x_1} \tag{46}$$

where the related $g(x_2)$ function is

$$g(x_2) = 2.0 - \exp\{-(\frac{x_2 - 0.2}{0.004})^2\} - 0.8\exp\{-(\frac{x_2 - 0.6}{0.4})^2\}. \tag{47}$$

Another two objectives function can be mathematically described by

$$\min f_1(x) = \begin{cases} -x & x \leqslant 1 \\ -2+x & 1 < x \leqslant 3 \\ 4-x & 3 < x \leqslant 4 \\ -4+x & x > 4 \end{cases} \tag{48}$$

$$\min f_2(x) = (x-5)^2. \tag{49}$$

6.2. Parameter Setting

The population number of all particles is set to 20 and the maximum number of generations in per run is set to 1000, respectively. Besides, the dimension of the whole solution space is 20. When the absolute velocity of each particle is strictly smaller than 0.001, it is reasonable to regard that the particles converge into the suboptimal or global optimum, taking into convergence state account. In addition, when the velocity or the position of one particle goes beyond the specified range, its current value should be reset to the random value in the specified range.

Finally, the number of generations in convergence process is compared in the presence of different crucial parameters, yet under same initial conditions including the same initial velocity, the same initial position, the same personal previous best position and the same previous best position of all particles, etc.

6.3. Discrepancy between Our Results and Existing Results

Convergence rate of the standard PSO algorithm determines the computational time and the efficiency in the real practical applications. To show the difference between the existing results and our results, the inertia weight is set to the constant parameter in the whole evolutionary process and random variables including r_1 and r_2 are considered to be time-varying variables. However, the inertia weight and random variables are regarded as constant parameters in the existing results.

There exist a number of results on the precision of particle swarm optimization reported in the literature, however, to the best knowledge of the authors, only a few of them are conducted before from the perspective of convergence rate, therefore, the following discussion mainly highlights the convergence rate and the corresponding comparison between previous works and our results. The common view is that the large convergence rate leads to the powerful exploration ability and the small convergence rate makes the powerful exploitation ability. If the convergence rate is set to too small value, such as 0.0, $\cdots$, 0.4, the particles easily converge into the local optimum. If the convergence rate is set to too large value, such as 0.9, $\cdots$, 0.99, the particles need large number of epochs to converge the suboptimal or global optimum.

In order to demonstrate the effectiveness of the above-mentioned results for single objective optimization problem, convergence rate under different parameters as well as the discrepancy between our results and the existing results is compared in Fig. 11(a), Fig. 11(b), Fig. 12(a), Fig. 12(b), Fig. 13(a), Fig. 13(b), Fig. 14(a), Fig. 14(b), Fig. 15(a) and Fig. 15(b), respectively. Taking Fig. 11(a) for example, each datum is calculated according to 0.001×

NG, where NG denotes the number of generations to achieve the convergence status. Furthermore, convergence rate is also dependent on the joint spectral radius, and large ϱ can result in the large convergence rate, and vice versa. However, the convergence rate is not in accordance with the existing result on convergence condition, specifically speaking, some parameters in the existing convergence condition finally give rise to divergence behavior, such as ω=0.95, $c_1 = c_2 = 1.90$, in addition, the convergence rate has the major difference of convergence rate when the spectral radius is the complex number. In order to show the detailed difference between our results involving random variables and the existing results regarding random variables as constant parameters, the corresponding comparison under the constant inertia weight is depicted in Fig. 11(b).

According to Fig. 11(a) - Fig. 15(b), we may conclude the following remarks.

1. The joint spectral radius is not greatly influenced by benchmark functions, but it closely relates to crucial parameters, which are composed of the inertia weight ω, acceleration coefficients c_1 and c_2. Convergence rate of every function is similar to each other from Fig. 11(a), Fig. 12(a), Fig. 13(a), Fig. 14(a) and Fig. 15(a).

2. Compared with the analysis of convergence rate in [31], there is the major discrepancy between the existing convergence rate and our results according to Fig. 11(b), Fig. 12(b), Fig. 13(b), Fig. 14(b) and Fig. 15(b). In the previous convergence analysis, convergence rate in Fig. 2(b) [31] is merely determined by inertia weight and it is not related to acceleration coefficients when its spectral radius is complex number. However, numerical results illustrate that two typical methods including inertia weight and acceleration coefficients have a significant effect on convergence rate.

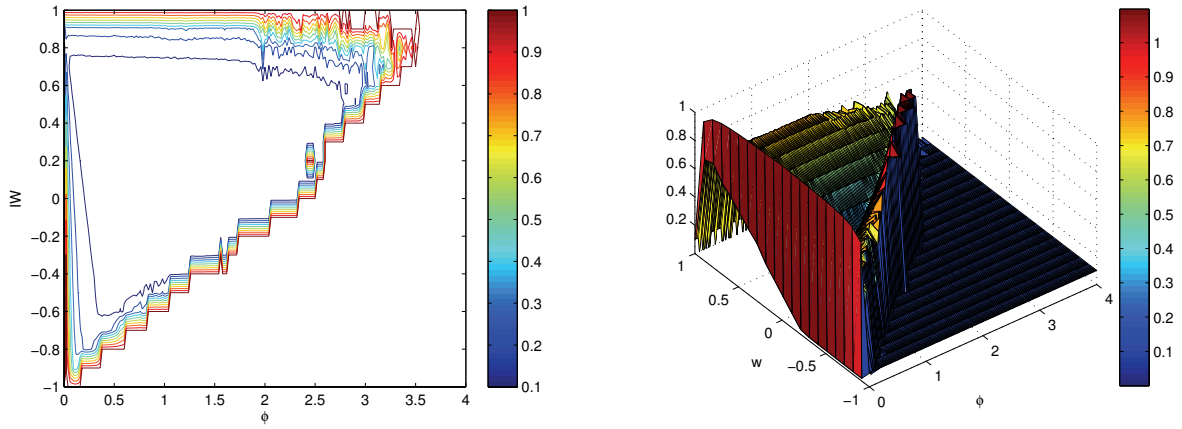
3. In addition, we may observe that some parameters, which subject to the previous convergence conditions according to (16), also lead to divergence behavior of the particles in the case of the above-mentioned benchmark functions. Notice that the shapes of convergence areas in Fig. 11(a), Fig. 12(a), Fig. 13(a), Fig. 14(a) and Fig. 15(a) do not coincide with those of Fig. 1, i.e. the areas **F1** and **F2**.

Additionally, the new convergence analysis of random PSO algorithm can be also applied for multi-objectives functions. In the PSO system, the convergence process can be divided into two substages. First substage, by utilizing the new convergence analysis, is to ensure that each particle can converge into its attractor according to the updating velocity and position equations, while second substage is to make that those attractors converge into one point or several points. As to the single objective function, the attractors of all particles converge into one point or the global optimum. Additionally, the corresponding attractors in the multi-objectives functions converge into several points, which are the crucial points in the Pareto front. In other words, the difference by using PSO algorithm between single objective optimization problem and multiple objectives optimization problem is the topology of all particles in random PSO system. The new convergence analysis is only applied in the first substage from the initial point to its attractor.

In the case of two testing multiple objectives benchmark functions, all particles with different topology in PSO algorithm can search for several discrete points, consisting of the Pareto front. The PSO algorithm with one connecting particular topology can find the Pareto front of Deb function in Fig. 16(a) and the last testing multi-objectives function in Fig. 16(b).

Compared with the existing PSO convergence analysis methods, such as the Poli's and Luis Fernandez-Martinez's convergence results, several remarks on the difference between their results and our convergence results can be concluded in the following.

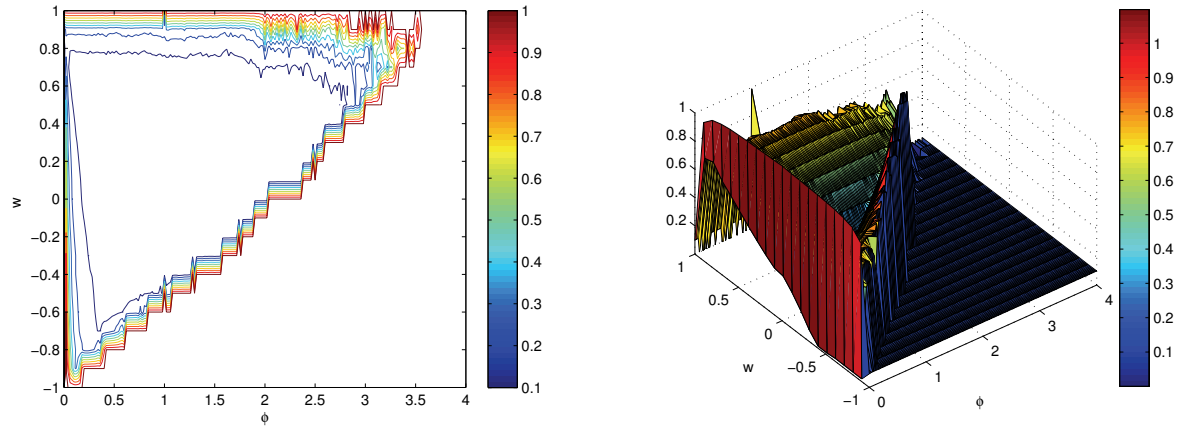
1. The existing convergence analysis methods utilize the expectation operator to deal with the randomness and obtain the convergence condition, while our method is based on the general evolutionary process. Specifically



(a) Convergence rate in Sphere function

(b) Comparison with the Trelea's result on Sphere function

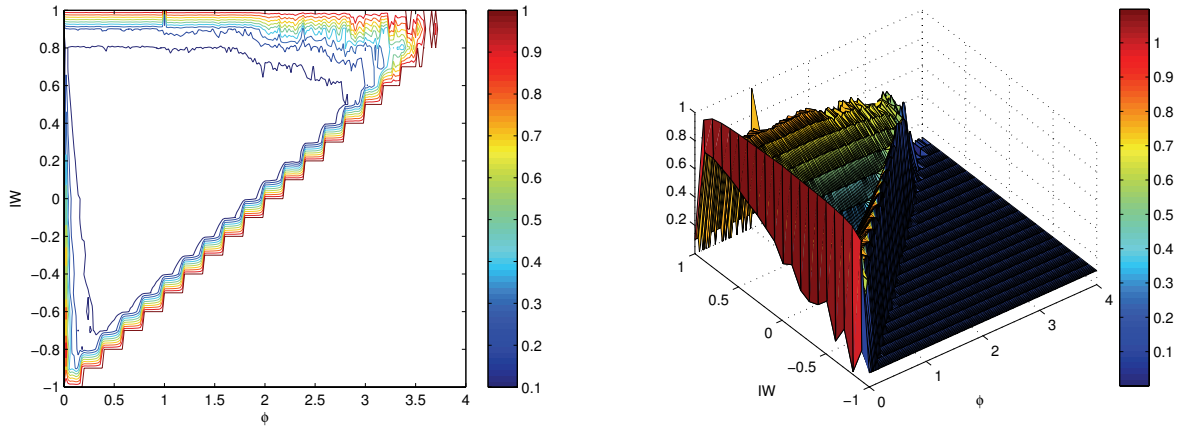
Figure 11: Convergence rate is mainly determined by the time-varying transfer matrix including random variables r_1 and r_2 in Sphere function. Larger convergence rate naturally needs too much computational time to converge, while smaller convergence rate needs too small number of generations to converge. The major discrepancy landscape is between Fig. 11(a) and the right subgraph of Fig. 2 in [31] on Sphere function. In essence, both landscapes mainly result from spectral radii by different methods, where the former is calculated by the time-varying transfer matrix, while the latter is directly based on the constant transfer matrix.



(a) Convergence rate in Griewank function

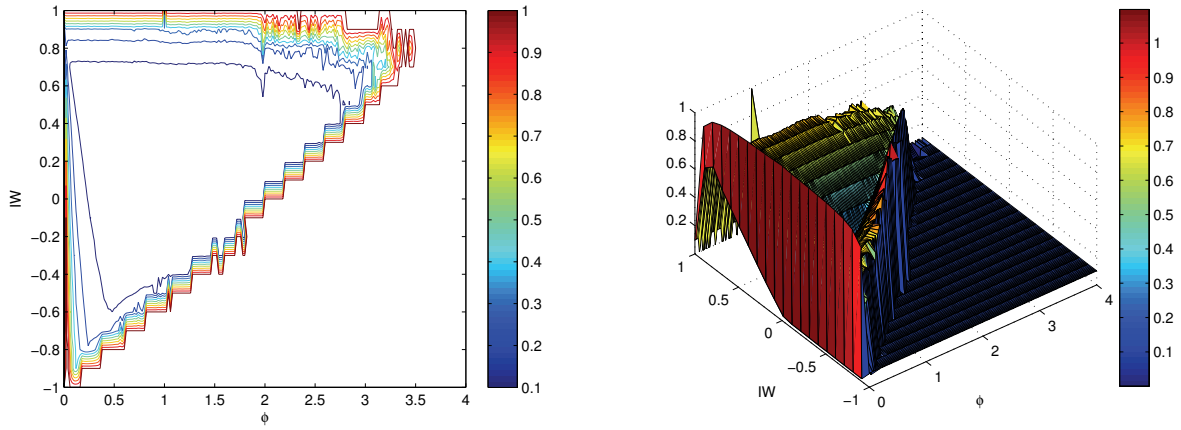
(b) Comparison with the Trelea's result on Griewank function

Figure 12: Convergence rate in Griewank function and the major discrepancy landscape between Fig. 12(a) and the right subgraph of Fig. 2 in [31] on Griewank function.



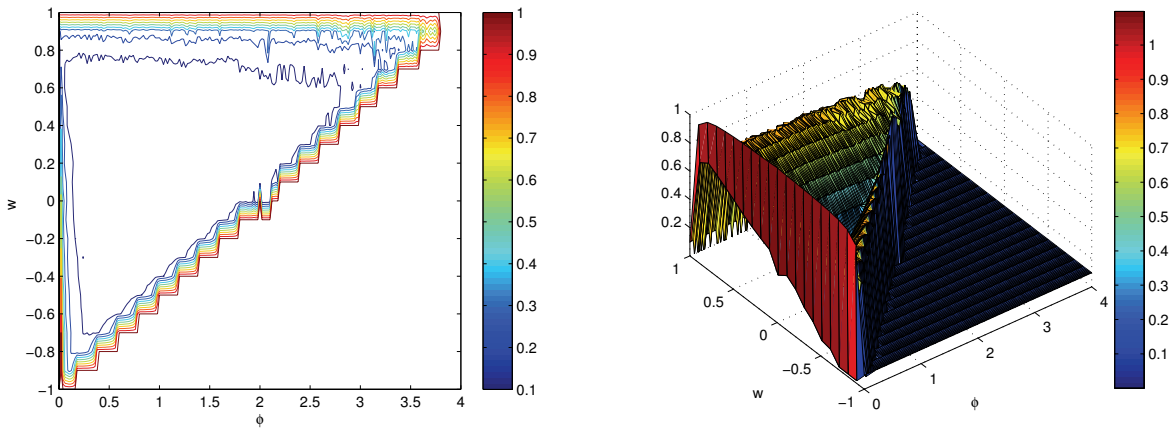
(a) Convergence rate in Rastrigin function (b) Comparison with the Trelea's result on Rastrigin function

Figure 13: Convergence rate in Rastrigin function and the major discrepancy landscape between Fig. 12(a) and the right subgraph of Fig. 2 in [31] on Rastrigin function.



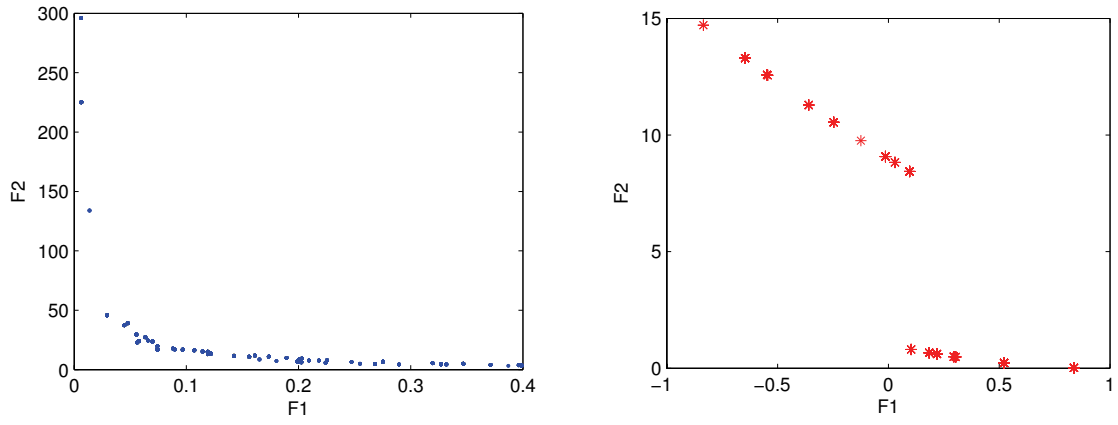
(a) Convergence rate in Rosenbrock function (b) Comparison with the Trelea's result on Rosenbrock function

Figure 14: Convergence rate in Rosenbrock function and the major discrepancy landscape between Fig. 12(a) and the right subgraph of Fig. 2 in [31] on Rosenbrock function.



(a) Convergence rate in Schaffer's f6 function (b) Comparison with the Trelea's result on Schaffer's f6 function

Figure 15: Convergence rate in Schaffer's f6 function and the major discrepancy landscape between Fig. 12(a) and the right subgraph of Fig. 2 in [31] on Schaffer's f6 function.



(a) Deb function (b) Another multi-objectives function

Figure 16: PSO algorithm with different topology handles with multi-objectives optimization problem.

speaking, the corresponding method calculates the joint spectral radius of the product of all transfer random matrices.

2. The joint spectral radius can denote the mathematical tradeoff between exploitation ability and exploration ability of PSO system.

3. By using our method, the setting parameters at the initialization process can compute the convergence speed of random PSO algorithm according to (39).

4. Because of the random time-varying transfer matrix, it is hard to obtain the mathematical convergence condition in our paper. However, the existing PSO convergence analysis method can get the concise convergence condition of first order and second order PSO system.

5. As to the corresponding result in this paper, it is also hard to analyze the particle's trajectory in the presence of random variable ϕ.

7. Concluding Remarks

In this contribution, from the perspective of the joint spectral radius, we are mainly concerned with analyzing the stability and discuss parameter selection of the standard PSO algorithm. One new convergence analysis on the random PSO algorithm is provided to discuss the difference between the existing results and our results. Essentially, its stability is governed by the product of all time-varying transfer matrices $\mathrm{M}(t)$ $(1 \leqslant t < \infty)$, while the existing convergence analysis for the standard PSO algorithm is conducted by regarding $\mathrm{M}(t)$ as a constant transfer matrix. Noticing that the product of time-varying transfer matrices is in fact very complicated due to the asymmetric transfer matrices involving random variables and it is very hard to analytically work out the explicit form of spectrum properties of the product, we discuss and analyze the relationship between $Mean(\varrho)$ and the parameters with the aid of Monte Carlo method.

In terms of extensive results of spectral radius in the simulations, the joint spectral radius, considering random variables including r_1 and r_2, maybe subject to a certain normal distribution which needs further theoretical analysis. This interesting and surprising result indicates that there may exist some intrinsic connections between the statistical parameters of the resulting normal distribution and the parameters of PSO algorithm, which may be critical to understand PSO system in depth. Additionally, the relationship between the mean spectral radius and the parameters from the view point of theoretic analysis still needs further investigation in the future.

Acknowledgements

This work is partially supported by National Nature Science Foundation under Grants 60974046, 61273150, 61011130163, 61004059, 61211130359 and 61374099. Also supported by Beijing Outstanding Talents Programme (2012D009011000003). Furthermore, this work also partially supported by the Research Fund for the Doctoral Program of Higher Education of China (No.20121101110029) and the China's Railway Corporation under Grants 2013X009-A-1, 2013X009-A-2 and 2012X010-A.

In addition, we would like to express our sincere thanks to the associate editor and the anonymous reviewers, whose valuable comments significantly help us to improve the quality of this paper.

References

[1] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995.

[2] R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43, 1995.

[3] M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73, 2002.

[4] F. Van den Bergh and A. P. Engelbrecht. A study of particle swarm optimization particle trajectories. *Information Sciences*, 176(8):937–971, 2006.

[5] R. Mendes, J. Kennedy, and J. Neves. The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation*, 8(3):204–210, 2004.

[6] R. C. Eberhart and Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization. In *Proceedings of the 2000 Congress on Evolutionary Computation*, pages 84–88, 2000.

[7] F. Van den Bergh and A. P. Engelbrecht. A cooperative approach to particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):225–239, 2004.

[8] S. Janson and M. Middendorf. A hierarchical particle swarm optimizer and its adaptive variant. *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 35(6):1272–1282, 2005.

[9] S. T. Hsieh, T. Y. Sun, C. C. Liu, and S. J. Tsai. Efficient population utilization strategy for particle swarm optimizer. *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 39(2):444–456, 2009.

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

[11] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj. Fractional particle swarm optimization in multidimensional search space. *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 40(2):298–319, 2010.

[12] K. E. Parsopoulos and M. N. Vrahatis. On the computation of all global minimizers through particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):211–224, 2004.

[13] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):256–279, 2004.

[14] T. Blackwell and J. Branke. Multi-swarm optimization in dynamic environments. *Applications of Evolutionary Computing*, 3005:489–500, 2004.

[15] Y. Jin and H. Branke. Evolutionary optimization in uncertain environments - A survey. *IEEE Transactions on Evolutionary Computation*, 9(3):303–317, 2005.

[16] T. Blackwell and J. Branke. Multiswarms, exclusion, and anti-convergence in dynamic environments. *IEEE Transactions on Evolutionary Computation*, 10(4):459–472, 2006.

[17] L. L. Liu, S. X. Yang, and D. W. Wang. Particle swarm optimization with composite particles in dynamic environments. *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 40(6):1634–1648, 2010.

[18] R. Brits, A. P. Engelbrecht, and F. Van den Bergh. Locating multiple optima using particle swarm optimization. *Applied Mathematics and Computation*, 189(2):1859–1883, 2007.

[19] R. C. Eberhart and Y. Shi. Tracking and optimizing dynamic systems with particle swarms. In *Proceedings of the 2001 Congress on Evolutionary Computation*, pages 94–100, 2001.

[20] D. Parrott and X. D. Li. Locating and tracking multiple dynamic optima by a particle swarm model using speciation. *IEEE Transactions on Evolutionary Computation*, 10(4):440–458, 2006.

[21] X. Q. Zhang, W. M. Hu, W. Qu, and S. Maybank. Multiple object tracking via species-based particle swarm optimization. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1590–1602, 2010.

[22] Y. H. Zheng and Y. Meng. Swarm intelligence based dynamic object tracking. In *Proceedings of 2008 IEEE Congress on Evolutionary Computation*, pages 405–412, 2008.

[23] A. A. A. Esmin, G. Lambert-Torres, and A. C. Z. de Souza. A hybrid particle swarm optimization applied to loss power minimization. *IEEE Transactions on Power Systems*, 20(2):859–866, 2005.

[24] C. F. Juang. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 34(2):997–1006, 2004.

[25] J. R. Zhang, J. Zhang, T. M. Lok, and M. R. Lyu. A hybrid particle swarm optimization-back-propagation algorithm for feedforward neural network training. *Applied Mathematics and Computation*, 185(2):1026–1037, 2007.

[26] G. Ciuprina, D. Ioan, and I. Munteanu. Use of intelligent-particle swarm optimization in electromagnetics. *IEEE Transactions on Magnetics*, 38(2):1037–1040, 2002.

[27] Z. L. Gaing. A particle swarm optimization approach for optimum design of PID controller in AVR system. *IEEE Transactions on Energy Conversion*, 19(2):384–391, 2004.

[28] L. Messerschmidt and A. P. Engelbrecht. Learning to play games using a PSO-based competitive learning approach. *IEEE Transactions on Evolutionary Computation*, 8(3):280–288, 2004.

[29] E. Ozcan and C. K. Mohan. Analysis of a simple particle swarm optimization system. In *Proceedings of Intelligent Engineering Systems Through Artificial Neural Networks*, volume 8, pages 253–258, 1998.

[30] E. Ozcan and C. K. Mohan. Particle swarm optimization: surfing the waves. In *Proceedings of the 1999 Congress on Evolutionary Computation*, volume 3, pages 1939–1944, 1999.

[31] I. C. Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85(6):317–325, 2003.

[32] V. Kadirkamanathan, K. Selvarajah, and P. J. Fleming. Stability analysis of the particle dynamics in particle swarm optimizer. *IEEE Transactions on Evolutionary Computation*, 10(3):245–255, 2006.

[33] M. Jiang, Y. P. Luo, and S. Y. Yang. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. *Information Processing Letters*, 102(1):8–16, 2007.

[34] R. Poli. Mean and variance of the sampling distribution of particle swarm optimizers during stagnation. *IEEE Transactions on Evolutionary Computation*, 13(4):712–721, 2009.

[35] M. R. Rapaic and Z. Kanovic. Time-varying PSO - convergence analysis, convergence related parameterization and new parameter adjustment schemes. *Information Processing Letters*, 109:548–552, 2009.

[36] J. L. Fernandez-Martinez and E. Garcia-Gonzalo. Stochastic stability analysis of the linear continuous and discrete PSO models. *IEEE Transactions on Evolutionary Computation*, 15(3):405–423, 2011.

[37] Y. Shi and R. C. Eberhart. A modified particle swarm optimizer. In *Proceedings of IEEE World Congress on Computational Intelligence*, pages 69–73, 1998.

[38] Y. Shi and R. C. Eberhart. Empirical study of particle swarm optimization. In *Proceedings of the 1999 Congress on Evolutionary Computation*, volume 3, pages 1945–1950, Piscataway, NJ, USA, 1999.