Computing and Informatics, Vol. 31, 2012, 371–392

DYNAMIC TRAJECTORY AND CONVERGENCE ANALYSIS OF SWARM ALGORITHM

Benxian Yue, Hongbo Liu*

School of Electronic and Information Engineering
Dalian University of Technology
Dalian 116023, China
e-mail: yuebenxian@vip.sina.com, lhb@dlut.edu.cn

Ajith Abraham

Machine Intelligence Research Labs – MIR Labs
Auburn, Washington 98071-2259, USA
e-mail: ajith.abraham@ieee.org

Communicated by Jozef Kelemen

Abstract. Swarm Intelligence (SI) is an innovative distributed intelligent paradigm whereby the collective behaviors of unsophisticated individuals interacting locally with their environment cause coherent functional global patterns to emerge. Although the swarm algorithms have exhibited good performance across a wide range of application problems, it is difficult to analyze the convergence. In this paper, we discuss the dynamic trajectory and convergence of the swarm intelligent model, namely the particle swarm algorithm. We explore the tradeoff between exploration and exploitation using differential analysis and Laplace transform. The trajectories are parsed into first-order inertial element and second-order oscillation element. Their transfer functions are derived, and the trajectories are described in explicit time functions. The first-order inertial element is helpful to maintain the trajectory's stability and algorithm convergence, while the second-order oscillation element trends to explore some new search spaces for the better solutions. The convergence regions of the swarm system are analyzed using the spectral radius and Lyapunov second theorem on stability.

Keywords: Swarm intelligence, swarm algorithm, convergence, stability

* corresponding author

1 INTRODUCTION

Swarm Intelligence (SI) is mainly inspired by social behaviour patterns of organisms that live and interact within large groups of unsophisticated autonomous individuals. In particular, it incorporates swarming behaviours observed in flocks of birds, schools of fish, or swarms of bees, colonies of ants, and even human social behavior, from which the intelligence is emerged [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. SI provides a framework to explore distributed problem solving without centralized control or the provision of a global model. The particle swarm model helps find optimal regions of complex search spaces through interaction of individuals in a population of particles [11]. It has exhibited good performance across a wide range of applications [12, 13, 14, 15, 16, 17, 5, 18, 19, 20, 21, 22, 23, 6, 24, 21, 25, 26, 27].

However, the executing efficiency and effectiveness of the algorithms are ignored in many important works, since it is difficult to evaluate the performance and convergence of considered algorithms. In the swarm intelligent model, its intelligent search must combine exploration of the new regions of the search space with evaluation of potential solutions already identified. Its performance and convergence are involved with the balancing exploration with exploitation. The exploitation of the swarm model emphasizes on searching around the best positions. In the swarm model, the good positions are the best position which each particle has found. Too much stress on exploration results in a pure random search whereas too much exploitation results in a pure local search. It is a fundamental problem in nature-inspired systems – the balance of system resources between exploration of the search space and exploitation of potentially good problem solutions. Particle swarm optimization is also believed to find an effective exploration/exploitation ratio. But it indeed strikes an effective exploration/exploitation balance and if so there are common principles that could provide a theoretical justification for this characteristic [28, 29].

In this paper, the dynamic trajectory and convergence of the swarm intelligent model, namely the particle swarm algorithm, are discussed in detail. We explore the particle's trajectory further using differential analysis and Laplace transform. The trajectories are parsed into first-order inertial elements and second-order oscillation element. Inertial elements of the trajectories are analyzed through the first-order differential equations, which depict the trajectories' tendency in one time-step. The trajectories exhibit oscillation element in the long times sense, which is implied in the first-order differential equations. Their transfer functions are derived, and the trajectories are described in explicit time functions. The first-order inertial element is helpful to maintain the trajectory's stability and algorithm convergence, while the second-order oscillation element trends to explore some new search spaces for the better solutions. The tradeoff between exploration and exploitation is crucial in search and optimization. The convergence regions of the swarm system are analyzed using the dynamical system theory. We proof the necessary and sufficient conditions of the convergence and the convergence regions are also illustrated.

The rest of the paper is organized as follows. Related works about dynamic trajectory and convergence analysis of swarm algorithm are reviewed in Section 2.

Particle swarm model is presented in Section 3. We analyse the first-order inertial element of the dynamic trajectories in Section 4. Two-order oscillation element of the swarm system is investigated using the second-order difference equations in Section 5. In Section 6, we discuss the convergence regions of the swarm system using the spectral radius and Lyapunov second theorem on stability, and finally conclusions are given in Section 7.

2 RELATED WORKS

The particle swarm algorithm has been applied to many real-world problems; but it is difficult to analyze the performance, stability and efficiency. What's more, sometime the algorithm has to be improved for the special applications [30, 31, 32, 33]. Many researchers explore the parameter selection, improvement guide, and mechanism analysis empirically and theoretically. Clerc and Kennedy analyze a particle's trajectory in discrete time and in continuous time. A five-dimensional depiction is developed, which describes the system completely. This analysis leads to a generalized model of the algorithm, containing a set of coefficients to control the system's convergence tendencies. Some results of the particle swarm optimizer, implementing modifications derived from the analysis, suggest methods for altering the original algorithm in ways that eliminate problems and increase the ability of the particle swarm to find optima of some well-studied test functions [11]. Ozcan and Mohan analyze closed form equations of particle swarm optimization algorithm for trajectories of particles in a multi-dimensional search space. The results show that in the general case, a particle does not "fly" in the search space, but rather "surfs" it on sine waves [34]. Trelea analyzes the particle swarm optimization algorithm using standard results from the dynamic system theory. Graphical parameter selection guidelines are derived. The exploration-exploitation tradeoff is discussed and illustrated [35]. Chen et al. investigate stabilities of the algorithm with constant parameters and time-varying parameters without Lipschitz constraint. Necessary and sufficient stability conditions for acceleration factor and inertia weight are presented [36]. Tan et al. analyse the evolutionary trajectories and the convergence properties based on the discrete time linear system theory, and the conditions for choosing the parameters are given [37]. Yasuda and Iwasaki analyse the stability analysis in order to obtain an understanding about how it searches a globally optimal solution and strategies about how to tune its parameters. Their work is carried out on the basis of both the eigenvalue analysis and the bounded input bounded output stability [38]. Van den Bergh and Engelbrecht overview the theoretical studies, and extend these studies to investigate particle trajectories for general swarms to include the influence of the inertia term. They also provide a formal proof that each particle converges to a stable point [39]. Kadirkamanathan et al. present the stability analysis of the particle swarm optimizer without this restrictive assumption using Lyapunov stability analysis and the concept of passive systems. Sufficient conditions for stability are derived [40]. Li et al. present the stability of particle's trajectory in particle swarm through difference equation and Z transform. They also discuss the influences of pBest, gBest and randomicity on particle's trajectory, and analyze the relationship between trajectory's stability and algorithm convergence [41]. Abraham et al. introduce some of the theoretical foundations of swarm intelligence. The design and implementation of the particle swarm optimization and ant colony optimization algorithms are provided for various types of function optimization problems, real world applications and data mining. Jiang et al. present a formal stochastic convergence analysis of the standard particle swarm optimization algorithm, which involves randomness. The stochastic convergent condition of the particle swarm system and corresponding parameter selection guidelines are derived [42]. Liu et al. [43] investigate the chaotic dynamic characteristics in swarm intelligence. The swarm intelligent model, namely the particle swarm (PS) is represented as an iterated function system. The dynamic trajectory of the particle is sensitive to the parameter values. The Lyapunov exponent and the correlation dimension are calculated and analyzed numerically for the dynamic system. The research results illustrate that performance of the swarm intelligent model depends on the sign of the maximum Lyapunov exponent. The particle swarm with a high maximum Lyapunov exponent usually achieves better performance, especially for multi-modal functions. Poli and Broomhead prevent the exact characterization of the sampling distribution of the swarm model [44]. Samal et al. present an alternative formulation of the PSO dynamics by a closed loop control system, and analyze the stability behavior of the system by using Jury's test and root locus technique [45]. Martínez and Gonzalo investigate stability, convergence and parameters for a generalized form of the particle swarm optimization algorithm presented [46]. Rapaić and Kanović investigate a formal convergence analysis of the conventional swarm algorithms with time-varying parameters. Several new schemes for parameter adjustment are introduced [47]. There are still two problems: why would the algorithm converge? And how to maintain the tradeoff between exploration and exploitation in the swarm model? It is necessary to investigate them from an explicitly theoretical rather than only heuristic perspective. In this paper, we discuss the dynamic trajectory and convergence of the swarm intelligent model, namely the particle swarm algorithm using differential analysis and Laplace transform.

3 SWARM ALGORITHM

A particle swarm model consists of a swarm of particles moving in a d-dimensional search space where the fitness f can be calculated as a certain quality measure. Each particle has a position represented by a position-vector $\vec{x}_i$ (i is the index of the particle), and a velocity represented by a velocity-vector $\vec{v}_i$. Each particle remembers its own best position so far in a vector $\vec{p}_i$, and its j^{th} dimensional value is $p_{i,j}$. The best position from the swarm thus far is then stored in a vector $\vec{p}_g$, and its j^{th} dimensional value is $p_{g,j}$. During the iteration time t, the update of the velocity from the previous velocity is determined by Equation (1a). Subsequently, the new position is determined by the sum of the previous position and the new velocity by Equation (1b)

$$\begin{aligned} v_{i,j}(t) &= \omega v_{i,j}(t-1) + c_1 r_1 (p_{i,j}(t-1) - x_{i,j}(t-1)) \quad &(1a)\\ &\quad + c_2 r_2 (p_{g,j}(t-1) - x_{i,j}(t-1)) \\ x_{i,j}(t) &= x_{i,j}(t-1) + v_{i,j}(t) \quad &(1b)\end{aligned}$$

where r_1 and r_2 are the random numbers, uniformly distributed within the interval $[0,1]$ for the j^{th} dimension of i^{th} particle; c_1 is a positive constant termed the coefficient of the self-recognition component; c_2 is a positive constant termed the coefficient of the social component. The variable ω is the inertia factor, whose value is typically set up to vary linearly from 1 to 0 during the iterated processing. From Equation (1a), a particle decides where to move next, considering its own experience, which is the memory of its best past position, and the experience of its most successful particle in the swarm. In the particle swarm model, the particle searches the solutions in the problem space within a range $[-s, s]$ (If the range is not symmetrical, it can be translated to the corresponding symmetrical range.) The pseudo-code for particle-search is illustrated in Algorithm 1.

Algorithm 1 Particle Swarm Algorithm

01. Initialize the size of the particle swarm n, and other
01.   parameters; Initialize the positions and the velocities
01.   for all the particles randomly.
02. While (the end criterion is not met) do
03. $t = t + 1$;
04.   Calculate the fitness value of each particle;
05. $\vec{p}_g(t) = argmin_{i=1}^{n}(f(\vec{p}_g(t-1), f(\vec{p}_1(t)),$
05. $f(\vec{p}_2(t)), \ldots, f(\vec{p}_i(t)), \ldots, f(\vec{p}_n(t)));$
06.   For i= 1 to n
07. $\vec{p}_i(t) = argmin_{i=1}^{n}(f(\vec{p}_i(t-1)), f(\vec{p}_i(t));$
08.   For $j = 1$ to d
09.   Update the j^{th} dimension value of $\vec{x}_i$ and $\vec{v}_i$
09.   according to Equations (1a) and (1b);
10.   Next j
11.   Next i
12. End While.

The particle swarm algorithm can be described generally as a population of vectors whose trajectories oscillate around a region which is defined by each individual's previous best success and the success of some other particle. The trajectory of a single particle is illustrated in Figure 1. Eberhart and Kennedy called the two basic methods "$g$best model" and "$l$best model" [2]. Some previous studies have shown that $g$best model converges quickly on problem solutions but has a weakness in becoming trapped in local optima, while $l$best model converges slowly on problem solutions but is able to "flow around" local optima, as the individuals explore different regions [48, 49].

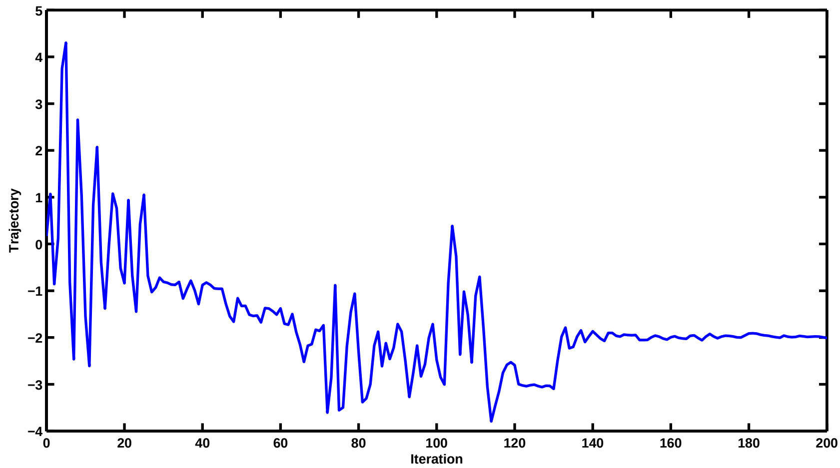

Fig. 1. Trajectory of a single particle

4 FIRST-ORDER INERTIAL ELEMENT

Observing Equation (1a), the particle decides where to move next, considering its own experience, which is the memory of the best past position, and the experience of the most successful particle in the swarm. We split the three terms in the RHS of Equation (1a) into three equations:

$$v_{i,j}(t) = \gamma_{1,1}\omega v_{i,j}(t-1) \tag{2}$$

$$v_{i,j}(t) = \gamma_{1,2}c_1r_1(p_{i,j}(t-1) - x_{i,j}(t-1)) \tag{3}$$

$$v_{i,j}(t) = \gamma_{1,3}c_2r_2(p_{g,j}(t-1) - x_{i,j}(t-1)) \tag{4}$$

where $\gamma_{1,1}$, $\gamma_{1,2}$ and $\gamma_{1,3}$ are the component constants within the interval $[0, 1]$. Let $\tau_{1,1} = \gamma_{1,1}\omega$, $\tau_{1,2} = \gamma_{1,2}c_1r_1$ and $\tau_{1,3} = \gamma_{1,3}c_2r_2$, reduce to a single dimension, and refer to Equation (1b), Equations (2), (3) and (4) can be deformed into one velocity equation and two difference equations as follows:

$$v_{i,j}(t) = \tau_{1,1}v_{i,j}(t-1) \tag{5}$$

$$x_{i,j}(t) - x_{i,j}(t-1) = \tau_{1,2}(p_{i,j}(t-1) - x_{i,j}(t-1)) \tag{6}$$

$$x_{i,j}(t) - x_{i,j}(t-1) = \tau_{1,3}(p_{g,j}(t-1) - x_{i,j}(t-1)) \tag{7}$$

Applying the Laplace transform to Equations (3) and (4), yield Equations (8) and (9), and rearranging results in Equations (10) and (11).

$$\begin{aligned} sX_i(s) + \tau_{1,2}X_i(s) &= \tau_{1,2}P_i(s) && (8)\\ sX_i(s) + \tau_{1,3}X_i(s) &= \tau_{1,3}P_g(s) && (9) \end{aligned}$$

$$\begin{aligned} G_{1,1}(s) &= \frac{X_i(s)}{P_i(s)} = \frac{\tau_{1,2}}{s+\tau_{1,2}} && (10)\\ G_{1,2}(s) &= \frac{X_i(s)}{P_g(s)} = \frac{\tau_{1,3}}{s+\tau_{1,3}} && (11) \end{aligned}$$

Taking the inverse Laplace transform, we can obtain: Equations (12) and (13).

$$\begin{aligned} g_{1,2}(t) &= \mathcal{L}^{-1}\{G_{1,1}(s)\}(t) = \tau_{1,2}e^{-\tau_{1,2}t} && (12)\\ g_{1,2}(t) &= \mathcal{L}^{-1}\{G_{1,2}(s)\}(t) = \tau_{1,3}e^{-\tau_{1,3}t} && (13) \end{aligned}$$

As illustrated in Equation (5), the velocity of the particle in the swarm model implies the inertial element. When $\tau_{1,1} < 1$, the velocity would be decreased gradually. The trajectories also contain inertial element in a short time-step, as described in Equations (12) and (13). Observing Figure 2, the particle swarm would collect the last position very fast. It is very helpful for the swarm algorithm to converge to the global optimum. However, it is possible for multi-modal problems involving high dimensions to suffer from a total implosion and ultimately fitness stagnation of the swarm. The oscillation element leads the particles to explore some new search space, which would be discussed in the next section.

5 TWO-ORDER OSCILLATION ELEMENT

We add a time step in Equations (1a) and (1b) and obtain recurrence relation in Equations (14a) and (14b).

$$\begin{aligned} v_{i,j}(t+1) &= \omega v_{i,j}(t) + c_1r_1(p_{i,j}(t) - x_{i,j}(t)) + c_2r_2(p_{g,j}(t) - x_{i,j}(t)) && (14a)\\ x_{i,j}(t+1) &= x_{i,j}(t) + v_{i,j}(t+1) && (14b) \end{aligned}$$

Substituting Equation (14a) into Equation (1b), we get Equation (15).

$$v_{i,j}(t+1) - (\omega+1)v_{i,j}(t) + (c_1r_1 + c_2r_2)(x_{i,j}(t) - x_{i,j}(t-1)) + \omega v_{i,j}(t-1) = 0 \quad (15)$$

Without loss of generality, we can analyze the swarm system by still observing the one-dimensional model. After elimination of position variables using Equation (1b), the velocity varying process yields an equivalent, second-order difference Equation (16).

$$v_{i,j}(t+1) + (c_1r_1 + c_2r_2 - \omega - 1)v_{i,j}(t) + \omega v_{i,j}(t-1) = 0 \quad (16)$$

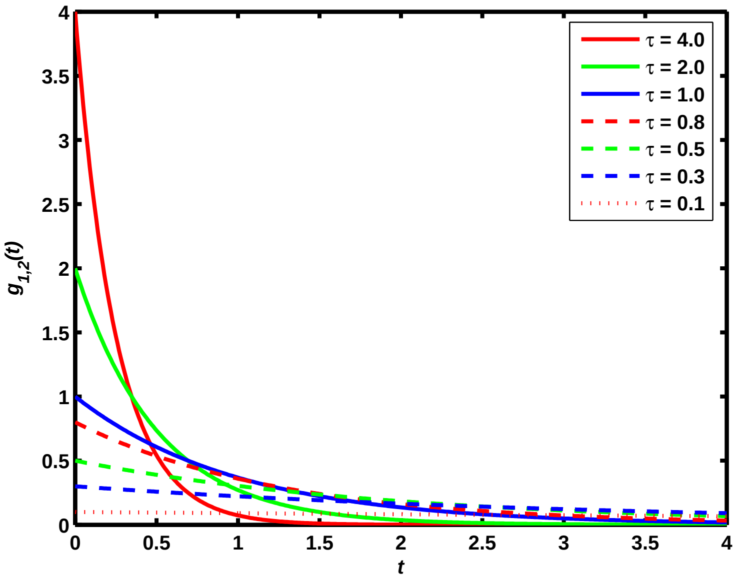

Fig. 2. First-order inertial element curve

Substituting Equations (1b) and (14b) into Equation (14a), the position varying process also yields an equivalent, second-order difference Equation (17).

$$x_i(t+1) + (c_1 r_1 + c_2 r_2 - \omega - 1)x_i(t) + \omega x_i(t-1) - c_1 r_1 p_i(t) - c_2 r_2 p_g(t) = 0 \quad (17)$$

Let $\varphi_1 = c_1 r_1$, $\varphi_2 = c_2 r_2$ and $\varphi = \varphi_1 + \varphi_2$, and reduce to a single dimension; then the second-order difference, non-homogeneous recurrence relations are obtained:

$$v_i(t+1) + (\varphi - \omega - 1)v_i(t) + \omega v_i(t-1) = 0 \quad (18a)$$
$$x_i(t+1) + (\varphi - \omega - 1)x_i(t) + \omega x_i(t-1) - \varphi_1 p_i(t) - \varphi_2 p_g(t) = 0 \quad (18b)$$

Taking the Laplace transform of Equation (18b), we obtain Equation (19).

$$X_i(s) = \frac{\varphi_1 s P_i(s) + \varphi_2 s P_g(s)}{s^2 + (\varphi_1 + \varphi_2 - \omega - 1)s + \omega} \quad (19)$$

Equation (19) can be split into two equtions:

$$X_i(s) = \frac{\gamma_{2,1}\varphi_1 s P_i(s)}{s^2 + (\varphi_1 + \varphi_2 - \omega - 1)s + \omega} \quad (20)$$
$$X_i(s) = \frac{\gamma_{2,1}\varphi_2 s P_g(s)}{s^2 + (\varphi_1 + \varphi_2 - \omega - 1)s + \omega} \quad (21)$$

where $\gamma_{2,1}$ and $\gamma_{2,2}$ are the component constants within the interval $[0,1]$. Let $\tau_{2,1} = \gamma_{2,1}\varphi_1$, $\tau_{2,2} = \gamma_{2,2}\varphi_2$ and $\varphi = \varphi_1 + \varphi_2$; Equations (20) and (21) can be rewritten as

$$G_{2,1}(s) = \frac{X_i(s)}{P_i(s)} = \frac{\tau_{2,1}s}{s^2 + (\varphi_1+\varphi_2-\omega-1)s+\omega} \tag{22}$$

$$G_{2,2}(s) = \frac{X_i(s)}{P_g(s)} = \frac{\tau_{2,2}s}{s^2 + (\varphi_1+\varphi_2-\omega-1)s+\omega} \tag{23}$$

i.e.,

$$G_{2,1}(s) = \frac{\tau_{2,1}s}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2 - \frac{4\omega-(\varphi-\omega-1)^2}{4}} \tag{24}$$

$$G_{2,2}(s) = \frac{\tau_{2,2}s}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2 - \frac{4\omega-(\varphi-\omega-1)^2}{4}} \tag{25}$$

Consider two cases.

1. If $(\varphi-\omega-1)^2 \geq 4\omega$, Equation (24) can be rewritten as

$$\begin{aligned} G_{2,1}(s) &= \tau_{2,1}\frac{s+\frac{\varphi-\omega-1}{2}-\frac{\varphi-\omega-1}{2}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2 - \frac{(\varphi-\omega-1)^2-4\omega}{4}} \\ &= \tau_{2,1}\frac{s+\frac{\varphi-\omega-1}{2}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2 - \left(\sqrt{\frac{(\varphi-\omega-1)^2-4\omega}{4}}\right)^2} \\ &\quad - \tau_{2,1}\frac{\frac{\varphi-\omega-1}{\sqrt{(\varphi-\omega-1)^2-4\omega}}\sqrt{\frac{(\varphi-\omega-1)^2-4\omega}{4}}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2 - \left(\sqrt{\frac{(\varphi-\omega-1)^2-4\omega}{4}}\right)^2} \end{aligned} \tag{26}$$

Applying the inverse Laplace transform, we can find the general solution in Equation (27).

$$\begin{aligned} g_{2,1}(t) &= \mathcal{L}^{-1}\{G_{2,1}(s)\}(t) \\ &= \tau_{2,1}e^{-\frac{\varphi-\omega-1}{2}t}\cosh\left(\frac{\sqrt{(\varphi-\omega-1)^2-4\omega}}{2}t\right) \\ &\quad - \tau_{2,1}\frac{\varphi-\omega-1}{\sqrt{(\varphi-\omega-1)^2-4\omega}}e^{-\frac{\varphi-\omega-1}{2}t}\sinh\left(\frac{\sqrt{(\varphi-\omega-1)^2-4\omega}}{2}t\right) \end{aligned} \tag{27}$$

2. If $(\varphi-\omega-1)^2 < 4\omega$, similarly Equation (25) can be rewritten as

$$\begin{aligned}
G_{2,2}(s) &= \tau_{2,2}\frac{s+\frac{\varphi-\omega-1}{2}-\frac{\varphi-\omega-1}{2}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2+\frac{4\omega-(\varphi-\omega-1)^2}{4}} \\
&= \tau_{2,2}\frac{s+\frac{\varphi-\omega-1}{2}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2+\left(\sqrt{\frac{4\omega-(\varphi-\omega-1)^2}{4}}\right)^2} \\
&\quad -\tau_{2,2}\frac{\frac{\varphi-\omega-1}{\sqrt{4\omega-(\varphi-\omega-1)^2}}\sqrt{\frac{4\omega-(\varphi-\omega-1)^2}{4}}}{\left(s+\frac{\varphi-\omega-1}{2}\right)^2+\left(\sqrt{\frac{4\omega-(\varphi-\omega-1)^2}{4}}\right)^2}
\end{aligned} \tag{28}$$

Applying the inverse Laplace transform, we can find the general solution in Equation (29).

$$\begin{aligned}
g_{2,2}(t) &= \mathcal{L}^{-1}\{G_{2,2}(s)\}(t) \\
&= \tau_{2,2}e^{-\frac{\varphi-\omega-1}{2}t}\cos\left(\frac{\sqrt{4\omega-(\varphi-\omega-1)^2}}{2}t\right) \\
&\quad -\tau_{2,2}\frac{\varphi-\omega-1}{\sqrt{4\omega-(\varphi-\omega-1)^2}}e^{-\frac{\varphi-\omega-1}{2}t}\sin\left(\frac{\sqrt{4\omega-(\varphi-\omega-1)^2}}{2}t\right)
\end{aligned} \tag{29}$$

As illustrated in Equations (27) and (29), the trajectory of the particle in the swarm model implies the oscillation element. The explicit time functions have shown that the trajectories of the particles oscillate as different sinusoidal (*sin* and *cos*) or hyperbolic sinusoidal (*sinh* and *cosh*) waves. Figures 3 and 4 illustrate the tendency of the trajectories. It is to be noted that the parameters are set to be constants and $\tau = 2.0$ in Figures 3 and 4. The sustained oscillation can be adjusted by tuning these parameters randomly or manually. The first-order inertial element is helpful to maintain the trajectory's stability and algorithm convergence, while the second-order oscillation element trends to explore some new search spaces for the better solutions. The tradeoff between exploration and exploitation is crucial in search and optimization, having a great effect on global optimization performance, e.g., accuracy and convergence speed of optimization algorithms [50, 51, 52].

6 CONVERGENCE REGIONS

Tan et al. presented analysis of the evolutionary trajectories and the convergence properties of PSO based on the discrete time linear system theory. They also discussed the conditions for choosing the parameters [37]. Rapaić and Kanović provided a formal convergence analysis of the conventional PSO algorithms with time-varying parameters [47]. Based on their analysis, the convergence-related parametric model

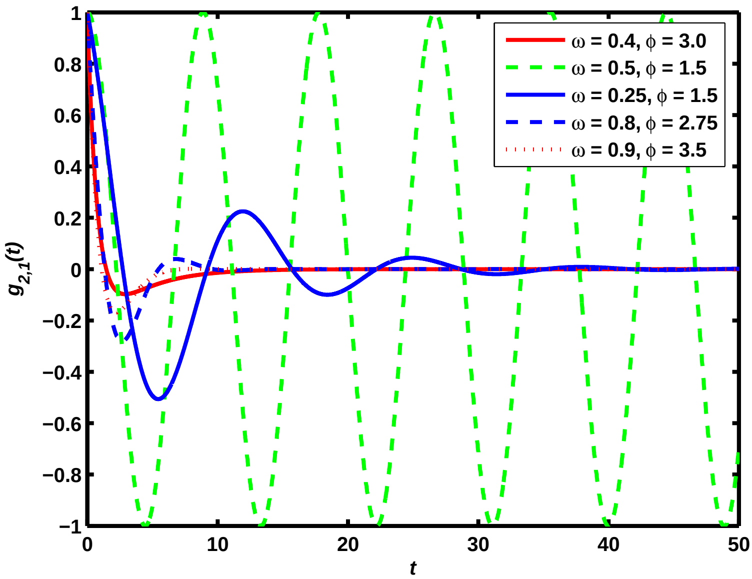

Fig. 3. Two-order inertial element Curve ($g_{2,1}(t)$)

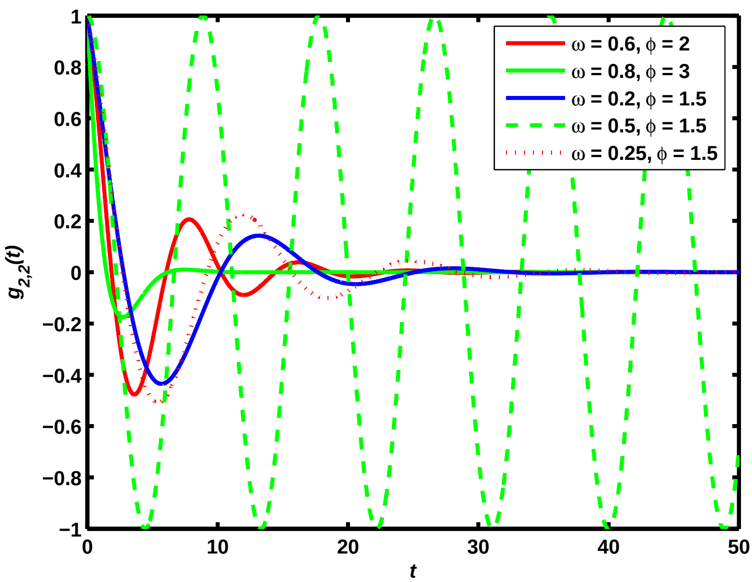

Fig. 4. Two-order inertial element Curve ($g_{2,2}(t)$)

for the conventional PSO was introduced. In this section, we further analyze the convergence regions of the swarm system based on the spectral radius and Lyapunov second theorem on stability.

The swarm system can be investigated by still observing the one-dimensional model:

$$\begin{aligned} v_i(t) &= \omega v_i(t-1) + c_1 r_1(p_i(t-1) - x_i(t-1)) + c_2 r_2(p_g(t-1) - x_i(t-1)) && \text{(30a)} \\ x_i(t) &= x_i(t-1) + v_i(t) && \text{(30b)} \end{aligned}$$

Equations (30a) and (30b) are rewritten as:

$$\begin{aligned} v_i(t) &= \omega v_i(t-1) - \varphi x_i(t-1) + \varphi_1 p_i(t-1) + \varphi_2 p_g(t-1) && \text{(31a)} \\ x_i(t) &= \omega v_i(t-1) + (1-\varphi) x_i(t-1) + \varphi_1 p_i(t-1) + \varphi_2 p_g(t-1) && \text{(31b)} \end{aligned}$$

This recurrence relation represents the system in state-space form:

$$\begin{bmatrix} v_i(t) \\ x_i(t) \end{bmatrix} = \begin{bmatrix} \omega & -\varphi \\ \omega & 1-\varphi \end{bmatrix} \cdot \begin{bmatrix} v_i(t-1) \\ x_i(t-1) \end{bmatrix} + \begin{bmatrix} \varphi_1 & \varphi_2 \\ \varphi_1 & \varphi_2 \end{bmatrix} \cdot \begin{bmatrix} p_i(t-1) \\ p_g(t-1) \end{bmatrix}$$

Let $\vec{y}(t) = \begin{bmatrix} v_i(t) \\ x_i(t) \end{bmatrix}$, $\vec{p}(t) = \begin{bmatrix} p_i(t) \\ p_g(t) \end{bmatrix}$, $G = \begin{bmatrix} w & -\varphi \\ w & 1-\varphi \end{bmatrix}$ and $B = \begin{bmatrix} \varphi_1 & \varphi_2 \\ \varphi_1 & \varphi_2 \end{bmatrix}$, then we have its discrete time varying system written in compact matrix form as follows:

$$\vec{y}(t) = G \cdot \vec{y}(t-1) + B \cdot \vec{p}(t-1). \tag{32}$$

When ω and φ are constants, the system converges if and only if $\rho(G) < 1$ where $\rho(G)$ represents the spectral radius of matrix G. Also, the rate of its convergence depends on $\rho(G)$ for the iterative process determined by Equation (32). Therefore, the spectral radius of the iteration matrix plays an important role in the comparison of the speed of convergence of different iterative process of the same system. The characteristic polynomial of the matrix G is expressed as

$$\lambda^2 + (\varphi - 1 - \omega)\lambda + \omega \tag{33}$$

where the scalar λ is an eigenvalue of G.

Lemma 1 (Necessary and Sufficient Condition of Convergence)**.** When ω and φ are constants, the system converges if and only if $0 < \varphi < 2\omega + 2$ and $\omega < 1$ are satisfied together, and the convergence rate is determined by $\rho(G)$.

Proof. When ω and φ are the constants, we can obtain its eigenvalues of the iteration matrix G in Equation (33). The eigenvalues are

$$\lambda_{1,2} = \begin{cases} \frac{1+\omega-\varphi \pm \sqrt{(\varphi-\omega-1)^2 - 4\omega}}{2} & \text{if } (\varphi-\omega-1)^2 \geq 4\omega \\ \frac{1+\omega-\varphi \pm i\sqrt{4\omega - (\varphi-\omega-1)^2}}{2} & \text{if } (\varphi-\omega-1)^2 < 4\omega. \end{cases} \tag{34}$$

The spectral radius $\rho(G)$ of matrix G is determined by the eigenvalues.

$$\rho(G) = \max_{1\le i\le 2} \|\lambda_i\| = \max\{\|\lambda_1\|, \|\lambda_2\|\} \tag{35}$$

where if λ is real number, $\|\lambda\|$ denotes its absolute value, and $\|\lambda\|$ denotes its norm if λ is complex number. Linear asynchronous iterations state that convergence occurs if and only if the spectral radius of the modulus matrix is less than 1. So the system converges if and only if $\rho(G) < 1$ [53, 54].

Consider two cases.

1. If $(\varphi - \omega - 1)^2 \ge 4\omega$, λ_1 and λ_2 are real numbers.

$$-2 < 1 + \omega - \varphi + \sqrt{(\varphi - \omega - 1)^2 - 4\omega} < 2 \tag{36a}$$

$$-2 < 1 + \omega - \varphi - \sqrt{(\varphi - \omega - 1)^2 - 4\omega} < 2 \tag{36b}$$

i.e.

$$1 + \omega - \varphi + \sqrt{(\varphi - \omega - 1)^2 - 4\omega} < 2 \tag{37a}$$

$$-2 < 1 + \omega - \varphi - \sqrt{(\varphi - \omega - 1)^2 - 4\omega}. \tag{37b}$$

We can obtain Equation (38a) from Equation (37a), and Equation (38b) from Equation (37b).

$$\varphi < 2\omega + 2 \tag{38a}$$

$$0 < \varphi \tag{38b}$$

We make Equation (38a), Equation (38b) and the precondition ($(\varphi - \omega - 1)^2 \ge 4\omega$) together. If $\omega > 0$, we have the following equations:

$$0 < \varphi \le 1 + \omega - 2\sqrt{\omega} \tag{39}$$

or

$$1 + \omega + 2\sqrt{\omega} \le \varphi < 2\omega + 2. \tag{40}$$

If $\omega < 0$, we get Equation (41).

$$0 < \varphi < 2\omega + 2. \tag{41}$$

2. If $(\varphi - \omega - 1)^2 < 4\omega$, λ_1 and λ_2 are complex numbers.

$$\|\lambda_{1,2}\| = \sqrt{\frac{(\omega + 1 - \varphi)^2}{4} + \frac{4\omega - (\varphi - \omega - 1)^2}{4}} = \sqrt{\omega}. \tag{42}$$

We make Equation (42) and the precondition ($(\varphi - \omega - 1)^2 < 4\omega$) together. It is concluded that:

$$\omega + 1 - 2\sqrt{\omega} < \varphi < \omega + 1 + 2\sqrt{\omega} \tag{43}$$

and

$$0 \leq \omega < 1. \tag{44}$$

Synthesize case (1) and case (2); it is concluded that:

$$\begin{cases} 0 < \varphi < 2\omega + 2 \\ \omega < 1. \end{cases} \tag{45}$$

□

Figure 5 illustrates the convergent range, in which A_1, A_2 and A_3 are determined by the real eigenvalues, A_4 is determined by the complex eigenvalues. The spectral radius increases from 0 to 1 as shown in Figures 6 and 7, and its contours are illustrated in Figure 8. The less the spectral radius is, the faster convergence rate the swarm system has.

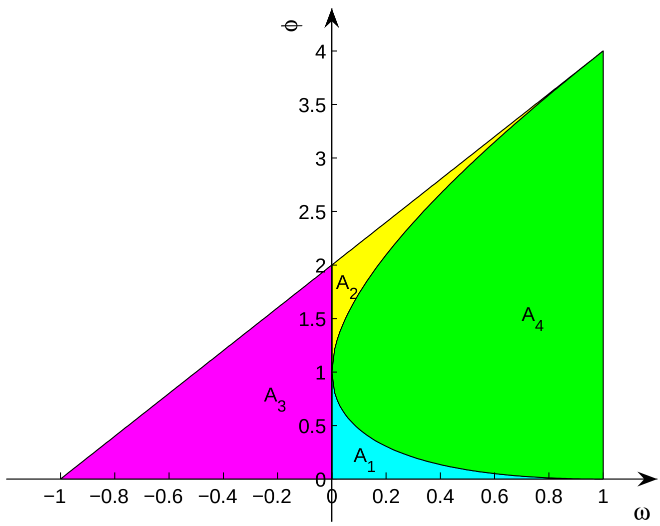

Fig. 5. Convergence regions

When ω and φ are also discrete time functions but not the constants, then we have the discrete time varying system:

$$\vec{y}(t) = G(t) \cdot \vec{y}(t-1) + B(t) \cdot \vec{p}(t-1). \tag{46}$$

Lemma 2 ((Lyapunov's Second Theorem on Stability) [55])**.** Let $\zeta(t) = 0$ be an equilibrium point of a nonlinear system. The equilibrium point is globally asymptotically stable (stability in the large) if there is a nonnegative scalar function $V(\zeta(t))$ which satisfies:

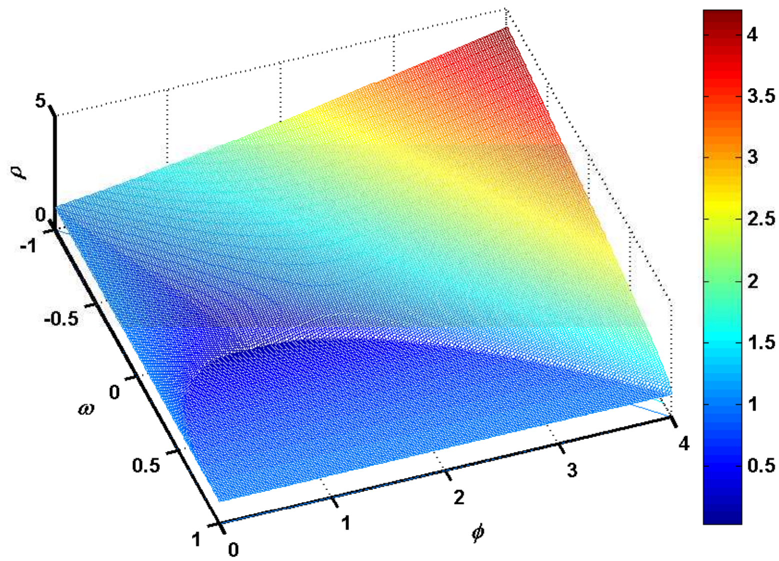

Fig. 6. Spectral radius in $(0, 70)^\circ$ view

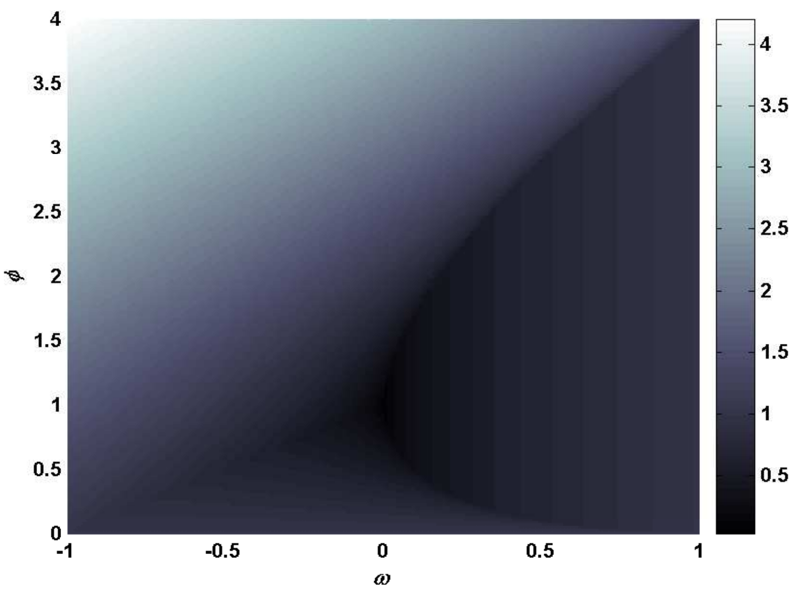

Fig. 7. Spectral radius in $(0, 90)^\circ$ view

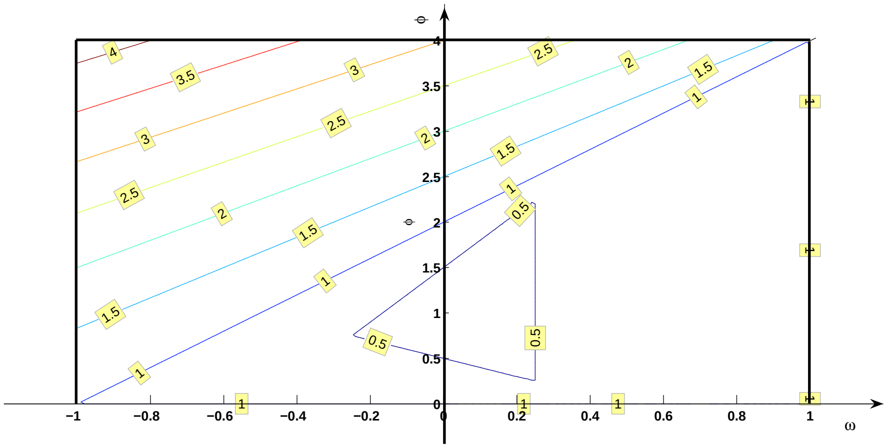

Fig. 8. Contour of the spectral radius

1.
$$V(0) = 0 \quad \text{and} \quad V(\zeta(t)) > 0 \quad \forall \zeta(t) \neq 0 \quad \text{(Positive Definite)} \tag{47}$$

2.
$$\Delta V(\zeta(t)) = V(\zeta(t+1)) - V(\zeta(t)) < 0 \quad \text{(Negative Definite)} \tag{48}$$

3.
$$\|\zeta(t)\| \to 0 \Rightarrow V(\zeta(t)) \to \infty. \tag{49}$$

The Lyapunov's second theorem on stability allows determining how far from the equilibrium point the trajectory can be and still converge to it as t approaches ∞ [56, 57, 58].

Lemma 3 (Sufficient Condition of Convergence)**.** When ω and φ are the discrete time functions, the discrete time varying system converges if $0 < \varphi < 2\omega + 2$ and $\omega < 1$ are satisfied together.

Proof. The "obvious" nonnegative scalar function to use in this context is the norm of the vector, i.e. let $\vec{y}(t) = \begin{bmatrix} v_i(t) \\ x_i(t) \end{bmatrix}$, and $V(\vec{y}(t)) = \|\vec{y}(t)\|$.

1. It is obvious that $V(\vec{y}(t)) \geq 0$, i.e. $V(\vec{y}(t))$ positive definite.
2.
$$\begin{aligned} \Delta V(\vec{y}(t)) &= V(\vec{y}(t+1)) - V(\vec{y}(t)) \\ &= \|\vec{y}(t+1)\| - \|\vec{y}(t)\| \\ &\leq \|G(t)\|\|\vec{y}(t+1)\| - \|\vec{y}(t)\| \\ &= (\|G(t)\| - 1)\|\vec{y}(t)\| \end{aligned} \tag{50}$$

Only if $\|G(t)\| < 1$, $\Delta V(\vec{y}(t))$ is negative definite. It means the norm of the eigenvalues of matrix G must be less than 1, i.e. ω and φ must satisfy the conditions in Equation (45).

3.

$$\|\vec{y}(t)\| \to 0 \Rightarrow V(\vec{y}(t)) \to \infty \tag{51}$$

Synthesize cases (1), (2) and (3); the lemma is proved up. □

Lemma 3 usually applied to the particle dynamics in determining sufficient conditions for asymptotic stability and, hence, convergence to the equilibrium point.

7 CONCLUSIONS

In this paper, we focused on the dynamic trajectory and convergence of swarm algorithm based on discrete time varying theory and probabilistic theory. We explored the tradeoff between exploration and exploitation using differential analysis and Laplace transform. The trajectories are parsed into first-order inertial element and second-order oscillation element. Their transfer functions are derived, and the trajectories are described in explicit time functions. The first-order inertial element is helpful to maintain the trajectory's stability and algorithm convergence, while the second-order oscillation element trends to explore some new search spaces for the better solutions. The convergence regions of the swarm system were analyzed using the dynamical system theory. We provided the necessary and sufficient conditions of the convergence. And the convergence regions were also illustrated.

Acknowledgment

The authors would like to thank Maurice Clerc for his scientific collaboration in this research work. The work described in this article was supported by the National Natural Science Foundation of China (Grant No. 60873054, 51005034, 61073056, 61173035, 61105117), the Program for New Century Excellent Talents in University, the Fundamental Research Funds for the Central Universities (Grant No. 2011-JC027), the Science and Technology Fund of Dalian (Grant No. 2010J21DW006).

REFERENCES

[1] Bonabeau, E.—Dorigo, M.—Theraulaz, G.: Swarm Intelligence: from Natural to Artificial Systems. Oxford University Press 1999.

[2] Kennedy, J.—Eberhart, R.—Shi, Y.: Swarm intelligence. Springer 2001.

[3] Engelbrecht, A.: Fundamentals of Computational Swarm Intelligence. Wiley 2005.

[4] Clerc, M.: Particle Swarm Optimization. ISTE Publishing Company 2006.

[5] Forestiero, A.—Mastroianni, C.—Spezzano, G.: So-Grid: A Self-Organizing Grid Featuring Bio-Inspired Algorithms. ACM Transactions on Autonomous and Adaptive Systems, Vol. 3, 2008, No. 2, pp. 1–37.

[6] Su, M.—Su, S.—Zhao, Y.: A Swarm-Inspired Projection Algorithm. Pattern Recognition, Vol. 42, 2009, No. 11, pp. 2764–2786.

[7] Lazinica, A.: Particle Swarm Optimization. InTech Education and Publishing 2009.

[8] Banks, A.—Vincent, J.—Phalp, K.: Natural Strategies for Search. Natural Computing, Vol. 8, 2009, No. 3, pp. 547–570.

[9] Schut, M. C.: On Model Design for Simulation of Collective Intelligence. Information Sciences, Vol. 180, 2010, No. 1, pp. 132–155.

[10] Jeya Mala, D.—Ruby, S.—Mohan, V.: A Hybrid Test Optimization Framework-Coupling Genetic Algorithm With Local Search Technique. Computing and Informatics, Vol. 29, 2010, No. 1, pp. 133–164.

[11] Clerc, M.—Kennedy, J.: The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation, Vol. 6, 2002, No. 1, pp. 58–73.

[12] Parsopoulos, K.—Vrahatis, M.: Recent Approaches to Global Optimization Problems Through Particle Swarm Optimization. Natural Computing, Vol. 1, 2002, No. 2, pp. 235–306.

[13] Vandenbergh, F.—Engelbrecht, A.: A Cooperative Approach to Particle Swarm Optimization. IEEE Transactions on Evolutionary Computation, Vol. 8, 2004, No. 3, pp. 225–239.

[14] Xu, R.—Wunsch, D.—Frank, R.: Inference of Genetic Regulatory Networks With Recurrent Neural Network Models Using Particle Swarm Optimization. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2007, pp. 681–692.

[15] Liu, H.—Abraham, A.: An Hybrid Fuzzy Variable Neighborhood Particle Swarm Optimization Algorithm for Solving Quadratic Assignment Problems. Journal of Universal Computer Science, Vol. 13, 2007, No. 9, pp. 1309–1331.

[16] Agrawal, S.—Dashora, Y.—Tiwari, M.—Son, Y.: Interactive Particle Swarm: A Pareto-Adaptive Metaheuristic to Multiobjective Optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and humans, Vol. 38, 2008, No. 2, pp. 258–277.

[17] Leong, W.—Yen, G.: PSO-Based Multiobjective Optimization With Dynamic Population Size and Adaptive Local Archives. IEEE Transactions on Systems, Man, and Cybernetics, Part B, Vol. 38, 2008, No. 5, pp. 1270–1293.

[18] Poli, R.: Analysis of the Publications on the Applications of Particle Swarm Optimisation. Journal of Artificial Evolution and Applications, 2008, pp. 1–10.

[19] Liu, H.—Abraham, A.—Wang, Z.: A Multi-Swarm Approach to Multi-Objective Flexible Job-Shop Scheduling Problems. Fundamenta Informaticae, 2009, 95, pp. 1–25.

[20] Kalivarapu, V.—Foo, J.—Winer, E.: Synchronous Parallelization of Particle Swarm Optimization With Digital Pheromones. Advances in Engineering Software, Vol. 40, 2009, No. 10, pp. 975–985.

[21] Al-Rashidi, M. R.—El-Hawary, M. E.: A Survey of Particle Swarm Optimization Applications in Electric Power Systems. IEEE Transactions on Evolutionary Computation, Vol. 13, 2009, No. 4, pp. 913–918.

[22] García-Villoria, A.—Pastor, R.: Introducing Dynamic Diversity into a Discrete Particle Swarm Optimization. Computers and Operations Research, Vol. 36, 2009, No. 3, pp. 951–966.

[23] Zielinski, K.—Weitkemper, P.—Laur, R.—Kammeyer, K. D.: Optimization of Power Allocation for Interference Cancellation With Particle Swarm Optimization. IEEE Transactions on Evolutionary Computation, Vol. 13, 2009, No. 1, pp. 128–150.

[24] Deep, K.—Bansal, J. C.: Mean Particle Swarm Optimisation for Function Optimisation. International Journal of Computational Intelligence Studies, Vol. 1, 2009, No. 1, pp. 72–92.

[25] Coelho, L. D. S.: Gaussian Quantum-Behaved Particle Swarm Optimization Approaches for Constrained Engineering Design Problems. Expert Systems with Applications, Vol. 37, 2010, No. 2, pp. 1676–1683.

[26] Babaoglu, I.—Findik, O.—Ilker, E.: A Comparison of Feature Selection Models Utilizing Binary Particle Swarm Optimization and Genetic Algorithm in Determining Coronary Artery Disease Using Support Vector Machine. Expert Systems with Applications, Vol. 37, 2010, No. 4, pp. 3177–3183.

[27] Zhao, X.: A Perturbed Particle Swarm Algorithm for Numerical Optimization. Applied Soft Computing, Vol. 10, 2010, No. 1, pp. 119–124.

[28] Bishop, M.—Nasuto, S.: Exploration Vs. Exploitation in Naturally Inspired Search. Proceedings of Adaptation in Artificial and Biological Systems, 2006.

[29] Čapkovič, F.: Modelling, Analysing and Control of Interactions Among Agents in MAS. Computing and Informatics, Vol. 26, 2007, No. 5, pp. 507–541.

[30] Li, X. C. Y.: A Modified Pso Structure Resulting in High Exploration Ability With Convergence Guaranteed. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 37, 2007, No. 5, pp. 1271–1289.

[31] Zhan, Z.—Zhang, J.—Li, Y.—Chung, H.: Adaptive Particle Swarm Optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 39, 2009, No. 6, pp. 1362–1381.

[32] Pedersen, M. E. H.—Chipperfield, A. J.: Simplifying Particle Swarm Optimization. Applied Soft Computing, Vol. 10, 2010, No. 2, pp. 618–628.

[33] Chen, T.—Chi, T. L.: On the Improvements of the Particle Swarm Optimization Algorithm. Advances in Engineering Software, Vol. 41, 2010, No. 2, pp. 229–239.

[34] Ozcan, E.—Mohan, C.: Particle Swarm Optimization: Surfing the Waves. Proceedings of the Congress on Evolutionary Computation, 1999, pp. 1939–1944.

[35] Trelea, I.: The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection. Information Processing Letters, Vol. 85, 2003, No. 6, pp. 317–325.

[36] Chen, J.—Pan, F.—Cai, T.—Tu, X.: Stability Analysis of Particle Swarm Optimization Without Lipschitz Constraint. Journal of Control Theory and Applications, Vol. 1, 2003, No. 1, pp. 86–90.

[37] Tan, Y.—Zeng, J.—Gao, H.: Analysis of Particle Swarm Optimization Based on Discrete Time Linear System Theory. Proceedings of the Fifth World Congress on Intelligent Control and Automation, 2004, pp. 2210–2213.

[38] Yasuda, K.—Iwasaki, N.: Analysis of the Dynamics of Particle Swarm Optimization. IEEJ Transactions on Electronics, Information and Systems, Vol. 125, 2005, No. 5, pp. 821–822.

[39] Van Den Bergh, F.—Engelbrecht, A.: A Study of Particle Swarm Optimization Particle Trajectories. Information Sciences, Vol. 176, 2006, No. 8, pp. 937–971.

[40] Kadirkamanathan, V.—Selvarajah, K.—Fleming, P.: Stability Analysis of the Particle Dynamics in Particle Swarm Optimizer. IEEE Transactions on Evolutionary Computation, Vol. 10, 2006, No. 3, pp. 245–255.

[41] Li, N.—Sun, D.—Zou, T.—Qin, Y.—Wei, Y.: Analysis for a Particle's Trajectory of Pso Based on Difference Equation. Chinese Journal of Computers, Vol. 29, 2006, No. 11, pp. 2052–2061.

[42] Jiang, M.—Luo, Y.—Yang, S.: Stochastic Convergence Analysis and Parameter Selection of the Standard Particle Swarm Optimization Algorithm. Information Processing Letters, Vol. 102, 2007, No. 1, pp. 8–16.

[43] Liu, H.—Abraham, A.—Clerc, M.: Chaotic Dynamic Characteristics in Swarm Intelligence. Applied Soft Computing Journal, Vol. 7, 2007, No. 3, pp. 1019–1026.

[44] Poli, R.—Broomhead, D.: Exact Analysis of the Sampling Distribution for the Canonical Particle Swarm Optimiser and Its Convergence During Stagnation. Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, 2007, pp. 134–141.

[45] Samal, N.—Konar, A.—Das, S.—Abraham, A.: A Closed Loop Stability Analysis and Parameter Selection of the Particle Swarm Optimization Dynamics for Faster Convergence. Proceedings of 2007 IEEE Congress on Evolutionary Computation, Washington 2007, pp. 1769–1776.

[46] Martínez, J.—Gonzalo, E.: The Generalized Pso: A New Door to Pso Evolution. Journal of Artificial Evolution and Applications, 2008, pp. 1–15.

[47] Rapaić, M.—Kanović, Z.: Time-Varying Pso-Convergence Analysis, Convergence-Related Parameterization and New Parameter Adjustment Schemes. Information Processing Letters, Vol. 109, 2009, No. 11, pp. 548–552.

[48] Liu, H.—Li, B.—Ji, Y.—Sun, T.: Particle Swarm Optimisation from Lbest to Gbest. Applied Soft Computing Technologies: The Challenge of Complexity, 2006, 34, pp. 537–545.

[49] Ghosh, S.—Kundu, D.—Suresh, K.—Das, S.—Abraham, A.—Panigrahi, B.—Snášel, V.: On Some Properties of the Lbest Topology in Particle Swarm Optimization. Proceedings of the 2009 Ninth International Conference on Hybrid Intelligent Systems, Washington, 2009, pp. 370–375.

[50] Olorunda, O.—Engelbrecht, A.: Measuring Exploration/Exploitation in Particle Swarms Using Swarm Diversity. Proceedings of IEEE Congress on Evolutionary Computation, Washington, IEEE Computer Society Press, 2008, pp. 1128–1134.

[51] Binkley, K.—Hagiwara, M.: Balancing Exploitation and Exploration in Particle Swarm Optimization: Velocity-Based Reinitialization. Information and Media Technologies, Vol. 3, 2008, No. 1, pp. 103–111.

[52] Chen, J.—Xin, B.—Peng, Z.—Dou, L.—Zhang, J.: Optimal Contraction Theorem for Exploration-Exploitation Tradeoff in Search and Optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part A, Vol. 39, 2009, No. 3, pp. 680–691.

[53] Wimmer, H.: Spectral Radius and Radius of Convergence. American Mathematical Monthly, Vol. 81, 1974, No. 6, pp. 625–627.

[54] Varga, R.: Matrix Iterative Analysis. Springer 2000.

[55] Lyapunov, A.: Stability of Motion. Academic Press 1966.

[56] Slotine, J.—Li, W.: Applied Nonlinear Control. Prentice-Hall - Englewood Cliffs 1991.

[57] Vidyasagar, M.: Nonlinear Systems Analysis. Society for Industrial Mathematics 2002.

[58] Rugh, W.: Linear System Theory. Prentice-Hall 1996.

Benxian Yue is a Ph. D. candidate in Dalian University of Technology, China. His research interests are in swarm intelligence, dynamic modeling, machine learning, and data mining. He has published several papers in the above areas.

Hongbo Liu is a Professor at School of Information Science and Technology at Dalian Maritime University, Dalian, China, with an affiliate appointment in the Computer Science and Biomedical Engineering Division at Dalian University of Technology. He received his Ph. D. degrees in computer science from Dalian University of Technology. His research interests are in system modeling and optimization involving soft computing, probabilistic model, nature-inspired computing, swarm intelligence, and other heuristics. Application areas include neuroinformatics, cognitive computing, computational grids, P2P computing, machine learning, data mining.

Ajith Abraham received the Ph. D. degree from Monash University, Melbourne, Australia in 2001. He is currently with the Center for Quantifiable Quality of Service in Communication Systems (Q2S), Center of Excellence, Norwegian University of Science and Technology (NTNU), Trondheim, Norway and Machine Intelligence Research Labs (MIR Labs). He has worldwide academic experience of nearly ten years with formal appointments in Monash University, Australia; Oklahoma State University; Chung- Ang University, Seoul; Jinan University, China; Rovira i Virgili University, Spain; Dalian Maritime University, China; Yonsei University, Seoul; Open University of Catalonia, Spain, and NTNU, Norway. For about two and a half years, he was working under the Institute of Information Technology Advancement Professorship Program funded by the South Korean Government. He was working with three multi-national companies: Keppel Engineering, Singapore; Hyundai Engineering, Korea; and Ashok Leyland Ltd, India where he was involved in different industrial research and development projects for nearly eight years. He has authored or co-authored more than 500 research publications in peer-reviewed reputed journals, book chapters, and conference proceedings. His primary research interests are in advanced computational intelligence, with a focus on using global optimization techniques for designing intelligent systems. Their application areas include Web services, information security, Web intelligence, social networks, financial modeling, multicriteria decision making, data mining, etc. He has given more than 25 plenary lectures and conference tutorials in the above areas. Dr. Abraham co-chairs the IEEE Systems Man and Cybernetics Society Technical Committee on Soft Computing. He is a regular reviewer of IEEE Intelligent Systems, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Neural Networks, IEEE Transactions on Fuzzy Systems, IEEE Transactions on Evolutionary Computation, IEEE Transactions on Systems, Man, And Cybernetics, and IEEE Transactions on Power Systems. He serves on the Editorial Boards of more than 30 international journals and has also guest-edited 30 special issues on various topics for international journals. Since 2001, he has been actively involved in the hybrid intelligent systems and the Intelligent Systems Design and Applications series of annual international conferences. He was the recipient of five Best Paper Awards.