
Chaos and Swarm Intelligence

Hongbo Liu¹ and Ajith Abraham²

¹ School of Computer Science and Engineering,
Dalian Maritime University, 116026 Dalian, China
lhb@dlut.edu.cn

<http://hongboliu.torrent.googlepages.com/>

² Norwegian Center of Excellence, Center of Excellence for Quantifiable Quality of
Service, Norwegian University of Science and Technology, Trondheim, Norway
ajith.abraham@ieee.org

<http://www.softcomputing.net/>

Summary. Swarm Intelligence (SI) is an innovative distributed intelligent paradigm whereby the collective behaviors of unsophisticated individuals interacting locally with their environment causing coherent functional global patterns to emerge. The intelligence emerges from a chaotic balance between individuality and sociality. The chaotic balances are a characteristic feature of the complex system. This chapter investigates the chaotic dynamic characteristics in swarm intelligence. The swarm intelligent model namely the Particle Swarm Optimization (PSO) algorithm is represented as an Iterated Function System (IFS). The dynamic trajectory of the particle is sensitive on the parameter values of IFS. The Lyapunov exponent and the correlation dimension are calculated and analyzed numerically for the dynamic system. Convergence of the swarm model is also analyzed. Our research findings illustrate that the 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. The research would be helpful to parameter selection and algorithm improvements for the swarm intelligence applications.

Keywords: Swarm Intelligence, Chaos, Particle Swarm, Iterated Function System.

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]. SI provides a framework to explore distributed problem solving without centralized control or the provision of a global model. The particle swarm model helps to find optimal regions of complex search spaces through interaction of individuals in a population of particles [4]. It has exhibited good performance across a wide range of applications [5, 6, 7, 8, 9, 10, 11].

In the swarm dynamic system, the intelligence emerges from a chaotic balance between individuality and sociality. The chaotic balances are a characteristic feature of the complex system. Bergh and Engelbrecht [34] overviewed current theoretical studies, and investigated particle trajectories for general swarms to include the influence of the inertia term. They also provided a formal proof that each particle converges to a stable point. Many studies on swarm intelligence have been presented and even some improved algorithms were proposed based on the chaotic search behavior. For a given energy or cost function, by following chaotic ergodic orbits [12], a chaotic dynamic system may eventually reach the global optimum or its good approximation with high probability. To enhance the performance of particle swarm optimization (one of the swarm intelligent models), Liu et al. [13] proposed hybrid particle swarm optimization algorithm by incorporating chaos. The proposed chaotic particle swarm optimization algorithm combined the population-based evolutionary search ability of particle swarm optimization and chaotic search behavior. Simulation results and comparisons with the standard particle swarm optimization and several other meta-heuristics illustrated that the approach could effectively enhance the search efficiency and significantly improve the search quality. Since chaotic mapping possesses properties of certainty, ergodicity and stochastic property, Jiang and Etorre [14, 15] introduced chaos mapping into the particle swarm optimization algorithm for reactive power optimization and short term hydroelectric system scheduling in a deregulated environment. Alatas et al. [16] embedded different chaotic maps to adapt the parameters of PSO algorithm. Empirical results demonstrated that the performance of the algorithms was improved obviously owing to its fast convergence and high precision.

However, not much work has been reported in the literature on the chaotic characteristics in swarm intelligence. In fact, several other studies in diverse fields indicated the analysis of the chaotic characteristics contributed to the understanding and applications of those complex systems. Chen [17] investigated the chaotic phenomena in macroeconomic systems, and offered an explanation of the multi-periodicity and irregularity in business cycles and of the low-dimensionality of chaotic monetary attractors. The empirical and theoretical results improved monetary control policy and the approaches to forecasting business cycles. Chialvo et al. [18] studied chaotic patterns of activation and action potential characteristics in the cardiac tissues. Empirical results indicated an apparent link between the mechanism of low dimensional chaos and the occurrence of reflected responses which could lead to more spatially disorganized phenomena. Frank et al. [19] analyzed the chaotic characteristics in the brain dynamics to predict changes of epileptic seizures. Goldberger et al. [20], Freeman [21] and Sarbadhikari and Chakrabarty [22] illustrated that chaos has a great important influence on brain and the evolutionary relationship between species. The investigations of chaotic dynamics in neural networks [23] promoted the development of neural networks and chaotic neural networks [24, 25]. The chaotic balances and their characteristic in swarm intelligence has become very

importance for its deeper understanding, application development and designing new computational models.

This chapter investigates the chaotic dynamic characteristics in swarm intelligence, and analyzes their relationship with the performance of SI. Particle swarm model is investigated as a case study. The swarm intelligent model is represented as an iterated function system (IFS) [26]. We simulate and analyze the dynamic trajectory of the particle based on the IFS. The Lyapunov exponent and the correlation dimension are calculated and analyzed numerically for the dynamic system. We attempt to discuss the relation between the algorithm convergence and the velocities of the particles. The dependence of the parameters is discussed analytically using several function optimization experiments.

The rest of the article is organized as follows. Particle swarm model is presented in Section 2. The concepts of iterative function system and its sensitivity is illustrated in Section 3. Dynamic chaotic characteristics are depicted and discussed in Section 4 and finally Conclusions are provided in Section 5.

2 Particle Swarm Model

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 \mathbf{x}_i (i is the index of the particle), and a velocity represented by a velocity-vector \mathbf{v}_i . Each particle remembers its own best position so far in a vector \mathbf{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 $\hat{\mathbf{p}}$, and its j -th dimensional value is \hat{p}_j . During the iteration time t , the update of the velocity from the previous velocity is determined by Eq. (1). Subsequently, the new position is determined by the sum of the previous position and the new velocity by Eq. (2).

$$v_{i,j}(t) = wv_{i,j}(t-1) + c_1r_1(p_{i,j}(t-1) - x_{i,j}(t-1)) + c_2r_2(\hat{p}_j(t-1) - x_{i,j}(t-1)) \quad (1)$$

$$x_{i,j}(t) = x_{i,j}(t-1) + v_{i,j}(t) \quad (2)$$

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 as the coefficient of the self-recognition component; c_2 is a positive constant termed as the coefficient of the social component. The variable w is the inertia factor, for which value is typically setup to vary linearly from 1 to 0 during the iterated processing. From Eq. (1), 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.) In order to guide the particles effectively in the search space, the maximum moving distance during one iteration is clamped in between the

maximum velocity $[-v_{max}, v_{max}]$ given in Eq. (3), and similarly for its moving range given in Eq. (4):

$$v_{i,j} = \text{sign}(v_{i,j})\min(|v_{i,j}|, v_{max}) \quad (3)$$

$$x_{i,j} = \text{sign}(x_{i,j})\min(|x_{i,j}|, x_{max}) \quad (4)$$

The value of v_{max} is $\rho \times s$, with $0.1 \leq \rho \leq 1.0$ and is usually chosen to be s , i.e. $\rho = 1$. The pseudo-code for particle-search is illustrated in Algorithm 1.

Algorithm 1. Particle Swarm Model

01. Initialize the size of the particle swarm n , and other parameters.
 02. Initialize the positions and the velocities for all the particles randomly.
 03. While (the end criterion is not met) do
 04. $t = t + 1$;
 05. Calculate the fitness value of each particle;
 06. $\hat{\mathbf{p}}(t) = \text{argmin}_{i=1}^n (f(\hat{\mathbf{p}}(t-1)), f(\mathbf{x}_1(t)), f(\mathbf{x}_2(t)), \dots, f(\mathbf{x}_i(t)), \dots, f(\mathbf{x}_n(t)))$;
 07. For $i=1$ to n
 08. $\mathbf{p}_i(t) = \text{argmin}_{i=1}^n (f(\mathbf{p}_i(t-1)), f(\mathbf{x}_i(t)))$;
 09. For $j = 1$ to d
 10. Update the j -th dimension value of \mathbf{x}_i and \mathbf{v}_i according to Eqs. (1), (3), (2), (4);
 11. Next j
 12. Next i
 13. End While.
-

3 Iterated Function System and Its Sensitivity

Clerc and Kennedy have stripped the particle swarm model down to a most simple form [27]. If the self-recognition component c_1 and the coefficient of the social-recognition component c_2 in the particle swarm model are combined into a single term c , i.e. $c = c_1 + c_2$, the best position \mathbf{p}_i can be redefined as follows:

$$\mathbf{p}_i \leftarrow \frac{(c_1 \mathbf{p}_i + c_2 \hat{\mathbf{p}})}{(c_1 + c_2)} \quad (5)$$

Then, the update of the particle's velocity is defined by:

$$\mathbf{v}_i(t) = \mathbf{v}_i(t-1) + c(\mathbf{p}_i - \mathbf{x}_i(t-1)) \quad (6)$$

The system can be simplified even further by using $\mathbf{y}_i(t-1)$ instead of $\mathbf{p}_i - \mathbf{x}_i(t-1)$. Thus, the reduced system is then:

$$\begin{cases} \mathbf{v}(t) = \mathbf{v}(t-1) + c\mathbf{y}(t-1) \\ \mathbf{y}(t) = -\mathbf{v}(t-1) + (1-c)\mathbf{y}(t-1) \end{cases}$$

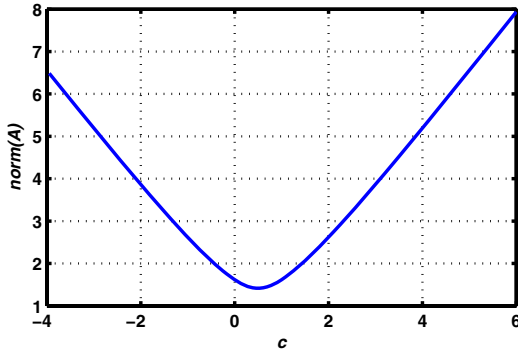


Fig. 1. Norm of A

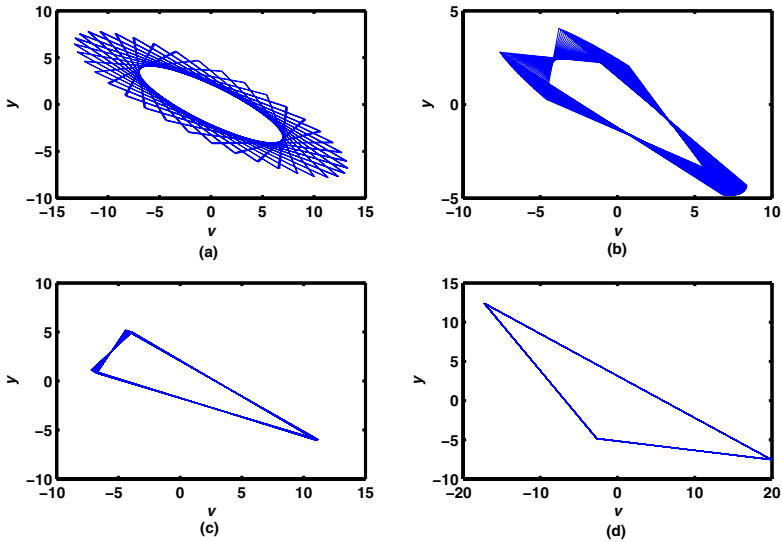


Fig. 2. Trajectory of the particle (a) $c = 2.9$, (b) $c = 2.999$, (c) $c = 2.999$, (d) $c = 2.9999$

This recurrence relation can be written as a matrix-vector product, so that

$$\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{y}(t) \end{bmatrix} = \begin{bmatrix} 1 & c \\ -1 & 1 - c \end{bmatrix} \cdot \begin{bmatrix} \mathbf{v}(t-1) \\ \mathbf{y}(t-1) \end{bmatrix}$$

Let

$$\mathbf{P}_t = \begin{bmatrix} \mathbf{v}_t \\ \mathbf{y}_t \end{bmatrix}$$

and

$$A = \begin{bmatrix} 1 & c \\ -1 & 1 - c \end{bmatrix}$$

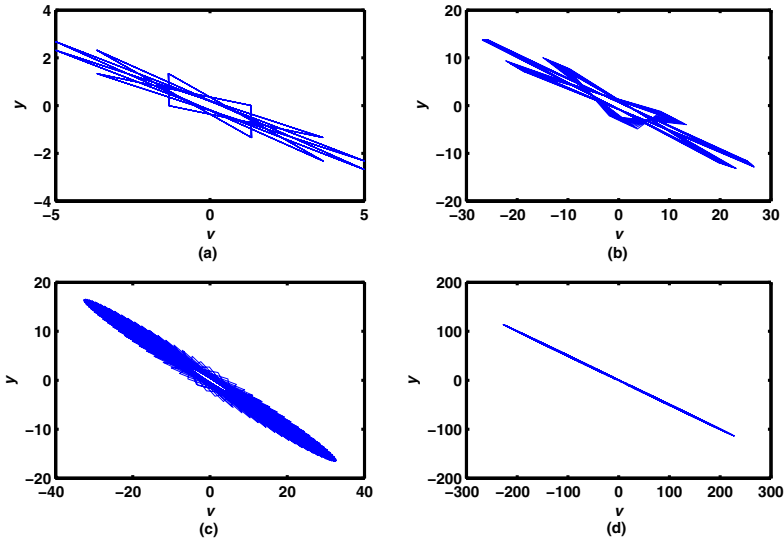


Fig. 3. Trajectory of the particle (a) $c = 3.7321$, (b) $c = 3.8$, (c) $c = 3.9$, (d) $c = 3.999$

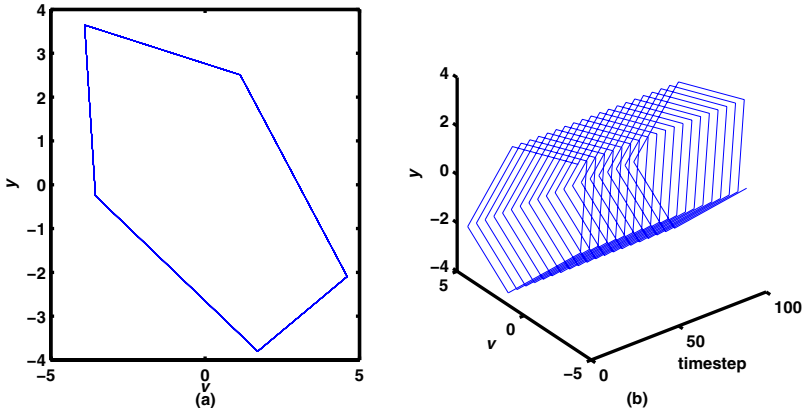


Fig. 4. 2D versus 3D (a) 2D: $c = 1.3820$, (b) 3D: $c = 1.3820$

we have an iterated function system for the particle swarm model:

$$\mathbf{P}_t = A \cdot \mathbf{P}_{t-1} \tag{7}$$

Thus, the system is completely defined by A . Its norm $\|A\|_2$ (also written $\|A\|$) is determined by c . The relationship of A and its dependence on c is illustrated in Figure 1.

IFS is sensitive to the values of c . It is possible to find different trajectories of the particle for various values of c . Figure 2(a) illustrates the system for a torus when $c=2.9$; Figure 2(b), a hexagon with spindle sides when $c=2.99$; Figure 2(c), a triangle with spindle sides when $c=2.999$; Figure 2(d), a simple triangle when

$c=2.9999$. As depicted in Figure 2, the iteration time step used is 100 for all the cases. Another system sensitivity instance is illustrated in Figure 3. It is to be noted that Figures 2 and 3 illustrate only some 2-dimensional representations of the iterated process. A comparison between 2D and 3D is illustrated in Figure 4.

4 Dynamic Chaotic Characteristics

Chaotic dynamics is defined by a deterministic system with non-regular, chaotic behavior [28]. They are both sensitive to initial conditions and computational unpredictability. The Lyapunov exponent and correlation dimension are most accessible in numerical computations based on the time-series of the dynamical system [29]. In this Section, we introduce the algorithm to compute the Lyapunov exponent and correlation dimension for quantitative observation of dynamic characteristics of the particles, and then analyze the relation between chaos and the swarm intelligent model.

4.1 Lyapunov Exponent

Lyapunov exponents provide a way to identify the qualitative dynamics of a system, because they describe the rate at which neighboring trajectories converge or diverge (if negative or positive, respectively) from one another in orthogonal directions. If the dynamics occur in an n -dimensional system, there are n exponents. Since the maximum exponent will dominate, this limit is practically useful only for finding the largest exponent. Chaos can be defined as the divergence between neighboring trajectories and the presence of a positive exponent could be considered as the diagnostic of chaos. For an IFS, Lyapunov exponents

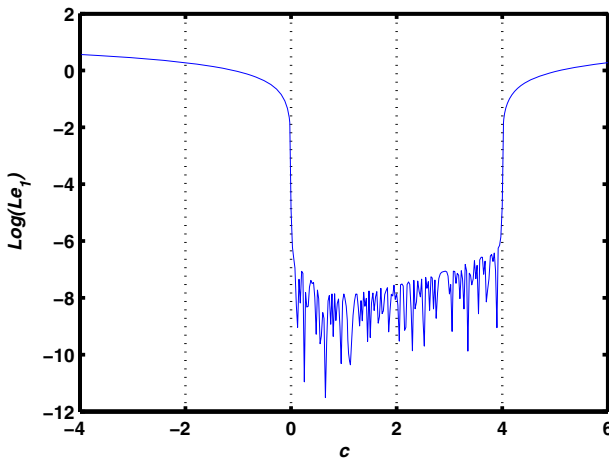


Fig. 5. Maximum Lyapunov exponent in PS

measure the asymptotic behavior of tangent vectors under iteration. The maximum Lyapunov exponent can be found using [30]:

$$Le_1 = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \ln\left(\frac{d_n}{d_1}\right) \quad (8)$$

where d_n is the distance between the n -th point-pair. Le_1 can be calculated using a programmable calculator to a reasonable degree of accuracy by choosing a suitably large value of “ N ”. Using the time series generated from the IFS Eq. (7), the maximum Lyapunov exponent Le_1 of the particle swarm model is calculated. The results are illustrated in Figure 5. The maximum Lyapunov exponent steadily increases with the value of c in the interval $[0.5, 4]$ and it bounds to reach a very high level when the value of c falls within the $[0, 4]$ interval.

4.2 Correlation Dimension

The dimension in a chaotic system is a measure of its geometric scaling property or its “complexity” and it has been considered as one of the most basic properties. Numerous methods have been proposed for characterizing the dimension produced by chaotic flows and one of the most common metrics is the correlation dimension, popularized by Grassberger and Procaccia [31]. It measures the probability that two points chosen at random will be within a certain distance of each other, and examines how this probability changes as the distance is increased. During the past decades, several investigators have undertaken nonlinear analysis using Grassberger and Procaccia’s algorithm (GP algorithm) to evaluate the correlation dimension of time-series data [32, 33].

Given by N points $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ from the iterated processes of IFS, the definition of the correlation integral is

$$C(r) = \lim_{N \rightarrow \infty} \frac{1}{N^2} \sum_{\substack{i,j=1 \\ i \neq j}}^N H(r - |\mathbf{x}_i - \mathbf{x}_j|) \quad (9)$$

where $H(x)$ is the Heaviside step function. When the limit exists, the correlation dimension is then defined as Eq. (10):

$$D_2 = \lim_{r,r' \rightarrow +0} \frac{\ln(C(r)/C(r'))}{\ln(r/r')} \quad (10)$$

In practice, $C(r)$ is calculated for several values of r and then a plot is constructed for $\ln C(r)$ versus $\ln(r)$ to estimate the slope, which then approximates the correlation dimension D_2 . In the particle swarm model, for $c = 3.9$, the slope, i.e. D_2 is illustrated in Figure 6 in the interval $[0, 4]$. The correlation dimension is depicted in Figure 7. There are no obvious differences for c values increasing within the interval of $[0, 4]$. D_2 is fluctuating mainly within 1 ± 0.2

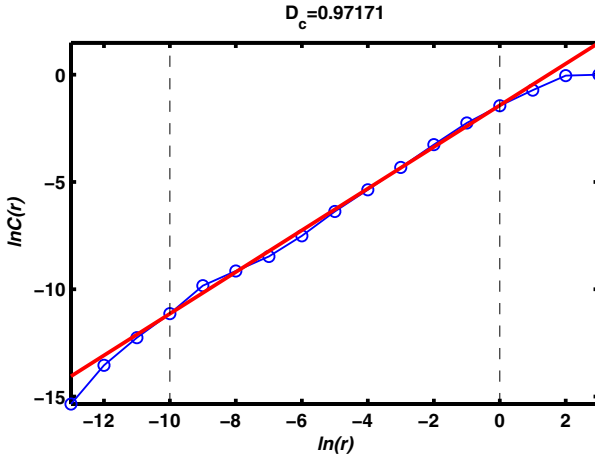


Fig. 6. Plot of $\ln C(r)$ vs. $\ln(r)$ for $c = 3.9$

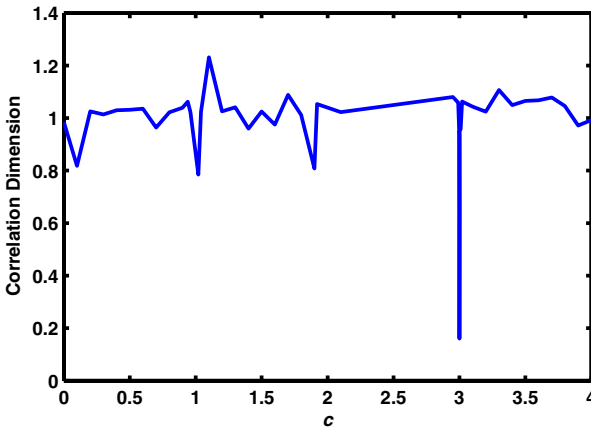


Fig. 7. Correlation dimension for varying values of c

and it is to be noted that the correlation dimension is very small when c is close to 3. Our experiment results also further validates the constant constriction coefficient w using the “classical” value $w = 1/(2 * \ln(2)) = 0.7213$ and the recommended value for $c = (w + 1)^2 = 2.9630$ [27].

For the iterated system determined by Eq. (7), the eigenvalues of A are λ_1 and λ_2 . We are looking for pair of values (c, k) so that

$$A^k = I \tag{11}$$

where I is the identity matrix. We have $\det(A) > 0$ (equal to 1, in fact), so it exists P so that

$$P^{-1}AP = \Lambda \tag{12}$$

where

$$A = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

Eq. (11) can then be rewritten

$$(PAP^{-1})^k = A^k = I \quad (13)$$

It means we must have

$$\lambda_1^k = \lambda_2^k = 1 \quad (14)$$

But we have

$$\begin{cases} \lambda_1 = 1 - \frac{c}{2} + \sqrt{\Delta} \\ \lambda_2 = 1 - \frac{c}{2} - \sqrt{\Delta} \end{cases} \quad (15)$$

with c is strictly positive, and

$$\Delta = \left(1 - \frac{c}{2}\right)^2 - 1$$

So it is possible for Eq. (14) only if the eigenvalues are true complex numbers, i.e. if Δ is strictly negative. It implies that c must be smaller than 4. It is easy to see that we have $|\lambda_1| = |\lambda_2| = 1$. So finally the only condition to have a perfect cycle of size k :

$$1 - \frac{c}{2} = \cos\left(\frac{2\pi}{k}\right) \quad (16)$$

i.e.

$$c = 2\left(1 - \cos\left(\frac{2\pi}{k}\right)\right) \quad (17)$$

There are an infinity of such cycles for small c values (smaller than 1), but in $[1, 4[$ the only possible ones are for cycle sizes $k = 6, 5, 4, 3$, i.e. $c = 1, 1.382, 2, 3$. It means in particular that if we generate a sequence of points in the particle swarm model by using one of these c values, the correlation dimension will be very small. On the contrary for other values we obtain a correlation dimension near to 1, which is the value for pure random distribution.

4.3 Convergence Analysis

We consider further the iterated system determined by Eq. (7) and the eigenvalues of A are λ_1 and λ_2 . Without loss of generality and to simplify the notation, the derivation is performed in one dimension only, using a single particle [34]. The explicit form of the recurrence relations in Eqs. (1) and (2) is then given by

$$x(t) = k_0 + k_1\lambda_1^t + k_2\lambda_2^t \quad (18)$$

A similar kind of expression for $v(t)$ is now produced:

$$v(t) = h_1\lambda_1^t + h_2\lambda_2^t \quad (19)$$

where k_0, k_1, k_2, h_1 and h_2 are constants determined by the initial conditions of the system. For the initial conditions,

$$\begin{aligned}
 k_0 &= \frac{c_1 r_1 p + c_2 r_2 \hat{p}}{c_1 r_1 + c_2 r_2} \\
 k_1 &= \frac{\lambda_2(x(0) - x(1)) - x(1) + x(2)}{\sqrt{\Delta}(\lambda_1 - 1)} \\
 k_2 &= \frac{\lambda_1(x(1) - x(0)) + x(1) - x(2)}{\sqrt{\Delta}(\lambda_2 - 1)} \\
 h_1 &= \frac{v(1) - \lambda_2 v(0)}{\lambda_1 - \lambda_2} \\
 h_2 &= \frac{\lambda_2 v(0) - v(1)}{\lambda_1 - \lambda_2}
 \end{aligned}$$

Since $|\lambda_1| = |\lambda_2| = 1$, we consider the value of $x(t)$ from Eq. (18) in the limit, thus:

$$\begin{aligned}
 \lim_{t \rightarrow +\infty} x(t) &= \lim_{t \rightarrow +\infty} (k_0 + k_1 \lambda_1^t + k_2 \lambda_2^t) \\
 &= k_0 + k_1 \lim_{t \rightarrow +\infty} \lambda_1^t + k_2 \lim_{t \rightarrow +\infty} \lambda_2^t \\
 &= k_0 + k_1 + k_2.
 \end{aligned}$$

Accordingly we consider the value of $v(t)$ from Eq. (19) in the limit as follows:

$$\begin{aligned}
 \lim_{t \rightarrow +\infty} v(t) &= \lim_{t \rightarrow +\infty} (h_1 \lambda_1^t + h_2 \lambda_2^t) \\
 &= h_1 \lim_{t \rightarrow +\infty} \lambda_1^t + h_2 \lim_{t \rightarrow +\infty} \lambda_2^t \\
 &= h_1 + h_2 \\
 &= 0.
 \end{aligned}$$

The results of the limits indicate the trajectory of the particle $x(t)$ will converge to a stable point and the velocity of the particle $v(t)$ gradually weakens until it stops. The gradual change of the particle’s velocity can be explained geometrically. During each iteration, the particle is attracted towards the location of the best fitness achieved so far by the particle itself and by the location of the best fitness achieved so far across the whole swarm. From Eq. (1), $v_{i,j}$ can attain a smaller value, but if the second term and the third term in the RHS of Eq. (1) are both small, it cannot resume a larger value and could eventually lose the exploration capabilities in the future iterations. Such situations could occur even in the early stages of the search. When the second term and the third term in the RHS of Eq. (1) are zero, $v_{i,j}$ will be damped quickly with the ratio of w . In other words, if a particle’s current position coincides with the global best position/particle, the particle will only move away from this point if its previous velocity and w are non-zero. If their previous velocities are very close to zero, then all the particles will stop moving once they catch up with the global best particle, which many lead to

premature convergence. In fact, this does not even guarantee that the algorithm has converged to a local minimum in some complex or dynamic environments, and it merely means that all the particles have converged to the best position discovered so far by the swarm. This state owes to the second term and the third term in the RHS of Eq. (1), the cognitive components of the PSO. But if the cognitive components of the PSO algorithm are invalidated, all particles always search the solutions using the initial velocities. Then the algorithm is merely a degenerative stochastic search without the characteristics of PSO.

Due to the limited space, we do not demonstrate the improvements to the performance of swarm intelligence in this chapter. Reader may consult CPSO, *Cw*PSO and CEPSO algorithms for the technical details. Liu, et al. [13] proposed hybrid particle swarm optimization algorithm by incorporating chaos. In *Cw*PSO algorithm [14], the objective was to introduce chaos theory so that chaotic mapping enjoys certainty, ergodicity and stochastic property to improve the exploration ability of the algorithm. Simulation results and comparisons with the standard particle swarm optimization and several other meta-heuristics have shown that the approach could effectively enhance the search efficiency and greatly improve the searching quality. The trajectory of particles was given a lot of importance rather than their velocities. The velocity could be driven by chaotic dynamic. It is also an ideal improvement approach, since the chaotic characteristics influence indirectly and softly the position/solution. Alatas, et al. [16] embedded different chaotic maps to adapt the parameters of PSO algorithm. Twelve chaos-embedded PSO methods have been proposed and eight chaotic maps have been analyzed in the benchmark functions. It have been detected that coupling emergent results in different areas, like those of PSO and complex dynamics, can improve the quality of results in some optimization problems and also that chaos may be a desired process. It has been also shown that, these methods have somewhat increased the solution quality, that is in some cases they improved the global searching capability by escaping the local solutions.

4.4 Model Performance Demonstration

In order to analyze the relationship between chaos and the swarm intelligent model, we optimized three unconstrained real-valued benchmark functions, and then investigated the performance of the model against the dynamic chaotic characteristics.

First, we considered the Rastrigin's function (f_1), given by Eq. (20). It is a continuous, multimodal function with multiple local minima. The function has a "large scale" curvature which guides the search towards the global minimum, $\mathbf{x}^* = (0, \dots, 0)$, with $f(\mathbf{x}^*) = 0$ in the interval $[-5.12, 5.12]$.

Next, we considered the Zakharov's function (f_2), given by Eq. (21). It is a continuous, multimodal function, and has the minimum, $\mathbf{x}^* = (0, \dots, 0)$, with $f(\mathbf{x}^*) = 0$ in the interval $[-10, 10]$.

Finally, we also evaluated the Levy's function (f_3), given by Eq. (22). It is a continuous, multimodal function with an offset, since it has the minimum, $\mathbf{x}^* = (1, \dots, 1)$, with $f(\mathbf{x}^*) = 0$ in the interval $[-10, 10]$.

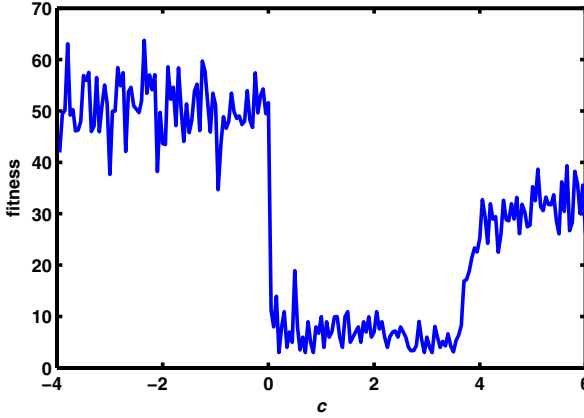


Fig. 8. The performance for 5-D Rastrigrin's function

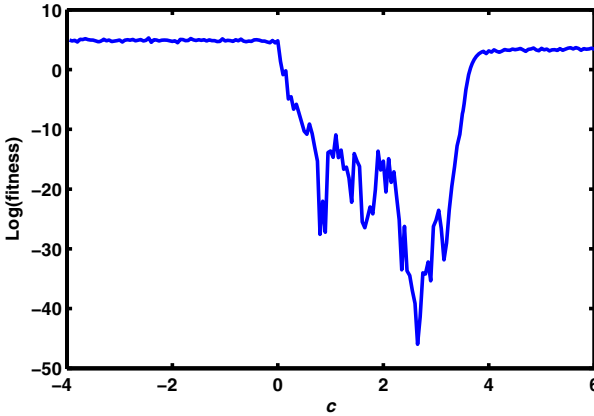


Fig. 9. The performance for 5-D Zakharov's function

$$f_1(\mathbf{x}) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10] \tag{20}$$

$$f_2(\mathbf{x}) = \sum_{i=1}^n x_i^2 + \left(\sum_{i=1}^n \frac{1}{2} i x_i \right)^2 + \left(\sum_{i=1}^n \frac{1}{2} i x_i \right)^4 \tag{21}$$

$$f_3(\mathbf{x}) = \frac{\pi}{n} \left(k \sin^2(\pi y_1) + \sum_{i=1}^{n-1} ((y_i - a)^2 (1 + k \sin^2(\pi y_{i+1}))) + (y_n - a)^2 \right);$$

$$y_i = 1 + \frac{1}{4}(x_i - 1); \quad k = 10; \quad a = 1. \tag{22}$$

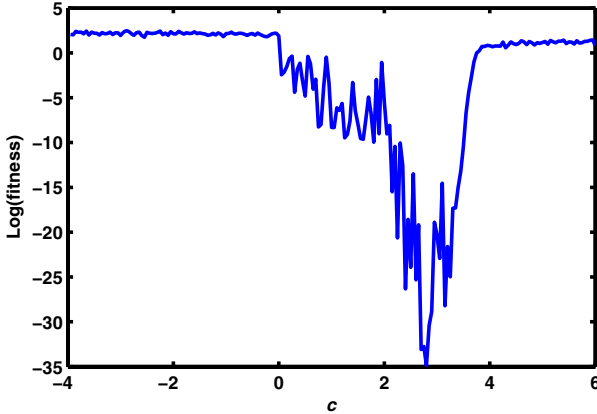


Fig. 10. The performance for 5- D Levy's function

The goal of the particle swarm algorithm is to find the global minimum for functions (20) to (22). All experiments for the functions were run 10 times, and the average fitness were recorded. The swarm size was set at 20, and 1000 iterations for the trials. The results are illustrated in Figure 8 for Rastrigin's function, Figure 9 for Zakharov's function and Figure 10 for Levy's function, respectively. It is obvious that the values of c within the interval $[0, 4]$ is fit for the model because the performance is much better than the other values of c . It is consistent with the Lyapunov exponent and the correlation dimension of the model as illustrated in Figures 5 and 7. In the interval $[0, 4]$, the particle swarm optimization algorithm with a high maximum Lyapunov exponent usually achieved better performance. The positive Lyapunov exponent describes the rate at which neighboring trajectories diverge. A high Lyapunov exponent in the particle swarm system implies that the particles are inclined to explore different regions and find better fitness values. But the big Lyapunov exponent would lead the system not to converge. The particles usually have to search solutions randomly because of the clamping of velocity and position. Compared to the correlation dimension of the system, the performance of the system is better when the c is close to 3.

5 Conclusion

In this chapter, we focused on the chaotic dynamic characteristics in swarm intelligence. Particle swarm was investigated as a case and the swarm model was represented by the Iterated Function System (IFS). The dynamic trajectory of the particle was sensitive on the value of the IFS parameters and the sensitivity of the system is illustrated. We introduced the algorithms to compute numerically the Lyapunov exponent and correlation dimension for quantitative observation of dynamic characteristics of the particles, convergence analysis, and then analyzed the dependence of the parameters using some function optimization experiments. The results illustrated that the performance of the swarm

intelligent model depended on the sign of the maximum Lyapunov exponent. The particle swarm optimization algorithm with a little high maximum Lyapunov exponent usually achieved better performance, especially for the multimodal functions. The correlation dimension of the system could recommend some values for the parameters.

Since the performance of the swarm intelligent model usually depends relatively on its Lyapunov exponent and correlation dimension, it would provide some new ideas for developing new swarm intelligent models. If we can design some models with a little higher maximum Lyapunov exponent, it might be possible to construct a new swarm intelligence model with better performance. The correlation dimension of the system would provide some suggestions for the parameter selection.

Acknowledgment

The authors would like to thank Maurice Clerc and Bo Li for their scientific collaboration in this research work. This work is supported partly by NSFC (60373095), DLMU (DLMU-ZL-200709).

References

1. Kennedy, J., Eberhart, R.: *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco (2001)
2. Bonabeau, E., Dorigo, M., Theraulaz, G.: *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York (1999)
3. Holland, J.H.: *Emergence: From Chaos to Order*. Addison-Wesley, Redwood City (1998)
4. Clerc, M.: *Particle Swarm Optimization*. ISTE Publishing Company, London (2006)
5. Parsopoulos, K.E., Vrahatis, M.N.: Recent Approaches to Global Optimization Problems through Particle Swarm Optimization. *Natural Computing* 1, 235–306 (2002)
6. Ursema, R.K., Vadstrup, P.: Parameter Identification of Induction Motors Using Stochastic Optimization Algorithms. *Applied Soft Computing* 4, 49–64 (2004)
7. Sousa, T., Silva, A., Neves, A.: Particle Swarm Based Data Mining Algorithms for Classification Tasks. *Parallel Computing* 30, 767–783 (2004)
8. Chang, B., Ratnaweera, A., Halgamuge, S.: Particle Swarm Optimisation for Protein Motif Discovery. *Genetic Programming and Evolvable Machines*, 5203–5214 (2004)
9. Du, F., Shi, W.K., Chen, L.Z., Deng, Y., Zhu, Z.F.: Infrared Image Segmentation with 2-D Maximum Entropy Method Based on Particle Swarm Optimization. *Pattern Recognition Letters* 26, 597–603 (2005)
10. Schute, J.F., Groenwold, A.A.: A Study of Global Optimization Using Particle Swarms. *Journal of Global Optimization* 31, 93–108 (2005)
11. Abraham, A., Guo, H., Liu, H.: Swarm intelligence: foundations, perspectives and applications. In: *Swarm Intelligent Systems. Studies in Computational Intelligence*, pp. 3–25 (2006)
12. Eckmann, J.-P., Ruelle, D.: Ergodic Theory of Chaos and Strange Attractors. *Reviews of Modern Physics* 57, 617 (1985)

13. Liu, B., Wang, L., Jin, Y., Tang, F., Huang, D.: Improved Particle Swarm Optimization Combined with Chaos. *Chaos, Solitons and Fractals* 25, 1261–1271 (2005)
14. Jiang, C., Etorre, B.: A hybrid Method of Chaotic Particle Swarm Optimization and Linear Interior for Reactive Power Optimisation. *Mathematics and Computers in Simulation* 68, 57–65 (2005)
15. Jiang, C., Etorre, B.: A Self-adaptive Chaotic Particle Swarm Algorithm for Short Term Hydroelectric System Scheduling in Deregulated Environment. *Energy Conversion and Management* 46, 2689–2696 (2005)
16. Alatas, B., Akin, E., Bedri Ozer, A.B.: Chaos Embedded Particle Swarm Optimization Algorithms. *Chaos, Solitons & Fractals* (2008), doi:10.1016/j.chaos.2007.09.063
17. Chen, P.: Empirical and Theoretical Evidence of Economic Chaos. *System Dynamics Review* 4, 81 (1988)
18. Chialvo, D.R., Gilmour Jr., R.F., Jalife, J.: Low Dimensional Chaos in Cardiac Tissue. *Nature* 342, 653–657 (1990)
19. Frank, G.W., Lookman, T., Nerenberg, M.A.H., Essex, C., Lemieux, J., Blume, W.: Chaotic Time Series Analysis of Epileptic Seizures. *Physica D* 46, 427 (1990)
20. Goldberger, A.L., West, B.J., Rigney, D.R.: Chaos and Fractals in Human Physiology. *Scientific American* 262, 42–49 (1990)
21. Freeman, W.J.: Brain dynamics: brain chaos and intentionality. In: Gordon, E. (ed.) *Integrative Neuroscience— Bringing Together Biological, Psychological and Clinical Models of the Human Brain*, pp. 163–171. Harwood Academic Publishers, Sydney (2000)
22. Sarbadhikari, S.N., Chakrabarty, K.: Chaos in the Brain: A Short Review Alluding to Epilepsy, Depression, Exercise And Lateralization. *Medical Engineering and Physics* 23, 445–455 (2001)
23. Bauer, M., Martienssen, W.: Lyapunov Exponents And Dimensions Of Chaotic Neural Networks. *Journal of Physics A: Mathematical and General* 24, 4557–4566 (1991)
24. Aihara, K., Takabe, T., Toyoda, M.: Chaotic Neural Networks. *Physical Letter A* 144, 333–340 (1990)
25. Dudul, S.V.: Prediction of A Lorenz Chaotic Attractor Using Two-layer Perceptron Neural Network. *Applied Soft Computing* 5, 333–355 (2005)
26. Sebastian, B., Pascal, H.: Logic Programs, Iterated Function Systems, and Recurrent Radial Basis Function Networks. *Journal of Applied Logic* 2, 273–300 (2004)
27. Clerc, M., Kennedy, J.: The Particle Swarm-explosion, Stability, and Convergence in A Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation* 6, 58–73 (2002)
28. Mullin, T.: *The Nature of Chaos*. Claredon Press, Oxford (1993)
29. Mosekilde, E.: *Topics in Nonlinear Dynamics*. World Science, London (1996)
30. Wolf, A., Swift, J.B., Swinney, H.L., Vastano, J.A.: Determining Lyapunov Exponents from A Time Series. *Physica D* 16, 285 (1985)
31. Grassberger, P., Procaccia, I.: Characterization of Strange Attractors. *Physical Review Letters* 50, 346–349 (1983)
32. Stefanovska, A., Strle, S., Kroselj, P.: On the Overstimation of the Correlation Dimension. *Physics Letters A* 235, 24–30 (1997)
33. Chlouverakis, K.E., Sprott, J.C.: A Comparison of Correlation and Lyapunov Dimensions. *Physica D* 200, 156–164 (2005)
34. van den Bergh, F., Engelbrecht, A.P.: A Study of Particle Swarm Optimization Particle Trajectories. *Information Sciences* 176(8), 937–971 (2006)