Chaos, Solitons and Fractals 42 (2009) 2450–2463

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

journal homepage: www.elsevier.com/locate/chaos

Detecting unstable periodic orbits of nonlinear mappings by a novel quantum-behaved particle swarm optimization non-Lyapunov way ☆

Fei Gao [a,b,*], Hongrui Gao [a], Zhuoqiu Li [a], Hengqing Tong [a], Ju-Jang Lee [b]

[a] Department of Mathematics, School of Science, Wuhan University of Technology, Luoshi Road 122, Wuhan, Hubei 430070, People's Republic of China
[b] Department of Electrical Engineering and Computer Science, Korean Advanced Institute of Science and Technology, 335 Gwahangno (373-1 Guseong-dong), Yuseong-gu, Daejeon 305-701, Republic of Korea

ARTICLE INFO

Article history:
Accepted 30 March 2009

ABSTRACT

It is well known that set of unstable periodic orbits (UPOs) can be thought of as the skeleton for the dynamics. However, detecting UPOs of nonlinear map is one of the most challenging problems of nonlinear science in both numerical computations and experimental measures. In this paper, a new method is proposed to detect the UPOs in a non-Lyapunov way. Firstly three special techniques are added to quantum-behaved particle swarm optimization (QPSO), a novel mbest particle, contracting the searching space self-adaptively and boundaries restriction (NCB), then the new method NCB–QPSO is proposed. It can maintain an effective search mechanism with fine equilibrium between exploitation and exploration. Secondly, the problems of detecting the UPOs are converted into a non-negative functions' minimization through a proper translation in a non-Lyapunov way. Thirdly the simulations to 6 benchmark optimization problems and different high order UPOs of 5 classic nonlinear maps are done by the proposed method. And the results show that NCB–QPSO is a successful method in detecting the UPOs, and it has the advantages of fast convergence, high precision and robustness.

1. Introduction

It is well known that periodic orbits play a vital important role in understanding the rich structures in a dynamical system, which was originally recognized by Poincaré and by many founders of modern dynamical system theory decades later [1].

And the set of unstable periodic orbits (UPOs) can be thought of as the skeleton for the dynamics. Actually one of the common characterizations of chaos is the positivity of the topological entropy, which is related to the exponential growth rate of the number of UPOs embedded within the attractor as one enumerates the orbits by their lengths [2,3]. UPOs are also important tools in affecting the behaviors of dynamical systems. Furthermore, many dynamical averages, for instances, the natural measure, the Lyapunov exponents, the fractal dimensions and etc., can be efficiently expressed in terms of a sum over the unstable periodic orbits [3–5].

However, detecting UPOs of nonlinear mapping is one of the most challenging problems of nonlinear science in both numerical computations and experimental measures because UPOs' inner unstable nature and the analytic expressions

☆ Supported by the National Science Foundation for Post-doctoral Scientists of China No. 20080431004, the NSFC No. 10647141, 60773210 from China, and the Brain Korea(BK)21 project from Korean government.

* Corresponding author. Address: Department of Mathematics, School of Science, Wuhan University of Technology, Luoshi Road 122, Wuhan, Hubei 430070, People's Republic of China. Tel./fax: +86 27 87651213.

E-mail addresses: hgaofei@gmail.com (F. Gao), gaohr-2009@163.com (H. Gao), science@whut.edu.cn (Z. Li), tonghengqing@126.com (H. Tong), jjlee@ee.kaist.ac.kr (J.-J. Lee).

URLs: http://hgaofei.googlepages.com/index.htm (F. Gao), http://public.whut.edu.cn/slx (H. Tong), http://iliad.kaist.ac.kr (J.-J. Lee).

0960-0779/$ - see front matter
doi:10.1016/j.chaos.2009.03.119

for evaluating periodic orbits can be obtained only if the chaos system is the nonlinear polynomial of low degree and the period is low.

In most experimental situations, a time series data which is usually the only available information from a dynamical system to determine the positions of UPOs [6–8]. Schmelcher and Diakonos have put an excellent algorithm (SD) [6], and as the simulations showed that it can be applied to detect higher periodic orbit and moreover it is globally converged. In addition, one modified version of SD has also been given to improve its converging rate [7]. Hu and etc. presented an efficient algorithm for computing UPOs. It can be applied to any finite dimensional chaotic systems including hyperchaotic systems [9].

To further analyze the system, one typically needs to dynamically reconstruct its phase space [10,11]. A basic scheme in detecting UPOs from this reconstructed data set is to look for recurrences of the iterated reconstructed map [4]. The sensitivity of this method in finding UPOs will naturally depend on the natural measure of the UPOs. An enhancement of the standard recurrence methods was proposed later [12].

There are also other more recent UPOs detection methods. Among them, there are a couple of notable ones: One is an adaptive control-based detection method proposed by Christini and Kaplan [13].

Unlike the above Lyapunov methods, the other method is a totally new one, through a swarm intelligence [14] non-Lyapunov way. It can succeed even in the case that the nonexistence of derivatives or poorly behaved partial derivatives in the neighborhood of the fixed points. But the results are not so satisfied and need to be progressed.

Consider the nonlinear map as below:

$$\Phi(x) = \begin{pmatrix} \Phi_1(x) \\ \vdots \\ \Phi_n(x) \end{pmatrix} : \Re^n \to \Re^n \tag{1}$$

where $x \in \Re, \Phi_i(x) : \Re^n \to \Re (i = 1, \ldots, n)$.

Let $\Phi^{(p)}(x) \triangleq \underbrace{\Phi(\Phi(\cdots\Phi(x)\cdots))}_{p \text{ times}}$, then the point $x \in \Re^n \quad \text{s.t.} \quad x = \Phi^{(p)}(x)$ is defined as a periodic point of period p of Φ.

Thus the detection of periodic orbits requires solving the system $x = \Phi^{(p)}(x)$, in Ref. [14,15] an optimization problem are defined by considering the objective function

$$G(x) = \left\| x - \Phi^{(p)}(x) \right\| = \|\xi_1, \ldots, \xi_n\| \tag{2}$$

where $\xi_i = x_i - \Phi_i^{(p)}(x),\ (i = 1, \ldots, n)$, $\|\cdot\|$ can be l_1, l_2, l_∞-norms arbitrarily.

Through the minimization of the nonnegative objective function (2), the detecting the global minimizers of the objective function is equivalent to computing the periodic orbits of period p [14,15].

Particle swarm intelligence (PSO) is a relatively new computational e1intelligence tool relation to Evolutionary Computation [16,17], developed by Dr. Eberhart and Dr. Kennedy in 1995 [18,19]. However, unlike GA, PSO does not have genetic operators, such as crossover and mutation. The dynamics of population in PSO resembles the collective behavior and self-organization of socially intelligent organisms. With the idea leading the population into the most potential district by the best individual, inspired by the PSO and quantum mechanics [20,21], a quantum-behaved PSO(QPSO) [22] has been put recently.

In this paper, with three novel techniques added to the QPSO in order to maintain an effective search mechanism with fine equilibrium between exploitation and exploration, a novel mbest particle, contracting the searching space self-adaptively and boundaries restriction (NCB), a NCB–QPSO is proposed for the nonlinear optimization problems. And we apply the put methods to detect the different UPOs at high order of chaotic systems. As the simulation results done compare to the QPSO show that the NCB–QPSO is a robust method.

The rest of this paper is organized as follows. In Section 2, the details of the classical and quantum-behaved PSO are introduced. In Section 3 the NCB–QPSO is given in details with the three new techniques, and simulations to 6 classical benchmark optimization problems are given too. In Section 4, firstly 5 famous nonlinear mappings are introduced, then simulations comparisons are done to detect the different UPOs at different high orders through QPSO and NCB–QPSO respectively, lastly the results are analyzed. And Section 5 summarizes the paper and discusses the future work.

2. Classical and quantum-behaved particle swarm optimization

2.1. Classical particle swarm optimization

Particle Swarm Optimization (PSO) belongs to the category of Swarm Intelligence methods closely related to the methods of Evolutionary Computation [14,16–19,23]. The PSO system is initialized with a population of random solutions and searches for optima by updating generations.

The procedure for implementing the global version of PSO is given as following [18,19].

At step k, Each particle $X_i(k) = (x_{i,1}(k), \ldots, x_{i,D}(k))$ $(i = 1, 2, \cdots M)$ keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved (self best) so far. (The fitness value is also stored.) This value is called pbest $P_i(k) = (p_{i,1}(k), \ldots, p_{i,D}(k))$. Another"best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbors of the particle, called lbest $L_i(t) = (l_{i,1}(t), \ldots, l_{i,D}(t))$. when a particle takes all the population as its topological neighbors, the best value is a global best and is called gbest $G(k) = ((g_1(k), \ldots, g_D(k))$.

The PSO concept consists of changing the velocity $V_i(k) = (v_{i,1}(k), \ldots, v_{i,D}(k))$ of each particle toward its pbest and gbest locations (PSO without neighborhood model), at each time step. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward pbest and gbest locations.

$$\begin{cases} A_{i,d}(k) = rand\ (0, c_1) \cdot [p_{i,d}(k) - x_{i,d}(k)] \\ B_{i,d}(k) = rand\ (0, c_2) \cdot [g_d(k) - x_{i,d}(k)] \\ v_{i,d}(k+1) = \chi[w \cdot v_{i,d}(k) + A_{i,d}(k) + B_{i,d}(k)] \\ x_{i,d}(k+1) = x_{i,d}(k) + v_{i,d}(k+1) \end{cases} \tag{3}$$

where w is called inertia weight, c_1, c_2 are cognition and social acceleration constant respectively, χ is constriction factor, normally $\chi = 0.9$. c_1 determines the effect of the distance between the current position of the particle and its best previous position P_i on its velocity. c_2 plays a similar role but it concerns the global best previous position $G(k)$, attained by any particle in the neighborhood. $rand(a, b)$ denotes random in $[a, b]$, in this way, the randomicity is introduced to PSO.

2.2. Quantum-behaved particle swarm optimization

Though PSO converges fast, sometimes it relapses into local optimum easily. Inspired by the classical PSO method and quantum mechanics, an improved edition [22] of PSO, that is QPSO, with the ideas from quantum computing [21] is also proposed to ameliorate convergence through maintaining more attractors.

For each particle $X_i(k)$, let

$$\begin{cases} \varphi_1 = rand\ (0, 1) \\ T_i(k) = \varphi_1 \cdot P_i(k) + (1 - \varphi_1) \cdot G(k) \end{cases} \tag{4}$$

Using the idea from the center-of-mass position, a new particle $S(k)$ for all $X_i(k)$ as mbest is defined

$$S(k) = \sum_{i=1}^{M} \frac{P_i}{M} \tag{5}$$

And the j-th dimension $x_{ij}(k+1)$ of the particle $X_i(k+1)$ is updated as following

$$\begin{cases} W_{ij}(k) = -\beta \cdot |S_{ij}(k) - X_{ij}(k)| \cdot \ln(1/u) \\ x_{ij}(k+1) = T_{ij}(k) + W_{ij}(k)(-1)^{round(t)} \end{cases} \tag{6}$$

where β is the extended coefficient which decreases linearly in interval $[1.3, 0.6]$, u, t are random in $(0, 1)$, $round(t)$ rounds the elements of t to the nearest integers.

3. An NCB particle swarm optimization

In this section, firstly three techniques, a novel mbest particle for QPSO, contracting the searching space self-adaptively boundaries restrictions strategy, are introduced. Then a novel QPSO is proposed and experiments are done for some benchmark optimization problems.

3.1. A novel Mbest particle

Though the particle using the idea from the center-of-mass position to define $S(k)$ by Eq. (5) is good, it is rather conservative and it may introduce much more inner information of all the particles, not some new information in the whole feasible space. Aiming this, we generate a novel mbest particle as below

$$\begin{cases} \text{if } rand\ (0,1) > 0.8 \\ \qquad S(k) = \sum\limits_{i=1}^{M} \mu_i P_i + \left(1 - \sum\limits_{i=1}^{M} \mu_i\right) \frac{\sum_{i=1}^{M} P_i}{M} \\ \text{else} \\ \qquad S(k) = \sum\limits_{i=1}^{M} \alpha_i P_i \\ \text{end} \end{cases} \tag{7}$$

where each $\mu_i, \alpha_i \in (-0.3, 1.2)$ randomly and $\sum_{i=1}^{M} \alpha_i = 1$.

With a relatively big probability 0.8, we using the similar idea from simplex method, but not the same. And with a relatively small probability 0.2, we introduce some new information in the whole space. Though this might cause infeasibility of the particle, but we do not use Eq. (7) solely, we have some more means following.

3.2. Contracting the searching space self-adaptively

For many experiment results reported suggest that too many generations do not bring the optimum better than the local one [16,19], a novel technique through a technique space contraction self-adaptively to avoid exploitation excessively in redundant space was put as below. In each contracted space, the normal QPSO has much fewer generations than the original feasible space.

If a local optimum Q_g is found, we define a new searching space $D^{(t+1)} = [a^{(t+1)}, b^{(t+1)}]$ centering Q_g from the current searching space $D^{(t)} = [a^{(t)}, b^{(t)}]$ as below:

$$\begin{cases} \text{if } rand\ (0,1) < 0.95 \\ \quad a_i^{(t+1)} = \max(x_i^{(t)} - \gamma_1 c_i^{(t)}, a_i^{(t)}), \\ \quad b_i^{(t+1)} = \min(x_i^{(t)} + \gamma_1 c_i^{(t)}, b_i^{(t)}), \\ \text{else} \\ \quad a_i^{(t+1)} = \max(\min(x_i^{(t)} - \gamma_2 c_i^{(t)}, a_i^{(t)}), a_i^{(1)}) \\ \quad b_i^{(t+1)} = \min(\max(x_i^{(t)} + \gamma_2 c_i^{(t)}, b_i^{(t)}), b_i^{(1)}) \\ \text{end} \end{cases} \tag{8}$$

where $c_i = (b_i - a_i)/2, i = 1, 2, \cdots, s$, pre−given contraction ratio $\gamma_1, \gamma_2 \in (0,1)$. Then we use QPSO to search in the new space to get a new optimum $Q\prime_g$, save the better one in $Q\prime_g$ and Q_g.

3.3. Boundaries restrictions strategy

Though v_i of PSO directs population to converge to the global optimum, sometimes it can make x_i out of feasible fields. When $x_{i,d}(k+1) > u_d$, it suggests that the individual has a trend to cross the u_d in its dimension d. To maintain the trend in some degree, we propose a novel strategy boundaries restrictions (BR) as below:

$$x_{i,d}(k+1) = \begin{cases} x_{i,d}(k+1), \quad \text{if } l_d \leqslant x_{i,d}(k+1) \leqslant u_d; \\ u_d - 0.3 \cdot rand\ (0,1) \cdot (u_d - lb_d), \text{if } x_{i,d}(k+1) > u_d; \\ l_d + 0.3 \cdot rand\ (0,1) \cdot (u_d - lb_d), \text{if } x_{i,d}(k+1) < l_d. \end{cases} \tag{9}$$

where $u = (u_1, u_2, \ldots, u_D)$ is the upper boundary and $l = (l_1, l_2, \ldots, l_D)$ is the lower boundary. In this way, the population will not cross the boundaries. Then the BR can eliminate the pneumonia of boundary-crossing and maintain the trend of crossing.

3.4. NCB quantum-behaved particle swarm optimization

With the techniques in Section 3.1–3.3, we can put a novel quantum particle swarm optimization. The procedure of NCB–QPSO is outlined in Algorithm 1:

Algorithm 1. NCB–QPSO

Initialize Parameters
repeat
  repeat
    Initialize Particle swarm and local and global best
    while Inner cycle Termination condition is not satisfied **do**
      Apply Generate Mbest by (7) and new swarm by (6)
      Check boundaries BR to new swarm by (9)
      Update swarm and local and global best Q_g
    end while
    Contract space Define a new region by (8)
  until $\max c_i < \delta$
  Output Global best Q_g
Until all the optimum gained

We have to explain that the number of inner cycle is much more less than the normal QPSO's iterations, for example, $1/10$ of QPSO's. Through NCB–QPSO, firstly it can avoid premature and find all the optimums sequentially with the help of outer cycle termination judging. Secondly NCB–QPSO can avoid exploitation excessively in redundant space and search in the most prospective space of the feasible field and avoid premature. Thirdly if QPSO in inner cycle is valid enough, NCB–QPSO will not contract the searching space, and in this sense QPSO is the special cases of NCB–QPSO.

To exhibit the performance of the proposed methods, we choose 6 classical benchmark functions in evolutionary methods [16]. All the functions to be simulated have the global minimum 0 on $(0, 0, \ldots, 0)$ except Rosenbrock function's on $(1, 1, \ldots, 1)$.

Example. 1: Griewank function

$$f(x) = \frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \prod_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{10}$$

It is a well known function in optimization. The minimum 0 is on $(0, 0, \ldots, 0)$, and almost not different from the numerous local minima around.

Example. 2: Rosenbrock function

$$f(x) = \sum_{i=1}^{N-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \tag{11}$$

It is quite difficult to find its global minimum as soon as the dimension is high with evolutionary methods. To this function, algorithms like methods using gradient may work very well, so it is by itself a challenge for stochastic methods like PSO, genetic algorithms etc.

Example. 3: Ackley function

$$f(x) = 20 + e - 20\exp\left(-0.2\sqrt{\sum_{i=1}^{N} x_i^2 \Big/ N}\right) - \exp\left(\sum_{i=1}^{N} 2\pi x_i \Big/ N\right) \tag{12}$$

The attraction basin of the global minimum of this function is quite narrow, so it is very difficult to find it.

Example. 4: Rastrigrin function

$$f(x) = \sum_{i=1}^{N} (x_i^2 - 10\cos 2\pi x_i + 10) \tag{13}$$

The function is often used to test the global searching performance of optimization algorithms.

Example. 5: Alpine function

$$f(x) = \sum_{i=1}^{N} |x_i \sin x_i + 0.1 x_i| \tag{14}$$

It is not really symmetrical. The function is nevertheless quite easy, and can be seen as a kind of ponsasinorum for many optimization algorithms.

Example. 6: Parabola function

$$f(x) = \sum_{i=1}^{N} x_i^2 \tag{15}$$

It is very easy to adjust its difficulty just by modifying the dimension N. Like Rosenbrock function, algorithms basing gradient may work very well,but it is a great challenge for stochastic methods.

For the functions optimization, let swarm sizes equal to $40, x \in [-100, 100]^{50}$, make the inner cycle 200 and contracted the searching space 10 times (that is, the whole iterations is 2000)for all the functions below for NCB–QPSO while the single QPSO's iterations are 2000. Let NQPS and OPSO run 100 times independently. And simulation results are reported below.

The number of success, mean, standard deviation, minimum and maximum number of required function evaluations are reported in Table 1 for QPSO and NCB–QPSO. And Figs. 1 and 2 show the comparisons between the two methods to function Ackley and Alpine.

From Table 1 we can see NCB–QPSO are much more efficient than the QPSO for functions Griewank, Ackley, Rastrigrin and Alpine, it can get the better results much easier, and it can get more chance to get the possible global minimum areas and avoid the local optimum areas. From Figs. 1 and 2 we can find that though space contraction introduce the worse particles with the new initial particle swarm, it result in the more decrease in every 100 cycles than QPSO by introducing the much more possible areas containing the global minimums. And the boundaries restrictions enhance the feasibilities of the particle swarm in the greatest deagress. Thus it can be concluded that the three techniques can enhance the performance of NCB–QPSO greatly.

The Figs. 1 and 2 show that NCB-QPSO have the advantages of fast convergence, high precision and robustness. Thus we can conclude that withdraw some local optimum and contracting the searching space with a new swarm particles and introduce some bad particles can make the objective functions' value decrease much more faster and arrive the best optimum space than QPSO just as the two figures show.

Table 1
Simulation results of NCB–QPSO and QPSO for benchmark problems.

Function	Method	Mean	StD	Min	Max
Griewank	QPSO	0.0102	0.0358	1.3467e−008	0.1826
	NCB–QPSO	0.0024	0.0048	4.9755e−005	0.0203
Rosenbrock	QPSO	208.9190	238.7240	46.3427	1.3580e+003
	NCB–QPSO	312.3845	563.8205	46.7507	3.0144e+003
Ackley	QPSO	13.6787	7.1478	0.0148	20.5998
	NCB–QPSO	2.0490	6.0661	0.0258	20.0804
Rastrigrin	QPSO	350.0404	37.3800	262.0888	428.1316
	NCB–QPSO	82.6610	63.6783	42.4634	291.3181
Alpine	QPSO	57.5566	17.8914	11.5837	89.7893
	NCB–QPSO	0.2613	0.3319	0.0530	1.7834
Parabola	QPSO	1.9590e−005	1.7608e−005	4.1423e−007	7.0206e−005
	NCB–QPSO	0.0314	0.0887	0.0014	0.5336

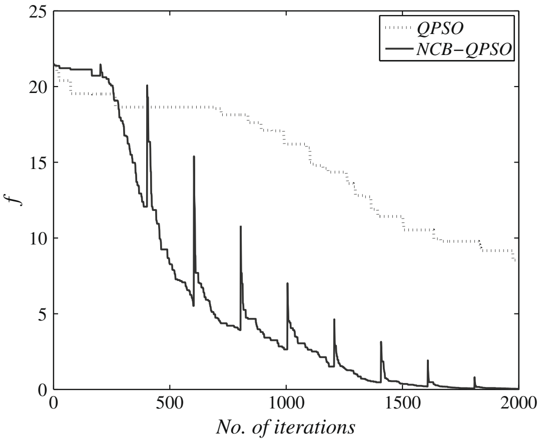

Fig. 1. Comparisons for function Ackley.

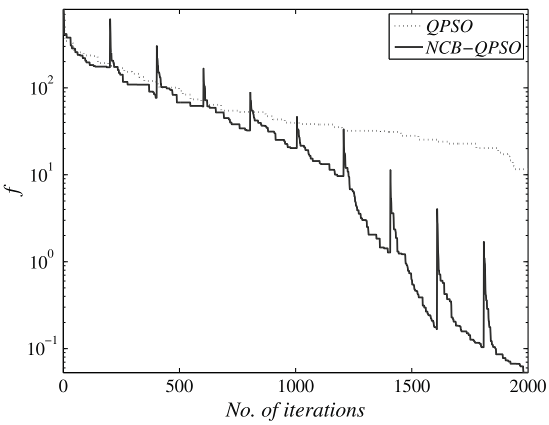

Fig. 2. Comparisons for function Alpine.

But the results in Table 1 for the function Rosenbrock and Parabola, NCB–QPSO are not as good as QPSO. The reason lies in two aspect, One is that the every 200 cycles of the 2000 of QPSO can make these two functions' value decrease sharply, while NCB–QPSO contracts the searching space at each 200 cycles with a new initial population, this may loosen the speed of function value decreasing in some degree. The other reason is the construction of this two functions, they only have polynomial without trigonometric or other function like the other testing functions. And they can be optimized easy by traditional local methods.

Thus for this two kind of functions, we can take QPSO as a special case of NCB–QPSO without contracting the searching space.

4. Detecting the chaotic system's unstable periodic orbits

The mappings considered in our simulations are:

Hénon map is a famous two-dimensional quadratic map given by the coupled equations with chaotic solutions proposed by the French astronomer Michel Hénon [24], as a simplified model of the Poincáre map for the Lorenz model. Usually it has two forms as below, one is

Example. 7: Hénon 2-dimensional map-1

$$\Phi(x) = \varphi(\alpha)\begin{pmatrix} x_1 \\ x_2 - x_1^2 \end{pmatrix}, \ \varphi(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \tag{16}$$

where $\alpha = \arccos^{-1}(0.24), \ x_1, x_2 \in [-1, 1]$.

Example. 8: Hénon 2-dimensional map-2

$$\Phi(x) = \begin{pmatrix} a + bx_2 - x_1^2 \\ x_1 \end{pmatrix} \tag{17}$$

where $x_1, x_2 \in [-1, 1]$ and a and b of system are the system parameters of system (17) when they are $a^* = 1.4, b^* = 0.3$, (17) is chaotic and it has strange attractor with a unstable fixed point in it [24,25].

Example. 9: Hénon 4-dimensional simplistic map. An extension of the Hénon 2-dimensional map-1 to the complex case [14,15,26]

$$\Phi(x) = \begin{pmatrix} \varphi(\alpha) & \theta_{2\times2} \\ \theta_{2\times2} & \varphi(\alpha) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 - x_1^2 + x_3^2 \\ x_3 \\ x_4 - 2x_1x_3 \end{pmatrix} \tag{18}$$

where $x \in [-3, 3]^4$ and $\alpha = \arccos^{-1}(0.24)$.

Example. 10: Predator–Prey map [14]

$$\Phi(x) = \begin{pmatrix} 3.6545 & -1 \\ 0 & 3.226 \end{pmatrix} \begin{pmatrix} x_1(1 - x_1) \\ x_1x_2 \end{pmatrix} \tag{19}$$

where $x \in [-2, 2]^2$.

Example. 11: Gingerbreadman map [27]

$$\Phi(x) = \begin{pmatrix} 1 - x_2 + |x_1| \\ x_1 \end{pmatrix} \tag{20}$$

where $x \in [-4, 8]^2$. The map is chaotic in the filled region above and stable in the six hexagonal regions.

With a new defined function (21) below as fitness function we can use NCB–QPSO and QPSO to get the different unstable period orbits.

$$F(x) = \left\| \Phi^{(p)}(x) - x \right\|_2^2 \tag{21}$$

As the objection is to minimize function (2), we use a simple square of the norm as in function (21) to decrease the time of $\|\cdot\|_2$ and enlarge the distance of the different local optimums of the system. And Figs. 3 and 4 show the figure of function $\lg(F)$ of the 11-period point objective function of system Hénon-2 and 7-period point objective function of system Gingerbreadman.

We can see the difficulties to optimize the function (21) from Figs. 3 and 4, it has so many local minimums and local maximums in its definition area, that is reason why the normal optimization methods based on gradients could not solve.

For the parameters of NCB–QPSO, let the inner cycle 100 and contracted the searching space 10 times (that is, the whole iterations is 1000) for all the functions below for NCB–QPSO while the single QPSO's iterations are 1000. Let NCB–QPSO and OPSO run 100 times independently.

Simulation results are reported below. Figs. 5–8 give the optimization process for the objective function of 11-period point of hénon system, 11-period point of Predator system, 13-period point of 4-D hénon system and 19-period point of Gingerbreadman system by NCB–QPSO and QPSO method respectively.

From the 4 figures above we can include that in the same iterations, NCB–QPSO has much more fine precision than QPSO for these four functions. We can find that NCB–QPSO is much more efficient than QPSO especially in this kind of objective functions, because it can avoid local minimums more easier and it can gain much more small value again and again in the whole 1000 iterations. While QPSO only manage in several time and the come into premature for a long time at the local points.

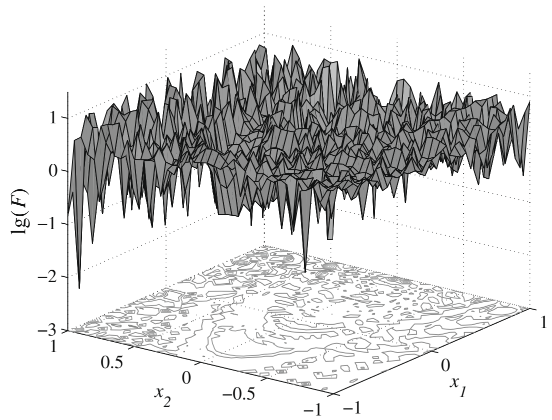

Fig. 3. $F(x)(p = 11)$ of Hénon-2.

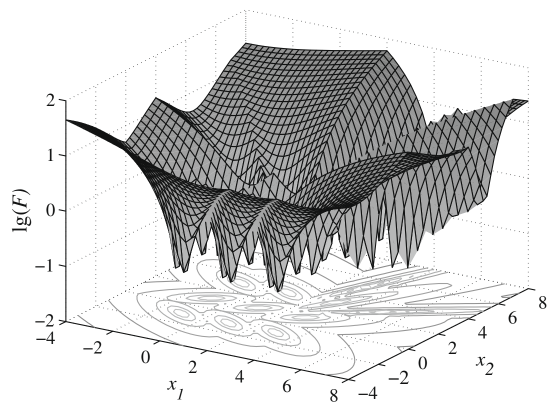

Fig. 4. $F(x)(p = 7)$ of Gingerbreadman.

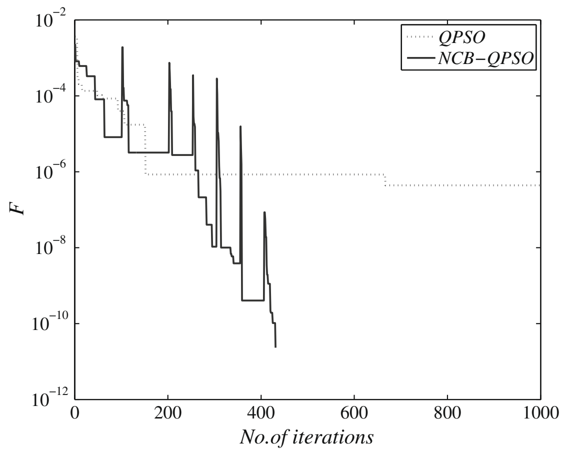

Fig. 5. Results of $F(x)(p = 11)$ for Hénon-2.

And we choose a random choose 40 results in the whole 100 times of all the simulations of optimization process by these two methods for the objective function of 61-period point of hénon system and 43-period point of Predator system, in the following figures.

From the Figs. 9 and 10 we can include that NCB–QPSO is much better than QPSO, because its results are much more less than QPSO in 95% of all the cases.

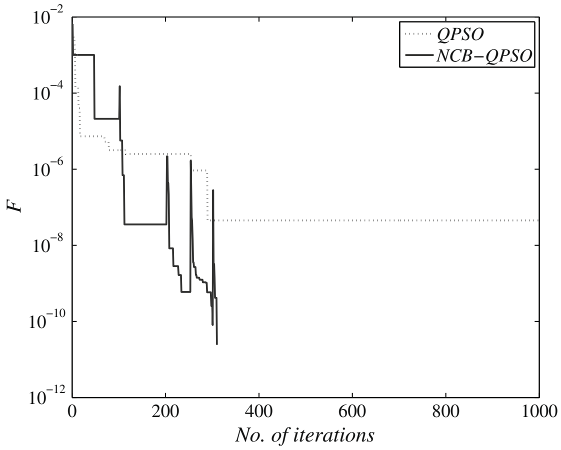

Fig. 6. Results of $F(x)(p = 11)$ for Predator.

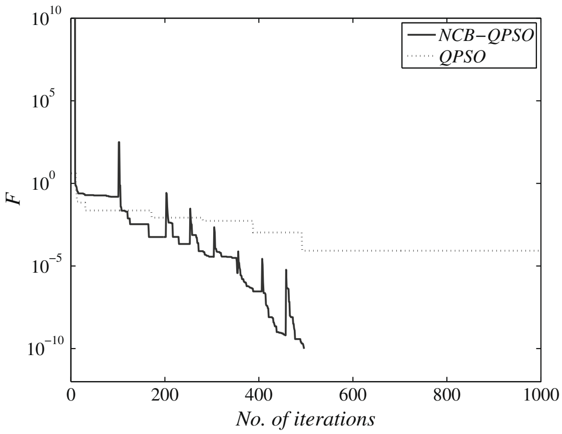

Fig. 7. Results of $F(x)(p = 13)$ for 4-Hénon.

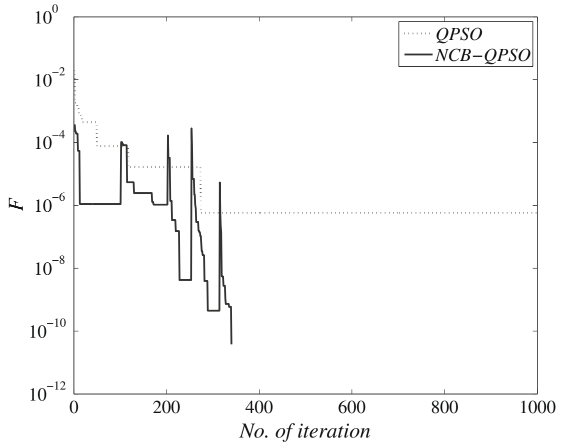

Fig. 8. Results of $F(x)(p = 19)$ for (20).

To compare globally we list the simulation of $F(x)$ for different order UPOs of the chaos systems above in following tables.

Note: "U" denotes the period orbits are unstable. And we use the method in Ref. [28] to judge the orbits' stabilities, which is listed in Appendix.

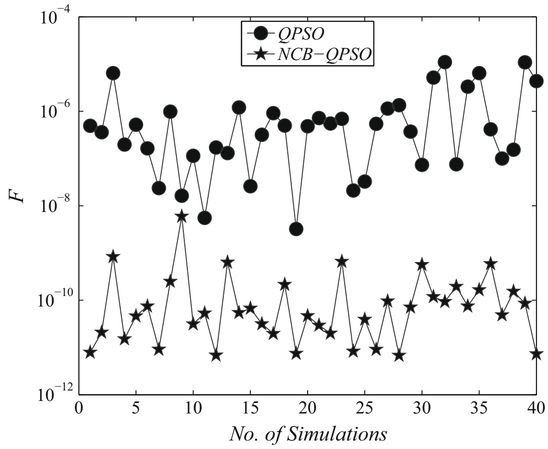

Fig. 9. Results of $F(x)(p = 61)$ for Hénon-1.

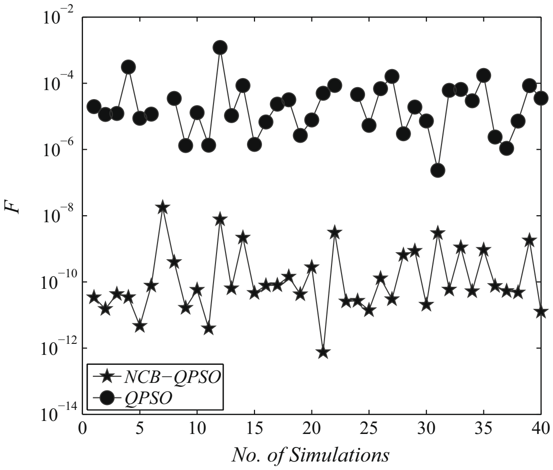

Fig. 10. Results of $F(x)(p = 43)$ for Predator.

From the Tables 2–6, we can see that, NCB–QPSO is much more efficient than QPSO for results of $F(x)$ for different order UPOs of all above chaotic mapping. It can get a much smaller value of $F(x)$ in most cases in 4 aspects: mean, max and min value and StD of all the results of correspondent simulations.

But for the Hénon system-2, NCB–QPSO have a little better value for high order $F(x)$ of UPOs than PSO, not like the cases in low order. We think the reason might lie in the function construction of Hénon system-2, for the same case occurs in the optimization of functions Rosenbrock and Parabola. Now we can say that our new NCB–QPSO are much more better than QPSO, but might have no advantage for some special functions that can be minimized easily by traditional optimization methods such as by or based on Newton methods.

Figs. 11–14 give the UPOs of Hénon map-1, Hénon map-2, Predator–prey and Gingerbreadman map respectively with the results from Tables above as below:

We can see that the points on the same order UPO are not always arranged in one bye one, but in a strange sequence, for instance, like Figs. 11–14.

5. Conclusions

Simulations above illustrate that NCB–QPSO is a unified operations that enables 3 aspects of different operations to the QPSO and it present a novel prospective way in optimization method. During the generations, NCB–QPSO withdraws some local optimum, contracts the searching space with a new swarm particles and introduce some bad particles every other generations. However these operations do not make the algorithm bad but lead a new decrease for the objective functions.

It also shows that NCB–QPSO can improve the global convergence and have the advantages of high precision and robustness to such a certain extent. And it is a successful method not only in optimization problems but also in detecting the unsta-

Table 2
Comparisons for results of $F(x)$ for different order UPOs of 2 Hénon-1.

P	Method	Stability	Mean	StD	Min	Max
5	NCB–QPSO		1.6125e−11	1.6364e−11	4.7635e−13	7.0486e−11
	QPSO		2.4885e−06	2.6906e−06	1.2335e−08	1.2111e−05
11	NCB–QPSO	U	4.9663e−11	1.1312e−10	7.0217e−13	7.2685e−10
	QPSO		9.4391e−07	1.0637e−06	1.7814e−08	4.6772e−06
16	NCB–QPSO	U	6.6521e−10	3.4092e−09	2.4285e−13	2.1611e−08
	QPSO		9.5126e−07	9.7555e−07	1.0735e−09	3.5773e−06
29	NCB–QPSO		1.2542e−12	1.9165e−12	4.9883e−16	7.9992e−12
	QPSO		9.9215e−07	2.3599e−06	2.5431e−12	1.1653e−05
43	NCB–QPSO		2.8320e−12	4.6193e−12	7.1723e−16	2.0781e−11
	QPSO		1.6554e−06	1.6821e−06	6.4419e−10	7.8360e−06
61	NCB–QPSO	U	2.8650e−10	9.4780e−10	6.7967e−12	5.9859e−09
	QPSO		1.5075e−06	2.7513e−06	3.2201e−09	1.0969e−05
83	NCB–QPSO	U	4.4991e−12	8.4193e−12	4.2273e−14	4.6773e−11
	QPSO		3.1691e−07	6.0974e−07	5.1898e−10	3.1903e−06
103	NCB–QPSO	U	1.4607e−11	2.8220e−11	7.1337e−15	1.2940e−10
	QPSO		7.0141e−07	1.0017e−06	8.1293e−10	3.6791e−06
197	NCB–QPSO	U	2.6350e−12	5.3045e−12	3.9304e−17	2.7230e−11
	QPSO		6.2712e−07	1.1763e−06	3.2874e−09	5.2388e−06
257	NCB–QPSO	U	3.1631e−12	4.7554e−12	3.0072e−15	1.8136e−11
	QPSO		2.0355e−06	2.8338e−06	1.2954e−10	1.1396e−05
359	NCB–QPSO	U	4.4341e−12	1.1626e−11	1.7860e−15	3.7399e−11
	QPSO		1.0476e−06	1.2335e−06	4.7144e−09	3.0978e−06

Table 3
Comparisons for results of $F(x)$ for different order UPOs of 2 Hénon-2.

P	Method	Stability	Mean	StD	Min	Max
1	NCB–QPSO	U	1.5874e−11	1.5448e−11	1.3023e−13	5.0857e−11
	QPSO		3.3793e−06	3.8761e−06	7.8769e−08	1.7987e−05
11	NCB–QPSO	U	4.1881e−07	2.2478e−06	4.3110e−11	1.4253e−05
	QPSO		1.7600e−05	1.5685e−05	4.3786e−08	5.9812e−05
13	NCB–QPSO	U	1.7430e−07	8.4356e−07	3.5971e−12	5.3035e−06
	QPSO		1.4267e−05	1.4646e−05	7.7260e−08	7.2038e−05
15	NCB–QPSO	U	1.9101e−05	3.9697e−05	5.0855e−10	1.6672e−04
	QPSO		2.0462e−05	1.7747e−05	7.4814e−07	8.0362e−05
17	NCB–QPSO	U	8.3311e−06	2.4600e−05	1.1712e−10	1.1992e−04
	QPSO		1.6105e−05	1.5301e−05	2.1485e−07	5.2903e−05
19	NCB–QPSO	U	3.7796e−05	2.0524e−04	7.3663e−09	0.0013
	QPSO		2.6308e−05	2.8351e−05	9.1229e−07	1.4859e−04
23	NCB–QPSO	U	7.0132e−06	1.3315e−05	1.0641e−09	6.9586e−05
	QPSO		2.0041e−05	2.1222e−05	6.5950e−07	1.1685e−04
29	NCB–QPSO	U	8.8086e−05	4.8598e−04	9.6866e−08	0.0031
	QPSO		2.2196e−05	2.4625e−05	4.6713e−07	1.0611e−04
31	NCB–QPSO	U	2.3025e−05	7.9924e−05	1.5960e−08	5.0087e−04
	QPSO		8.6276e−04	6.4267e−04	3.0380e−04	0.0024
37	NCB–QPSO	U	1.0466e−06	1.4639e−06	4.2945e−09	5.7683e−06
	QPSO		1.5944e−05	1.6134e−05	1.2991e−07	5.9229e−05

ble UPOs of chaos system, especially in some higher order cases of UPOs. This is a new Non-Lyaponov way in detecting the UPOs.

We still have to improve the performance of NCB–QPSO for some special functions like Rosenbrock, Parabola and etc. Only in this way, it can have a large application areas.

And the idea from NCB–QPSO can be easily introduced into other evolutionary computation methods.

Table 4
Comparisons for results of $F(x)$ for different order UPOs of 4-Hénon.

P	Method	Stability	Mean	StD	Min	Max
1	NCB–QPSO	U	5.1249e−5	2.2686e−4	1.0396e−10	0.0010
	QPSO		0.0746	0.0351	0.0258	0.1728
2	NCB–QPSO	U	4.5020e−5	2.8407e−4	3.2130e−12	0.0018
	QPSO		0.0227	0.0137	0.0037	0.0668
3	NCB–QPSO	U	4.2683e−4	0.0013	6.9373e−7	0.0042
	QPSO		0.0394	0.0174	0.0143	0.0771
4	NCB–QPSO	U	1.5830e−9	6.8213e−9	1.1488e−11	4.1436e−8
	QPSO		0.0494	0.0197	0.0087	0.0873
5	NCB–QPSO	U	1.4562e−9	2.0899e−9	5.7964e−11	7.6580e−9
	QPSO		0.0711	0.0352	0.0013	0.1384
9	NCB–QPSO	U	9.4387e−8	1.3202e−7	8.1876e−11	5.2525e−7
	QPSO		0.1054	0.0409	0.0211	0.2115
11	NCB–QPSO		4.9505e−8	1.9997e−7	1.4663e−11	1.1868e−6
	QPSO		0.3107	0.5765	0.0029	3.5210

Table 5
Comparisons for results of $F(x)$ for different order UPOs of Predator–Prey.

P	Method	Stability	Mean	StD	Min	Max
1	NCB–QPSO	U	1.0114e−11	8.0431e−12	5.6799e−13	3.0225e−11
	QPSO		9.2930e−5	9.5415e−5	1.5528e−8	4.0223e−4
2	NCB–QPSO	U	6.0605e−9	2.1132e−8	9.3486e−14	1.2586e−7
	QPSO		0.0025	4.5483e−4	8.4895e−4	0.0029
3	NCB–QPSO	U	7.3715e−8	2.7290e−7	1.9836e−12	1.3590e−6
	QPSO		5.4474e−6	5.0824e−6	4.8906e−7	2.2332e−5
11	NCB–QPSO	U	3.0966e−11	4.1116e−11	1.3735e−12	2.5714e−10
	QPSO		2.1827e−5	4.4301e−5	2.3795e−10	2.4509e−4
43	NCB–QPSO	U	1.0269e−9	3.0567e−9	7.5781e−13	1.7795e−8
	QPSO		7.0988e−5	1.9749e−4	2.3432e−7	0.0012
59	NCB–QPSO	U	1.0960e−8	1.0448e−8	4.0744e−11	3.1578e−8
	QPSO		7.2946e−5	9.9663e−5	1.2321e−6	4.1800e−4
97	NCB–QPSO	U	7.6408e−9	1.6756e−8	1.7691e−11	5.5037e−8
	QPSO		6.9743e−5	1.4950e−4	3.9233e−9	7.7521e−4
131	NCB–QPSO	U	4.3849e−6	7.4829e−6	3.3662e−7	2.5473e−5
	QPSO		0.0019	0.0033	1.0033e−4	0.0111

Table 6
Comparisons for results of $F(x)$ for different order UPOs of Gingerbreadman.

P	Method	Stability	Mean	StD	Min	Max
1	NCB–QPSO		3.3861e−11	2.3394e−11	6.3772e−13	7.7319e−11
	QPSO		0.1084	0.0952	0.0048	0.3123
5	NCB–QPSO	U	9.1469e−11	2.7829e−10	1.6473e−12	1.6755e−9
	QPSO		2.4525e−4	1.9935e−4	7.0313e−6	7.6499e−4
11	NCB–QPSO	U	1.8645e−11	1.5346e−11	2.6042e−14	6.6605e−11
	QPSO		0.0069	0.0070	9.0528e−5	0.0346
19	NCB–QPSO		3.0895e−11	4.6981e−11	5.2412e−13	2.6982e−10
	QPSO		6.4926e−6	6.6681e−6	1.3866e−8	3.1605e−5
23	NCB–QPSO	U	2.3336e−4	4.0934e−4	1.5393e−12	9.3344e−4
	QPSO		0.1098	0.0988	6.2656e−5	0.2682
29	NCB–QPSO	U	1.7193e−10	4.2982e−10	1.2778e−12	2.1994e−9
	QPSO		6.7558e−5	6.0052e−5	2.9128e−6	2.4323e−4

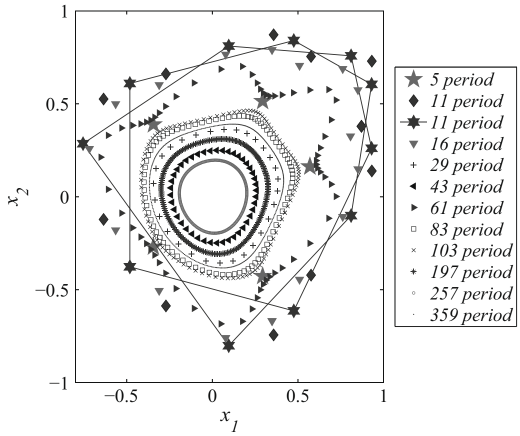

Fig. 11. UPOs of Hénon map-1.

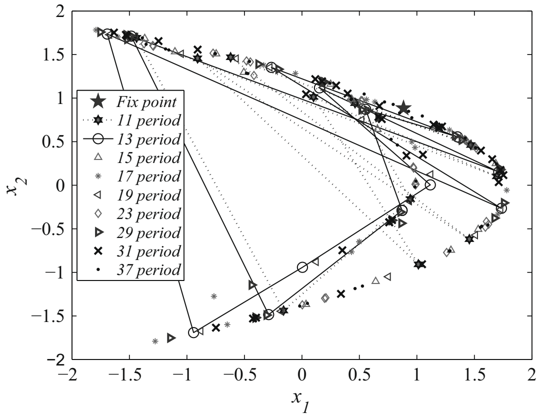

Fig. 12. UPOs of Hénon map-2.

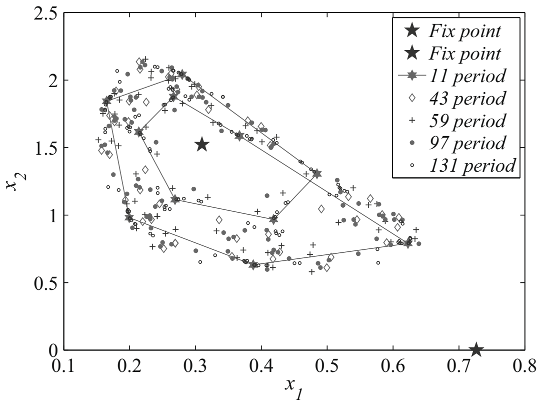

Fig. 13. UPOs of Predator map.

Algorithm 2. Algorithm for judging a periodic orbit's stability [28]

1: **Initialize** The mapping Φ, the initial X_1 and the period p;
2: Let J be the Jacobian matrix of the mapping. Set $J = Jacobian\Phi(X_1)$;
3: **For** $i \leftarrow 2 : p$ **do**

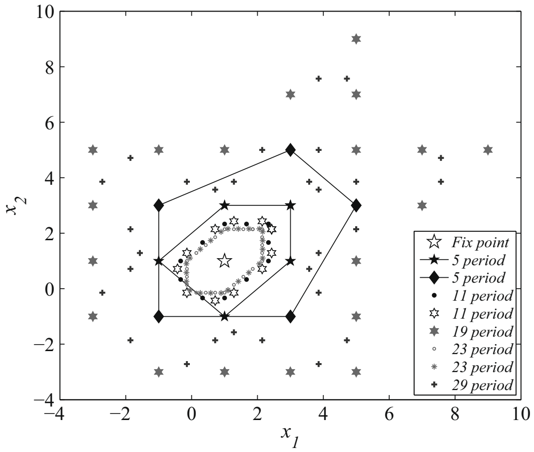

Fig. 14. UPOs of Gingerbreadman map.

$X_i = \Phi(X_{i-1}), J = J \cdot Jacobian\Phi(X_i)$;
end for
Let λ_1 is an eigenvalues of the final matrix J;
if $\lambda_1 \neq 0, Im(\lambda_1) \neq 0$ & $\|\lambda_1\|_2 \approx 1$
the orbits are stable
else
the orbits are Unstable
end if

References

[1] Lorenz EN. The essence of chaos. Seattle,WA: University of Washington Press; 1996.
[2] Katok A, Hasselblatt B. A first course in dynamics with a panorama of recent developments. Cambridge University Press; 2003.
[3] Ott E. Chaos in dynamical systems. New York: Cambridge University Press; 1993.
[4] Pierson D, Moss F. Phys Rev Lett 1995;75:2124.
[5] Dhamala M, Lai YC. Phys Rev E 1999;60:6176.
[6] Schmelcher P, Diakonos FK. Phys Rev Lett 1997;78:4733.
[7] Davidchack RL, Lai YC. Phys Rev E 1999;60:6172.
[8] Awrejcewicz J, Dzyubak L, Grebogi C. Chaos, Solitons & Fractals 2004;19:503.
[9] Bu SL, Wang BH, Jiang PQ. Detecting unstable periodic orbits in chaotic systems by using an efficient algorithm. Chaos, Solitons & Fractals 2004;22:237.
[10] Takens F. Dynamical systems and turbulence. In: DRand, Young LS, editors. Berlin: Spinger-Verlag; 1981.
[11] So P, Ott E, Schiff SJ, Kaplan DT, Sauer T, Grebogi C. Phys Rev Lett 1996;76:4705.
[12] So P, Francis JT, Netoff TI, Gluckman BJ, Schiff SJ. Periodic orbits: a new language for neuronal dynamics. Biophys J 1998;74:2776.
[13] Christini DJ, Kaplan DT. Adaptive estimation and control method for unstable periodic dynamics in spike trains. Phys Rev E 2000;61:5149.
[14] Parsopoulos KE, Vrahatis MN. Computing periodic orbits of nondifferentiable discontinuous mappings through particle swarm optimization. In: IEEE 2003 swarm intelligence symposium (SIS 2003). Indianapolis, USA; 2003. p. 34–41.
[15] Petalas YG, Parsopoulos KE, Vrahatis MN. Entropy-based Memetic particle swarm optimization for computing periodic orbits of nonlinear mappings. In: IEEE 2007 congress on evolutionary computation, Singapore: 2040; 2007.
[16] Yao X, Xu Y. J Comp Sci Tech 2006;21:1.
[17] Gao F, Lee J-J, Li ZQ, Tong HQ. Chaos, Chaos Solitons Fractals 2009;42:1286–91.
[18] Eberhart RC, Shi YH. Proceedings of the 2000 congress on evolutionary computation. Piscataway, NJ: IEEE Service Center; 2000. 84.
[19] Gao F, Tong HQ. ACTA Phys Sin 2006;55:577.
[20] Schweizer W. Numerical quantum dynamics (progress in theoretical chemistry and physics). Hingham (MA, USA): Springer; 2001. 95.
[21] Tad Hogg, Dmitriy Portnov. Quantum optimization. Inform Sci 2000;128:181.
[22] Sun J, Xu WB, Feng B. In: Proceedings of IEEE international conference on systems, man and cybernetics, Big Island (HI, USA), 3049; 2005.
[23] Gao F, Li ZQ, Tong HQ. Chinese Phys Sin 2008;17:1196.
[24] Hénon M. Numerical study of quadratic area-preserving mappings. Quart Appl Math 1969;27:291.
[25] Eric W. Weisstein, 2008, Hénon Map. http://mathworld.wolfram.com/HenonMap.html.
[26] Vrahatis MN. An efficient method for locating and computing periodic orbits of nonlinear mappings. J Comp Phys 1995;119:105.
[27] Devaney RL. A piecewise linear model for the zones of instability of an area preserving map. Physica D 1984;10:387.
[28] Parsopoulos KE, Vrahatis MN. Computing periodic orbits of nonlinear mappings through particle swarm optimization. In: Proceedings of the 4th GRACM congress on computational mechanics, Patras, Greece; 2002.