

Generalized Form of Parrondo's Paradoxical Game with Applications to Chaos Control

MARIUS-F. DANCA

*Department of Mathematics and Computer Science,
Avram Iancu University,
400380 Cluj-Napoca, ROMANIA
and
Romanian Institute for Science and Technology,
400487 Cluj-Napoca, ROMANIA,
danca@rist.ro*

MICHAL FEČKAN

*Department of Mathematical Analysis and Numerical Mathematics,
Comenius University,
Mlynská dolina, 842 48 Bratislava, SLOVAKIA,
and
Mathematical Institute,
Slovak Academy of Sciences,
Štefánikova 49, 814 73 Bratislava, SLOVAKIA,
Michal.Feckan@fmph.uniba.sk*

MIGUEL ROMERA

*Instituto de Tecnologías Físicas y de la Información (ITEFI)
Consejo Superior de Investigaciones Científicas (CSIC)
Serrano 144, 28006 Madrid, SPAIN,
miguel@iec.csic.es*

Received (to be inserted by publisher)

In this paper we show that a generalized form of Parrondo's paradoxical game can be applied to discrete systems, working out the logistic map as a concrete example, to generate stable orbits. Written in Parrondos' terms, this reads: $chaos_1 + chaos_2 + \dots + chaos_N = order$, where $chaos_i$, $i = 1, 2, \dots, N$, are denoted the chaotic behaviors generated by N values of the parameter control, and by $order$ one understands some stable behavior. The numerical results are sustained by quantitative dynamics generated by Parrondo's game. The implementation of the generalized Parrondo's game is realized here via the parameter switching (PS) algorithm for continuous-time systems [Danca, 2013] adapted to the logistic map. Some related results for more general maps on averaging, which represent discrete analogies of the PS method for ODE, are also presented and discussed.

Keywords: Parrondo's paradox, Chaos control, Parameter switching algorithm

1. Introduction

Named after the Spanish physicist J. M. R. Parrondo in 1996, Parrondo’s paradox affirms that two losing games together can be set up to give a winning scenario [Harmer & Abbott, 1999a], [Harmer & Abbott, 1999b]. Parrondo showed that alternating deterministically (or even randomly) two losing gains, one can obtain a positive gain, i.e.: “losing + losing = winning”.

This apparent contradiction is known now as Parrondo’s paradox, becoming an active area of research in such as discrete-time ratchets [Amengual *et al.*, 2004], minimal Brownian ratchet [Lee *et al.*, 2003], game theory in the sense of Blackwell [Blackwell & Girshick, 1954], molecular transport [Heath *et al.*, 2002], biology [Danca & Lai, 2012], and so on. For example, the sacrifice of some chess pieces can lead to winning a game, therefore there are suggestions that Parrondos game could offer a possibility to make money by investing into losing stocks. Even more, there are theories which affirm that this kind of mechanisms could explain the origin of life [Abbot, 2010].

Zeilberger, said on Response to the Award of the 1998 Steele Prize for Seminal Contributions to Research: “Combining different and sometimes opposite approaches and viewpoints will lead to revolutions. So the moral is: Don’t look down on any activity as inferior, because two ugly parents can have beautiful children”.

Since then, a huge number of scientific papers have been published and, despite the fact that not all scientists agree with the principle of Parrondo’s paradox ([e.g. Iyengar & Kohli, 2003]), alternations like losing–winning, weakness–strength, order–chaos and so on, can be found in many mathematical systems, control theory, physics, biology, quantum systems, and so on, where combined processes may lead to counterintuitive dynamics. Moreover, it has been found that this apparently trivial phenomenon is typical not only in computational experiments but also in nature: there are many interactions which are due to accidental or intentional switches of some parameters characterizing the underlying systems.

In this paper, we show that via the numerical parameter switching (PS) algorithm for continuous-time systems [Danca, 2013], a generalization of the Parrondo’s game can be implemented for the logistic map to obtain stable orbits. PS algorithm allows to direct trajectories of some considered continuous nonlinear system, to wherever one wants within a targeted attractor (see also [Danca, 2013]).

The paper is organized as follows: Section 2 describes the PS algorithm for continuous systems revealing his interpretation as a generalization of the Parrondo’s game. Section 3 introduces the PS algorithm for the logistic map and general discrete systems. In Section 4, the generalization of the Parrondo’s game is applied via the PS algorithm to the logistic map, to obtain stable orbits. The Conclusion Section closes this paper and the Appendix contains the proof of one result for discrete maps presented in Section 3.

2. PS algorithm as generalization of the Parrondo’s game

Let us consider the following class of continuous systems modeled by the initial value problem:

$$f(x(t)) = f(x(t)) + pAx(t), \quad x(0) = x_0, \quad t \in I = [0, \infty), \quad (1)$$

with $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ a nonlinear (at least) continuous vector valued function, $x_0 \in \mathbb{R}^n$, A a square real matrix, and $p \in \mathbb{R}$ the control parameter.

The class of the systems modeled by (1) comprises systems such as Lorenz, Chen, Chua, Rössler and many other known chaotic systems.

To construct the PS algorithm, a set of parameter values $\mathcal{P}_N = \{p_1, p_2, \dots, p_N\}$ is necessary, where $p_i \in \mathbb{R}$, $i = 1, 2, \dots, N$, for $N \geq 2$, have “weights” m_i , with m_i being some positive integers and, also, we need a numerical method for ODEs with the single fixed step size h to integrate the system.

With these ingredients, while the initial value problem is numerically integrated, parameter p is switched within \mathcal{P}_N in the following manner: for the first m_1 integration steps, $p = p_1$, for the next m_2 steps, $p = p_2$, and so on, until the last m_N steps, when $p = p_N$, after which the algorithm repeats on the next subintervals, until I is covered.

Symbolically, once N , \mathcal{P}_N and the weights m_i , $i = 1, 2, \dots, N$, together with the step size h are set, PS algorithm will be denoted as follows:

$$[m_1 p_1, m_2 p_2, \dots, m_N p_N]. \quad (2)$$

For example, $[3p_1, 2p_2]$ means that while the problem is numerical integrated, the first 3 integration steps $p = p_1$, then for the next two steps $p = p_2$, after which, again, for three times with $p = p_1$, then two times $p = p_2$, and so on.

In this way, once the scheme (2) is applied, the “switched” solution, obtained with the PS algorithm, will converge to the “averaged” solution obtained from the following “averaged equation”:

$$\dot{x}(t) = g(x(t)) + p^* Ax(t), \quad (3)$$

where the “averaged” value p^* is given by

$$p^* = \frac{\sum_{i=1}^N m_i p_i}{\sum_{i=1}^N m_i}. \quad (4)$$

Two different proofs of the convergence of the PS algorithm are presented in [Danca, 2013] and [Mao *et al.*, 2010], respectively.

PS is useful when we want to obtain the numerical approximation of some attractor, whose underlying parameter value, for some reasons, cannot be set or attained. Then, by choosing \mathcal{P}_N and m_i , so that the value p^* obtained via (4) equals the searched value and by applying the PS algorithm, one obtains a solution which finally leads to the targeted attractor, corresponding to p^* .

For example, let us consider the Chen system [Chen & Ueta, 1999]

$$\begin{aligned} \dot{x}_1 &= a(x_2 - x_1), \\ \dot{x}_2 &= (p - a)x_1 - x_1 x_3 + p x_2, \\ \dot{x}_3 &= x_1 x_2 - b x_3, \end{aligned} \quad (6)$$

where $a = 35$ and $b = 3$, while the control parameter is chosen to be p , with

$$f(x) = \begin{pmatrix} a(x_2 - x_1) \\ -ax_1 - x_1 x_3 \\ x_1 x_2 - b x_3 \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

If we intend to approximate, for example, the stable limit cycle corresponding to $p = 26.09$, we can choose, the scheme (2) with $N = 2$: $[m_1 p_1, m_2 p_2]$, $\mathcal{P}_2 = \{25.75, 26.26\}$ and $m_1 = 1$, $m_2 = 2$ which, via (4), gives $p^* = (1 \times 25.75 + 2 \times 26.26)/(1 + 2) = 26.09$. By applying the PS algorithm, the obtained attractor, denoted by A^* (plotted in red in Fig. 1), approximates the averaged attractor (blue), denoted by A_{p^*} (the beginning transients have been neglected). The attractors corresponding to p_1 and p_2 are chaotic and are denoted by A_1 and A_2 (Fig. 1 top)¹, respectively.

Obviously, there are several ways to obtain the same attractor using the PS algorithm. For example, the same stable attractor approximated above, can be obtained by switching $N = 3$ parameter values, with $\mathcal{P}_3 = \{24.75, 25.8, 27.08\}$ using the scheme $[2p_1, 1p_2, 3p_3]$. Again, the relation (4) leads to $p^* = 26.09$.

Other examples, for continuous, discontinuous systems of integer, and fractional-order systems, can be found in [Danca *et al.*, 2012].

Remark 2.1.

(i) The simplicity of the PS algorithm, which allows the approximation of any solution of (1), resides in the linear dependence on p (The term $pAx(t)$).

¹The numerical method utilized here is the standard RK with step size $h = 0.001$.

(ii) The PS algorithm can be considered as easy to implement a chaos control-like algorithm, in the following sense: suppose that one intends to obtain a stable limit cycle which corresponds to some p^* , but due to some reasons this value cannot be set in the considered problem. Then, by choosing an adequate scheme (2) for PS such that the right-hand side of (4) gives the desired value p^* , the PS algorithm will approximate the desired stable cycle. By its nature, the PS algorithm can also be used as an anticontrol-like method, if the attractor we want to approximate, corresponding to p^* , is chaotic (see [Danca et al., 2012]).

Hereafter, in this paper, by a little abuse of the notions of chaos control and anticontrol, these concepts will be used when stable or chaotic orbits, respectively, are obtained with the PS algorithm.

An interesting fact is that a stable or chaotic attractor can be approximated with the PS algorithm, whatever the behavior of the considered attractors (corresponding to p_i) is. This property helps us to implement Parrondo's game.

Let us suppose that we want to approximate with the PS algorithm some stable limit cycle, using the simplest scheme $[1p_1, 1p_2]$, by p_1, p_2 corresponding to chaotic attractors. Then, if we denote the chaotic motions corresponding to $p_{1,2}$ with $chaos_{1,2}$ and, the obtained stable behavior by $order$, then the PS algorithm reveals Parrondo's paradoxical game in the following form: $chaos_1 + chaos_2 = order$ (i.e. chaos control).

Moreover, the PS algorithm can be considered as a generalization of Parrondo's game, since the scheme (2) can be considered for $N > 2$ values of $p_i, i = 1, 2, \dots, N$, corresponding to chaotic motions:

$$chaos_1 + chaos_2 + \dots + chaos_N = order.$$

If $p_{1,2}$ correspond to stable motions, and the obtained behavior is chaotic, then the PS algorithm leads to the following variant of Parrondo's game: $order_1 + order_2 = chaos$ (i.e. anticontrol).

Remark 2.2. The weights m_i can be omitted since $m_i \times chaos_i$, corresponding to the term $m_i p_i$, can be viewed as (another) chaotic motion - - an *absorption* like property.

For the case of $N = 2$, there are the possibilities presented in Table 1, where only the first two cases present the paradoxical character of Parrondo's game.

Table 1. Possible results with PS algorithm for $N = 2$.

	Control	Observations
$chaos_1 + chaos_2 = order$	Chaos control	Parrondo's game
$order_1 + order_2 = chaos$	Anticontrol	Parrondo's game
$order_1 + chaos = order_2$	Chaos control	-
$order + chaos_1 = chaos_2$	Anticontrol	-
$order_1 + order_2 = order_3$	Chaos control	-
$chaos_1 + chaos_2 = chaos_3$	Anticontrol	-

Similarly, one can imagine the same possibilities for the general case of $N > 2$.

3. PS algorithm applied to maps

In this section, some quantitative aspects of Parrondo's game are analyzed.

While for continuous systems modeled by (1), by applying Parrondo's game, the switched solution obtained with the PS algorithm applied to (1) converges to its averaged solution obtained by integration of (1) with p being replaced with p^* . However, for the discrete systems, things look different.

Let us consider the following discrete variant of the PS algorithm applied to (1)

$$x_{k+1} = f(x_k) + q_k Ax_k, \quad (5)$$

where, as stated in Section 2, $p_i \in \mathcal{P}_N$ with weights m_i for $i = 1, \dots, N$, $A \in L(\mathbb{R}^n)$, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ satisfies

$$|f(x_1) - f(x_2)| \leq L|x_1 - x_2| \quad \forall x_{1,2} \in \mathbb{R}^n,$$

for some $L > 0$. Also, in order to implement the switching of the parameter q_k with the PS algorithm, let us consider $\{q_k\}_{k \in \mathbb{N}}$ as a sequence of T -periodic piecewise constant, given by $q_k = p_i$ for $k \in [M_{i-1} + 1 + nT, M_i + nT]$, $M_0 = 0$, $M_i := \sum_{j=1}^i m_j$, $1 \leq i \leq N$, $n \in \mathbb{N}$, the period being $T := M_N$. The partition repeats periodically, covering the desired number of iterations in (5). For example, if we intend to apply the scheme $[2p_1, 3p_2]$, we have the partition $[1, 2]$, $[3, 5]$, $[6, 7]$, $[8, 10]$, ..., and for q : for $k \in [1, 2]$: $q_1 = p_1$, $q_2 = p_1$, for $k \in [3, 5]$: $q_3 = p_2$, $q_4 = p_2$, $q_5 = p_2$, and so on.

Unfortunately, compared to the case of continuous time systems modeled by (1), here there is no any relationship between the switched equation (5) and the averaged variant

$$x_{k+1} = f(x_k) + p^* Ax_k,$$

with p^* given, as in Section 2, by (4).

On the other hand, if one considers the following discrete version of the PS algorithm:

$$x_{k+1} = x_k + h(f(x_k) + q_k Ax_k), \quad (6)$$

whose averaged form is

$$\bar{x}_{k+1} = \bar{x}_k + h(f(\bar{x}_k) + p^* A\bar{x}_k), \quad (7)$$

then there exists already an averaging theory for difference equations (see e.g. [Dumas *et al*, 2004]). It allows us to introduce the following result which characterizes quantitatively the solutions of (6) and (7).

Theorem 1. *If*

$$p_{\max} := \max\{|p_j|, j = 1, \dots, N\},$$

$$\bar{p}_{\max} = \max \left\{ \left| \sum_{j=1}^k (q_j - p^*) \right|, k = 1, \dots, T \right\},$$

then

$$|x_k - \bar{x}_k| \leq \left(|x_1 - \bar{x}_1| + h\bar{p}_{\max} \|A\| (|\bar{x}_1| + (N+h)|f(0)|) e^{N(L+p^*\|A\|)} \right) \times e^{N(L+p_{\max}\|A\|)}, \quad (8)$$

for any $1 \leq k \leq \frac{N}{h} + 1$ and h a small positive real number.

See the proof in Appendix.

In order to implement Parrondo's game to the logistic map $f : [0, 1] \rightarrow [0, 1]$, $f(x) = px(1-x)$, $p \in [0, 4]$, using the scheme (2), the following simplest form will be used:²

$$x_{k+1} = q_k x_k (1 - x_k), \quad k = 0, 1, \dots \quad (9)$$

where, as defined above, $q_k = p_i$ for $k \in [M_{i-1} + 1 + nT, M_i + nT]$, $1 \leq i \leq N$ and $n \in \mathbb{N}$. For example the scheme $[2p_1, 1p_2]$ means that, by iterating with PS algorithm, one obtains $x_1 = p_1 x_0 (1 - x_0)$, $x_2 = p_1 x_1 (1 - x_1)$, $x_3 = p_2 x_2 (1 - x_2)$, $x_4 = p_1 x_3 (1 - x_3)$, $x_5 = p_1 x_4 (1 - x_4)$, ...

Consider now a sequence of logistic maps, $f_{p_n}(x) := p_n x(1-x)$ with $p_n \in (0, 4]$, $n \geq 0$.³ Let $f_{p^*}(x) := p^* x(1-x)$, p^* being some value within $(0, 4]$, $p^* \in (0, 4]$ and suppose that it has an exponentially stable

²In [Almeida *et al*, 2005] and [Fulai, 2012], particular forms of this kind of switching has been used to study behavior of the alternating orbits of the more accessible quadratic real (Mandelbrot) map $x_{k+1} = x_k^2 + p$, while in [Danca *et al*, 2009] the switches have been utilized to study the connectivity of alternating Julia sets.

³ p_n are switched with the PS algorithm via q_n as described above.

orbit $\{\bar{x}_n\}_{n=0}^\infty$, i.e. $\bar{x}_{n+1} = f_{p^*}(\bar{x}_n)$, $n \geq 0$, with $\prod_{i=0}^n |f'_{p^*}(\bar{x}_i)| \leq Ke^{-\alpha n}$ for any $n \geq 0$ and some positive constants K, α . We intend to find conditions that f_{p^*} controls the local dynamics near \bar{x}_0 of the PS system given by $\{f_{p_n}\}_{n=0}^\infty$. This is described in the next result.

Theorem 2. *Suppose the above assumptions hold. Set $P^* := \sup_{i \geq 0} |p_i - p^*|$. Let $r \geq \delta \geq 0$ satisfy*

$$K\delta + \frac{KP^*e^\alpha}{4(e^\alpha - 1)} + \frac{Kp^*e^\alpha}{e^\alpha - 1}r^2 \leq r. \quad (10)$$

If $|x_0 - \bar{x}_0| \leq \delta$, then $|x_n - \bar{x}_n| \leq r$ for any $n \geq 0$, where $x_{n+1} = f_{p_n}(x_n)$.

Proof. We have

$$\begin{aligned} |x_{n+1} - \bar{x}_{n+1}| &= |f_{p_n}(x_n) - f_{p^*}(\bar{x}_n)| \leq |f_{p^*}(x_n) - f_{p^*}(\bar{x}_n)| + |f_{p_n}(x_n) - f_{p^*}(x_n)| \\ &\leq |f'_{p^*}(\bar{x}_n)||x_n - \bar{x}_n| + p^*(x_n - \bar{x}_n)^2 + \frac{|p_n - p^*|}{4}, \end{aligned}$$

which implies

$$\begin{aligned} |x_n - \bar{x}_n| &\leq \prod_{i=0}^{n-1} |f'_{p^*}(\bar{x}_i)||x_0 - \bar{x}_0| + \sum_{k=0}^{n-1} \prod_{i=0}^{n-1-k} |f'_{p^*}(\bar{x}_i)| \left(p^*(x_k - \bar{x}_k)^2 + \frac{|p_n - p^*|}{4} \right) \\ &\leq Ke^{-\alpha(n-1)}|x_0 - \bar{x}_0| + \sum_{k=0}^{n-1} Ke^{-\alpha(n-1-k)} \left(p^*(x_k - \bar{x}_k)^2 + \frac{P^*}{4} \right) \\ &\leq K|x_0 - \bar{x}_0| + \frac{KP^*e^\alpha}{4(e^\alpha - 1)} + Kp^* \sum_{k=0}^{n-1} e^{-\alpha(n-1-k)}(x_k - \bar{x}_k)^2, \end{aligned} \quad (11)$$

for $n \geq 1$ and $P^* := \sup_{i \geq 0} |p_i - p^*|$.

By assumption, $|x_0 - \bar{x}_0| \leq \delta \leq r$. Suppose $|x_k - \bar{x}_k| \leq r$ for any $1 \leq k \leq n-1$. Then, (10) and (11) give

$$|x_n - \bar{x}_n| \leq K\delta + \frac{KP^*e^\alpha}{4(e^\alpha - 1)} + \frac{Kp^*e^\alpha}{e^\alpha - 1}r^2 \leq r.$$

So, the mathematical induction principle completes the proof. \blacksquare

Note that inequality (10) has a solution $r = 0$ for $\delta = 0$ and $P^* = 0$. As δ or P^* is increasing, r is also increasing, but there are upper bounds for δ and P^* , when (10) has no solution. We do not analyze this since it is elementary but awkward in notations. We note that the above arguments work also on finite-length orbit $\{\bar{x}_n\}_{n=0}^m$ when $\prod_{i=0}^n |f'_{p^*}(\bar{x}_i)| \leq Ke^{-\alpha n}$ for any $m \geq n \geq 0$ and some positive constants K, α . Then of course the statement of Theorem 2 is reduced on this finite orbit.

Remark 3.1.

(i) Theorems 2 and 1 show how the PS method acts differently for ODE and for maps. Accordingly, we are entitled to state, for the first time to our knowledge, that by applying Parrondo's game to the logistic map, the obtained (ordered or chaotic) dynamic is not a typical behavior for the logistic map, but only a close (similar) one.

(ii) It should be mentioned that this result can be directly extended to general maps, in higher dimensions. Then, the stability of the orbit can be replaced by its hyperbolicity, i.e. by an exponential dichotomy of the linearized equation. Moreover, we think that our result is also related to synchronization, since we try to find a map which locally synchronizes the PS system.

4. Chaos control of the logistic map with Parrondo's game

In this section, since the quantitative aspects of PS algorithm implementation have been explored in the previous section, we shall see qualitatively, aided by extensive numerical computations, that by applying Parrondo's game (via the PS algorithm), chaos control can be achieved.

To underline the stable character of the obtained orbits, bifurcation diagrams, time series, cobweb plots and first return map are plotted. The cobweb plots are drawn, by connecting alternately with horizontal and vertical lines the diagonal line and the N graphs corresponding to the considered N values.

Throughout the rest of this paper, all the orbits are plotted after the first thousand points have been discarded, and the behavior corresponding to the reached orbit with the PS algorithm will be denoted by *order**, or *chaos**. The orbits corresponding to p_i will be denoted by $\{x^i\}$.

In order to better understand the way in which the PS algorithm is used to implement Parrondo's game, let us first consider the simplest case of $N = 2$ with the scheme $[1p_1, 1p_2]$.

Since chaos control expressed in terms of Parrondo's game envisages the underlying behaviors of the orbits $\{x^i\}$, $i = 1, 2$, to be chaotic, we will focus mostly on those values of p_1 and p_2 which generate chaotic orbits. In order to facilitate their choice, for the particular case of $N = 2$, *switched bifurcation diagrams* will be used, which can be visualized by applying the PS algorithm for all p_1 , and p_2 , within the chaotic range $p \in [3.6, 4]$, where chaotic windows prevail (see Fig. 2 where, for the sake of the image quality, only nine values of p_1 have been considered).

One of these switched bifurcation diagrams, corresponding to $p_1 = 3.815$, is plotted in Fig. 3. There, one can see that by applying the PS algorithm using the scheme $[1p_1, 1p_2]$ with $p_2 = 3.857$, the result is a stable orbit as plotted in Fig. 4, where it is indicated by the dotted red line in the bifurcation diagram. In this case, we have $chaos_1 + chaos_2 = order^*$, i.e. we have achieved chaos control with Parrondo's game.

We recall that this bifurcation diagram (Fig. 3) is obtained with the PS algorithm but not in the classical way, where the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to $p = p_2 = 3.857$, are chaotic.

A more helpful tool to identify the points (p_1, p_2) , where Parrondo's game applies, is to plot the "Parrondo's basins", where the points verify, under the PS algorithm, one of the conditions: $chaos_1 + chaos_2 = order$ (chaos control), or $order_1 + order_2 = chaos$ (anticontrol).

Because we are more interested in chaos control with the Parrondo's game (see Table 1), the lattice of points (p_1, p_2) we have chosen is $[3.6, 4] \times [3.6, 4]$ (Fig. 5), where chaos prevails. Therefore, the points (p_1, p_2) , whose components $p_{1,2}$ generate chaotic orbits by iterating the logistic map in this lattice, are the majority and are plotted in green. The rest of the points (p_1, p_2) which, separately, generate stable orbits by iterating the logistic map, are plotted in white.

The points (p_1, p_2) , which were obtained via the PS algorithm, generate stable orbits (i.e. chaos control) and are plotted in red.

The distinction of stable\chaotic obits has been determined by calculating the Lyapunov exponent.

As can be seen, there are red-green points, where the red points are plotted over the green points, and red-white points, where the red points are plotted over the white points.

"Parrondo's chaos control basins" obtained by using the PS algorithm within this lattice, are the red-green points .

As can be seen from the numerically obtained results shown in Fig. 5 (see also the detailed D presented in Fig. 6), Parrondo's basins verify the following result.

Property 3. The sets of points (p_1, p_2) verifying Parrondo's paradox have the following properties:

- (a) they form a fractal structure;
- (b) they form connect regions.

The same properties are suspected to hold for the general case of $N > 2$.

Now, we can see that the scheme used above, $[1p_1, 1p_2]$ with $p_1 = 3.815$ and $p_2 = 3.857$ (see the corresponding point $M_1(p_1, p_2)$ in Fig. 5), leads to Parrondo's game and it belongs to a red-green basin, i.e. belongs to Parrondo's chaos control basin.

If one chooses a point within a (rectangular) red-white region (e.g. the point $M_2(3.84, 3.844)$), a stable cycle is obtained with the PS algorithm (due to the red region), but the underlying values p_1 and p_2 , by iterating the logistic map, generate stable orbits (due to the white region). The stable orbit is plotted in Fig. 7 a, where, because the orbits are almost identic, they are slightly translated. Now, we have $order_1 + order_2 = order^*$, which means that we have achieved chaos control (in the sense defined above), but not Parrondo's game.

A stable orbit with a higher period obtained with Parrondo's game is plotted in Fig. 8, which was obtained with $p_1 = 3.6075$ and $p_2 = 3.7$. The corresponding point, M_3 , belongs to a red-green region (Fig. 5).

As expected, Parrondo's game can be used to model anticontrol too. For example, for the scheme $[1p_1, 1p_2]$ with $p_1 = 3.74$, $p_2 = 3.845$, chosen in some white and exterior of red regions (point M_4 in Fig. 5), the obtained chaotic orbit is plotted in Fig. 9, emphasizing the paradoxical character of the game in this case.⁴

An example of anticontrol, where Parrondo's game does not apply, is presented in Fig. 10, which corresponds to the point $M_5(3.8, 3.7)$ situated in a green region.

Remark 4.1. As can be seen, there exists a positive probability to realize chaos control by using the (generalized) Parrondo's game. For example, for the considered lattice scanned with step size 0.001 (which gives 16×10^4 points), the chance to find a pair (p_1, p_2) generating a stable cycle via Parrondo's game, is about 15%. This property is favored by the fractal structure of the lattice (Property 3 (a)), which can be seen in the enlarged detailed D in Fig. 6. Therefore, almost everywhere, there are points (p_1, p_2) generating stable orbits with the PS algorithm. Moreover, due to the connectivity of Parrondo's basins (Property 3 (2)), near every point verifying Parrondo's game, there are infinitely many other points verifying that property.

Not only the "classical" form of Parrondo's game (implemented with the scheme $[1p_1, 1p_2]$) can be used to obtain stable orbits. For example, for $N = 6$, with $m_i = 1, i = 1, \dots, 6$, and with $\mathcal{P}_6 = \{3.6, 3.7, 3.75, 3.8, 3.86, 3.9\}$, one obtains a stable 6-periodic orbit (Fig. 11a). Therefore, in this case, we have a generalized Parrondo's game with $\sum_{i=1}^6 chaos_i = order^*$, since all the orbits entering into these relations are chaotic. If we change the weights m_i , $m_1 = 3, m_2 = 1, m_3 = 5, m_4 = 2, m_5 = 1, m_6 = 10$, and maintain the \mathcal{P}_6 set as before, or we use the set $\mathcal{P}_6 = \{3.4, 2.35, 3.5, 2.8, 1.9, 0.85\}$ with $m_1 = 10, m_2 = 1, m_3 = 5, m_4 = 10, m_5 = 1, m_6 = 10$, we obtain the periodic bursts shown in Fig. 11b and 11c, respectively. In this case, we do not have Parrondo's game, since few of the orbits corresponding to p_i are stable but most others are chaotic.

Conclusion

In this paper, we have shown that Parrondo's game can be implemented for the logistic map, via the PS algorithm for continuous-time systems, to obtain stable orbits. As stated by Theorem 2, the obtained switching orbits are close to, but different from the generic orbits of the logistic map.

Numerically, we found that there exists a positive probability to obtain stable orbits with Parrondo's game and that the set of points (p_1, p_2) , which lead to chaos control with Parrondo's game, have a fractal structure.

Since several parameter values ($N > 2$) can be used, the PS algorithm can be considered as a way to obtain a generalized variant of Parrondo's game $chaos_1 + chaos_2 + \dots + chaos_N = order$.

A generalized Parrondo's game for anticontrol can also be implemented, if the underlying values generate stable orbits and the obtained orbit is chaotic: $order_1 + order_2 + \dots + order_N = chaos$.

We also show that the PS algorithm can be used for some other classes of general discrete systems.

Appendix

Proof. [Proof of Theorem 1] Let $1 \leq k \leq \frac{N}{h} + 1$. From (7), we have

$$|\bar{x}_{k+1}| \leq (1 + h(L + |p^*| \|A\|)) |\bar{x}_k| + h|f(0)|,$$

⁴The "Parrondo's anticontrol basins" are situated within the white regions, but they have not been plotted due to the scope of this work and in order not to overload the image.

which implies

$$\begin{aligned} |\bar{x}_{k+1}| &\leq |x_1| + h(L + |p^*||A|) \sum_{j=1}^k |\bar{x}_j| + kh|f(0)| \\ &\leq |\bar{x}_1| + (N + h)|f(0)| + h(L + |p^*||A|) \sum_{j=1}^k |\bar{x}_j|. \end{aligned}$$

Then applying discrete Gronwall inequality [Elaydi, 2005], we obtain

$$\begin{aligned} |\bar{x}_k| &\leq (|\bar{x}_1| + (N + h)|f(0)|)e^{(k-1)h(L+|p^*||A|)} \\ &\leq (|\bar{x}_1| + (N + h)|f(0)|)e^{N(L+|p^*||A|)}. \end{aligned} \quad (12)$$

Next, (6), (7) and (12) give

$$\begin{aligned} |x_{k+1} - \bar{x}_{k+1}| &\leq |x_1 - \bar{x}_1| + h \sum_{j=1}^k L|x_j - \bar{x}_j| \\ &\quad + h \left| \sum_{j=1}^k (q_j - p^*)||A||\bar{x}_j| + h \sum_{j=1}^k |q_j||A||x_j - \bar{x}_j| \right. \\ &\leq |x_1 - \bar{x}_1| + h\bar{p}_{\max}\|A\|(|\bar{x}_1| + (N + h)|f(0)|)e^{N(L+|p^*||A|)} \\ &\quad \left. + h(L + p_{\max}\|A\|) \sum_{j=0}^k |x_j - \bar{x}_j|, \end{aligned} \quad (13)$$

since $k \rightarrow \sum_{j=1}^k (q_j - p^*)$ is T -periodic on \mathbb{N} . Then, again, applying discrete Gronwall inequality to (13), we obtain

$$\begin{aligned} |x_k - \bar{x}_k| &\leq \left(|x_1 - \bar{x}_1| + h\bar{p}_{\max}\|A\|(|\bar{x}_1| + (N + h)|f(0)|)e^{N(L+|p^*||A|)} \right) \\ &\quad \times e^{h(k-1)(L+p_{\max}\|A\|)}, \end{aligned}$$

which implies (8). The proof is completed. ■

Remark 4.2.

(i) We derive from the above proof that

$$|x_k - \bar{x}_k| \leq (|x_1 - \bar{x}_1| + h\bar{p}_{\max}\|A\| \max_{1 \leq j \leq k} |\bar{x}_j|)e^{(k-1)h(L+p_{\max}\|A\|)} \quad (14)$$

for any $k \in \mathbb{N}$.

(ii) We can consider, in the above discussion, any globally Lipschitz map $A : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

References

- Abbot, D. [2010] “Asymmetry and disorder: a decade of Parrondo’s paradox,” *Fluct Noise Lett* **9**, 129-156.
- Almeida, J., Peralta-Salas, D., and Romera, M., [2005] “Can two chaotic systems give rise to order?” *Physica D* **200**, 124-132.
- Amengual, P., Allison, A., Toral, R., & Abbott, D. [2004] “Discrete-time ratchets, the Fokker-Planck equation and Parrondos paradox,” *Proc R Soc Lond Ser A Math Phys Eng Sci* **460**, 2269-84.
- Blackwell D. & Girshick M.A. [1954] *Theory of Games and Statistical Decisions* (John Wiley & Sons).
- Casti, J.L. [2001] “Losing games for your winning plays,” *Complexity* **6**(6), 11-14.
- Chen, G. & Ueta, T. [1999] “Yet another chaotic attractor,” *Int. J. Bifurcation and Chaos* **9**, 1465–1466.
- Danca, M.-F., Romera, M. Pastor, G., [2009] “Alternated Julia sets and connectivity properties,” *Int. J. Bifurcation and Chaos* **19**(6), 2123-2129.
- Danca, M.-F., Lai, D. [2012] “Parrondo’s game model to find numerically the stable attractors of a tumor growth model,” *Int. J. Bifurcation and Chaos* **22**(10), 1250258.
- Danca, M.-F., Romera, M. Pastor, G. & Montoya, F. [2012] “Finding attractors of continuous-time systems by parameter switching,” *Nonlinear Dynamics* **67**(4), 2317-2342.
- Danca, Marius-F. [2013] “Convergence of a parameter switching algorithm for a class of nonlinear continuous systems and a generalization of Parrondos paradox,” *Commun Nonlinear Sci Numer Simulat* **18**, 500-510.
- Davies, P. C. W. [2001] “Physics and life: the Abdus Salam memorial lecture,” in *6th Trieste Conference on Chemical Evolution. Kluwer Academic Publishers, Chela-Flores, J., Tobias, D. and Raulin F. (Eds.)*, 13-20.
- Dumas, H. S., Ellison, J. A. & Vogt, M. [2004] “First-order averaging principles for maps with applications to accelerator beam dynamics”, *SIAM J. Appl. Dyn. Sys.* **3**, 409–432.
- Elaydi, S. [2005] *An Introduction to Difference Equations*, 3rd ed. Springer.
- Fulai, W. [2012] “Improvement and empirical research on chaos control by theory of chaos+chaos=order”, *CHAOS* **22**, 043145.
- Harmer, G.P. & Abbott, D. [1999a] “Game theory: Losing strategies that can win by Parrondos paradox,” *Nature* **402**, 864.
- Harmer, G.P. & Abbott, D. [1999b] “Parrondos paradox,” *Statistical Sci* **14**(2), 206-213.
- Heath, D., Kinderlehrer D, & Kowalczyk M. [2002] “Discrete and continuous ratchets: from coin toss to molecular motor,” *Discr Cont Dynam Syst-B* **2**, 153-67
- Raghuram, I. & Rajeev K. [2003] “Why Parrondos paradox is irrelevant for utility theory, stock buying, and the emergence of life; Can losses be combined to give gains?” *Complexity* **9**(1), 23–27.
- Mao, Y., Tang W.K.S. & Danca, Marius-F. [2010] “An averaging model for chaotic system with periodic time-varying parameter,” *Applied Mathematics and Computation* **217**(1), 355–362.
- Lee Y, Allison A, Abbott D, Stanley H.E. [2003] “Minimal Brownian ratchet: an exactly solvable model,” *Phys Rev Lett* **91**, 220601.
- Parrondo, J.M.R., Harmer, G.P., Abbott, D. [2000] “New paradoxical games based on Brownian ratchets,” *Phys Rev Lett* **85**(24), 5226-5229.

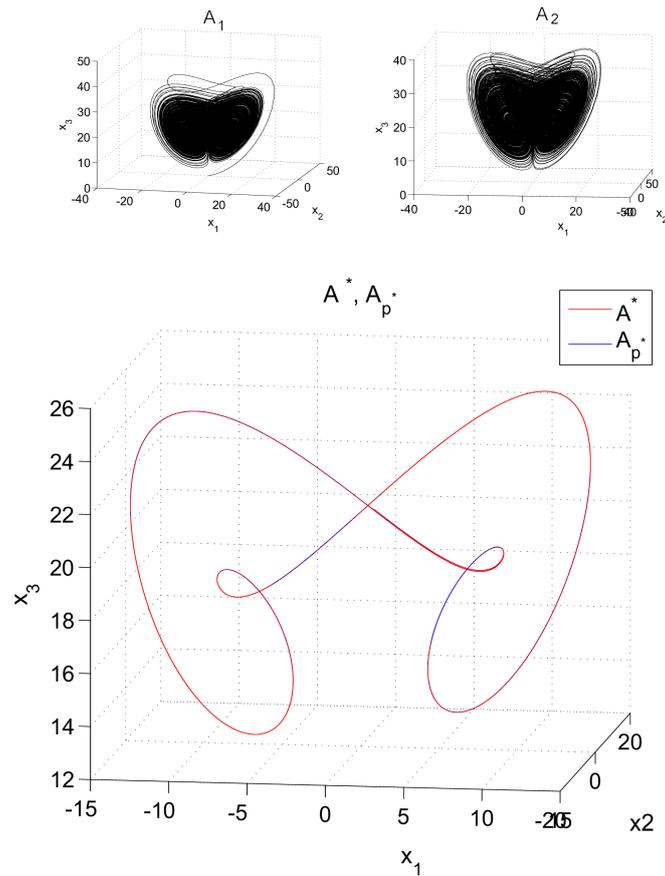


Fig. 1. Stable limit cycle of the Chen system (6), corresponding to $p = 26.09$, and approximated by the PS algorithm with the scheme $[1p_1, 2p_2]$, $p_1 = 25.75$ and $p_2 = 26.26$. Up: the chaotic attractors corresponding to p_1 and p_2 . Down: the switched attractor A^* (red) and averaged attractor A_{p^*} (blue).

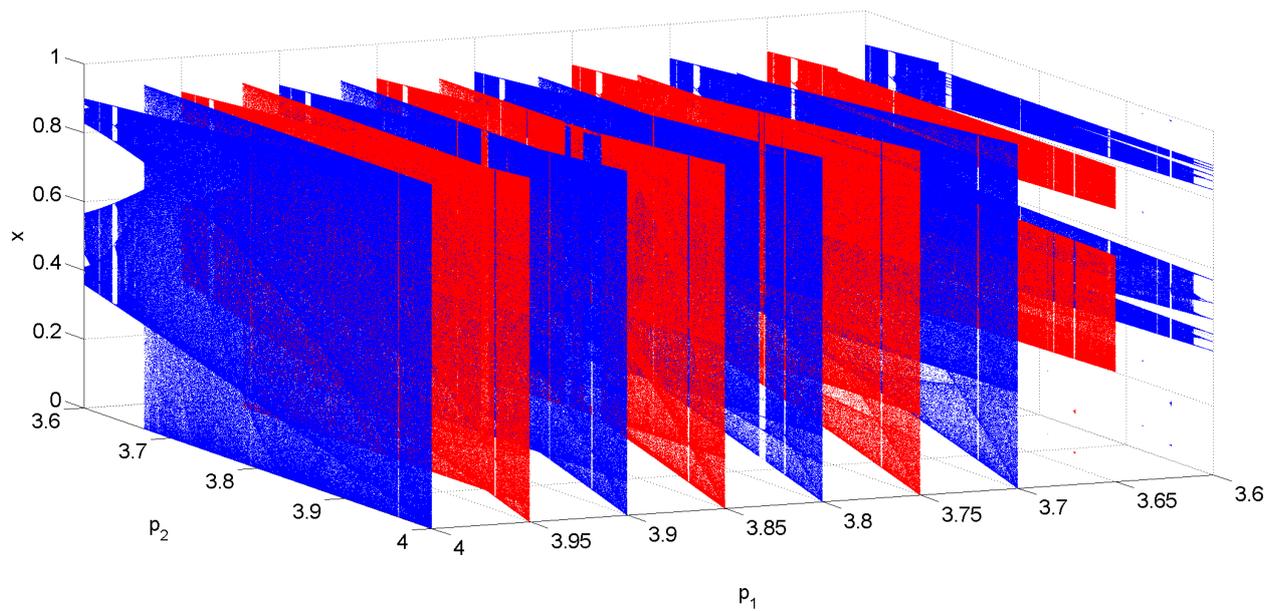


Fig. 2. Switched bifurcation diagram for $p_{1,2} \in [3.6, 4]$. For the sake of clarity, only few diagrams, corresponding to nine values of p_1 , have been plotted.

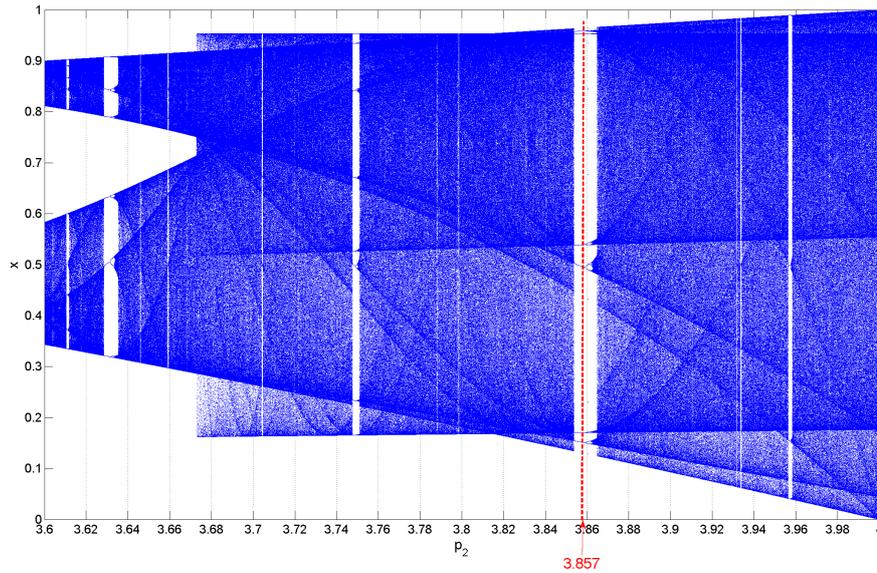


Fig. 3. A switched bifurcation diagram, corresponding to $p_1 = 3.815$. This diagram reveals the fact that for $p_1 = 3.815$ and $p_2 = 3.857$, the obtained switching orbit is stable (even $p_{1,2}$ generate chaotic orbits).

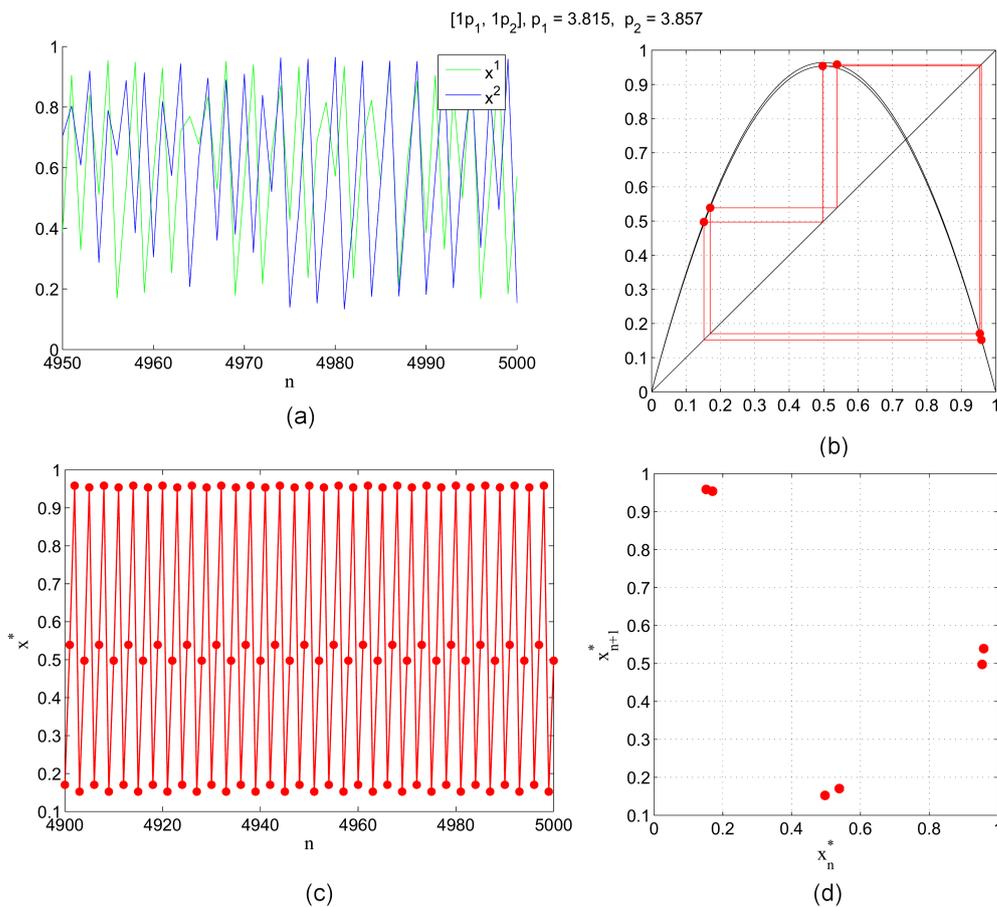


Fig. 4. Stable orbit obtained with Parrondo's game: $Chaos_1 + Chaos_2 = Order^*$, via the scheme $[1p_1, 1p_2]$ with $p_1 = 3.815$ and $p_2 = 3.857$ (see the corresponding point M_1 in the red-green region, Fig. 5). (a) Time series of the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to p_1 and p_2 . (b) Cobweb plot of the switching orbit. (c) Time series of the switching orbit. (d) First return map of the switching orbit.

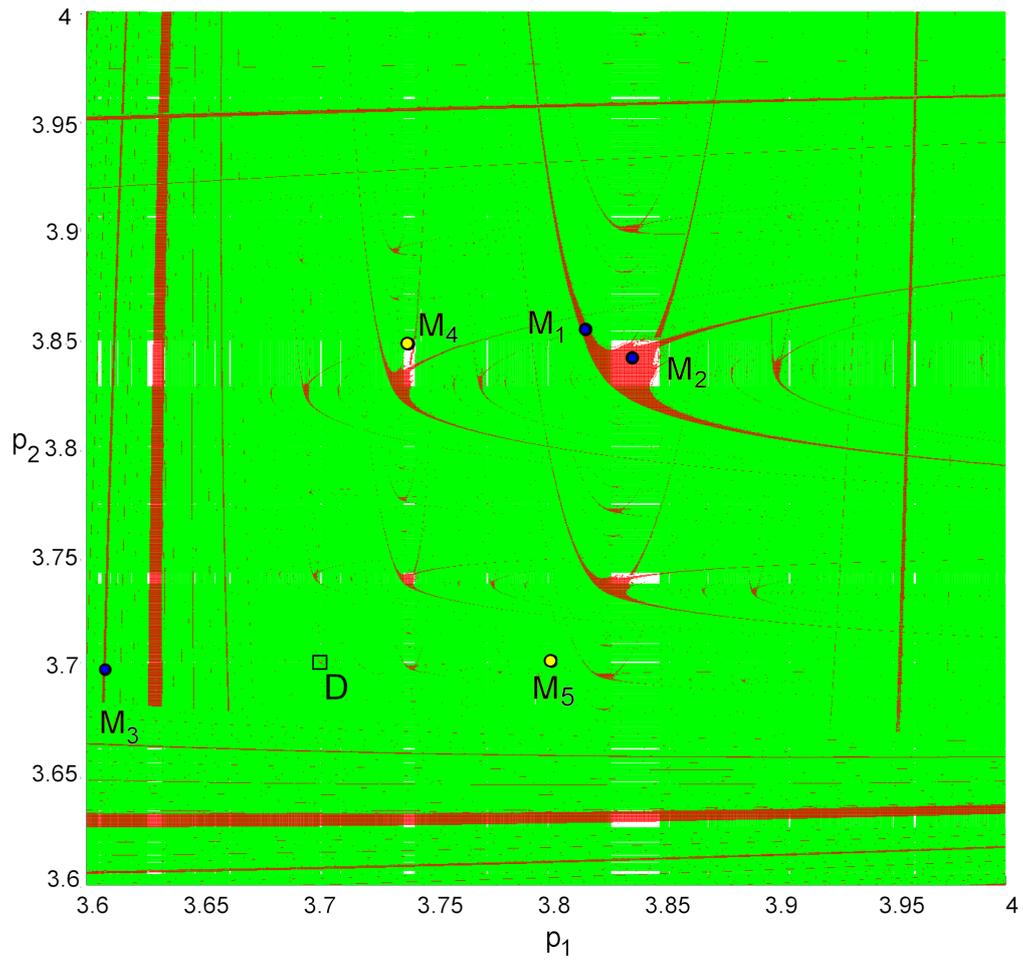


Fig. 5. The lattice of points $(p_1, p_2) \in [3.6, 4] \times [3.6, 4]$, showing the (red-green) points (p_1, p_2) , which lead to Parrondo's game.

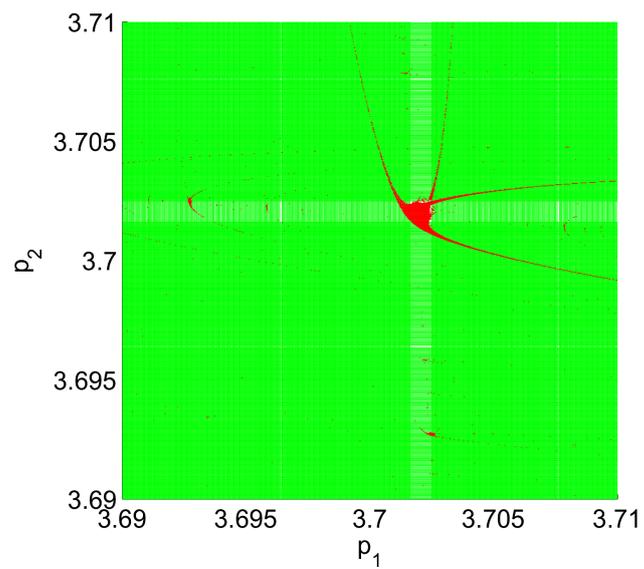


Fig. 6. Enlarged view of the detailed D in the lattice in Fig. 5.

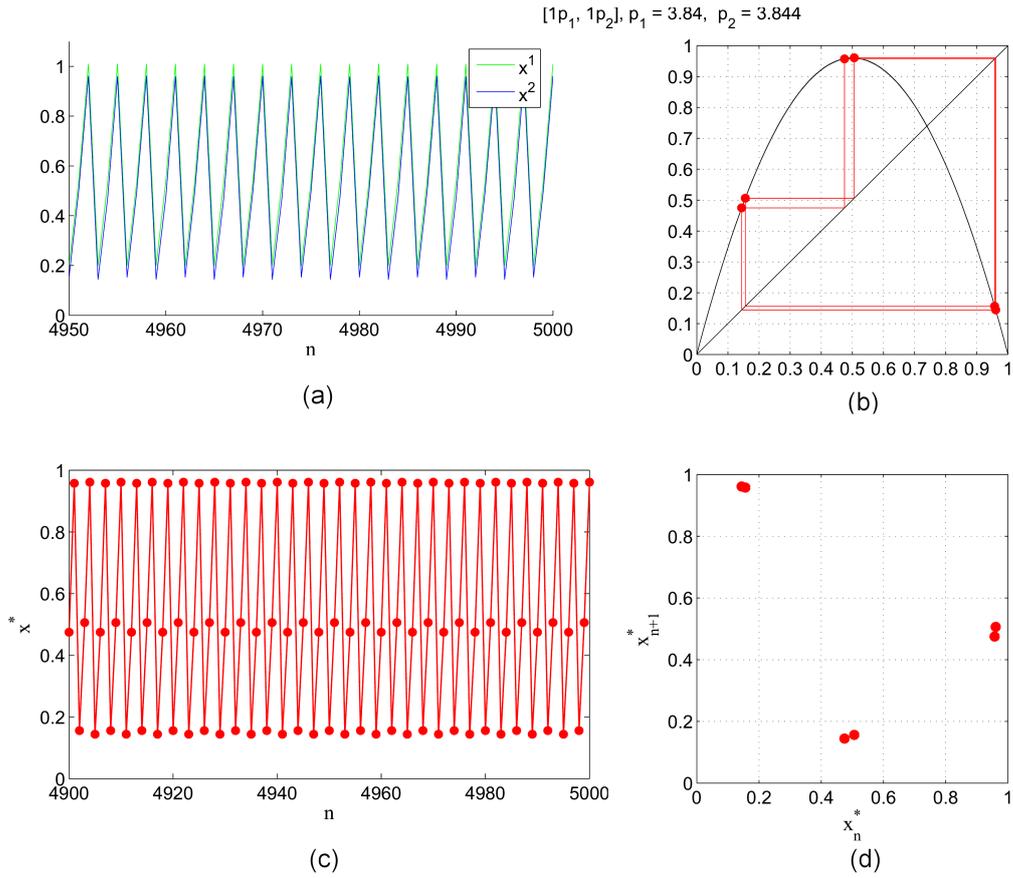


Fig. 7. Stable orbit obtained with the scheme $[1p_1, 1p_2]$ with $p_1 = 3.84$ and $p_2 = 3.844$ (see the corresponding point M_2 in the red-white region, Fig. 5). Now, $Order_1 + Order_2 = Order^*$ and Parrondo's game do not apply in this case. (a) Time series of the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to p_1 and p_2 . (b) Cobweb plot of the switching orbit. (c) Time series of the switching orbit. (d) First return map of the switching orbit.

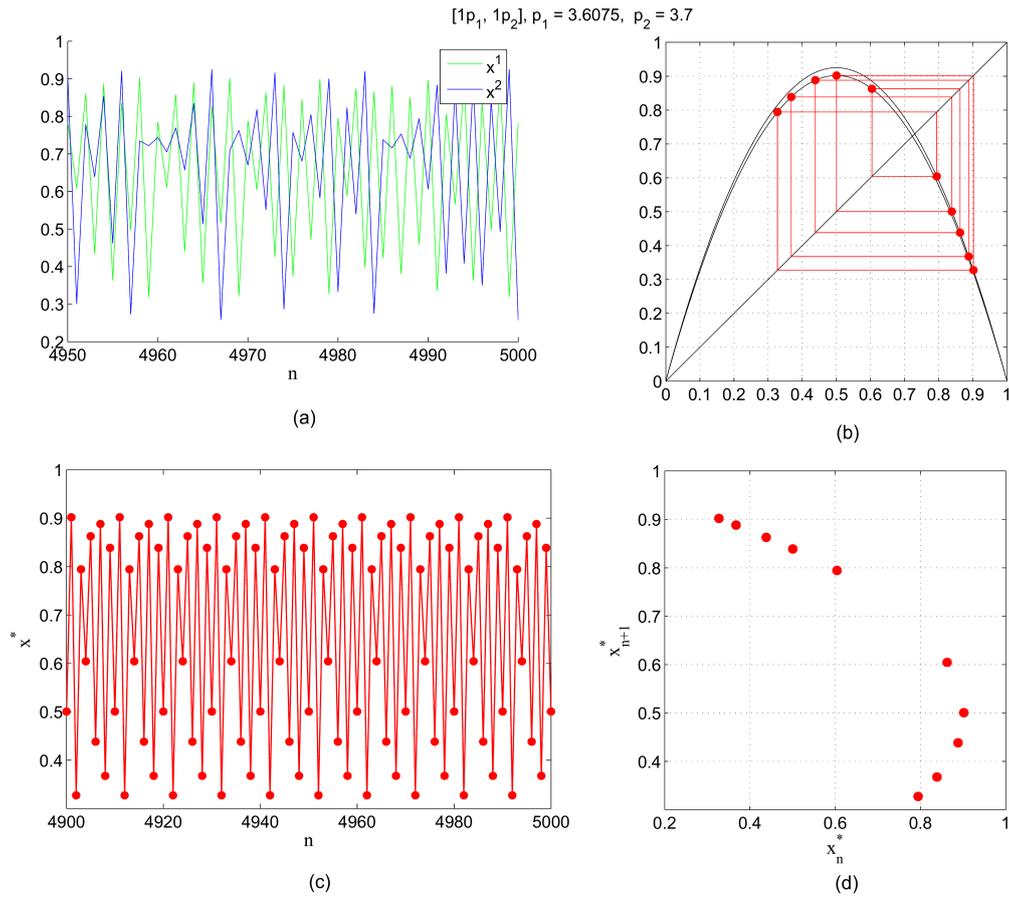


Fig. 8. Higher periodic stable orbit obtained by Parrondo's game: $Chaos_1 + Chaos_2 = Order^*$, using the scheme $[1p_1, 1p_2]$, with $p_1 = 3.6075$ and $p_2 = 3.7$ (see the corresponding point M_3 in red-green region, Fig. 5). (a) Time series of the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to p_1 and p_2 . (b) Cobweb plot of the switching orbit. (c) Time series of the switching orbit. (d) First return map of the switching orbit.

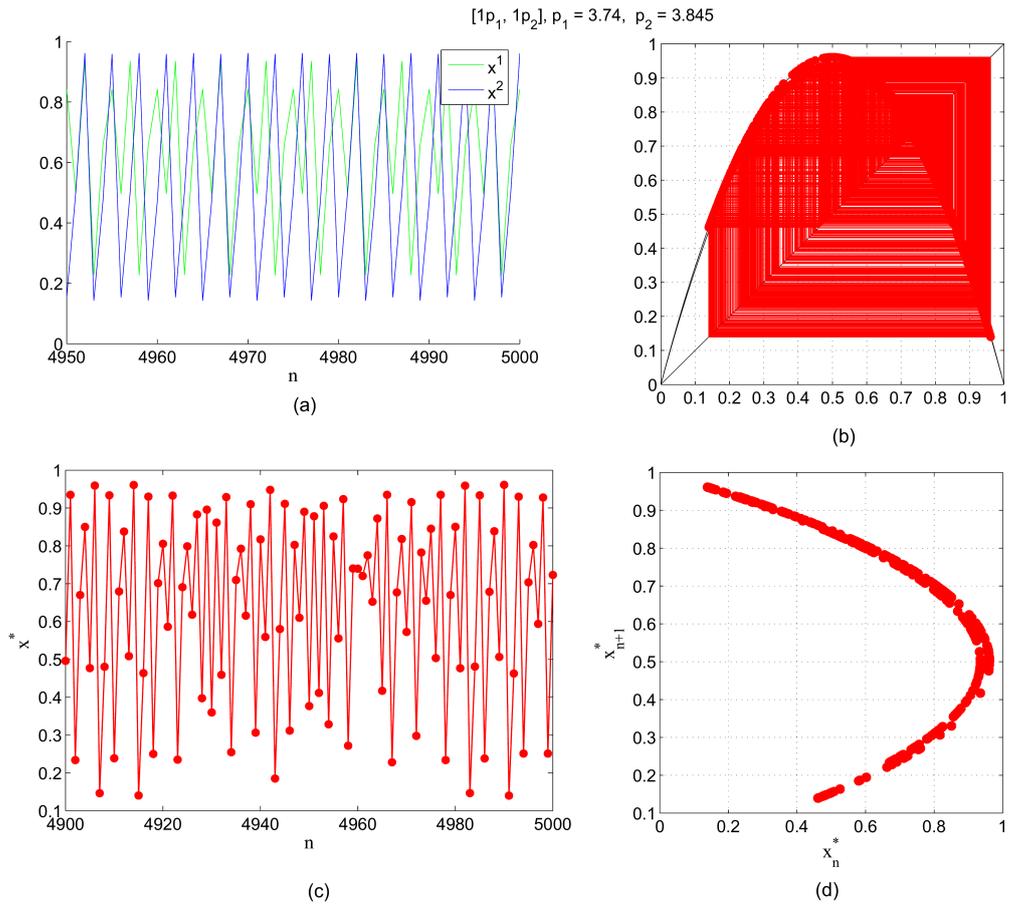


Fig. 9. Chaotic orbit obtained with Parrondo's (anticontrol) game: $Order_1 + Order_2 = Chaos^*$, using the scheme $[1p_1, 1p_2]$, with $p_1 = 3.74$, $p_2 = 3.845$ (see the corresponding point M_4 in white region, Fig. 5). (a) Time series of the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to p_1 and p_2 . (b) Cobweb plot of the switching orbit. (c) Time series of the switching orbit. (d) First return map of the switching orbit.

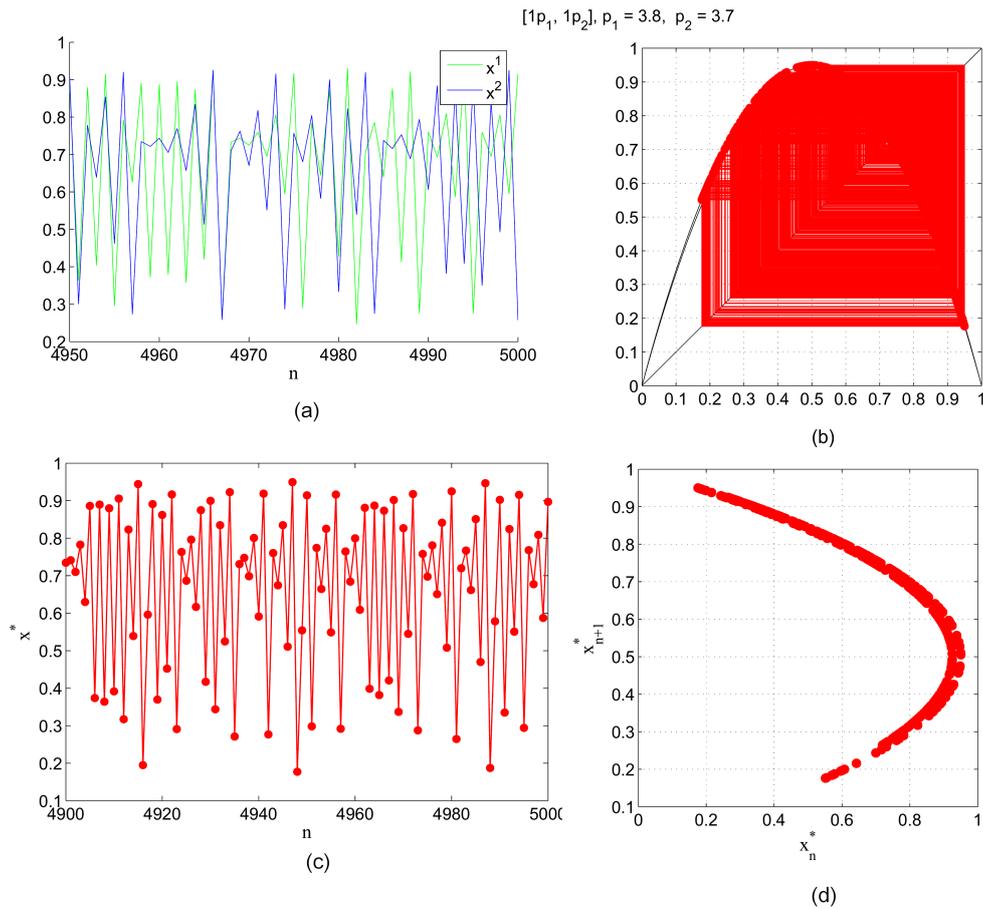


Fig. 10. Chaotic orbit obtained by the scheme $[1p_1, 1p_2]$ with $p_1 = 3.8$ and $p_2 = 3.7$ (see the corresponding point M_5 in green region Fig. 5). Now, $Chaos_1 + Chaos_2 = Chaos^*$ and Parrondo's game do not apply in this case. (a) Time series of the orbits $\{x^1\}$ and $\{x^2\}$, corresponding to p_1 and p_2 . (b) Cobweb plot of the switching orbit. (c) Time series of the switching orbit. (d) First return map of the switching orbit.

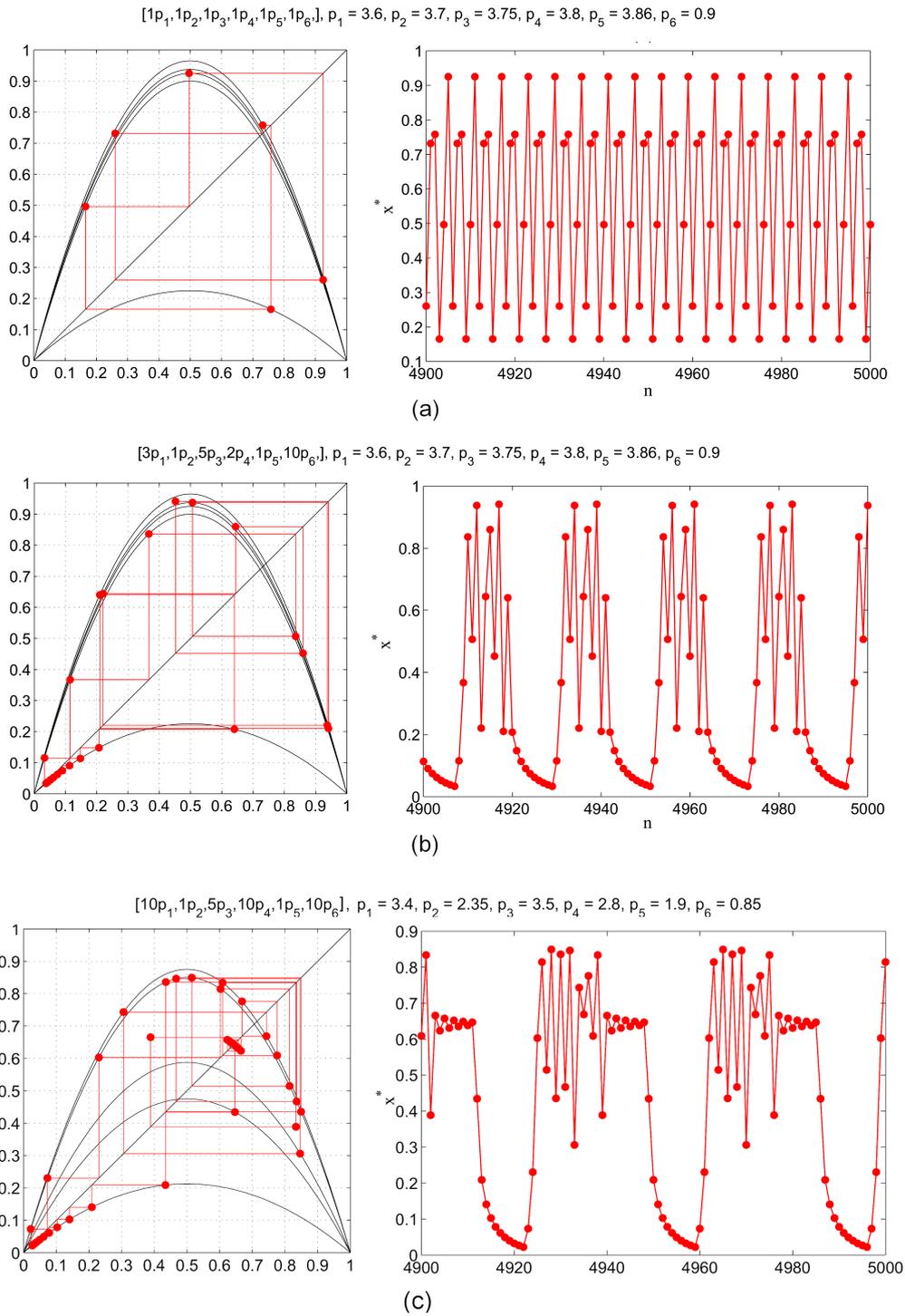


Fig. 11. Stable orbits obtained with the generalized Parrondo's game for $N = 6$ using scheme $[m_1p_1, m_2p_2, \dots, m_6p_6]$ ($\sum_{i=1}^6 chaos_i = order^*$). (a) $\mathcal{P}_6 = \{3.6, 3.7, 3.75, 3.8, 3.86, 3.9\}$ and $m_i = 1, i = 1, \dots, 6$. (b) Same set $\mathcal{P}_6 = \{3.6, 3.7, 3.75, 3.8, 3.86, 3.9\}$ with $m_1 = 3, m_2 = 1, m_3 = 5, m_4 = 2, m_5 = 1, m_6 = 10$. (c) $\mathcal{P}_6 = \{3.4, 2.35, 3.5, 2.8, 1.9, 0.85\}$ and $m_1 = 10, m_2 = 1, m_3 = 5, m_4 = 10, m_5 = 1, m_6 = 10$.