

Convergence of a parameter switching algorithm for a class of nonlinear continuous systems and a generalization of Parrondo's paradox

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*

Abstract

In this paper, we prove the convergence of a numerical algorithm that switches in some deterministic or random manner, the control parameter of a class of continuous-time nonlinear systems while the underlying initial value problem is numerically integrated. The numerically obtained attractor is a good approximation of the attractor obtained when the control parameter is replaced with the average of the switched values. In this way, a generalization of Parrondo's paradox can be obtained. As an application, the Lorenz and Rabinovich–Fabrikant systems are used for illustration.

Keywords: Parameter Switching, Parrondo's paradox, Attractor, Lorenz system, Rabinovich–Fabrikant system
2000 MSC: 37M05, 37D45, 65L20

1. Introduction

As is well known, Parrondo's paradox was named after the Spanish physicist J.M.R. Parrondo in 1996 and affirms that two losing games together can be set up to produce a winning scenario.

This paper is concerned with the convergence of a parameter switching algorithm for a class of continuous-time nonlinear systems and its connection

Email address: danca@rist.ro (Marius-F. Danca)

with Parrondo's paradox.

In [1], Parrondo et al. showed that alternating randomly or deterministically the losing gains of two games, one can actually obtain a winning game with a positive gain, i.e.: "losing + losing = winning" (see also [2]) or, in other words: "two ugly parents can have beautiful children" (Zeilberger, on receiving the 1998 Leroy P. Steele Prize). Since its discovery, this apparent contradiction has been known as *Parrondo's paradox*, and has become an active area of research for example in minimal Brownian ratchet [3], discrete-time ratchets [4], molecular transport [5], and so on. In its original form, Parrondo's game can be considered as game theory in the Blackwell sense [6]. Recently, in [7, 8] Parrondo's original games were extended to include player strategy (for a recent review of the history of Parrondo's paradox, developments, and connections to related phenomena, see [9]).

This kind of alternation weakness-strength, order-chaos, losing-winning and so on, can be found or induced in many physical, biological, quantum, mathematical systems, control theory, or even fractals, where combining processes may lead to counterintuitive dynamics. This apparently trivial phenomenon seems to be typical not only for theoretical systems but also in nature, where there are many interactions due to some accidental or intentional parameter switches. Even more, there is a belief that this kind of mechanism could be used even as a possible explanation of the origin of life [10].

While almost all the systems in the mentioned applications are of discrete-time (possibly chaotic) and imply only a dichotomic like alternation between two states: losing and winning, some intensive numerical tests have been realized on some classes of nonlinear continuous-time systems in the state space \mathbb{R}^n with $n \geq 2$, of integer or fractional order, continuous or discontinuous with respect to the state variable and depending linearly on a single real parameter p (defined in Section 2). It was shown that alternations of $N \geq 2$ different behaviors (with N being of order ten or even higher), can lead to a generalized Parrondo like paradox (as can be seen in [11]). By dealing with $N \geq 2$ behaviors, however, one actually has no "alternations" but "switchings".

Let us consider an Initial Value Problem modeling a nonlinear continuous-time system. By switching p while numerically integrating the IVP, via an algorithm called Parameter Switching (PS) (described in Section 2), inside a chosen set of values, under some reasonable mild conditions for most chaotic systems (such as uniqueness and boundedness), one obtains a behavior which

is approximately identical to the one obtained by integrating the IVP with p being replaced by the average of the switched values (several examples can be found in [11]).

If we replace in Parrondo's paradox the word "loosing" with "chaos" and, by a minor abuse of notation "winning" with "regular" (as the opposite of "chaos")¹ then, by switching p within two values using the PS algorithm we might have one of the following, perhaps surprising, situations: "chaos+chaos=regular", "regular+regular=chaos", and some other possible (maybe less spectacular) combinations; in other words, a Parrondo's like paradox for continuous-time systems.

Moreover, if with the PS algorithm p is switched within the set $P_N = \{p_1, p_2, \dots, p_N\}$, for some positive integer $N \geq 2$, which corresponds to the regular or chaotic behaviors $A(p_i)$ (distinct for different i), then we obtain a generalization of Parrondo's game. As shown in Section 4, this could be written formally as follows: $\sum_{i=1}^N A(p_i) = A^*$, where A^* is a well determined behavior corresponding to $p = p^*$ (the average value of p_i (see Section 2)). The values p^* , p_i , and also the behaviors A^* , A_i are, respectively, different from each other.

Here, we note that compared to discrete-time systems, where alternations do not always lead to characteristic behaviors and may modify the underlying system (as shown in Danca et al. [12], Loskutov [13], Almeida et al. [14] and Romera et al. [15]), for the considered continuous-time cases, the PS algorithm always enhances one of the possible behaviors of the considered system [11].

It is easy to understand that the PS algorithm can be used as a kind of control or anticontrol (chaotification) technique, in the sense that any kind of possible chaotic or regular behaviors of some system in concern can be enhanced. Compared to the classical algorithms for control or anticontrol of chaos, where tedious calculus is necessary, the PS algorithm can be easily implemented (Section 4).

Besides the fact that the PS algorithm allows to extend Parrondo's algorithm to general classes of systems, its utility resides in the possibility to control or anticontrol chaos in the following sense: suppose that some desired targeted value p^* for p (for which the considered system behaves regular or

¹Apparently, a most adequate notation would be "chaos" and "order" but, as known, there is order although few decades ago scientists thought there is only chaos or disorder.

chaotically) cannot be accessed. Then, as shown in Section 2, by choosing an appropriate set P_N of other (accessible) values for p , p^* can be easily obtained with the PS algorithm so that the intended control or anticontrol is achieved. Reversely, given some set P_N , by switching p within this set in whatever manner (deterministically or randomly), one obtains one a possible behavior of the considered system.

The article is organized as follows: Section 2 presents the PS algorithm, Section 3 proves the PS convergence and, in Section 4, the Lorenz and Rabinovich–Fabrikant systems are considered as an application. Finally, conclusions and open problems are presented.

2. Preliminaries and PS algorithm

2.1. Utilized notions

A mathematical model of the considered class of continuous nonlinear systems, is generated by a system of autonomous ordinary differential equations on \mathbb{R}^n . The corresponding Initial Value Problem (IVP) has the following defining equations and initial conditions

$$\dot{x}(t) = f(x(t)) := g(x(t)) + pAx(t), \quad t \in I = [0, T], \quad x(0) = x_0, \quad (1)$$

where f is an n -dimensional vector-valued function with $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ a nonlinear (at least) continuous vector-valued function, $x_0 \in \mathbb{R}^n$, $A \in \mathcal{M}_n(\mathbb{R})$ a non-zero square matrix with real entries, and $p \in \mathbb{R}$ the control (bifurcation) parameter.

Most known chaotic dynamical systems such as Lorenz, Rösler, Chen, Lotka-Volterra, Rabinovich–Fabrikant (system presented by Danca and Chen in [16]), Hindmarsh-Rose, Lü, some classes of minimal networks and many others, can be modeled by the IVP (1).

Because in practical examples, there is a large number of nonlinear systems which do not satisfy globally Lipschitz condition, such as the Lorenz-like systems, for uniqueness a locally Lipschitz condition and apriori bound on the solutions (such that the solutions do not blow up in any finite time) required (see, for example, [17]). As known, unboundedness usually causes difficulty in defining an appropriate notion of attractor (e.g. omega-limit sets can be empty). Therefore, we consider the following common assumption

Assumption 1. *Given an initial condition x_0 , to every p there corresponds a unique bounded solution to IVP (1).*

Remark 1.

- i) As known, attractors present continuous dependence on a parameter. Roughly speaking, the dependence of the solution of the IVP (1) on the parameter p is continuous as long as the function f is so (see e.g. [18] or [19, p. 83]). Therefore, under the above assumption, the PS algorithm does not affect the solution continuity;*
- ii) Instead of the boundness condition, one can use the dissipative property, which is also an essential prerequisite for the existence of global attractors (see e.g. [20], [21]). Thus, if $f \in C^1$, the simplest form for dissipativity is given via the divergence (Gauss) theorem, i.e. the trace of the Jacobian: $\sum_{i=1}^n \frac{\partial f(x)}{\partial x_i} < 0$ (see, for instance, Chapter 16.9 in Stewart's book [22]). It is to mention that all analyzed systems in this paper are dissipative.*

Roughly speaking, in this paper the *global attractor* (notion borrowed from PDEs) will be considered in a state space region of a dynamical system characterizing all the long-time dynamics of the underlying equation. Note that the trajectories can enter but not leave, containing no smaller such region (for the background of attractors see, for example, [23], [24], [21], [25]). In other words, it can be considered as containing all the solutions, including chaotic solutions as well as stable stationary and periodic solutions.

The term *local attractor* (as mentioned by Stuart and Humphries in [21, p. 83]) is used sometimes for attractors which are not global attractors. For a given parameter p , function on initial conditions, the global attractors may contain several local attractors. Therefore, a global attractor can be viewed as being "composed" of the set of all local attractors separated by the attraction basins (i.e. the initial conditions). If the local attractor is unique, then it coincides with the global attractor (called, in this case, *maximal attractor*).

The best way to simulate Parrondo's paradox for systems modeled by (1) is to approximate numerically the system attractors, while the PS algorithm is applied (see, for example, [26] for numerically approximations of global attractors).

Already it is well known that if a single trajectory is numerically approximated over a long time, the approximating trajectory can diverge from the true trajectory. Nevertheless, we can suppose that the numerical solution remains close enough to some exact solution for a reasonably long time (see, for instance, Coomes [33] or Eden et al. [23]).

Hereafter, for brevity, and without affecting the PS algorithm implementation, by *attractor* one understands the global attractor numerically approximated by a (unique) solution through a chosen initial condition x_0 , after the transients have been neglected. Under Assumption 1, one can consider that to each p , for a given initial condition, there corresponds uniquely an attractor which can be either regular (i.e., in this paper, stable fixed points and limit cycles) or chaotic. If some global attractor consists in several local attractors, they will be identified by an adequate choice of initial conditions.

2.2. PS algorithm

Next, we show how to implement the PS algorithm. For this purpose, let us choose some set $P_N = \{p_1, p_2, \dots, p_N\}$ and an equidistant grid of the time interval I , with nodes $k\Delta t$, $k = 0, 1, \dots$ (the mesh size being Δt) and $I = \bigcup_{j=1,2,\dots} (\bigcup_{i=1}^N I_{ij})$ (Fig. 1), where the subintervals I_{ij} have, for every $j = 1, 2, \dots$, lengths $m_i\Delta t$, $i = 1, 2, \dots, N$, with m_i being some positive integers. On these subintervals, consider p as a piecewise constant function $p(t) = p_i$ for $t \in I_{ij}$. Then, for a fixed Δt , the PS algorithm can be implemented using the following scheme

$$S_{\Delta t} \triangleq [p_1|_{I_{1j}}, p_2|_{I_{2j}}, \dots, p_N|_{I_{Nj}}], \text{ for } j = 1, 2, \dots \quad (2)$$

which means that while the underlying IVP is integrated, for $j = 1$, p will first take the value p_1 for $t \in I_{11}$ ($i = 1$), then p_2 for $t \in I_{21}$ ($i = 2$), and so on, until $t \in I_{N1}$ when $p = p_N$ and $i = N$ (see Fig. 1), after which $j = 2$ and the algorithm repeats in the next set of N intervals I_{i2} , $i = 1, 2, \dots, N$, and so on until $t \geq T$.

In most applications, the largest length of the time subintervals I_{ij} , can be chosen to be of order $20h$ (and even $100h$ for few particular cases), and, for h in general $10^{-1} - 10^{-3}$, depending on the systems characteristics.

For simplicity, once Δt is fixed, we introduce the following simplified form for the scheme (2)

$$S \triangleq [m_1 p_1, m_2 p_2, \dots, m_N p_N]. \quad (3)$$

3. PS Convergence

Based on the convergence of some numerical schemes with single constant step size for ODEs (e.g. the standard Runge-Kutta (RK4) method), we prove in this section that applying the PS algorithm to the IVP (1), the obtained

solution is close enough to the solution obtained when p is replaced with the averaged values of P_N , p^* , given by

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

which, under the considered assumptions, converges to the exact solution. For this purpose, it is sufficient to prove that the global error tends to zero as the integration step $h \rightarrow 0$.

Theorem 1. *(PS algorithm convergence). Let Assumption 1 on the IVP (1) hold, and consider some given set P_N and Δt . Then, the solution obtained with the PS algorithm converges to the solution corresponding to $p = p^*$, with p^* given by (4), for every initial condition x_0 .*

Remark 2.

- i) The proof does not explicitly require an estimation for the step size Δt , because the bounds for Δt are considered to be implicitly stated by the utilized convergent numerical scheme (for the RK4 utilized in this paper, we refer e.g. to [21]).*
- ii) To avoid the potential problems in choosing, for the two solutions, different initial conditions (but in the same attraction basin), and for simplicity, let us consider without loss of generality that both solutions pass through the same initial conditions x_0 . However, the convergence can also be proven when the initial conditions are different (x_0 and y_0 respectively) but satisfies $\|x_0 - y_0\| \leq \varepsilon$ with ε small enough.*

Proof. Since under the considered assumptions any solution through x_0 converges to the exact solution, the proof is simple and implies few iteration steps in calculus. Therefore, the only thing we have to verify, is the global error tends to zero, as $h \rightarrow 0$. For simplicity, we next drop the time variable t and use the following notations: $\Delta t = h$ and the solution obtained with the PS algorithm is x_n (approximation to $x(nh)$), while the solution corresponding to p^* is y_n (approximation to $y(nh)$).

The proof is based on the RK4 convergence of the obtained solution to the exact solution to the IVP (1), under the considered assumptions (see for instance Theorem 3.4.7 on page 239 in [21]), and consists in finding numerical solutions by applying inductively RK4, firstly by using the scheme (3), and then by considering $p = p^*$. Finally, the global error is determined.

First, recall that RK4 method applied to the IVP (1) is given by the following equation

$$x_{n+1} = x_n + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4),$$

where x_n is the RK4 approximation of $x(t_n)$ and, for our autonomous case of systems (1),

$$\begin{aligned} K_1 &= hf(x_n), \\ K_2 &= hf\left(x_n + \frac{1}{2}K_1\right), \\ K_3 &= hf\left(x_n + \frac{1}{2}K_2\right), \\ K_4 &= hf(x_n + K_3). \end{aligned}$$

Consider next h sufficiently small, with x_0 and P_N , for some $N > 1$, all given.

I) *Solution with PS algorithm*

Let us first determine the numerical solution y_n of the IVP (1), starting from x_0 , by using some scheme S given by (3). Thus, for the first m_1 integration steps, $p = p_1$, the next m_2 steps, $p = p_2$, and so on. f depends on the two variables y and p : $f = f(y, p)$.

For $p = p_1$, after one step one obtains

$$\begin{aligned} K_1 &= hf(x_0, p_1) = hg(x_0) + hp_1Ax_0 = hp_1Ax_0 + \mathcal{O}(h), \\ K_2 &= hf\left(x_0 + \frac{K_1}{2}, p_1\right) = hg\left(x_0 + \frac{K_1}{2}\right) + hp_1Ax_0 \\ &\quad + \frac{1}{2}h^2p_1Ag(x_0) + \frac{1}{2}h^2p_1^2Ax_0 = hp_1Ax_0 + \mathcal{O}(h), \\ K_3 &= hf\left(x_0 + \frac{K_2}{2}, p_1\right) = hp_1Ax_0 + \mathcal{O}(h), \\ K_4 &= hf(x_0 + K_3, p_1) = hp_1Ax_0 + \mathcal{O}(h), \end{aligned}$$

and

$$y_1 = x_0 + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) = x_0 + hp_1Ax_0 + \mathcal{O}(h).$$

Next, for the following iteration step with $p = p_1$,

$$\begin{aligned} K_1 &= hf(y_1, p_1) = hp_1Ax_0 + hg(y_1) = hp_1Ax_0 + \mathcal{O}(h), \\ K_2 &= hf\left(y_1 + \frac{K_1}{2}, p_1\right) = hp_1Ax_0 + \mathcal{O}(h), \\ K_3 &= hf\left(y_1 + \frac{K_2}{2}, p_1\right) = hp_1Ax_0 + \mathcal{O}(h), \\ K_4 &= hf(y_1 + K_3, p_1) = hp_1Ax_0 + \mathcal{O}(h), \\ y_2 &= x_0 + 2hp_1Ax_0 + \mathcal{O}(h), \end{aligned}$$

and, after m_1 iterations with $p = p_1$,

$$y_{m_1} = x_0 + m_1hp_1Ax_0 + \mathcal{O}(h).$$

Next, $p = p_2$, and we have

$$\begin{aligned} K_1 &= hf(y_{m_1}, p_2) = (m_1p_1 + p_2)hAx_0 + \mathcal{O}(h), \\ K_2 &= hf\left(y_{m_1} + \frac{K_1}{2}, p_2\right) = (m_1p_1 + p_2)hAx_0 + \mathcal{O}(h), \\ K_3 &= hf\left(y_{m_1} + \frac{K_2}{2}, p_2\right) = (m_1p_1 + p_2)hAx_0 + \mathcal{O}(h), \\ K_4 &= hf(y_{m_1} + K_3, p_2) = (m_1p_1 + p_2)hAx_0 + \mathcal{O}(h), \end{aligned}$$

and

$$y_{m_1+1} = x_0 + (m_1p_1 + p_2)hAx_0 + \mathcal{O}(h),$$

which, after m_2 iterations with $p = p_2$, becomes

$$y_{m_1+m_2} = x_0 + (m_1p_1 + m_2p_2)hAx_0 + \mathcal{O}(h).$$

By iterating the next m_3 steps with $p = p_3$, for m_4 times with $p = p_4$, and so on until the N th subinterval, for m_N times, $p = p_N$. At the end of the first N subintervals I_{i_1} , $i = 1, 2, \dots, N$, $j = 1$, one have

$$y_{m_1+m_2+\dots+m_N} = x_0 + (m_1p_1 + m_2p_2 + \dots + m_Np_N)hAx_0 + \mathcal{O}(h).$$

Finally, by repeating the algorithm on n consecutive subintervals I_{ij} (i.e. $j = 1, 2, \dots, n$, with $n \in \mathbb{N}^*$), the numerical solution becomes

$$y_{n(m_1+m_2+\dots+m_N)} = x_0 + n(m_1p_1 + m_2p_2 + \dots + m_Np_N)hAx_0 + \mathcal{O}(h), \quad (5)$$

which represents the solution obtained with the PS algorithm.

II) *Solution for $p = p^*$*

Let us now consider $p = p^*$ with p^* being some real number. Now, $f = f(x)$, and we obtain

$$K_1 = hf(x_0) = hp^*Ax_0 + hg(x_0) = p^*Ax_0h + \mathcal{O}(h),$$

$$\begin{aligned} K_2 &= hf\left(x_0 + \frac{K_1}{2}\right) = hg\left(x_0 + \frac{K_1}{2}\right) + hp^*Ax_0 + \frac{1}{2}h^2p^*Ag(x_0) + \frac{1}{2}h^2p^{*2}Ax_0 \\ &= hp^*Ax_0 + \mathcal{O}(h), \end{aligned}$$

$$\begin{aligned} K_3 &= hf\left(x_0 + \frac{K_2}{2}\right) = hg\left(x_0 + \frac{K_2}{2}\right) + hp^*Ax_0 + \frac{1}{2}h^2p^*Ag\left(x_0 + \frac{K_1}{2}\right) \\ &\quad + \frac{1}{2}h^2p^{*2}A^2x_0 + \frac{1}{2}h^3p^{*3}A^3x_0 = hp^*Ax_0 + \mathcal{O}(h), \end{aligned}$$

$$K_4 = hf(x_0 + K_3) = hp^*Ax_0 + \mathcal{O}(h),$$

and

$$x_1 = x_0 + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) = x_0 + hp^*Ax_0 + \mathcal{O}(h).$$

Finally, after $n(m_1 + m_2 + \dots + m_N)$ similar steps, we obtain

$$x_{n(m_1+m_2+\dots+m_N)} = x_0 + n(m_1 + m_2 + \dots + m_N)hp^*Ax_0 + \mathcal{O}(h), \quad (6)$$

i.e. the solution for $p = p^*$.

III) *Global error*

Next, in order to obtain the global error, e_n , after $n(m_1 + m_2 + \dots + m_N)$ steps, between the solutions $y_{n(m_1+m_2+\dots+m_N)}$ and $x_{n(m_1+m_2+\dots+m_N)}$, given by (5) and (6) respectively, we have to calculate

$$\begin{aligned} e_n &= \|x_{n(m_1+m_2+\dots+m_N)} - y_{n(m_1+m_2+\dots+m_N)}\| \\ &\leq nh\|A\|x_0 \left| \sum_{i=1}^N m_i p_i - p^* \sum_{i=1}^N m_i \right| + \mathcal{O}(h), \end{aligned}$$

which, for $p^* = \sum_{i=1}^N m_i p_i / \sum_{i=1}^N m_i$, leads to the following estimation

$$\|x_{n(m_1+m_2+\dots+m_N)} - y_{n(m_1+m_2+\dots+m_N)}\| \leq \mathcal{O}(h),$$

Taking into account the convergence of the RK4 method, with which x_n converges to the exact solution, y_n also converges to the exact solution, and the proof is completed. \square

Remark 3.

- (i) In [27], Mao et al. have proved the convergence of the PS algorithm based on the averaging method (see, for averaging theory, e.g. the book of Sander and Verhulst [28]);
- (ii) The proof presented in this paper does not depend on the periodicity of p switches, as the form of S in (3) suggests. Therefore, due to the convexity property of p^* , which is behind (4) (denoting $\alpha_i = m_i / \sum_{i=1}^N m_i$, one obtains $\sum_{i=1}^N \alpha_i = 1$), it is easy to see that not only periodic switches can be used but, such as shown in [29], also random switches can be implemented to achieve the PS algorithm;
- (iii) The convergence of the PS algorithm is clearly supported by characteristic tools for dynamical systems, such as histograms, Poincaré sections, time series, phase plots or Hausdorff distance, as shown in [11];
- (iv) Because the round-off error depends on the computer on which the algorithm is implemented, it was not considered in the numerical analysis.

4. Application

Let us focus on a representative case of the famous Lorenz system describing a meteorological phenomenon

$$\begin{aligned}
 \dot{x}_1 &= 10(x_2 - x_1), \\
 \dot{x}_2 &= -x_1x_3 - x_2 + px_1, \\
 \dot{x}_3 &= x_1x_2 - \frac{8}{3}x_3,
 \end{aligned}
 \tag{7}$$

whose dynamics can be demonstrated in Fig. 2, where the bifurcation diagram was drawn for the first component x_1 , for $p \in [30, 220]$. The integration step-size for this example was $h = 10^{-2}$ and $T = 75$.

Example 1. Let $P_2 = \{90, 96\}$ for which, as shown by the bifurcation diagram, the system behaves chaotically. If one applies the PS algorithm with the scheme $S = [1p_1, 1p_2]$, for which p^* , given by the relation (4), is $p^* = 93$, one finally obtains a stable limit cycle, denoted by A^* in Fig. 3 (red plot). Superimposed in the same figure (in order to underline the match) is the plotted *average attractor* A_{p^*} corresponding to the average value p^* (blue plot). The tests for the Hausdorff distance revealed a value of order 10^{-5} (see the Falconer's book [30, Chapter 9] for the Hausdorff distance).

For this system, in [11], besides phase plots and Hausdorff distance, time series, Poincaré sections, histograms and cross-correlation are used to underline the match between the two attractors, A^* and A_{p^*} .

In this case we can formulate Parrondo's paradox as: "chaos+chaos=regular", i.e. a chaos control like phenomenon.

Remark 4. "Summing" (regular or chaotic) attractors, such as this example would suggest is, to our knowledge, at least a strange, if not impossible, idea. However, as mentioned before, we just use, as usually for Parrondo's paradox, symbolic forms.

Example 2. As shown in [11], there are others possible schemes, $S = [m_1p_1, m_2p_2]$ with $p_1 = 90, p_2 = 96$ and $m_{1,2} > 1$, which give the same result (i.e. $p^* = 93$) and, obviously, more schemes with $N \geq 3$ to approximate A_{p^*} . For example, with $N = 4$ and the scheme $S = [3p_1, 3p_2, 2p_3, 3p_4]$ with $p_1 = 126, p_2 = 131, p_3 = 170$ and $p_4 = 220$, one obtains $p^* = 161$ which belongs to another periodic window (see Fig. 2). Both attractors A^* and A_{p^*} are plotted in Fig. 4. The Hasudorff distance is of same order as in Example 1.

Now, considering the behaviors corresponding to these values of p , i.e. chaos, denoted $C_i, i = 1, 2, 3$, for the first three values $p_{1,2,3}$, and regular R for the fourth one, p_4 (see the bifurcation diagram), we have $C_1 + C_2 + C_3 + R = R^*$.

Moreover, if we take account the "weight" of each motion (i.e. the m_i values) we obtain: $\underbrace{C_1 + C_1 + C_1}_{m_1 \text{ times}} + \underbrace{C_2 + C_2 + C_2}_{m_2 \text{ times}} + \underbrace{C_3 + C_3}_{m_3 \text{ times}} + \underbrace{R + R + R}_{m_4 \text{ times}} = R^*$, where R^* represents the obtained regular motion corresponding to $p^* = 161$.

Example 3. In order to obtain a chaotic attractor (i.e. anticontrol), we can slightly modify, for example, the scheme in Example 2. Thus, by changing only m_4 : $m_4 = 8$, p^* is "pulled" away (due to the mentioned convexity, see Remark 3 ii) from the periodic window where it was before (for $m_4 = 3$), to a new value $p^* = 179.4375$, which corresponds to a chaotic motion (Fig. 5). As expected, for anticontrol, a larger time interval I is now necessary, since in these cases the chaotic attractors are, theoretically, rigorously obtained only for $t \rightarrow \infty$. Also, compared to the above cases, the Hausdorff distance is bigger now (in this case, of order 10^{-3}). Anyway, even for relative large time subintervals and smaller step sizes (the length of I_{4j} is $m_4h = 8 \times 0.0001 = 0.0008$ ms, with a small step size necessary to maintain

the underlying trajectory of A^* to be sufficiently smooth and close to A_{p^*}), the match between the two attractors is still well realized.

Example 4. In order to underline the accuracy of the PS algorithm, let us now consider the Rabinovitch-Fabrikant system [16], which presents strong nonlinearities

$$\begin{aligned}\dot{x}_1 &= x_2(x_3 - 1 + x_1^2) + ax_1, \\ \dot{x}_2 &= x_1(3x_3 + 1 - x_1^2) + ax_2, \\ \dot{x}_3 &= -2x_3(b + x_1x_2).\end{aligned}\tag{8}$$

One can see that the system belongs to our defined classes of systems, either a or b is considered as control parameter. Let us fix $a = -1$ and switch $p := b$ with scheme $[1p_1, 1p_2]$ for $p_1 = -0.3$ and $p_2 = -0.1$. The obtained attractor A^* is chaotic and approximates the attractor A_{p^*} corresponding to $p^* = -0.2$ (Fig. 6). To underline the match, we plotted the Poincaré sections with plane $x_3 = 3$ (Fig. 7) and cross-correlation for the first variable x_1 (Fig. 8). The negative cross-correlation values show, as remarked in [11] for all studied systems, that besides the fact that the two time series are correlated, they are also delayed.

Some other examples of chaos control and anticontrol can be found in [11].

5. Conclusions and Open Problems

In this paper, we have proven that by switching the control parameter of a system modeled by the IVP (1) within a chosen set of values P_N , one obtains an attractor which is typical for the considered system. In other words, the PS algorithm presents a structural like stability over the parameter switching region. Moreover, as has been pointed out, this attractor matches the attractor obtained when p is replaced with the average of the switched values. In this paper, the convergence is obtained via the standard Runge-Kutta scheme and we can conclude that the PS algorithm is numerically stable, as also underlined by the numerous simulations on several examples. Another important characteristic is that the PS algorithm allows to obtain a generalization of Parrondo's paradox. Also, the PS algorithm can be used to explain what happens with some systems in nature, when accidentally (or not) switchings appear.

Regarding possible applications, it is to mention the possibility to use the PS algorithm to realize control or anticontrol of chaos. Compared to the

known algorithms, such as the OGY method for chaos control (introduced by Ott, Grebogy and Yorke in [31]), or anticontrol (see e.g. [32]), by avoiding tedious calculations, the PS algorithm is much easier to implement. However, it is about a kind of control and anticontrol since only stable limit cycles and fixed points or existing chaotic motions, respectively, can be approximated, while with e.g. the OGY algorithm, unstable periodic orbits (UPO) can be stabilized. Also, the PS algorithm allows to consider any attractor of a system modeled by the IVP (1) as an infinite sequence of symbols "C" (Chaotic) and "R" (Regular). For example, if one considers the scheme $[2p_1, 3p_2]$ for a given h , and considers that p_1 corresponds to a chaotic motion and p_2 to some regular motion, the the obtained attractor A could be symbolized as $A = CCRRRCCRRRCC\dots$ (see also Example 1 and Example 2).

Among several existing open problems, one can mention the proofs for other already numerically verified classes of systems, such as discontinuous systems, or continuous/discontinuous systems of fractional order (some steps in this direction have already been made by the author and collaborators in [34, 35, 36]). Another interesting open problem is to study what happens when two initial conditions are arbitrarily nearby, asymptotically approach two different attractors i.e. the case of *riddle basins* (see, for example, the paper of Lai and Grebogy [37], regarding the notion of riddle basins). This problem becomes important especially when approximating chaotic attractors, such as the last example. Also, finding a (4)-like formula for p^* for the (however rare) case of systems, modeled by IVPs depending nonlinearly on p or for Hamiltonian systems, is a challenging open problem. An analytical proof of the existence of a bijection between a set of p values P_N and a set of N attractors, may lead to a very exciting idea: over the set of all attractors, some binary operations could be induced from the ordered real set P_N (see Remark 4).

Acknowledgement

The author wish to thank the reviewers for their constructive comments.

References

- [1] Parrondo JMR, Harmer GP, Abbott D. New paradoxical games based on Brownian ratchets. Phys Rev Lett 2000;85:5226–5229.

- [2] Percus OE, Percus JK. Can two wrongs make a right? coin-tossing games and parrondo's paradox. *Math Intelligencer* 2002;24:68–72.
- [3] Lee Y, Allison A, Abbott D, Stanley HE. Minimal Brownian ratchet: an exactly solvable model. *Phys Rev Lett* 2003;91:220601.
- [4] Amengual P, Allison A, Toral R, Abbott D. Discrete-time ratchets, the Fokker-Planck equation and Parrondo's paradox. *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.*, 2004;460:2269–2284.
- [5] Heath D, Kinderlehrer D, Kowalczyk M. Discrete and continuous ratchets: from coin toss to molecular motor. *Discrete Continuous Dynam Systems - B* 2002;2:153–167.
- [6] Blackwell D, Girshick MA. *Theory of Games and Statistical Decisions*. New York: John Wiley & Sons; 1954.
- [7] Groeber P. On Parrondos games as generalized by Behrends. *Lect. Notes Cont. Inform. Sci.* 2006;341:223–230.
- [8] Abbott D, Davies PCW, Shalizi CR. Order from disorder: The role of noise in creative processes: A special issue on game theory and evolutionary processes overview. *Fluctuation and Noise Letters* 2002;2:C1-C12.
- [9] Abbot D. Assymmetry and disorder: A decade of Parrondo's paradox *Fluctuation and Noise Letter* 2010;9:129156.
- [10] Davies PCW. *Physics and life: The Abdus Salam Memorial Lecture*. 6th Trieste Conference on Chemical Evolution. Kluwer Academic Publishers; 2001, pp. 13-20.
- [11] Danca M-F, Romera M, Pastor G, Montoya F. Finding attractors of continuous-time systems by parameter switching. *Nonlinear Dynam.* 2012;67(4):23172342.
- [12] Danca M-F, Romera M, Pastor G. Alternated julia sets and connectivity properties. *Internat J Bifur Chaos Appl Sci Engrg* 2009;19:2123–2129.
- [13] Loskutov A. Parametring perturbations and non-feedback controlling chaotic motion. *Discrete Continuous Dynam Systems - B* 2006;6:11157–1174.

- [14] Almeida J, Peralta-Salas D, Romera M. Can two chaotic systems give rise to order?. *Physica D* 2005;200:124–132.
- [15] Romera M, Small M, Danca M-F. Deterministic and random synthesis of discrete chaos. *Appl Math Comput* 2007;192:283–297.
- [16] Danca M-F, Chen G. Bifurcation and chaos in a complex model of dissipative medium. *Internat J Bifur Chaos Appl Sci Engrg* 2004;14:3409–3447.
- [17] Hartman P. *Ordinary Differential Equations*. New York: John Wiley & Sons; 1973.
- [18] Hale JK. *Asymptotic behavior of dissipative systems*, Mathematical Surveys and Monographs Vol. no. 25. Amer Math Soc Providence R. I. 1988.
- [19] Perko L. *Differential Equations and Dynamical Systems*. New York: Springer-Verlag; 1991.
- [20] Andronov A, Vitt A, Khaikin S. *Theory of oscillations*. Oxford: Pergamon Press; 1966.
- [21] Stuart AM, Humphries AR. *Dynamical systems and numerical analysis*. Cambridge: Cambridge University Press; 1996.
- [22] Stewart J. *Calculus: Early Transcendentals*. 6th edition. Belmont: Thomson Brooks/Cole; 2008.
- [23] Eden A, Foias C, Nicolaenko B Temam R. *Exponential Attractors for Dissipative Evolution Equations*. New York: John Wiley; 1994.
- [24] Kapitanski L, Rodnianski I. Shape and morse theory of attractors. *Comm Pure Appl Math* 2000;53:218–242.
- [25] Temam R. *Infinite-dimensional Dynamical Systems in Mechanics and Physics*. 2nd edition. New York: Springer; 1997.
- [26] Foias C, Jolly MS. On the numerical algebraic approximation of global attractors. *Nonlinearity* 1995;8:295-319.

- [27] Mao Y, Tang WKS, Danca M-F. An averaging model for chaotic system with periodic time-varying parameter. *Appl Math Comput* 2010;217:355–362.
- [28] Sanders JA, Verhulst F. *Averaging Methods in Nonlinear Dynamical Systems*. 2nd edition. New York: Springer-Verlag; 2007.
- [29] Danca M-F. Random parameter-switching synthesis of a class of hyperbolic attractors. *Chaos* 2008;18:033111.
- [30] Falconer K. *Fractal Geometry, Mathematical Foundations and Applications*. Chichester: John Wiley & Sons; 1990.
- [31] Grebogi C, Ott E, Yorke JA. Controlling chaos. *Phys Rev Lett* 1990;64:1196 – 1199.
- [32] Wang XF, Chen G, Yu X. Anticontrol of chaos in continuous-time systems via timedelay feedback. *Chaos* 2000;10:771–779.
- [33] Coomes BA. Shadowing orbits of ordinary differential equations on invariant submanifolds. *Trans Amer Math Soc* 1997;349:203–216.
- [34] Danca M-F. On the uniqueness of solutions to a class of discontinuous dynamical systems. *Nonlinear Anal Real World Appl* 2010;11:1402–1412.
- [35] Danca M-F, Diethlem K. Fractional-order attractors synthesis via parameter switchings. *Commun Nonlinear Sci Numer Simul* 2010;15:3745–3753.
- [36] Danca M-F. Numerical approximation of a class of discontinuous systems of fractional order. *Nonlinear Dynam* 2011;66(1):133–139.
- [37] Lai Y-C, Grebogi C. Intermingled basins and two-state on-off intermittency. *Phys Rev E* 1995;52:R3313–R3316.

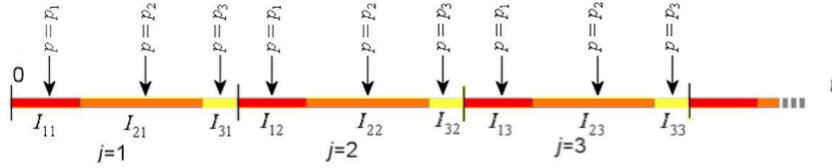


Figure 1: Partition of the time interval I (sketch).

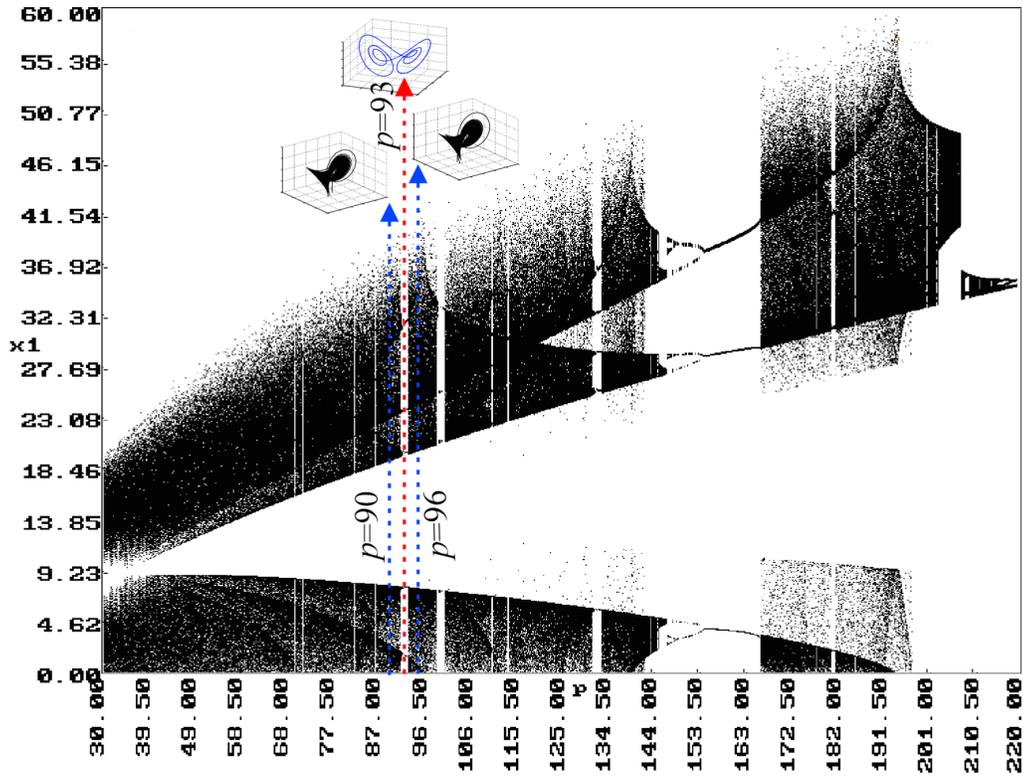


Figure 2: Bifurcation diagram of the first variable x_1 , for the Lorenz system.

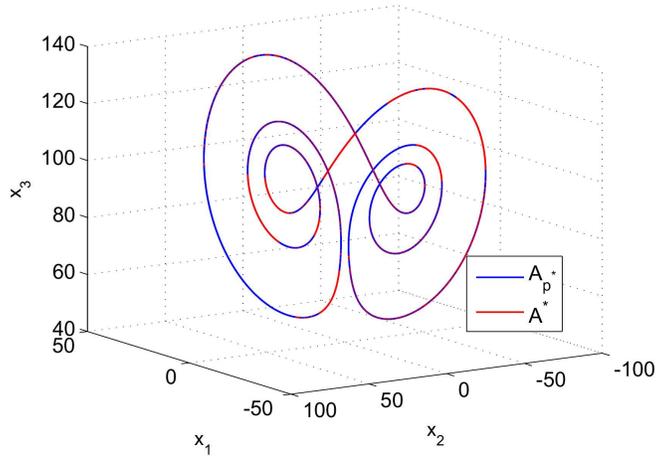


Figure 3: Plots of the attractors A^* (red) and A_{p^*} (blue) obtained with the scheme $S = [1p_1, 1p_2]$ with $p_1 = 90$, $p_2 = 96$ and $p^* = 93$.

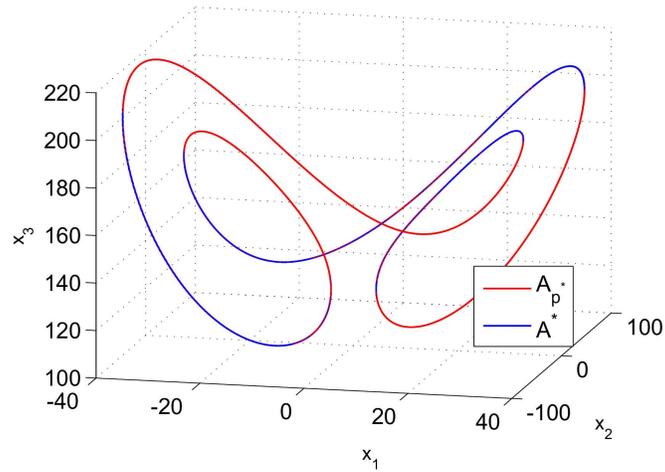


Figure 4: Plots of the attractors A^* (red) and A_{p^*} (blue) obtained with the scheme $S = [3p_1, 3p_2, 2p_3, 3p_4]$ with $p_1 = 126$, $p_2 = 131$, $p_3 = 170$ and $p_4 = 220$; $p^* = 161$.

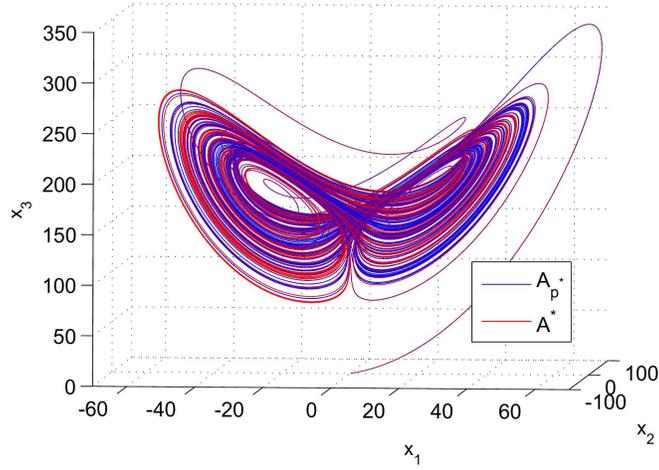


Figure 5: Plots of the chaotic attractors A^* (red) and A_{p^*} (blue) obtained with the scheme $S = [3p_1, 3p_2, 2p_3, 8p_4]$ with $p_1 = 126$, $p_2 = 131$, $p_3 = 170$ and $p_4 = 220$; p^* is now $p^* = 179.4375$.

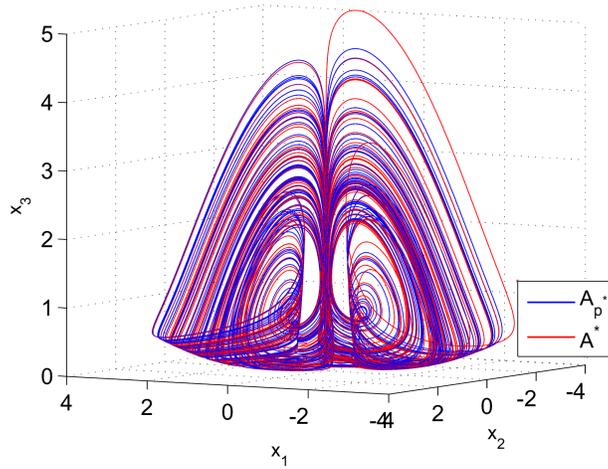


Figure 6: Plots of the chaotic attractors A^* (red) and A_{p^*} (blue) obtained with the scheme $S = [1p_1, 1p_2]$ with $p_1 = -0.3$, $p_2 = -0.1$; $p^* = -0.2$.

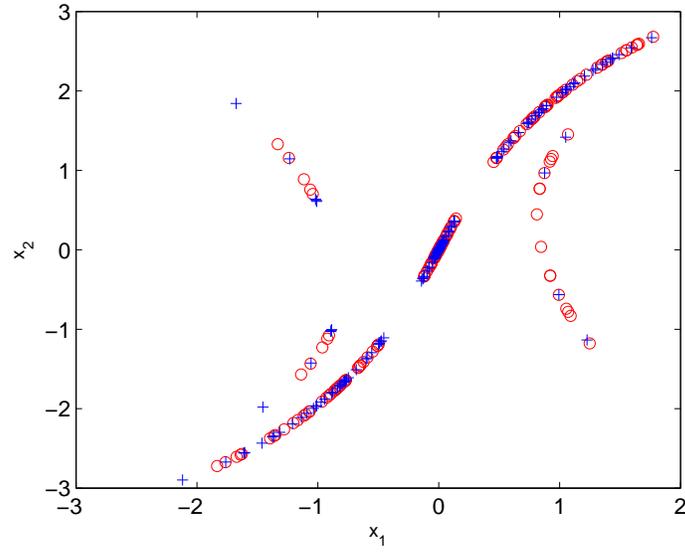


Figure 7: Poincaré sections with plane $x_3 = 0.3$.

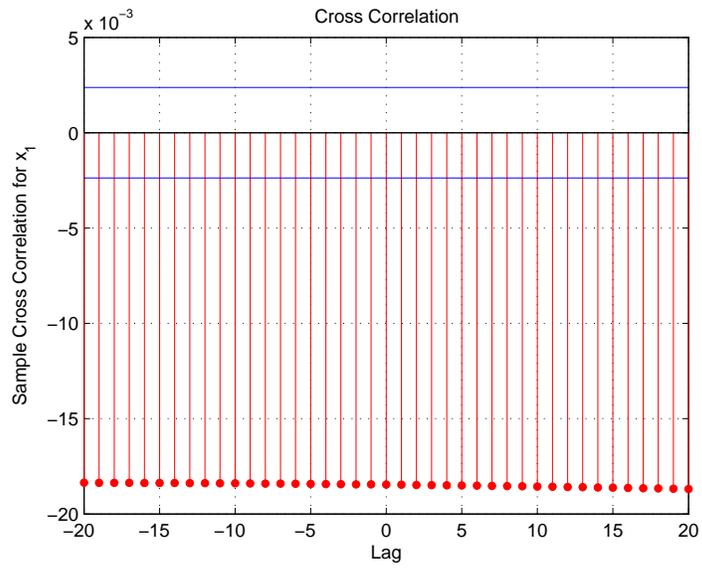


Figure 8: Cross-correlations for x_1 .