



# A lower bound error for free-run simulation of the polynomial NARMAX

E. G. Nepomuceno & S. A. M. Martins

To cite this article: E. G. Nepomuceno & S. A. M. Martins (2016) A lower bound error for free-run simulation of the polynomial NARMAX, Systems Science & Control Engineering, 4:1, 50-58, DOI: [10.1080/21642583.2016.1163296](https://doi.org/10.1080/21642583.2016.1163296)

To link to this article: <https://doi.org/10.1080/21642583.2016.1163296>



© 2016 The Author(s). Published by Taylor & Francis



[View supplementary material](#)



Published online: 06 Apr 2016.



[Submit your article to this journal](#)



Article views: 2197



[View related articles](#)



[View Crossmark data](#)



Citing articles: 6 [View citing articles](#)

## A lower bound error for free-run simulation of the polynomial NARMAX

E. G. Nepomuceno and S. A. M. Martins

Department of Electrical Engineering, Federal University of São João del-Rei, São João del-Rei, Brazil

### ABSTRACT

A lower bound error for free-run simulation of the polynomial NARMAX (Nonlinear AutoRegressive Moving Average model with eXogenous input) is introduced. The ultimate goal of the polynomial NARMAX is to predict an arbitrary number of steps ahead. Free-run simulation is also used to validate the model. Although free-run simulation of the polynomial NARMAX is essential, little attention has been given to the error propagation to round off in digital computers. Our procedure is based on the comparison of two pseudo-orbits produced from two mathematical equivalent models, but different from the point of view of floating point representation. We apply successfully our technique for three identified models of the systems: sine map, Chua's circuit and Duffing–Ueda oscillator. This technique may be used to reject a simulation, if a required precision is greater than the lower bound error, increasing the numerical reliability in free-run simulation of the polynomial NARMAX.

### ARTICLE HISTORY

Received 26 October 2015  
Accepted 4 March 2016

### KEYWORDS

NARMAX; prediction; discrete time systems; nonlinear systems; numerical methods

### 1. Introduction

The polynomial NARMAX (Nonlinear AutoRegressive Moving Average model with eXogenous input) is widely applied to represent all sorts of systems (Billings, 2013; Chen & Billings, 1989). This mathematical representation may be seen as a well-structured type of recursive function, in which terms and parameters are carefully selected (Korenberg, Billings, Liu, & Mcilroy, 1988). Recursive functions are also used to describe and solve many types of systems and problems (Feigenbaum, 1978) and play an essential role in solving the majority of nonlinear dynamics systems (Hammel, Yorke, & Grebogi, 1987; Lozi, 2013; Nepomuceno, 2014).


Numerous researchers are confident in subscribing a chaotic behaviour to various systems through numerical solutions (Ott, 2002). These solutions make use of popular software and computers easily accessible to most researchers. However, Lozi (2013, p. 63–64) says that 'there are many published works whose reliability numerical results is not carefully evaluated'. For example, Lozi (2013, p. 64) says that 'In the simple case of a dynamic discrete system (of Henon map) there are doubts as to the nature of the 'computational results: long unstable pseudo-orbits or strange attractors?'

Nepomuceno (2014) shows, using double precision in a popular software, that a simple sequence of iterations may generate wrong results in steady state, that

is, converged to the wrong answer. In that work, a careful way is presented to calculate error propagation in the computation of recursive function. Investigation of propagation error is not a recent issue (Galias, 2013; Faranda, Mestre, & Turchetti, 2012; Goldberg, 1991; Hammel et al., 1987). In fact, there are many works based on deterministic or stochastic tools that provide some confidence in simulation of recursive functions. Many of those techniques are not practical for nonlinear recursive functions with many terms, such as the polynomial NARMAX, or for non-expert users, who like to measure or at least estimate the error.

To deal with the numerical questions in nonlinear dynamic systems, many researchers have used the shadowing property (Chow & Palmer, 1992, 1991; Hammel et al., 1987; Sauer, Grebogi, & Yorke, 1997). This property ensures that a pseudo-orbit (a sequence of points calculated from a recursive function map) is a homeomorphism that remains close to the real orbit (Faranda et al., 2012). What fewer researchers report is the fact that this property is valid only for uniformly hyperbolic systems (Lozi, 2013). We present two works that have been developed to address this situation. First, the work of Hammel et al. (1987) used computational effort to prove a theorem which states that a pseudo-orbit of the logistic map (May, 1976) remains at a distance of  $10^{-7}$  from the true orbit up to 10 million iterations using double

**CONTACT** E.G. Nepomuceno  [nepomuceno@ufsj.edu.br](mailto:nepomuceno@ufsj.edu.br)

 Supplemental data for this article can be accessed here. <http://dx.doi.org/10.1080/21642583.2016.1163296>

precision. The computational effort is still a strong tool to support results of analysis of nonlinear dynamic systems. Liao & Wang (2014) used 1200 CPUs and an integration algorithm based on an expansion order of 3500 Taylor series, and 4180 digits of precision to perform simulation of the Lorenz equations. The second example is a variant of the shadowing property, in which the domain is analysed by intervals, resulting in property *parameter-shifted shadowing* (Lozi, 2013).

Despite advances in numerical computation for non-hyperbolic cases, the two cases presented still have limitations. In the first case (Hammel et al., 1987), an issue if the computer test constitutes a sufficient condition can be raised, and therefore the theorem is proved for all cases, or whether it is a necessary condition, the result may not be valid for all cases. This issue of necessary or sufficient condition on computer simulation has recently come up in Nepomuceno (2014). In the second example, Lozi (2013) states that the parameter-shifted shadowing applied to the map tent (Monteiro, 2002) is valid for almost all values of the parameter  $a$ , but not for all values.

Hammel et al. (1987) present a technique that may be used to estimate an upper bound error of a simulation. In that work, the authors show that for a specific parameter and initial condition for the logistic map, the computed orbit is within a distance of  $10^{-8}$  for  $10^7$  iterates. This is an important computational result, as the logistic map is a non-hyperbolic system. However, this result is presented for few specific initial conditions and parameters, as well in a specific hardware, the Cray X-MP computer. It is easy to show that the logistic map is a particular case of a polynomial NARMAX. It is indeed interesting to find the upper bound error for computational simulations, but it is not always possible. We can now point out two main facts to justify the relevance of this paper. First, we could not find any paper that deals with the error propagation due to finite precision for the polynomial NARMAX. Second, we present an approach to evaluate a lower bound error for recursive functions, as alternative tool to increase the reliability of computational simulation. The technique is based on the fact that the interval extensions (Moore & Moore, 1979) are mathematically equivalent but may exhibit different computer simulation results. The result of using multiple extensions brings a new concept of 'multiple pseudo-orbits'. The technique presents a tool that may be used to reject a simulation. It means that when a simulation presents a lower bound error greater than the expected precision, we have to stop this simulation and undertake other procedures, such as an increase in the precision of software. We applied our technique in three identified models of the systems: sine map, Chua's circuit and Duffing–Ueda oscillator, focusing on systems which present nonlinear behaviour. It was possible to

show a moment of the simulation in which there is no more numerical reliability regarding a required precision.

## 2. Preliminary concepts

### 2.1. The polynomial NARMAX

A NARMAX model can be written as Chen & Billings (1989):

$$y_{n+1} = \mathcal{F}^\ell[y_n, \dots, y_{n-1-k_y}, u_n, \dots, u_{n-1-k_u}, e_n, \dots, e_{n-1-k_e}], \quad (1)$$

where  $y_n$ ,  $u_n$  and  $e_n$  are, respectively, the output, input and noise terms at the discrete time  $n \in \mathbb{N}$ . The parameters  $k_y$ ,  $k_u$  and  $k_e$  are the maximum lag considered for output, input and noise. Terms of  $e_n$  are frequently included in the parameter estimation process, to avoid bias of an estimator. In this work,  $\mathcal{F}^\ell[\cdot]$  is assumed to be a polynomial with non-linearity degree  $\ell \in \mathbb{Z}^+$ . Non-linear systems are frequently modelled using particular cases of the polynomial NARMAX, such as NAR, NARX and NARMA. In this paper, we name them all as polynomial NARMAX.

Although structure selection and parameter detection are decisive steps in system identification, they are not the main aim of this paper. Billings (2013) presents a extensive list of these techniques. In general, model structure is selected using orthogonal techniques as the error reduction ratio and parameters are estimated using prediction error minimization-based techniques. Free-run simulation is often used as a criterion to validate the polynomial NARMAX (Barbosa, Aguirre, Martinez, & Braga, 2011; Billings, 2013). In many cases,  $n$  step-ahead free predictions are performed and compared to a specific validation data set. To the best of our knowledge, little attention has been given to the error propagation in such simulations.

### 2.2. Recursive functions

Let  $n \in \mathbb{N}$ , a metric space  $M \subset \mathbb{R}$ , the relation

$$x_{n+1} = f(x_n), \quad (2)$$

where  $f : M \rightarrow M$  is a recursive function or a map of a state space into itself and  $x_n$  denotes the state at the discrete time  $n$ . The sequence  $\{x_n\}$  obtained by iterating Equation (2) starting from an initial condition  $x_0$  is called the orbit of  $x_0$  (Gilmore & Lefranc, 2012).

It is clear that if we take  $x \in \mathbb{R}^n$ ,  $\mathcal{F}^\ell$  of Equation (1) can be seen as a specific case of  $f$  in Equation (2).

### 2.3. Interval extension

Let  $f$  be a function of real variable  $x$ . Moore & Moore (1979) present the following definition.

**Definition 2.1:** An interval extension of  $f$  is an interval valued function  $F$  of an interval variable  $X$ , with the property

$$F(x) = f(x) \quad \text{for real arguments,} \quad (3)$$

where by an interval we mean a closed set of real numbers  $x \in \mathbb{R}$  such that  $X = [\underline{X}, \bar{X}] = \{x : \underline{X} \leq x \leq \bar{X}\}$ .

Let us give an example.

**Example 2.1:** Let  $x_{n+1} = f(x_n) = rx_n(1 - x_n)$ . We present a few examples of interval extension of  $f$ :

$$X_{n+1} = F(X_n) = r(X_n(1 - X_n)), \quad (4)$$

$$= rX_n - rX_n^2, \quad (5)$$

$$= r(X_n - X_n^2), \quad (6)$$

$$= rX_n - (rX_n)X_n. \quad (7)$$

In Example 2.2, Equations (4)–(7) are mathematically equivalent, but they have different sequences of its basic arithmetic operations (Rudolph-Lilith & Muller, 2014; Yabuki & Tsuchiya, 2013). As it is well known that on floating point standard, we do not have commutative and distributive operations, they may exhibit different pseudo-orbits (Goldberg, 1991; Institute of Electrical and Electronics Engineers (IEEE), 2008; Overton, 2001). This difference is also explained when we deal with an interval representation of a number. In such cases, there are interval extensions that are equivalent. This leads us to the following definition.

**Definition 2.2:**  $G$  and  $H$  are equivalent interval extensions if

$$G(X) = H(X) \quad \text{for interval arguments.}$$

Consider the following example of Definition 2.2.

**Example 2.2:** Let the following extension intervals:

$$G(X) = rX(1 - X),$$

$$H(X) = r(X(1 - X)),$$

$$L(X) = rX - rX^2.$$

Considering  $r = 3$  and  $X = [0.3, 0.4]$ , then we have

$$G([0.3, 0.4]) = 3[0.3, 0.4](1 - [0.3, 0.4]) = [0.54, 0.84],$$

$$H([0.3, 0.4]) = 3([0.3, 0.4](1 - [0.3, 0.4]))$$

$$= [0.54, 0.84] \quad \text{and}$$

$$L([0.3, 0.4]) = 3[0.3, 0.4] - 3([0.3, 0.4]^2) = [0.42, 0.93].$$

In this example, only  $G(X)$  and  $H(X)$  are equivalent interval extensions.

### 2.4. Orbits and pseudo-orbits

Associated with a map we may define an orbit as follows.

**Definition 2.3:** An orbit is a sequence of values of a map, represented by  $\{x_n\} = [x_0, x_1, \dots, x_n]$ .

The calculation of this orbit is generally carried out by a finite-precision computer. There is no unique pseudo-orbit, as there are different hardware, software and numerical precision standard, such as IEEE 754-2008, which may yield different output for each extension interval.

**Definition 2.4:** Let  $i \in \mathbb{N}$  represents a pseudo-orbit, which is defined by an initial condition, an interval extension of  $f$ , some specific hardware, software and numerical precision standard. A pseudo-orbit is an approximation of an orbit and we represent as

$$\{\hat{x}_{i,n}\} = [\hat{x}_{i,0}, \hat{x}_{i,1}, \dots, \hat{x}_{i,n}],$$

such that

$$|x_n - \hat{x}_{i,n}| \leq \delta_{i,n}, \quad (8)$$

where  $\delta_{i,n} \in \mathbb{R}$  is the error and  $\delta_{i,n} \geq 0$ .

A pseudo-orbit define an interval where the true orbit relies. Hence, we may define an interval associated with each value of a pseudo-orbit

$$I_{i,n} = [\hat{x}_{i,n} - \delta_{i,n}, \hat{x}_{i,n} + \delta_{i,n}]. \quad (9)$$

From Equations (8) and (9), it is clear that

$$x_n \in I_{i,n} \quad \text{for all } i \in \mathbb{N}. \quad (10)$$

## 3. Method

### 3.1. Lower bound error

From what has been presented on the previous section, we may establish the following lemma.

**Lemma 3.1:**  $I_{a,n} \cap I_{b,n} \neq \emptyset \forall a, b, n \in \mathbb{N}$ .

**Proof:** Let us assume that there is an intersection equals to empty set for some  $a, b, n$ . It means that at least one point  $n$  belongs to only a pseudo-orbit  $a$  or only a pseudo-orbit  $b$ , such that  $I_{a,n} \cap I_{b,n} = \emptyset$ . Hence,  $x_n \in I_{a,n}$  and  $x_n \notin I_{b,n}$  or  $x_n \in I_{b,n}$  and  $x_n \notin I_{a,n}$ . But it implies that either  $I_{a,n}$  or  $I_{b,n}$  is not an pseudo-orbit according to Equation (10), which is a contradiction. ■

Now we present the following theorem.

**Theorem 3.2:** If  $I_{a,n} \cap I_{b,n} \neq \emptyset$ , then  $|\hat{x}_{a,n} - \hat{x}_{b,n}| \leq \delta_{a,n} + \delta_{b,n}$ .

**Proof:** Let  $I_{a,n} = [\hat{x}_{a,n} - \delta_{a,n}, \hat{x}_{a,n} + \delta_{a,n}]$  and  $I_{b,n} = [\hat{x}_{b,n} - \delta_{b,n}, \hat{x}_{b,n} + \delta_{b,n}]$ .  $I_{a,n} \cap I_{b,n} \neq \emptyset$  if and only if

$$\hat{x}_{b,n} - \delta_{b,n} \leq \hat{x}_{a,n} + \delta_{a,n} \quad \text{and} \quad \hat{x}_{a,n} - \delta_{a,n} \leq \hat{x}_{b,n} + \delta_{b,n}. \quad (11)$$

Equation (11) may be rewritten as

$$\begin{aligned} \hat{x}_{b,n} - \hat{x}_{a,n} &\leq \delta_{b,n} + \delta_{a,n} \quad \text{and} \quad \hat{x}_{a,n} - \hat{x}_{b,n} \leq \delta_{a,n} + \delta_{b,n}, \\ \hat{x}_{a,n} - \hat{x}_{b,n} &\geq -(\delta_{b,n} + \delta_{a,n}) \quad \text{and} \quad \hat{x}_{a,n} - \hat{x}_{b,n} \leq \delta_{a,n} \\ &+ \delta_{b,n}, \quad -(\delta_{a,n} + \delta_{b,n}) \leq \hat{x}_{a,n} - \hat{x}_{b,n} \leq \delta_{a,n} + \delta_{b,n}, \\ |\hat{x}_{a,n} - \hat{x}_{b,n}| &\leq \delta_{a,n} + \delta_{b,n}. \end{aligned}$$

And that completes the proof. ■

Now, we define the lower bound error in the following theorem.

**Theorem 3.3:** Let two pseudo-orbits  $\{\hat{x}_{a,n}\}$  and  $\{\hat{x}_{b,n}\}$  derived from two interval extensions. Let  $\delta_{\alpha,n} = |\hat{x}_{a,n} - \hat{x}_{b,n}|/2$  be the lower bound error of a map  $f(x)$ , then  $\delta_{a,n} \geq \delta_{\alpha,n}$  or  $\delta_{b,n} \geq \delta_{\alpha,n}$ .

**Proof:** Suppose that  $\delta_{a,n} < \delta_{\alpha,n}$  and  $\delta_{b,n} < \delta_{\alpha,n}$ . It implies that

$$\delta_{a,n} + \delta_{b,n} < 2\delta_{\alpha,n} = |\hat{x}_{a,n} - \hat{x}_{b,n}|,$$

which is a contradiction to Theorem 3.2. And that completes the proof. ■

Theorem 3.2 establishes that at least one of the two pseudo-orbits must have an error greater or equal to the lower bound error. This has a practical meaning. If this lower bound error is greater than the required precision, we cannot go on with the simulation without a further analysis. To guarantee the confidence in the simulation, we must prove that one of the pseudo-orbits has the required accuracy. We believe that the lower bound error is a simple and practical tool to increase the reliability of computational simulation for dynamical systems.

Besides that, if we take  $G$  and  $H$  as equivalent interval extension there is no reason to state that  $\delta_{a,n} \gg \delta_{b,n}$  neither  $\delta_{b,n} \gg \delta_{a,n}$ . So we may expect that there are simulations which both pseudo-orbits present an error greater than lower bound error, and that was what happened in three studied cases in this paper at a specific time  $n$ , as we can see in Section 4.

### 3.2. Stop simulation criterion

One of direct consequences of our approach is to yield information that allows the user to stop the simulation when a required numerical precision is no longer satisfied. We present here an example of such a criterion based on the relative error of a sequence of values. This criterion may be easily adapted for different user needs. Let  $\varepsilon_{\alpha,n}$  be the relative precision at iteration  $n$  defined by

$$\begin{aligned} \varepsilon_{\alpha,n} &= \frac{\delta_{\alpha,n}}{\frac{|\hat{x}_{a,n} + \hat{x}_{b,n}|}{2}} = \frac{\frac{|\hat{x}_{a,n} - \hat{x}_{b,n}|}{2}}{\frac{|\hat{x}_{a,n} + \hat{x}_{b,n}|}{2}}, \\ &= \frac{|\hat{x}_{a,n} - \hat{x}_{b,n}|}{|\hat{x}_{a,n} + \hat{x}_{b,n}|}, \end{aligned} \quad (12)$$

where  $n \in \mathbb{N}$ . A user must define the minimum precision,  $\varepsilon$ , required by simulation. It implies that the simulation must be interrupted at any moment when  $\varepsilon_{\alpha,n} > \varepsilon$ . In this work, we adopt  $\varepsilon = 0.001$ .

### 3.3. Routines

All routines are performed in Scilab 5.5.2 in a double precision and the results, up to discrete time  $n = 10$ , are verified by means symbolic computation performed in wxMaxima 13.04.2 with 1000 digits of accuracy. We used a computer with a processor Intel i5-4440 @ 3.1GHz and a Windows 8 operating system. Scilab and wxMaxima are free and may be downloaded from the internet. In the supplemental material, we present all routines used to produce the results of this paper.

## 4. Results

In this section, we present the lower bound error applied for three case studies, which exhibit nonlinear dynamics, such as chaos. The idea is to produce two pseudo-orbits from two different interval extensions of models obtained from the literature. We select two interval extensions that are equivalent, according to Definition 2.2. The three chosen models are for the systems sine map (Nepomuceno, Takahashi, Amaral, & Aguirre, 2003), Chua's circuit (Aguirre, 1997) and Duffing–Ueda (Aguirre & Billings, 1994).

### 4.1. Sine map

A unidimensional sine map is defined as

$$x_{n+1} = \alpha \sin(x_n), \quad (13)$$

where  $\alpha = 1.2\pi$ . A polynomial NARMAX identified for this system is given by (Nepomuceno et al., 2003)

$$y_{n+1} = 2.6868y_n - 0.2462y_n^3. \quad (14)$$



**Table 1.** Numerical computation in Scilab of the first 10 values of  $G(X_n)$  and  $H(X_n)$  as presented in Equations (15) and (16).

| $G(X_n)$               | $H(X_n)$               | $\delta_{\alpha,n}$   |
|------------------------|------------------------|-----------------------|
| 2.684338000000000D-01  | 2.684338000000000D-01  | 0.000000000000000D+00 |
| 7.164658213860300D-01  | 7.164658213860300D-01  | 0.000000000000000D+00 |
| 1.834453282361167D+00  | 1.834453282361167D+00  | 0.000000000000000D+00 |
| 3.408933569627769D+00  | 3.408933569627769D+00  | 0.000000000000000D+00 |
| -5.939994265069455D-01 | -5.939994265069437D-01 | 8.881784197001252D-16 |
| -1.544358084012885D+00 | -1.544358084012880D+00 | 2.220446049250313D-15 |
| -3.242538381762892D+00 | -3.242538381762889D+00 | 1.776356839400250D-15 |
| -3.185457394195996D-01 | -3.185457394196174D-01 | 8.881784197001252D-15 |
| -8.479107015420703D-01 | -8.479107015421167D-01 | 2.320366121466577D-14 |
| -2.128081089747603D+00 | -2.128081089747703D+00 | 4.973799150320701D-14 |

Note: The third column is the lower bound error of these two pseudo-orbits according to Theorem 3.3.

Let us consider two equivalent interval extensions of the model (14):

$$G(X_n) = 2.6868X_n - \underline{0.2462X_n^3}, \quad (15)$$

$$H(X_n) = 2.6868X_n - \underline{(0.2462X_n)X_n^2}, \quad (16)$$

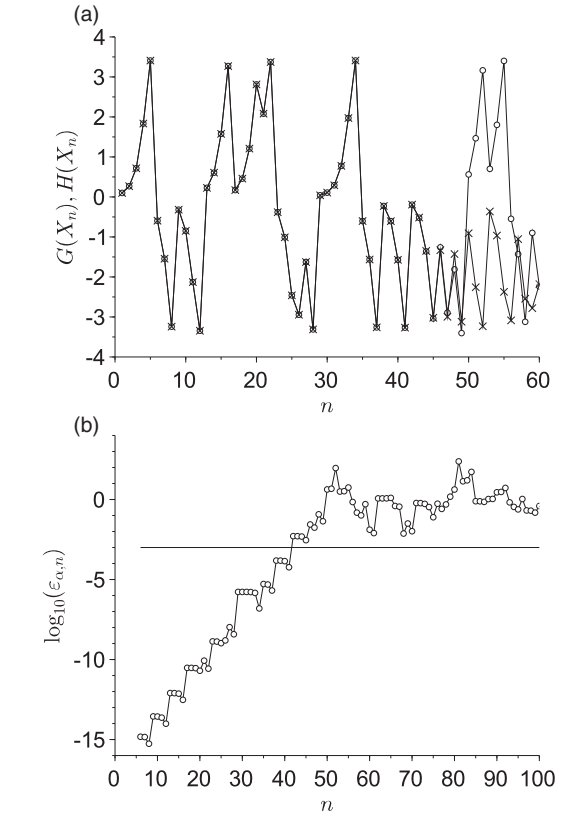
in which the underline was used only to stress the difference between  $G(X_n)$  and  $H(X_n)$ . The difference between the two natural extensions was also underlined in the examples of Chua's circuit and Duffing–Ueda oscillator. As already mentioned in Definition 2.3 and Example 2.4, Equations (15) and (16) are mathematically equivalent, but they represent a different sequence of arithmetical operations, which may produce different outcome. These extensions were simulated using  $X_0 = 0.1$  as an initial condition. By iterating Equations (15) and (16), we produce two pseudo-orbits  $\{\hat{y}_{G,n}\}$  and  $\{\hat{y}_{H,n}\}$  of the model (14), respectively.

Table 1 shows the first 10 values of a simulation performed in Scilab and the lower bound error, whereas Figure 1(a) shows the results for  $n$  up to 60. The relative error is presented in Figure 1(b), in which values are in a logarithm scale. The precision required is no longer satisfied when  $n = 42$ . The lower bound error is checked up to the 10th iteration by means the method suggested in Section 3.3. We found  $\delta_{G,10} \geq \delta_{\alpha,10}$  and  $\delta_{H,10} \geq \delta_{\alpha,10}$ , which agrees with Theorem 3.3 and additionally points out for the possibility of cases in which the lower bound error occurs for both pseudo-orbits.

## 4.2. Chua's circuit

Chua's circuit (Chua, Wu, Huang, & Zhong, 1993) is an electrical circuit able to reproduce different regimes, such as periodic and chaotic oscillations. The circuit is composed of a nonlinear Chua's diode, which plays the nonlinear role. Equations (17) and (18) present the white-box model of Chua's circuit

$$C_1 \frac{dv_{C_1}}{dt} = \frac{v_{C_2} - v_{C_1}}{R} - i_d(v_1),$$



**Figure 1.** Sine map: (a) free-run simulation and (b) evolution of relative error. (a) Simulation of Equations (15) and (16), with results for  $G(X_n)$  ( $-x-$ ) and  $H(X_n)$  ( $-o-$ ) and the same initial condition, that is,  $X_0 = 0.1$ .  $n$  stands for the number of iterations. (b) Evolution of relative error  $\epsilon_{\alpha,n}$  of Equation (15) compared to required precision  $\epsilon = 0.001$ . The values are plotted using the  $\log_{10}$ . When  $n \geq 44$  the  $\epsilon_{\alpha,n} > \epsilon = \log_{10}(0.001) = -3$ , and the simulation no longer satisfies the adopted precision.

$$C_2 \frac{dv_{C_2}}{dt} = \frac{v_{C_1} - v_{C_2}}{R} - i_L, \quad (17)$$

$$L \frac{dv_{iL}}{dt} = -v_{C_2},$$

$$i_d(v_1) = \begin{cases} m_0 v_1 + B_p(m_0 - m_1), & v_1 < -B_p, \\ m_1 v_1, & |v_1| \leq B_p, \\ m_0 v_1 + B_p(m_1 - m_0), & v_1 > B_p. \end{cases} \quad (18)$$

A polynomial NARMAX identified for Chua’s circuit is given by (Aguirre, 1997)

$$\begin{aligned}
 y_{n+1} = & 3.523y_n - 4.2897y_{n-1} - 0.2588y_{n-3} - 1.7784y_n^3 \\
 & + 2.0652y_{n-2} + 6.1761y_n^2y_{n-1} \\
 & + 0.1623y_ny_{n-1}y_{n-3} - 2.7381y_n^2y_{n-2} \\
 & - 5.5369y_n^2y_{n-1}^2 + 0.1031y_{n-1}^3 + 0.4623y_{n-3}^3 \\
 & - 0.5247y_{n-1}^2y_{n-3} - 1.8965y_n^2y_{n-2}^2 \\
 & + 5.4255y_ny_{n-1}y_{n-2} + 0.7258y_{n-1}y_{n-3}^2 \\
 & - 1.7684y_{n-2}y_{n-3}^2 + 1.1800y_{n-2}^2y_{n-3}, \quad (19)
 \end{aligned}$$

where the time interval between  $n$  and  $n+1$  is  $12 \mu$  s.

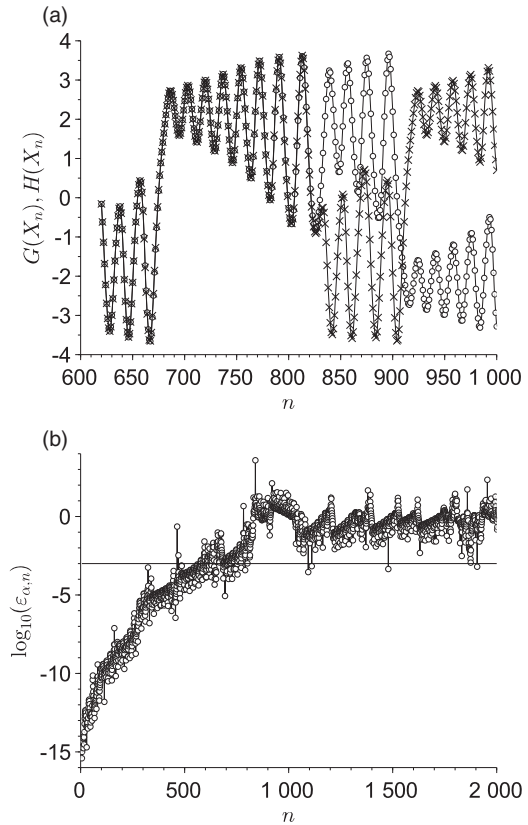
Considering two equivalent interval extensions of model (19), we have

$$\begin{aligned}
 G(X_n) = & 3.523X_n - 4.2897X_{n-1} - 0.2588X_{n-3} \\
 & - 1.7784X_n^3 + 2.0652X_{n-2} + 6.1761X_n^2X_{n-1} \\
 & + 0.1623X_nX_{n-1}X_{n-3} - 2.7381X_n^2X_{n-2} \\
 & - 5.5369X_nX_{n-1}^2 + 0.1031X_{n-1}^3 + 0.4623X_{n-3}^3 \\
 & - 0.5247X_{n-1}^2X_{n-3} - 1.8965X_nX_{n-2}^2 \\
 & + 5.4255X_nX_{n-1}X_{n-2} + 0.7258X_{n-1}X_{n-3}^2 \\
 & - 1.7684X_{n-2}X_{n-3}^2 + 1.1800X_{n-2}^2X_{n-3}, \quad (20)
 \end{aligned}$$

$$\begin{aligned}
 H(X_n) = & 3.523X_n - 4.2897X_{n-1} - 0.2588X_{n-3} \\
 & - 1.7784X_n^3 + 2.0652X_{n-2} + 6.1761X_nX_nX_{n-1} \\
 & + 0.1623X_nX_{n-1}X_{n-3} - 2.7381X_n^2X_{n-2} \\
 & - 5.5369X_nX_{n-1}^2 + 0.1031X_{n-1}^3 + 0.4623X_{n-3}^3 \\
 & - 0.5247X_{n-1}^2X_{n-3} - 1.8965X_nX_{n-2}^2 \\
 & + 5.4255X_nX_{n-1}X_{n-2} + 0.7258X_{n-1}X_{n-3}^2 \\
 & - 1.7684X_{n-2}X_{n-3}^2 + 1.1800X_{n-2}^2X_{n-3}. \quad (21)
 \end{aligned}$$

These interval extensions were simulated using the initial condition  $X_p = 1$  for  $p = 0 \dots 3$ . By iterating Equations (20) and (21), we produce two pseudo-orbits  $\{\hat{y}_{G,n}\}$  and  $\{\hat{y}_{H,n}\}$  of the model (14), respectively.

Table 2 presents 10 first values and the lower bound error of a simulation performed using Scilab. The results up to  $n=10$  were verified using 1000 digits precision and are presented as a supplemental material. Figure 2(a) shows a specific window for the performed simulation,



**Figure 2.** Chua’s circuit: (a) free-run simulation and (b) evolution of relative error. (a) Simulation of Equations (20) and (21), with results for  $G(X_n)$  ( $-x-$ ) and  $H(X_n)$  ( $-o-$ ) and the same initial condition, that is,  $X_p = 1$ , for  $p = 0 \dots 3$ .  $n$  stands for the number of iterations. The interval of each iteration is  $12 \times 10^{-6}$  s. We present in this figure the interval of simulation between 7.2 and 12 ms. (b) Evolution of relative error  $\epsilon_{\alpha,n}$  of Equation (15) compared to required precision  $\epsilon = 0.001$ . The values are plotted using the  $\log_{10}$ . When  $n \geq 464$  the  $\epsilon_{\alpha,n} > \epsilon$ , and the simulation no longer satisfies this requirement. This means that after around 5.6 ms the simulation does not satisfy the required.

**Table 2.** Numerical computation in Scilab of the first 10 values of  $G(X_n)$  and  $H(X_n)$  as presented in Equations (20) and (21).

| $G(X_n)$              | $H(X_n)$              | $\delta_{\alpha,n}$   |
|-----------------------|-----------------------|-----------------------|
| 1.031800000000000D+00 | 1.031800000000000D+00 | 0.000000000000000D+00 |
| 1.132163046694930D+00 | 1.132163046694931D+00 | 4.440892098500626D-16 |
| 1.330695186038330D+00 | 1.330695186038333D+00 | 1.332267629550188D-15 |
| 1.632133203065402D+00 | 1.632133203065409D+00 | 3.108624468950438D-15 |
| 2.000018513977929D+00 | 2.000018513977946D+00 | 8.215650382226158D-15 |
| 2.374908880012700D+00 | 2.374908880012723D+00 | 1.176836406102666D-14 |
| 2.708412651735185D+00 | 2.708412651735223D+00 | 1.909583602355269D-14 |
| 2.960871265482912D+00 | 2.960871265482957D+00 | 2.220446049250313D-14 |
| 3.100523935792458D+00 | 3.100523935792515D+00 | 2.842170943040401D-14 |
| 3.106916026153829D+00 | 3.106916026153840D+00 | 5.329070518200751D-15 |

Note: The third column is the lower bound error of these two pseudo-orbits according to Theorem 3.3.

using the criterion presented in Section 3.2, when  $n = 464$  the relative error is greater than the minimum precision (see Figure 2(b)). In other words, using double precision and a relative error of  $\varepsilon = 0.001$  we may simulate the polynomial NARMAX equation (20) up to around 5.6 ms in order to guarantee the precision adopted. In this case, we also found that  $\delta_{G,10} \geq \delta_{\alpha,10}$  and  $\delta_{H,10} \geq \delta_{\alpha,10}$ .

### 4.3. Duffing–Ueda oscillator

Let us consider a damped, periodically forced nonlinear Duffing–Ueda oscillator (Billings, 2013):

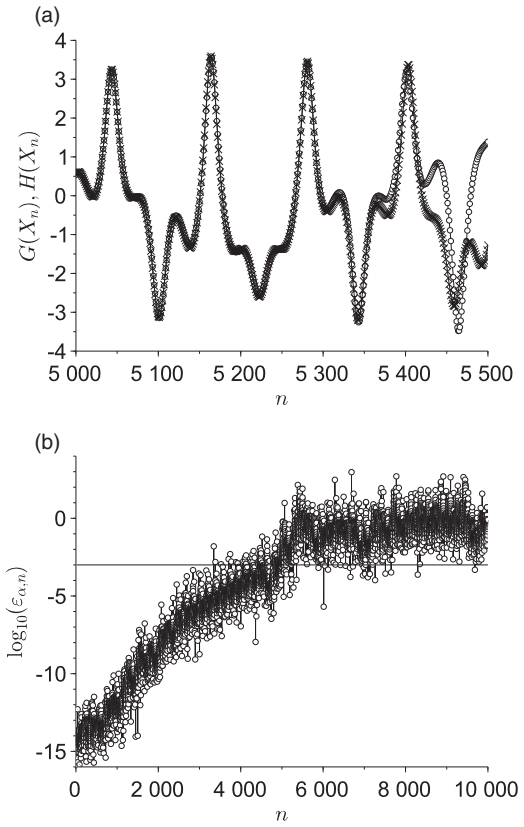
$$\frac{d^2y}{dt^2} + k \frac{dy}{dt} + \mu y^3 = A \cos(t). \quad (22)$$

A polynomial NARMAX for the Duffing–Ueda oscillator was identified by (Aguirre and Billings, 1994)

$$\begin{aligned} y_{n+1} = & 2.1579y_n - 1.3203y_{n-1} + 0.16239y_{n-2} \\ & + 0.0003416u_n + 0.0019463u_{n-1} - 0.0048196y_n^3 \\ & + 0.003523y_n^2y_{n-1} - 0.0012162y_ny_{n-1}y_{n-2} \\ & + 0.0002248y_{n-2}^3, \end{aligned} \quad (23)$$

where  $u = A \cos(kT_s)$ ,  $n \in \mathbb{N}$  and  $T_s = \pi/60$ . Let us consider two equivalent extensions of the model (23), which were simulated using  $X_p = 2$  for  $p = 0 \dots 2$  as initial conditions given by Equations (24) and (25). By iterating Equations (24) and (25), we produce two pseudo-orbits  $\{\hat{y}_{G,n}\}$  and  $\{\hat{y}_{H,n}\}$  of the model (14), respectively.

Table 3 presents the first 10 values of a simulation of these interval extensions, performed in Scilab and verified using 1000 digits precision in wxMaxima. The codes are provided as a supplemental material. Figure 3(a) presents a specific window from  $n = 5000$  to  $n = 5500$ . Figure 3(b) presents the evolution of relative error, which reaches the adopted precision when  $n = 3345$ . Thus, after 175.1 s the simulation of Duffing–Ueda by means of Equation (24) presents a relative error superior than the minimum precision adopted of 0.001. Again,  $\delta_{G,10} \geq \delta_{\alpha,10}$  and  $\delta_{H,10} \geq$



**Figure 3.** Duffing–Ueda oscillator: (a) Free-run simulation and (b) Evolution of relative error. (a) Simulation of Equations (24) and (25), with results for  $G(X_n)$  ( $-x-$ ) and  $H(X_n)$  ( $-o-$ ) and the same initial condition, that is,  $X_p = 2$ , for  $p = 0 \dots 2$ .  $n$  stands for the number of iterations. The interval of each iteration is  $\pi/60$  s. We present in this figure the interval of simulation between approximately 262.8 and 288.0 s. (b) Evolution of relative error  $\varepsilon_{\alpha,n}$  of Equation (15) compared to required precision  $\varepsilon = 0.001$ . The values are plotted using the  $\log_{10}$ . When  $n \geq 3345$ , the  $\varepsilon_{\alpha,n} > \varepsilon$ , and the simulation no longer satisfies this requirement. This means that after around 175.1 s the simulation does not satisfy the required precision.

$\delta_{\alpha,10}$  are noticed for this example.

$$\begin{aligned} G(X_n) = & \underline{2.1579X_n - 1.3203X_{n-1} + 0.16239X_{n-2}} \\ & \underline{+ 0.0003416U_n + 0.0019463U_{n-1}} \\ & \underline{- 0.0048196X_n^3 + 0.003523X_n^2X_{n-1}} \end{aligned}$$

**Table 3.** Numerical computation in Scilab of the first 10 values of  $G(X_n)$  and  $H(X_n)$  as presented in Equations (24) and (25).

| $G(X_n)$              | $H(X_n)$              | $\delta_{\alpha,n}$   |
|-----------------------|-----------------------|-----------------------|
| 2.004509613429507D+00 | 2.004509613429507D+00 | 2.220446049250313D-16 |
| 2.013981518787148D+00 | 2.013981518787149D+00 | 6.661338147750939D-16 |
| 2.028012881721737D+00 | 2.028012881721740D+00 | 1.332267629550188D-15 |
| 2.045874049002760D+00 | 2.045874049002764D+00 | 1.776356839400250D-15 |
| 2.066612504060602D+00 | 2.066612504060607D+00 | 2.442490654175344D-15 |
| 2.089094700680393D+00 | 2.089094700680398D+00 | 2.664535259100376D-15 |
| 2.112044628115711D+00 | 2.112044628115717D+00 | 3.108624468950438D-15 |
| 2.134088617826492D+00 | 2.134088617826500D+00 | 3.774758283725532D-15 |
| 2.153807268094331D+00 | 2.153807268094339D+00 | 3.996802888650564D-15 |
| 2.169793078733805D+00 | 2.169793078733812D+00 | 3.552713678800501D-15 |

Note: The third column is the lower bound error of these two pseudo-orbits according to Theorem 3.3.



$$- 0.0012162X_nX_{n-1}X_{n-2} + 0.0002248X_{n-2}^3; \quad (24)$$

$$H(X_n) = \frac{0.0003416U_n + 0.0019463U_{n-1} + 2.1579X_n}{-1.3203X_{n-1} + 0.16239X_{n-2}} - 0.0048196X_n^3 + 0.003523X_n^2X_{n-1} - 0.0012162X_nX_{n-1}X_{n-2} + 0.0002248X_{n-2}^3; \quad (25)$$

## 5. Final remarks

We presented a method to calculate a lower bound error for free-run simulation of the polynomial NARMAX. Although there are other methods to estimate the error propagation, based on the Taylor expansion or shadowing properties, these methods are complex from the point of view of computation and mathematical approach. In general, these methods try to calculate a value of errors instead of a lower bound. In our case, a lower bound is useful as it consists in a way to interrupt simulation when a minimum precision is no longer satisfied.

Our method is based on the comparison of two pseudo-orbits of the same models, but derived from two extension intervals. This brings a new idea to be further explored, the very fact that a polynomial NARMAX has multiple pseudo-orbits. It is clear that it also happens with other mathematical representation, whatever it makes use of recursive functions, which increases the relevance of this observation. When we compare two pseudo-orbits that are equivalent from the point of view of interval analysis, we proved a theorem that the sum of absolute error of each pseudo-orbit is at least equal to the distance of these two pseudo-orbits. This theorem works for each iteration  $n$ . We also presented a theorem that states that at least one of the two pseudo-orbits presents an error greater than a value, called as lower bound error, which can be easily calculated as a half of the distance of the two pseudo-orbits in each iteration  $n$ . This strategy may also be explored for more than two pseudo-orbits, which belongs to the set of future works of our research group.

We applied the methodology in three cases, which are examples of identified systems obtained from literature, by means of the polynomial NARMAX. The sine map, Chua's circuit and Duffing–Ueda oscillator are well known chaotic systems and have been identified using the polynomial NARMAX. These models have been validated using free-run simulation, but little attention has been given to the error propagation of such models due to round off and truncation operations, which are intrinsic at floating point software that uses IEEE 754-2008 standard. Although, one can think that we can simulate these models for an arbitrary  $n$  steps ahead, our results

show that it is not the case. For the sine map, we see that when  $n = 42$  iterations the relative error reaches the minimum precision adopted, that is,  $\varepsilon = 0.001$ . Also, for Chua's circuit, after 464 iterations, which means a time of  $t \approx 5.6$  ms, the relative error is superior than  $\varepsilon$ . Finally, it is not different from the Duffing–Ueda system, in which  $n = 3345$ , or approximately  $t = 175.1$  s, takes the simulation to a relative error greater than  $\varepsilon$ . It is interesting to notice that for all these three cases, as the number of iterations increases the relative error reaches the same magnitude of the pseudo-orbit, which means that we have lost the significance of all digits. Moreover, the relative error increases exponentially, as can be seen in Figures 1(b), 2(b) and 3(b). This fact also deserves further investigation.

We have found, for the three studied cases, that both pseudo-orbits present an error greater than the lower bound error at  $n = 10$ . We have used wxMaxima to check it. This is a stronger result than that established in Theorem 3.3. We believe that it is related to the fact that we have chosen two interval extensions that are equivalent according to Definition 2.3. We aim at investigating if it would be possible to generalize the results of Theorem 3.3. Finally, we also intend to analyse if the concepts present in this paper may be used to develop a strategy to calculate an upper bound for the simulation error of polynomial NARMAX.

It is important to emphasize that in a great number of published works, the double precision is used to perform the simulation, without taking into account the limits of the precision of these recursive simulations. Therefore to guarantee the reliability of the use of these models, it is recommended that a required precision should be fixed, which may imply in a maximum number of iterations for a specific software using double precision or any other finite precision system. Another important aspect, which explains the fact that in the Supplemental Material we present all of software codes, is that without the access of the software and algorithm it may be impossible to reproduce the results.

## Acknowledgments

The authors are thankful to the Brazilian agencies CNPq/INERGE, FAPEMIG [TEC APQ 01746/14] and CAPES.

## Disclosure statement

No potential conflict of interest was reported by the authors.

## References

- Aguirre, L. A. (1997). Recovering map static nonlinearities from chaotic data using dynamical models. *Physica D*, 100(1–2), 41–57.

- Aguirre, L. A., & Billings, S. A. (1994). Validating identified non-linear models with chaotic dynamics. *International Journal of Bifurcation and Chaos*, 4(1), 109–125.
- Barbosa, B. H. G., Aguirre, L. A., Martinez, C. B., & Braga, A. P. (2011). Black and gray-box identification of a hydraulic pumping system. *IEEE Transactions on Control Systems Technology*, 19(2), 398–406.
- Billings, S. A. (2013). *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. West Sussex: John Wiley & Sons.
- Chen, S., & Billings, S. A. (1989). Representations of non-linear systems: The NARMAX model. *International Journal of Control*, 49(3), 1013–1032.
- Chow, S.-N., & Palmer, K. J. (1991). On the numerical computation of orbits of dynamical systems: The one-dimensional case. *Journal of Dynamics and Differential Equations*, 3(3), 361–379.
- Chow, S.-N., & Palmer, K. J. (1992). On the numerical computation of orbits of dynamical systems: The higher dimensional case. *Journal of Complexity*, 8(4), 398–423.
- Chua, L. O., Wu, C. W., Huang, A., & Zhong, G. (1993). A universal circuit for studying and generating chaos – part I: Routes to chaos. *IEEE Transactions on Circuits and Systems I Fundamental Theory and Applications* 11/1993, 40(10), 732–744. doi:10.1109/81.246149
- Faranda, D., Mestre, M., & Turchetti, G. (2012). Analysis of round off errors with reversibility test as a dynamical indicator. *International Journal of Bifurcation and Chaos*, 22(9), 1250215-1–1250215-14.
- Feigenbaum, M. J. (1978). Quantitative universality for a class of nonlinear transformations. *Journal of Statistical Physics*, 19(1), 25–52.
- Galias, Z. (2013). The dangers of rounding errors for simulations and analysis of nonlinear circuits and systems? and how to avoid them. *IEEE Transactions on Circuits and Systems Magazine*, 13(3), 35–52.
- Gilmore, R., & Lefranc, M. (2012). *The topology of chaos: Alice in stretch and squeeze land*. Weinheim: Wiley-VCH.
- Goldberg, D. (1991). What every computer scientist should know about floating-point arithmetic. *Computing Surveys*, 23(1), 5–48.
- Hammel, S. M., Yorke, J. A., & Grebogi, C. (1987). Do numerical orbits of chaotic dynamical processes represent true orbits?. *Journal of Complexity*, 3(2), 136–145.
- Institute of Electrical and Electronics Engineers (IEEE). (2008). 754-2008 – IEEE standard for floating-point arithmetic. IEEE, p. 1–58.
- Korenberg, M., S. A. Billings, Y. P. Liu, & P. J. Mcilroy (1988). Orthogonal parameter estimation algorithm for non-linear stochastic systems. *International Journal of Control*, 48(1), 193–210.
- Liao, S., & Wang, P. (2014). On the mathematically reliable long-term simulation of chaotic solutions of Lorenz equation in the interval [0,10000]. *Science China: Physics, Mechanics and Astronomy*, 57(2), 330–335.
- Lozi, R. (2013). Can we trust in numerical computations of chaotic solutions of dynamical systems?. In Letellier Christophe (Ed.), *Topology and dynamics of chaos: In celebration of Robert Gilmore's 70th birthday* (pp. 63–98, Vol. 1). Singapore: World Scientific Publishing.
- May, R. M. (1976). Simple mathematical models with very complicated dynamics. *Nature*, 261, 459–467.
- Monteiro, L. H. A. (2002). *Sistemas dinâmicos*. São Paulo: Editora Livraria da Física.
- Moore, R. E., & Moore, R. E. (1979). *Methods and applications of interval analysis, Vol. 2*. Philadelphia: SIAM.
- Nepomuceno, E. G. (October 2014). Convergence of recursive functions on computers. *The Journal of Engineering*, 1–3. <http://digital-library.theiet.org/content/journals/10.1049/joe.2014.0228>.
- Nepomuceno, E. G., Takahashi, R. H. C., Amaral, G. F. V., & Aguirre, L. A. (2003). Nonlinear identification using prior knowledge of fixed points: A multiobjective approach. *International Journal of Bifurcation and Chaos*, 13(5), 1229–1246.
- Ott, E. (2002). *Chaos in dynamical systems*. Cambridge: Cambridge University Press.
- Overton, M. L. (2001). *Numerical computing with IEEE floating point arithmetic*. Philadelphia: SIAM.
- Rudolph-Lilith, M., & Muller, L. E. (2014). On a representation of the Verhulst logistic map. *Discrete Mathematics*, 324, 19–27. <http://www.sciencedirect.com/science/article/pii/S0012365X14000302>.
- Sauer, T. B., Grebogi, C. D., & Yorke, J. A. (1997). How long do numerical chaotic solutions remain valid?. *Physical Review Letters*, 79(1), 59–62.
- Yabuki, M., & Tsuchiya, T. (May 2013). Double precision computation of the logistic map depends on computational modes of the floating-point processing unit. ArXiv e-prints.