Using Scilab for nonlinear dynamic systems

Diego Salomão Penha Pereira
Universidade Federal de São João del-Rei
salomao@estudante.ufsj.edu.br

Samuel Maier Kurcbart
Universidade Federal de São João del-Rei
samuel@ufsj.edu.br

Erivelton Geraldo Nepomuceno
Universidade Federal de São João del-Rei
nepomuceno@ufsj.edu.br

Abstract. *Modeling a system means to develop an mathematical representation who mimics some characteristics observed in a real systems. These systems are usually nonlinear, therefore nonlinear dynamic models should be used. Softwares, such Matlab, Maple and Mathematica, are often used to obtain and analyze the models. However, these softwares are unaffordable for the majority of public institutions. In this context, this paper aims at presenting the Scilab as a free tool for nonlinear system analysis. Scilab is an environment directed toward the development of software used in numerical problems. We applied the Scilab in the study of two nonlinear dynamic systems: Chua's circuit and Duffing's oscillator. The Scilab was adequate in the simulations, being an alternative to paid softwares.*

Keywords: *Scilab, Chua's circuit, Duffing's oscillator, nonlinear dynamic systems, simulation, free software.*

1. Introduction

Modeling a system means to develop an mathematical representation who mimics some characteristics observed in a real systems. The dynamic systems when further analyzed are generally nonlinear. Usually linear approaches are enough for the analysis, but in many cases they are not satisfactory and nonlinear models should be used (what it increases the algorithm complexity). The evolution law of the dynamic systems can assume diverse forms: iterations, equations, partial differential equations, transformations or flows, stochastic equations. In this point the computational methods are essential. Softwares for the numerical resolution and qualitative analysis of dynamic systems in the education of engineering and applied sciences simplify the resolution and analysis of these systems.

The massive use of softwares generated a great business. There are several commercial softwares, and among the most important are the Matlab, Maple And Mathematica (Chonacky and Winch, 2005). Just to give an idea, the most comprehensive bibliographic database, the Web of Science[1], presents the following number of articles which contains the aforementioned softwares: 4,805 for Maple, 2,267 for Matlab and 1,062 for Mathematica.

Although the credibility and efficiency of these softwares are undeniable, their cost are not affordable for the majority of public Brazilian universities. The prices are US \$ 1,245.00, US \$ 1,569.99, US\$ 500,00 for Maple, Mathematica and Matlab, respectively. In the case of Matlab, this cost refers only to 10 licences for academic use, without any toolbox. These conditions are usually unsatisfactory, as for teaching and research purposes more licences and other toolboxes, which cost around US \$ 300,00 each, are necessary.

This paper aims at presenting the Scilab [2], as a friendly, free software and alternative tool for the nonlinear systems analysis. Comparing with other commercial softwares, Scilab is in its infancy. In the Web of Science, there are only 11 citations. There are works presenting applications in Physics (Papasotiriou and Geroyannis, 2002, 2003; Emiris and Nikitopoulos, 2005), Economy (Mrkaic, 2000). Chemistry (Fang et al., 1999) Control (Pendharkar, 2005; Pakshin and Ryabov, 2004). Mrkaic (2000) calls Scilab as a free Matlab-like programming system, which can be used for data analysis and applied numerical work in both research and teaching.

The user interaction with Scilab can be given of two forms. In the first form, the commands are typed directly in prompt (functioning as a calculator). In prompt we also see the error messages. In the second form, the commands (routines), using its editor (Scipad), can be saved as a *.sci* file. This file is taken for prompt and executed. The results are shown in prompt or by means of graphs, that when called for commands, they appear automatically in the screen.

The remain of this paper is organized as follows. Section 2 presents a brief overview of Scilab. In section 3, basic features of nonlinear systems are summarized. Section 4 and 5 present some simulations of a modified Duffing's oscillator

[1]More information can be obtained in the homepage: http://isi02.isiknowledge.com

[2]Scilab is a trademark of INRIA, "Copyright (c) 1989-2005. INRIA ENPC". Scilab license allows you to: use freely Scilab for non commercial use; use freely Scilab for commercial use if you do not use it as a derived software (ie a modified Scilab) or a composite software (ie Scilab included in another software). It can be downloaded from www.scilab.org

and Chua's Circuit, respectively. Section 6 brings some final remarks about this work.

2. A Brief Overview of Scilab

In this section a brief overview of Scilab is presented[3]. Scilab has been developed since 1990 by INRIA and ENPC researchers, and it is now maintained and developed by Scilab Consortium since its creation in May 2003. Scilab has been developed for system control and signal processing applications. It is freely distributed in source code format and is made of three distinct parts: an interpreter, libraries of functions (Scilab procedures) and libraries of Fortran and C routines. A key feature of the Scilab syntax is its ability to handle matrices: basic matrix manipulations such as concatenation, extraction or transpose are immediately performed as well as basic operations such as addition or multiplication. Scilab also aims at handling more complex objects than numerical matrices. For instance, control people may want to manipulate rational or polynomial transfer matrices. This is done in Scilab by manipulating lists and typed lists which allows a natural symbolic representation of complicated mathematical objects such as transfer functions, linear systems or graphs.

Polynomials, polynomials matrices and transfer matrices are also defined and the syntax used for manipulating these matrices is identical to that used for manipulating constant vectors and matrices. Scilab provides a variety of powerful primitives for the nonlinear system analysis. Integration of explicit and implicit dynamic systems can be accomplished numerically. The Scicos toolbox allows the graphic definition and simulation of complex interconnected hybrid systems. There exist numerical optimization facilities for non linear optimization (including non differentiable optimization), quadratic optimization and linear optimization.

Scilab has an open programming environment where the creation of functions and libraries of functions is completely in the hands of the user. Functions are recognized as data objects in Scilab and, thus, can be manipulated or created as other data objects. For example, functions can be defined inside Scilab and passed as input or output arguments of other functions. In addition Scilab supports a character string data type which, in particular, allows the on-line creation of functions. Matrices of character strings are also manipulated with the same syntax as ordinary matrices. Finally, Scilab is easily interfaced with Fortran or C subprograms. This allows use of standardized packages and libraries in the interpreted environment of Scilab.

Scilab general philosophy is to provide the following sort of computing environment:

1. To have data types which are varied and flexible with a syntax which is natural and easy to use.

2. To provide a reasonable set of primitives which serve as a basis for a wide variety of calculations.

3. To have an open programming environment where new primitives are easily added. A useful tool distributed with Scilab is intersci that is a tool for building interface programs to add new primitives i.e. to add new modules of Fortran or C code into Scilab.

4. To support library development through "toolboxes" of functions devoted to specific applications (linear control, signal processing, network analysis, non-linear control, etc.)

Scilab also provides an easy desktop to develop brief computations or for a first test of functions (see Figure 1),and has also an integrated editor (see Figure 2).

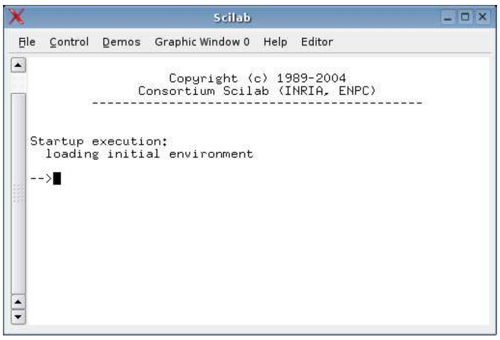

Figure 1. Desktop of Scilab.

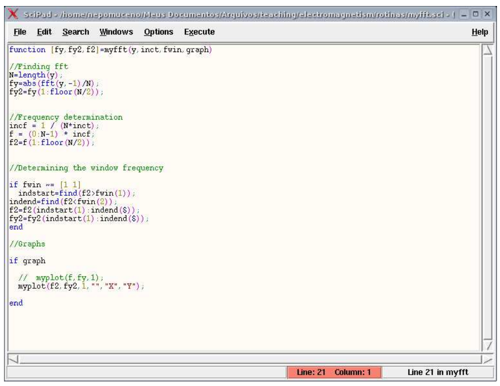

Figure 2. Editor of Scilab.

Another useful tool of Scilab is the Scicos: a dynamic systems modeler and simulator. Scicos is a Scilab toolbox included in the Scilab Package. Written in Fortran, C and Scilab language, it comes with complete source code. Scicos is a free software and its features are: a user-friendly GUI-based editor for modeling dynamic systems as block diagrams;

[3] Most of this overview was obtained from http://www.scilab.org

hierarchical block diagram structure (Super blocks); a large number of standard blocks available in various palettes; possibility of defining new blocks using C, Fortran (dynamic link) or Scilab Language; powerful formalism for modeling hybrid systems; diagram compilation and fast simulation; and C code generation. Figures 3 and 4 show an example of Scicos file and its simulation result.

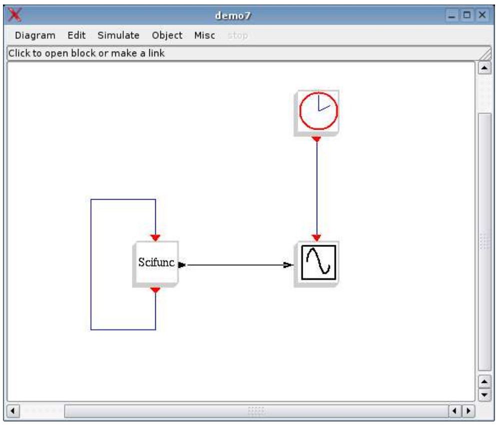

Figure 3. Scicos: a simulation toolkit based on blocks.

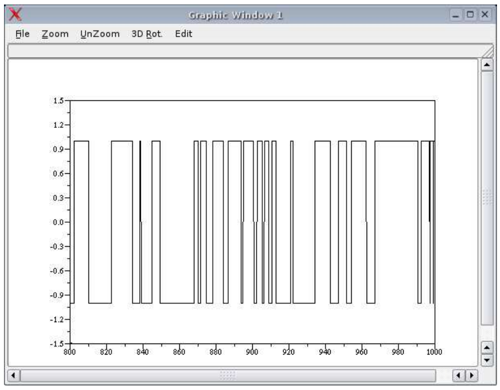

Figure 4. A simulation example of Scicos.

In Scilab there are some mathematical functions and commands of interest in the analysis of dynamic systems. There is the *ode* function, which allows the numerical solution of differential equation, given the initial conditions. The used methods are: Adams (standard); Stiff equations; Adaptive Runge-Kutta of order 4; Fehlberg's Runge-Kutta of order 4 and 5. For all of these methods, the precision can be controlled.

3. Some Concepts of Nonlinear Dynamic Systems

The time series data associated to engineering systems are often modelled by ordinary differential equations (ODE). The study of ODE allows us to determine properties of these data as pattern or transitions. In the analytical studies of the engineering models, some quantities are usefully described by the linearization of ODE and the nonlinearity in the model can be neglected. In the past, the undergraduate engineering curriculum was based on the study of linear techniques. Nevertheless nonlinear phenomena as pattern formation or solitons or chaos are not adequately understood by the linearization of the equations of motion.

When we study a nonlinear model of a engineering system, fundamental tools of linear analysis are no longer available as the principle of superposition of the solutions. There is no single general approach to study the nonlinear systems but, instead, a collection of different approaches. One is the study of the qualitative behavior of the model. In this paper, we will present Scilab as a tool for the study the qualitative behavior of engineering nonlinear systems.

We begin the qualitative study of nonlinear system with some fundamental concepts (Guckenheimer and Holmes, 1983; Fiedler-Ferrara and Prado, 1994). We consider the equations of motion that govern the evolution of a N degrees of freedom system. We now convert these equations into a set of N first-order differential equations. The system of these first-order ODE with its initial conditions is called a dynamic system. In this study we have considered dynamic systems of the following form

$$\dot{\mathbf{x}} \quad = \quad f(\mathbf{x}, t, \lambda), \tag{1}$$

where $\dot{\mathbf{x}}$, $\mathbf{x}$ are vectors with N components, λ is a vector and t is a positive, real number . The time t is the independent variable. A dynamic system can have discrete or continuous time. If the functions $f(\cdot)$ in (1) do not depend explicitly on the variable t then the system is called autonomous. The set of dependent variables forms the phase space. At each point in this space the functions f give a vector and we say that the f characterize a flow on the phase space. The solutions of the dynamic system correspond to curves in phase space that are everywhere tangent to the flow. These curves are the trajectories or solutions. To understand the qualitative behavior of the system we need construct and analyze the portrait of all trajectories of the flow on the phase space. This portrait is called phase portrait.

A fixed point or an equilibrium point or a steady state, denoted by x^*, is a solution of the dynamic system such that $f(x^*, t) = 0$. This solution does not change in time. A fixed point is stable if solutions close to the this point, x^*, at a given instant remain close to this point for later times. A fixed point is asymptotic stable if solutions close to this point converge to it as the time approaches infinity. To determine the stability of a fixed point, we do the linearization of the system near this point. In this process we obtain a linearized version of the dynamic system. If we can expand f in Taylor series and retaining only the linear terms, we get a linear system with a linear, constant coefficient ODE system formed by the derivative f', the Jacobian matrix about the fixed points. The stability properties of the fixed points can be characterized by the eigenvalues of the Jacobian. If all eigenvalues of this matrix have the negative real component then the fixed point is asymptotic stable.

In the qualitative study of the dynamic systems we investigate the solution curves in the phase space and how these curves change when the parameters system are varied continuously. The "sudden qualitative" changes in the phase curves that occur, when the system parameters λ are changed continuously, are called bifurcations or branchings.

In the next sections we want to show how Scilab can be used to construct the phase portrait of modified Duffing's oscillator and Chua's circuit.

4. The modified Duffing's oscillator

This version of the Duffing's oscillator[4] describes a particle moving along the x-axis, subject to a cubic force $\omega_0^2 x - \beta x^3$, a linear damping and a time-dependent, harmonic driving force (Aguirre and Sanjuan, 2002). This dynamic system is given by (2) e (3)

$$\dot{x} = y \tag{2}$$
$$\dot{y} = -\delta y - (\beta x^3 \omega_0^2 x) + \gamma cos(\omega t + \phi), \tag{3}$$

where δ, β, ω_0 and γ are real parameters. The parameter δ can be interpreted as the damping strength, ω_0 as the frequency and γ as the amplitude of the forcing. If there is no damping and no driving force this system is conservative and it has no attracting fixed point (Guckenheimer and Holmes, 1983). If we consider the linear damping to the unforced Duffing's oscillator, this system has no closed orbit (Guckenheimer and Holmes, 1983). The harmonic driving, damped Duffing's oscillator presents explicit time dependence and it is classified as a non-autonomous system. The parameter γ is the control parameter associated to the bifurcations in the dynamics.

Simulating the equations (4) and (5) we have gotten the time series and the phase portrait (see figures (5) to (8). We have fixed δ=0.18, β=1, ω_0=1 ω=0.8, ϕ=0 and have chosen certain values of γ. In those figures, one can see the influence of the bifurcation parameter, γ, in the flow on the phase space. There also are duplications of periods, which are well-known as a route to chaos (Fiedler-Ferrara and Prado, 1994; Guckenheimer and Holmes, 1983).

$$\dot{x} = y; \tag{4}$$
$$\dot{y} = -\delta y - \beta x^3 - \omega_0^2 x + \gamma cos(\omega t). \tag{5}$$

5. Chua's Circuit

The figure 9 schematically show the Chua's circuit (Torres and Aguirre, 2000). This circuit is not difficult to build in the laboratory, a simulated inductor L, a resistor R, capacitors C_1 and C_2 and the Chua's diode make up it. Its active element is the Chua's diode and gives the nonlinearity in the circuit. The current-voltage characteristic of this diode is given by the function

$$f(x) = Ri_d(x), \tag{6}$$

where

$$i_d(x) = \begin{cases} m_0 x + B_p(m_0 - m_1), & \text{if} \quad x < -B_p \\ m_1 x, & \text{if} \quad |x| \leq B_p \\ m_0 x + B_p(m_1 - m_0), & \text{if} \quad x > B_p \end{cases}, \tag{7}$$

and m_0, m_1 and Bp are real parameters of the Chua's diode. The current voltage characteristic is a piecewise-linear function. We have made simulations, which are not shown in this paper, with a cubic function that fits the piecewise-linear function of Chua's diode. This procedure was also successfully undertaken by means of Scilab (see figure 10).

Using Kirchhoff's current and voltage laws, we have obtained the evolution equations given by (8)

$$\begin{aligned} \dot{x} &= p[y - x - f(x)] \\ \dot{y} &= x - y + z \\ \dot{z} &= -qy + bz, \end{aligned} \tag{8}$$

[4]The original Duffing's oscillator describes a particle subjected to a double well potential (Guckenheimer and Holmes, 1983).

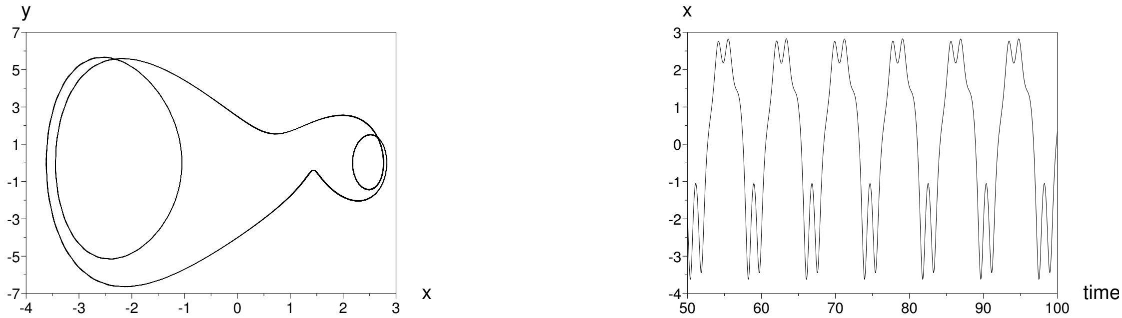

Figure 5. Phase portrait of Duffing's oscillator with γ=19.0 (period 1) and the time series.

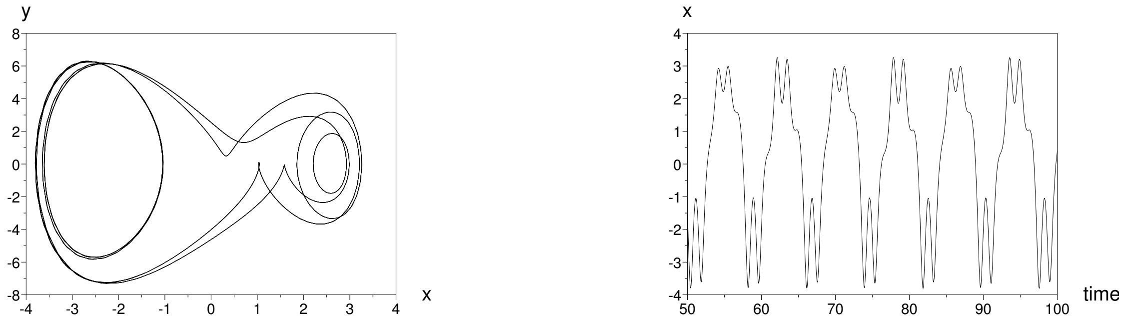

Figure 6. Phase portrait of Duffing's oscillator with γ=21.2 (period 2) and the time series

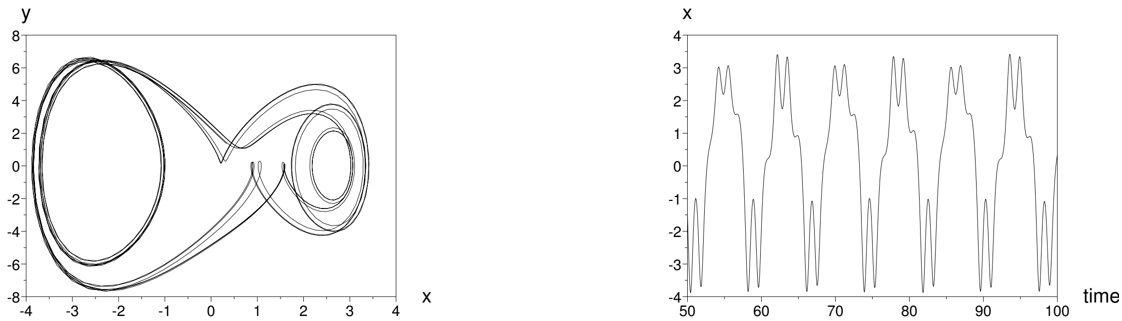

Figure 7. Phase portrait of Duffing's oscillator with γ=22.0 (period 4) and the time series

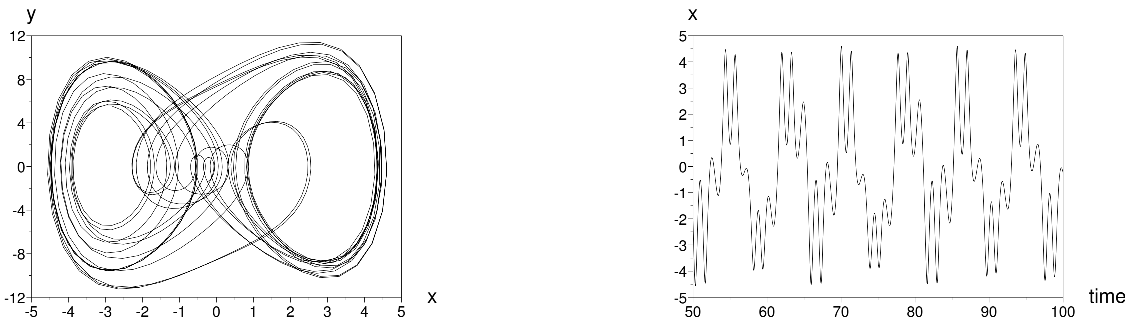

Figure 8. Phase portrait with γ=30.0 (chaos) and the time series

where $p = C_2/C_1$, $q = R^2C_2/L$ and $b = -r_L/(RL)$. The overdot in these equations means time derivative. However time t' has been scaled by $t = t'/RC$. In our implementation of this circuit we have used a simulated inductor with no associated resistance, r_L. This circuit can exhibit different nonlinear behaviors such as bifurcations, spirals, double scroll, bistability and excitability (Mira, 1997).

Simulating the Chua's circuit equation in Scilab, the following time series of the voltage in the capacitor C_1, C_2 and of the current in Chua's diode have been gotten and illustrated in the figures 11 to 15. The R value was varied in order to get

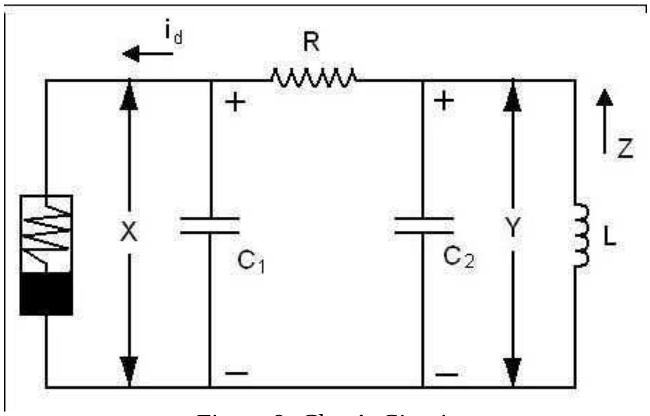

Figure 9. Chua's Circuit

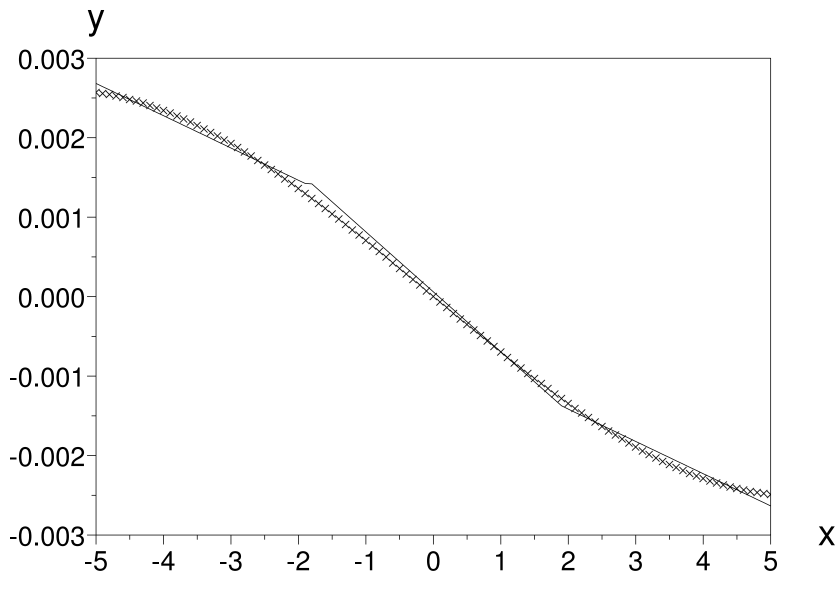

Figure 10. Linearization of Chua's diode curve using *linsolve* function of Scilab((–) simulated curve, (+) linearized curve). Axis *x* is voltage. Axis *y* is current.

different dynamic behaviors. For the Chua's circuit graphics, it can be seen that with a small variation in the parameters of the circuit, the resistance R in this case, we can get completely different behaviors.

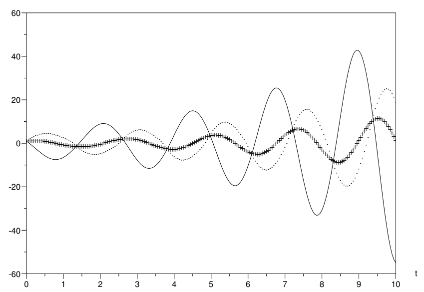
Figure 11. $R = 1500\Omega$, (–) voltage in C_1, $(\cdots)$ voltage in C_2 and (+) current in diode.

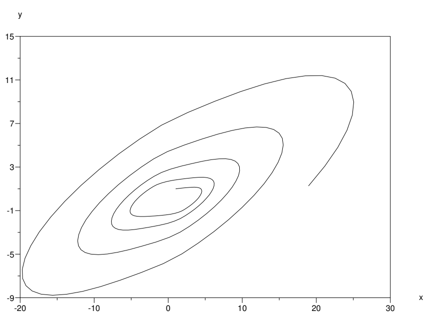
Figure 12. $R = 1500\Omega$, voltage in capacitor C_1 x voltage in capacitor C_2.

Copyright © 2005 by ABCM

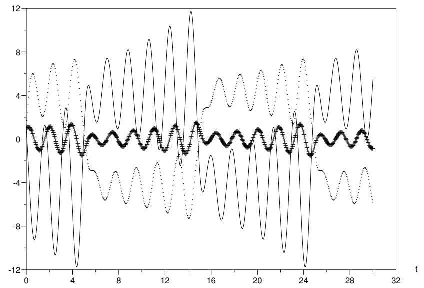

Figure 13. $R = 1800\Omega$, (–) voltage in C_1, $(\cdots)$ voltage in C_2 and (+) current in diode.

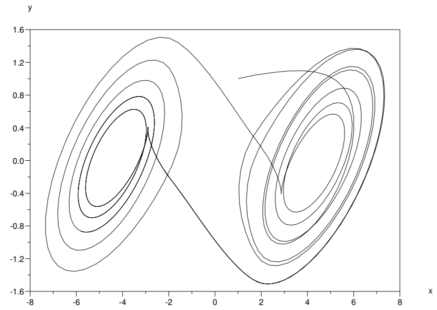

Figure 14. $R = 1800\Omega$, voltage in capacitor C_1 x voltage in capacitor C_2.

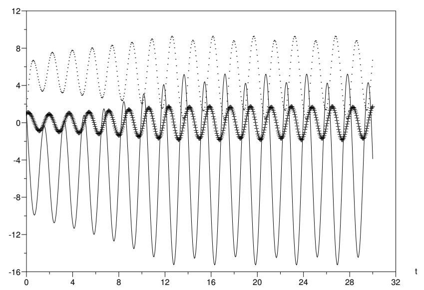

Figure 15. $R = 1900\Omega$, (–) voltage in C_1, $(\cdots)$ voltage in C_2 and (+) current in diode.

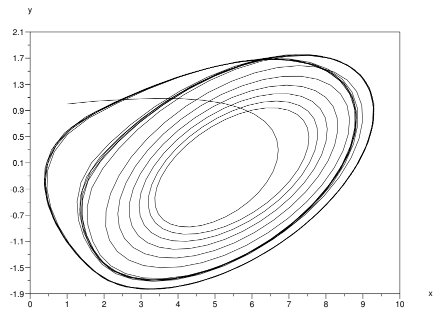

Figure 16. $R = 1900\Omega$, voltage in capacitor C_1 x voltage in capacitor C_2.

6. Final Remarks

This article has presented a brief overview of Scilab as an alternative tool for studying of nonlinear dynamic systems. In the two systems case, Chua's Circuit and Duffing's Oscillator, we see that Scilab functions are suitable for some nonlinear analysis. The help command is a simple tool, that allows us to consult the syntax functions. This help also allows a new-user a fast learning, as the syntax presents a good overview of the functions, as well as some examples.

Scilab was adequate in the simulations, being an alternative to paid softwares, as Matlab for example. The emphasis of this article has been to present a Scilab to the community of engineers and researchers. As Pendharkar (2005) pointed out "readers are invited to try out the package and evaluate it for themselves".

7. Acknowledgements

The authors are grateful to CNPq for its financial support.

8. References

Aguirre, J. and Sanjuan, M. A. F. (2002). Unpredictable behavior in the Duffing oscillator: Wada basins. *Physica D-Nonlinear Phenomena*, 171(1–2):41–51.

Chonacky, N. and Winch, D. (2005). 3Ms for instruction - Reviews of Maple, Mathematica, and Matlab. *Computing in Science & Engineering*, 7(3):7 – 13.

Emiris, I. and Nikitopoulos, T. (2005). Molecular conformation search by distance matrix perturbations. *Journal of Mathematical Chemistry*, 37(3):233 – 253.

Fang, C., Horng, W., and Lee, L. (1999). Pole clustering inside a disk for generalized state-space systems - An LMI approach. *Journal of the Chinese Institute of Engineers*, 22(4):519 – 527.

Fiedler-Ferrara, N. and Prado, C. P. C. (1994). *Caos - uma Introdução*. Editora Edgard Blücher.

Guckenheimer, J. and Holmes, P. (1983). *Nonlinear oscillations, dynamical systems, and bifurcation of vector fields*. Springer-Verlag, New York.

Mira, C. (1997). Chua's circuit and the qualitative theory of dynamical systems. *Journal of the Franklin Institute*, 334(5–6):737–744.

Mrkaic, M. (2000). Scilab as an econometric programming system.

Pakshin, P. and Ryabov, A. (2004). A static output feedback control for linear systems. *Automation and Remote Control*, 65(4):559 – 566.

Papasotiriou, P. and Geroyannis, V. (2002). A Scilab program for computing rotating magnetic compact objects. *International Journal of Modern Physics C*, 13(3):297 – 314.

Papasotiriou, P. and Geroyannis, V. (2003). A Scilab program for computing general-relativistic models of rotating neutron stars by implementing Hartle's perturbation method. *International Journal of Modern Physics C*, 14(3):321 – 350.

Pendharkar, S. (2005). Rltool for Scilab. *IEEE Control Systems Magazine*, 25:23–25.

Torres, L. A. B. and Aguirre, L. A. (2000). Inductorless Chua's circuit. *Electronics Letters*, 36(23):1915–1916.

9. Responsibility notice

The authors are the only responsible for the printed material included in this paper