Investigation on Evolutionary Chaos Controller Synthesis for Hénon Map Stabilization

[(1)]Roman Senkerik, [(1)]Zuzana Oplatkova, [(2)]Ivan Zelinka

[(1)]*Tomas Bata University in Zlin , Faculty of Applied Informatics, Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic*
[(2)]*Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic*

Abstract. This research deals with the synthesis of control law by means of Analytic Programming (AP) for the Hénon Map, which is discrete chaotic system. The tool for symbolic regression (AP) is used for the purpose of stabilization of stable state and higher periodic orbits, which represent oscillations between several values of chaotic system. For experimentation, Self-Organizing Migrating Algorithm (SOMA) with AP and Differential Evolution (DE) as the second algorithm for meta-evolution were used.

Keywords: Chaos Control, Analytic programming, optimization, evolutionary algorithms.
PACS: 02.30.Oz

INTRODUCTION

There is a growing interest about the interconnection between evolutionary techniques and control of chaotic systems. The first steps were done in [1] - [3], where the control law was based on the Pyragas method: Extended delay feedback control (ETDAS) [4]. These papers were concerned with tuning several parameters inside the control technique for chaotic system. Compared to this, current presented research shows a possibility as to how to generate the whole control law (not only to optimize several parameters) for the purpose of stabilization of a chaotic system. The synthesis of control law is inspired by the Pyragas's delayed feedback control TDAS and ETDAS [5], [6].

Analytic programming (AP) is used in this research. AP is a superstructure of EAs and is used for synthesis of analytic solution according to the required behavior. Control law from the proposed system can be viewed as a symbolic structure, which can be synthesized according to the requirements for the stabilization of the chaotic system. The advantage is that it is not necessary to have some "preliminary" control law and to estimate its parameters only. This system will generate the whole structure of the law even with suitable parameter values.

This research is an extension of previous work [7] focused on stabilization of simple p-1 orbit (stable state).

ANALYTIC PROGRAMMING

Basic principles of the AP were developed in 2001 [8]. Till then, only Genetic Programming (GP) and Grammatical Evolution (GE) had existed. GP uses Genetic Algorithms (GA) while AP can be used with any EA, independently on individual representation.

The core of AP is based on a special set of mathematical objects and operations. The set of mathematical objects is a set of functions, operators and so-called terminals (as well as in GP), which are usually constants or independent variables. This set of variables is usually mixed together and consists of functions with different number of arguments. Because of a variability of the content of this set, it is termed the "general functional set" (GFS). The structure of GFS is created by subsets of functions according to the number of their arguments. For example GFS_{all} is a set of all functions, operators and terminals, GFS_{3arg} is a subset containing functions with only three arguments, GFS_{0arg} represents only terminals, etc. The subset structure presence in GFS is vitally important for AP. It is used to

Numerical Analysis and Applied Mathematics ICNAAM 2011
AIP Conf. Proc. 1389, 1027-1030 (2011); doi: 10.1063/1.3637785

avoid synthesis of pathological programs, i.e. programs containing functions without arguments, etc. The content of GFS is dependent only on the user. Various functions and terminals can be mixed together.

The second part of the AP core is a sequence of mathematical operations, which are used for the program synthesis. These operations are used to transform an individual of a population into a suitable program. Mathematically stated, it is a mapping from an individual domain into a program domain. This mapping consists of two main parts. The first part is called Discrete Set Handling (DSH) and the second one stands for security procedures, which do not allow synthesizing pathological programs. The method of DSH, when used, allows handling arbitrary objects including nonnumeric objects like linguistic terms {hot, cold, dark…}, logic terms (True, False) or other user-defined functions. In the AP, DSH is used to map an individual into GFS and together with security procedures creates the above-mentioned mapping, which transforms arbitrary individual into a program.

AP needs some EA that consists of a population of individuals for its execution. Individuals in the population consist of integer parameters, i.e. an individual is an integer index pointing into the GFS. The individual contains numbers, which are indices into GFS. AP exists in 3 versions: AP_{basic} without constant estimation, AP_{nf} – estimation by means of nonlinear fitting package in *Mathematica* environment and AP_{meta} – constant estimation by means of another evolutionary algorithms; meta implies meta-evolution.

PROBLEM DESIGN

Selected Chaotic System

The chosen example of chaotic system was the two dimensional Hénon map in form (1):

$$\begin{aligned} x_{n+1} &= a - x_n^2 + by_n \\ y_{n+1} &= x_n \end{aligned} \tag{1}$$

The map depends on two parameters, a and b, which for the canonical Hénon map have values of $a = 1.4$ and $b = 0.3$. For these canonical values the Hénon map is chaotic.

TDAS and ETDAS Control Method

This work is focused on explanation of application of AP for synthesis of a whole control law instead of demanding tuning of TDAS or EDTAS method control laws to stabilize desired Unstable Periodic Orbits (UPO). In this research desired UPOs were p-1 (stable state) and p-2 (higher periodic orbit – oscillation between two values).

Within the research concentrated on synthesis of control law only for p-1 orbit (a fixed point), an inspiration for preparation of sets of basic functions and operators for AP was simpler TDAS control method in its discrete form given in (2):

$$F_n = K\left(x_{n-m} - x_n\right). \tag{2}$$

For the purpose of stabilization of higher periodic orbits, ETDAS method was obviously an inspiration for preparation of sets of basic functions and operators for AP. The original control method – ETDAS in the discrete form suitable for two-dimensional Hénon Map has the form (3):

$$\begin{aligned} x_{n+1} &= a - x_n^2 + by_n + F_n, \\ F_n &= K\left[(1-R)S_{n-m} - x_n\right], \\ S_n &= x_n + RS_{n-m}, \end{aligned} \tag{3}$$

where: K and R are adjustable constants, F is the perturbation; S is given by a delay equation utilizing previous states of the system and m is the period of m-periodic orbit to be stabilized. Due to the recursive attributes of delay equation S utilizing previous states of the system in discrete ETDAS (3), the data set for AP had to be expanded and

cover a longer system output history (x_n to x_{n-9} .), thus to imitate inspiring control method for the successful synthesis of control law securing the stabilization of higher periodic orbits.

Cost Function

The used CF is in general based on searching for the desired stabilized periodic orbit and thereafter calculation of the difference between desired and found actual periodic orbit on the short time interval - τ_s (20 iterations – p-1 orbit and 40 iterations – p-2 orbit) from the point, where the first min. value of difference between desired and actual system output is found. Such a design of CF should secure the successful stabilization of either p-1 orbit (stable state) or higher periodic orbit anywise phase shifted. The CF_{Basic} has the form (4).

$$CF_{Basic} = pen_1 + \sum_{t=\tau 1}^{\tau 2} \left| TS_t - AS_t \right|, \tag{4}$$

where: TS - target state, AS - actual state, τ_1 - the first min value of difference between TS and AS, τ_2 – the end of optimization interval ($\tau_1 + \tau_s$), $pen_1 = 0$ if $\tau_i - \tau_2 \geq \tau_s$; $pen_1 = 10*(\tau_i - \tau_2)$ if $\tau_i - \tau_2 < \tau_s$ (i.e. late stabilization).

Used Evolutionary Algorithms

This research used two evolutionary algorithms: Self-Organizing Migrating Algorithm (SOMA) [9] and Differential Evolution (DE) [10]. SOMA is a stochastic optimization algorithm that is modelled on the social behavior of cooperating individuals. DE is a population-based optimization method that works on real-number-coded individuals. Both algorithms were chosen because it has been proven that they have the ability to converge towards the global optimum

SIMULATION RESULTS

In this research AP_{meta} version was used. Meta-evolutionary approach means usage of one main evolutionary algorithm for AP process and second algorithm for coefficient estimation. SOMA algorithm was used for the main AP process and DE was used in the second evolutionary process. Settings of EA parameters for both processes were based on performed numerous experiments with chaotic systems and simulations with AP_{meta}. Basic set of elementary functions for AP:

GFS_{2arg} = +, -, /, *, ^
GFS_{0arg} = $data_{n-1}$ to $data_n$, K (for p-1 orbit), GFS_{0arg} = $data_{n-9}$ to $data_n$, K (for p-2 orbit).

The simulation results in Table 1 represent the best examples of synthesized control laws for the p-1 and p-2 orbit stabilization. Description of the two selected simulation results covers direct output from AP representing the synthesized control law without coefficients estimated; further the notation with simplification after estimation by means of second algorithm DE and corresponding CF value.

TABLE 1. Simulation results.

UPO	Column Header Goes Here	Column Header Goes Here	CF Value
p-1 UPO	$F_n = \left(x_n - K_1 x_{n-1}\right)\left(x_{n-1} - x_n - \frac{x_n - x_{n-1}}{x_{n-1}}\right)$	$F_n = \left(x_n - 1.46956 x_{n-1}\right)\left(x_{n-1} - x_n - \frac{x_n - x_{n-1}}{x_{n-1}}\right)$	$1.3323.10^{-15}$
p-2 UPO	$F_n = -\frac{x_{n-1}\left(K_2 - x_{n-3} - x_n\right)}{K_1}$	$F_n = 0.342699 x_{n-1}\left(0.7 - x_{n-3} - x_n\right)$	$3.8495.10^{-12}$

Simulations depicted in Figure 1 lend weight to the argument, that AP is able to synthesize a new control laws securing very quick and very precise stabilization. In the first case, the CF Value was $1.3323.10^{-15}$, which means that average error between actual and required system output was only $6.66.10^{-17}$ per iteration (CF Value calculated on 20 iterations). In the second case the final CF Value $3.8495.10^{-12}$ represents the average error $9.6237.10^{-14}$ per iteration (CF Value calculated on 40 iterations).

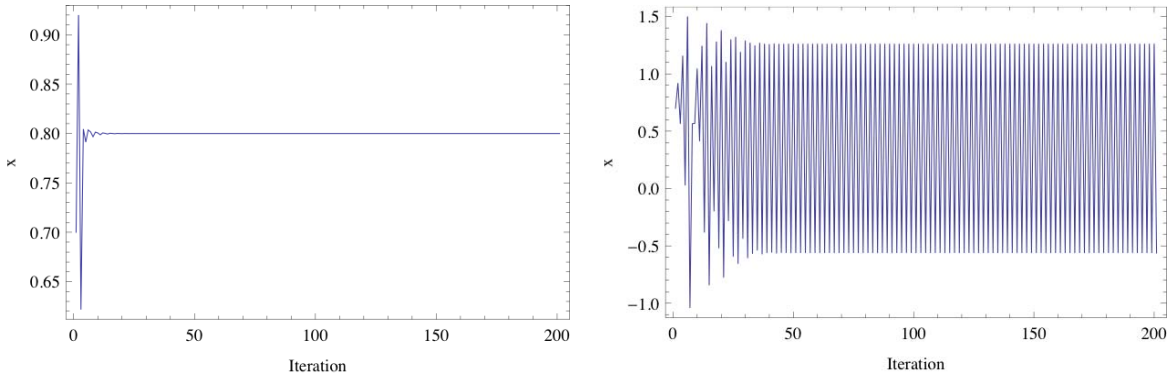

FIGURE 1. Simulation results for the best new synthesized control laws: p-1 orbit (left) and p-2 orbit (right).

CONCLUSION

This paper deals with a synthesis of a control law by means of AP for the stabilization of Hénon Map, which was selected as an example of discrete chaotic system. Obtained results reinforce the argument that AP is able to solve this kind of difficult problems and to produce a new synthesized control law in a symbolic way securing desired behavior of chaotic system. Precise stabilization lends weight to the argument, that AP is a powerful symbolic regression tool, which is able to strictly and precisely follow the rules given by cost function and synthesize any symbolic formula, in the case of this research, to synthesize the feedback controller for chaotic system. The question of energy costs and more precise stabilization will be included into future research together with the development of better cost functions, different AP data set, and performing of numerous simulations to obtain more results and produce better statistics, thus to confirm the robustness of this approach.

ACKNOWLEDGMENTS

This work was supported by the grant NO. MSM 7088352101 of the Ministry of Education of the Czech Republic and by grants of Grant Agency of Czech Republic GACR 102/09/1680 and by European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

REFERENCES

1. R. Senkerik, I. Zelinka, D. Davendra, Z. Oplatkova, "Evolutionary Design of Chaos Control in 1D*"*, in. *Evolutionary Algorithms and Chaotic Systems*, edited by I. Zelinka, S. Celikovski., H. Richter., G. Chen, SpringerVerlag Berlin, 2010, pp.165 - 190.
2. R. Senkerik, I. Zelinka, D. Davendra, Z. Oplatkova., "Utilization of SOMA and differential evolution for robust stabilization of chaotic Logistic equation", *Computers & Mathematics with Applications* **60**, pp. 1026-1037 (2004).
3. I. Zelinka., R. Senkerik, E. Navratil, "Investigation on evolutionary optimization of chaos control", *Chaos, Solitons & Fractals* **40**, pp. 111-129 (2009).
4. K. Pyragas, "Control of chaos via extended delay feedback", *Physics Letters A* **206**, pp. 323-330 (1995).
5. K. Pyragas, "Continuous control of chaos by self-controlling feedback", *Physics Letters A* **170**, 421-428 (1992).
6. W. Just, "Principles of Time Delayed Feedback Control", in *Handbook of Chaos Control*, edited by H.G. Schuster Wiley-Vch, 1999.
7. R. Senkerik, Z. Oplatkova, I. Zelinka, D. Davendra, R. Jasek, "Synthesis Of Feedback Controller For Chaotic Systems By Means Of Evolutionary Techniques", Proceeding of Fourth Global Conference on Power Control and Optimization, Sarawak, Borneo, 2010.
8. I. Zelinka, Z. Oplatkova, L. Nolle, "Boolean Symmetry Function Synthesis by Means of Arbitrary Evolutionary Algorithms-Comparative Study", *International Journal of Simulation Systems, Science and Technology* **6**, pp. 44 – 56 (2005).
9 I. Zelinka, "SOMA – Self Organizing Migrating Algorithm", in *New Optimization Techniques in Engineering*, edited by B.V. Babu, G. Onwubolu, chapter 7, 33, Springer-Verlag, 2004.
10 K. Price, R. M. Storn, J. A. Lampinen, "*Differential Evolution: A Practical Approach to Global Optimization"*, Natural Computing Series*,* Springer, 1995.