

Research Article

Chaos Enhanced Differential Evolution in the Task of Evolutionary Control of Selected Set of Discrete Chaotic Systems

**Roman Senkerik,¹ Ivan Zelinka,² Michal Pluhacek,¹
Donald Davendra,² and Zuzana Oplatková Kominková¹**

¹ Faculty of Applied Informatics, Tomas Bata University in Zlin, Náměstí T.G. Masaryka 5555, 760 01 Zlin, Czech Republic

² Faculty of Electrical Engineering and Computer Science, VŠB-Technical University of Ostrava, 17 Listopadu 15, Poruba, 708 33 Ostrava, Czech Republic

Correspondence should be addressed to Roman Senkerik; senkerik@fai.utb.cz

Received 6 March 2014; Accepted 7 July 2014; Published 26 August 2014

Academic Editor: Gerhard-Wilhelm Weber

Copyright © 2014 Roman Senkerik et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Evolutionary technique differential evolution (DE) is used for the evolutionary tuning of controller parameters for the stabilization of set of different chaotic systems. The novelty of the approach is that the selected controlled discrete dissipative chaotic system is used also as the chaotic pseudorandom number generator to drive the mutation and crossover process in the DE. The idea was to utilize the hidden chaotic dynamics in pseudorandom sequences given by chaotic map to help differential evolution algorithm search for the best controller settings for the very same chaotic system. The optimizations were performed for three different chaotic systems, two types of case studies and developed cost functions.

1. Introduction

In many engineering applications, one of the most innate tasks is the controlling of highly nonlinear dynamical systems in order to either eliminate or synchronize the chaos. The first pioneering approach to control chaotic dynamics by means of a simple analytical linearization method was introduced in 1990s by Ott et al. (i.e., OGY method) [1]. Subsequently, the rapid development of methods for stabilizing of nonlinear chaotic dynamics has arisen and many advanced techniques have been applied for chaos control and chaos synchronization including methods from the artificial intelligence field.

During recent years, usage of new intelligent systems in engineering, technology, modeling, computing, and simulations has attracted the attention of researchers worldwide. The most current methods are mostly based on soft computing, which is a discipline tightly bound to computers, representing a set of methods of special algorithms, belonging to the artificial intelligence paradigm. The most popular of these methods are neural networks, evolutionary algorithms (EAs), fuzzy logic, and tools for symbolic regression like genetic programming. Currently, EAs are known as a powerful set

of tools for almost any difficult and complex optimization problem.

The interest about the interconnection between evolutionary techniques and control of chaotic systems is rapidly spreading. The initial research was conducted in [2], whereas [3, 4] was more concerned with the tuning of parameters inside the chaos control technique based on the Pyragas method: extended delay feedback control (ETDAS) [5]. When compared to the aforementioned research, later works [6, 7] show a possibility as to how to generate the entire control law (not only how to optimize several parameters) for the purpose of stabilization of a chaotic system. Such approach also may overcome the possible sensitivity to initial conditions which may lead to stability issues. The synthesis of control is inspired by the Pyragas' delayed feedback control technique [8, 9]. This method is very advantageous for the evolutionary computation, due to its amount of easy accessible control parameters, which can be easily tuned by means of EAs.

Other approaches utilizing the EAs for stabilizing chaotic dynamics have mostly applied the particle swarm optimization (PSO) algorithm [10] and multi-interval gradient

method [11] or minimum entropy control technique [12]. EAs have been also frequently used in the task of synchronization of chaos [13–15]. In [16], an EA for optimizing local control of chaos based on a Lyapunov approach is presented.

Evolutionary techniques were also used for the synthesis of new complex discrete type structures with chaotic behavior [17] as well as the synthesis (identification) of a mathematical model of chaotic system based on the measured data [18].

Another example of interconnection between deterministic chaos and EAs represents the research focused on the embedding of chaotic dynamics into the EAs. Recent research in chaos driven heuristics has been fueled with the predisposition that, unlike stochastic approaches, a chaotic approach is able to bypass local optima stagnation. This one clause is of deep importance to EAs. A chaotic approach generally uses the chaotic map in the place of a pseudorandom number generator [19]. This causes the heuristic to map unique regions, since the chaotic map iterates to new regions. The task is then to select a very good chaotic map as the pseudorandom number generator.

The initial concept of embedding chaotic dynamics into EAs is given in [20]. Later, the initial study [21] was focused on the simple embedding of chaotic systems in the form of chaos pseudorandom number generator (CPRNG) for differential evolution (DE) [22] and self-organizing migrating algorithm (SOMA) [23] in the task of optimal PID tuning. Also, the PSO algorithm with elements of chaos was introduced as the CPSO [24]. This field of research was later extended with the successful experiments with chaos driven DE [25] in real domain as well as in combinatorial problems domain [26, 27].

At the same time, the chaos embedded PSO with inertia weigh strategy was closely investigated [28], followed by the introduction of a PSO strategy driven alternately by two chaotic systems [29] and novel chaotic multiple choice PSO (chaos MC-PSO) strategy [30]. Recently, the chaotic firefly algorithm was also introduced [31].

Finally, the last example represents the research focusing on the EAs and the edge of chaos. An unconventional approach of the edge of chaos and its application to discrete systems and evolutionary algorithms in terms of stagnation avoidance is presented in [32].

The organization of this paper is as follows: firstly, used evolutionary technique, which is DE, is described and is followed by the description of the ChaosDE concept. Thereafter, the problem design and appropriate corresponding cost functions are investigated and proposed. Results and conclusion follow afterwards.

2. Motivation

This paper extends the research of evolutionary chaos control optimization by means of ChaosDE algorithm [33–35]. Recent studies have shown that differential evolution [22] is one of the most potent heuristics and it has been used for a number of optimization tasks; [36–38] have explored DE for combinatorial problems; [39, 40] have hybridized DE whereas [41–43] have developed self-adaptive DE variants.

In this paper, the DE/rand/1/bin strategy driven by different chaotic maps (systems) was utilized to solve the issue of

evolutionary optimization of chaos control for the very same chaotic system used as a CPRNG in the particular case study. Thus, the idea was to utilize the hidden chaotic dynamics in pseudorandom sequences given by chaotic map to help differential evolution algorithm search for the best controller settings for the very same chaotic system. Since the very positive contribution of the chaotic dynamics to the performance of DE in the task of evolutionary chaos control optimization was proven in comparison with original canonical DE within the initial study [44], this paper is not primarily focused on the performance comparisons with different heuristic. But this research extends the initial work with the aforementioned idea and with the several case studies combining different chaotic systems and different utilized cost functions.

3. Used Heuristic: Differential Evolution (DE)

DE is a simple and powerful population-based optimization method that works either on real-number-coded individuals or with small modifications on discrete type individuals [22, 45, 46]. DE is quite robust, fast, and effective, with global optimization ability. This global optimization ability has been proven in many interdisciplinary researches. It works well even with noisy and time-dependent objective functions. The canonical basic principle is as follows.

For each individual $\vec{x}_{i,G}$ in the current generation G, DE generates a new trial individual $\vec{x}'_{i,G}$ by adding the weighted difference between two randomly selected individuals $\vec{x}_{r1,G}$ and $\vec{x}_{r2,G}$ to a randomly selected third individual $\vec{x}_{r3,G}$. The resulting individual $\vec{x}'_{i,G}$ is crossed over with the original individual $\vec{x}_{i,G}$. The fitness of the resulting individual, referred to as a perturbed vector $\vec{u}_{i,G+1}$, is then compared with the fitness of $\vec{x}_{i,G}$. If the fitness of $\vec{u}_{i,G+1}$ is greater than the fitness of $\vec{x}_{i,G}$, then $\vec{x}_{i,G}$ is replaced with $\vec{u}_{i,G+1}$; otherwise, $\vec{x}_{i,G}$ remains in the population as $\vec{x}_{i,G+1}$.

Please refer to (1) for notation of crossover and to [22] for the detailed description of used DERand1Bin strategy and all other DE strategies:

$$u_{i,G+1} = x_{r1,G} + F \cdot (x_{r2,G} - x_{r3,G}). \quad (1)$$

4. Concept of ChaosDE

This section contains the description of discrete dissipative chaotic maps, which can be used as the chaotic pseudorandom generators for DE as well as the main principle of the ChaosDE concept. In this research, direct output iterations of the chaotic maps were used for the generation of real numbers in the process of crossover based on the user defined CR value and for the generation of the integer values used for selection of individuals.

4.1. Chaotic Pseudorandom Number Generator. The general idea of ChaosDE and CPRNG is to replace the default PRNG with the discrete chaotic map. As the discrete chaotic map is a set of equations with a static start position, we created a random start position of the map, in order to have different start position for different experiments (runs of EAs). This

random position is initialized with the default PRNG, as a one-off randomizer. Once the start position of the chaotic map has been obtained, the map generates the next sequence using its current position.

The first possible way is to generate and store a long data sequence (approximately 50–500 thousands numbers) during the evolutionary process initialization and keep the pointer to the actual used value in the memory. In case of the using up of the whole sequence, the new one will be generated with the last known value as the new initial one.

The second approach is that the chaotic map is not reinitialized during the experiment and any long data series is not stored; thus it is imperative to keep the current state of the map in memory to obtain the new output values.

As two different types of numbers are required in ChaosDE, real and integers, the use of modulo operators is used to obtain values between the specified ranges, as given in the following equations (2):

$$\text{rndreal} = \text{mod}(\text{abs}(\text{rndChaos}), 1.0) \tag{2}$$

$$\text{rndint} = \text{mod}(\text{abs}(\text{rndChaos}), 1.0) \times \text{Range} + 1,$$

where *abs* refers to the absolute portion of the chaotic map generated number *rndChaos*, *mod* is the modulo operator, and *Range* specifies the value (inclusive) till where the number is to be scaled.

4.2. Selected Chaotic Systems. This subsection contains the mathematical and graphical description of the three selected discrete dissipative systems, which served both as for CPRNGs and also as the examples of systems to be evolutionary controlled.

4.2.1. Burgers Map. The Burgers mapping is a discretization of a pair of coupled differential equations which were used by Burgers to illustrate the relevance of the concept of bifurcation to the study of hydrodynamics flows. The map equations are given in (3) with control parameters $a = 0.75$ and $b = 1.75$ as suggested in [47]:

$$\begin{aligned} X_{n+1} &= aX_n - Y_n^2, \\ Y_{n+1} &= bY_n + X_nY_n. \end{aligned} \tag{3}$$

4.2.2. Delayed Logistic. The delayed logistic is a simple two-dimensional discrete system similar to the one-dimensional logistic equation. The map equations are given in (4). The parameter used in this work is $A = 2.27$ as also suggested in [47]:

$$\begin{aligned} X_{n+1} &= AX_n(1 - Y_n), \\ Y_{n+1} &= X_n. \end{aligned} \tag{4}$$

4.2.3. Lozi Map. The Lozi map is a simple discrete two-dimensional chaotic map. The map equations are given in (5). The parameters used in this work are as follows: $a = 1.7$ and $b = 0.5$ as suggested in [47]. For these values, the system exhibits typical chaotic behavior and, with this parameter

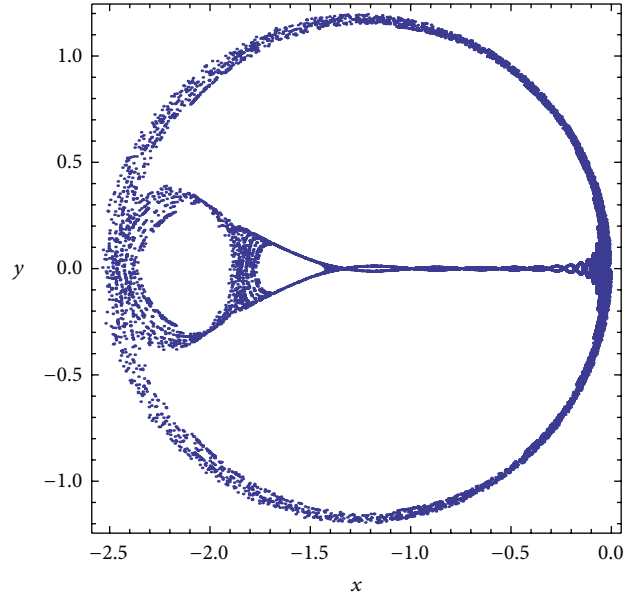


FIGURE 1: *x, y* plot of the Burgers map.

setting, it is used in most research papers and other literature sources [48]:

$$\begin{aligned} X_{n+1} &= 1 - a|X_n| + bY_n, \\ Y_{n+1} &= X_n. \end{aligned} \tag{5}$$

4.3. Graphical Examples. Three chaotic maps were selected for the CPRNG concept. The *x, y* plots of the selected maps are depicted in Figure 1 (Burgers map), Figure 4 (delayed logistic map), and Figure 7 (Lozi map). The chaotic behavior of the chaotic maps, represented by the examples of direct output iterations, is depicted in Figures 2, 5, and 8. Finally, the illustrative histograms of the distribution of real numbers transferred into the range $\langle 0-1 \rangle$ generated by means of chaotic maps are in Figures 3, 6, and 9.

5. Design of Cost Functions for Chaotic System Stabilization

The proposal of the basic cost function (CF_{Simple}) is in general based on the simplest CF, which could be used problem-free only for the stabilization of $p-1$ orbit. The idea was to minimize the area created by the difference between the required state and the real system output on the whole simulation interval, τ_i (6). This CF design is very convenient for the evolutionary searching process due to the relatively favorable CF surface. Nevertheless, this simple approach has one big disadvantage, which is the including of initial chaotic transient behavior of not stabilized system into the cost function value. As a result of this, the very tiny change of control method setting for extremely sensitive chaotic system (given by the very small change of CF value) can be

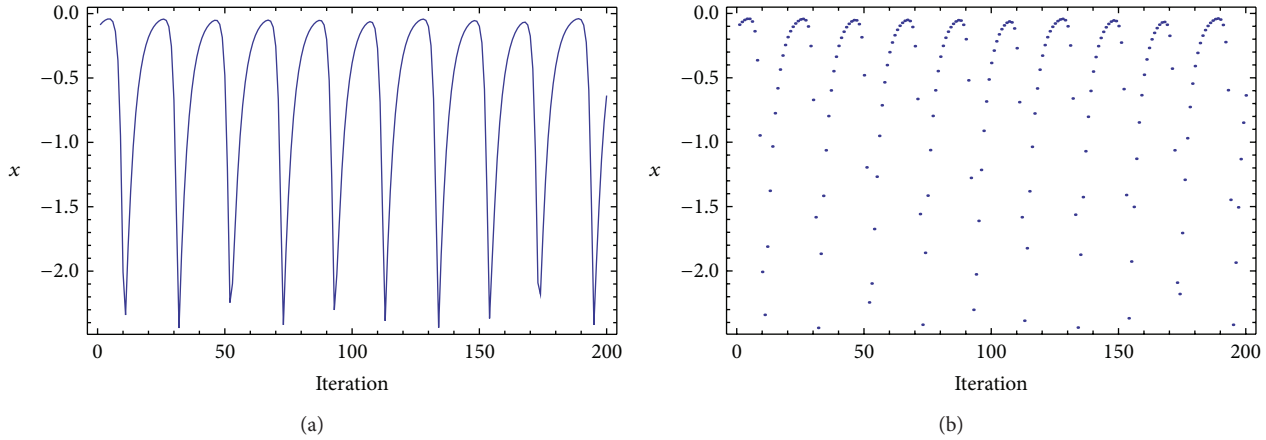


FIGURE 2: Iterations of the Burgers map (variable x) (a), iterations of the Burgers map (variable x), point-plot (b).

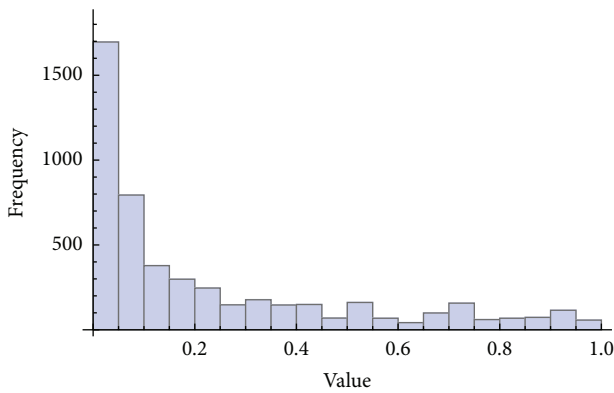


FIGURE 3: Histogram of the distribution of real numbers transferred into the range (0-1) generated by means of the chaotic Burgers map, 5000 samples.

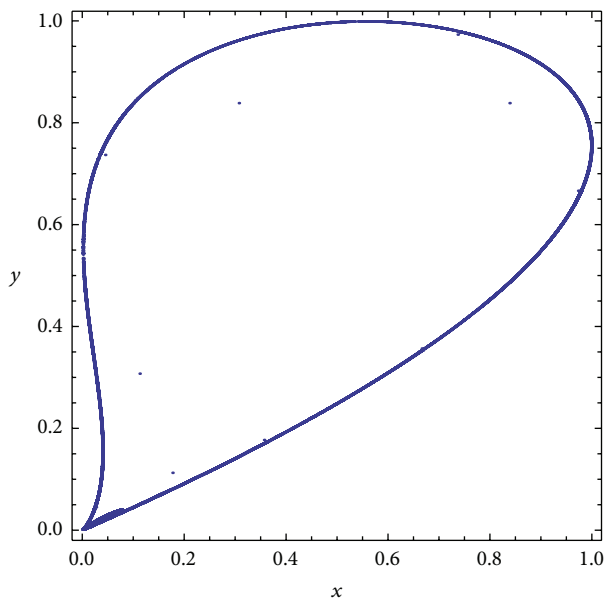


FIGURE 4: x, y plot of the delayed logistic map.

suppressed by the aforementioned inclusion of initial chaotic transient behavior. Consider

$$CF_{\text{Simple}} = \sum_{t=0}^{\tau_t} |TS_t - AS_t|, \quad (6)$$

where TS is target state and AS is actual state.

Different type of universal cost function without any selection rules is purely 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), from the point where the first minimal value of difference between desired and actual system output is found (i.e., floating window for minimization; see Figure 10).

Such a design of universal CF should secure the successful stabilization of either p-1 orbit (stable state) or any higher periodic orbit anyway phase shifted. Furthermore, due to CF values converging towards zero, this CF also allows the use of decision rules, avoiding very time demanding simulations. This rule stops EA immediately, when the first individual with good parameter structure is reached; thus the value of CF is lower than the acceptable (CF_{acc}) one. Based on the numerous experiments, typically $CF_{\text{acc}} = 0.001$ at time interval $\tau_s = 20$ iterations; thus the difference between desired and actual output has the value of 0.0005 per iteration, that is, successful stabilization for the used control technique. The CF_{UNI} has the following form:

$$CF_{\text{UNI}} = pen_1 + \sum_{t=\tau_1}^{\tau_2} |TS_t - AS_t|, \quad (7)$$

where τ_1 is the first min value of the difference between TS and AS and τ_2 is 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).

The issue of pure searching for periodic orbits causes very chaotic, erratic, and discrete type CF surfaces.

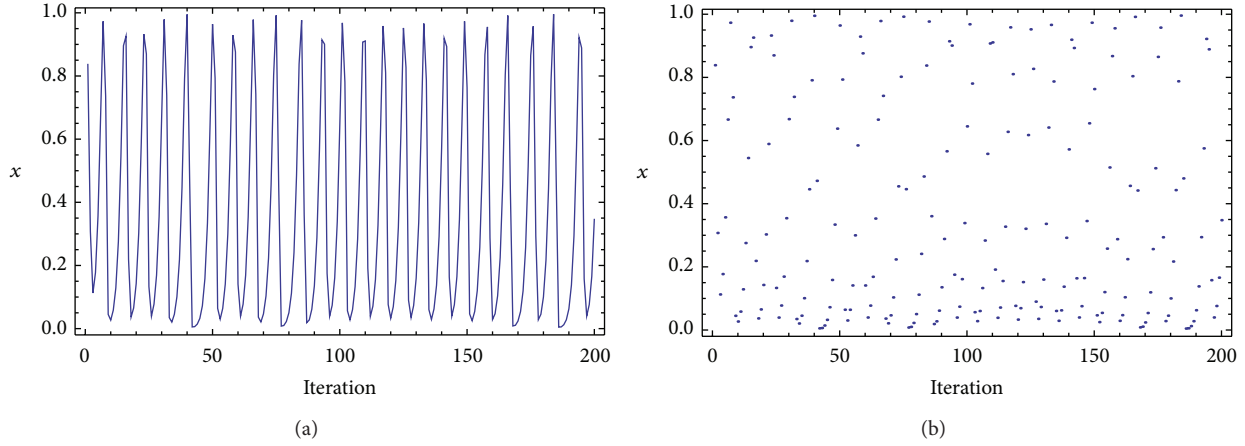


FIGURE 5: Iterations of the delayed logistic (variable x) (a), iterations of the delayed logistic (variable x), point-plot (b).

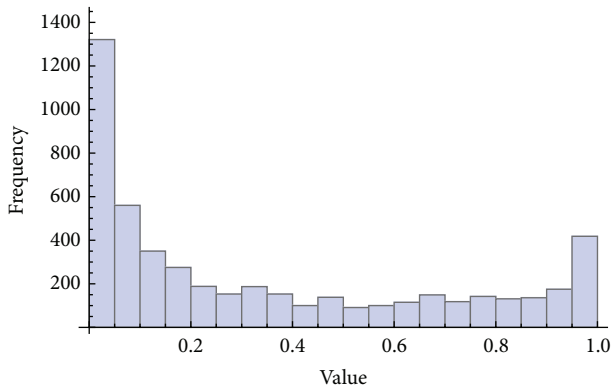


FIGURE 6: Histogram of the distribution of real numbers transferred into the range (0-1) generated by means of the chaotic delayed logistic map, 5000 samples.

6. Experimental Design

This research encompasses six case studies. Six different sets of discrete chaotic systems as CPRNGs/to be controlled and two different cost functions were combined in the following form:

- (i) case study 1: Burgers map as CPRNG/controlled system with CF_{Simple} ,
- (ii) case study 2: Burgers map as CPRNG/controlled system with CF_{UNI} ,
- (iii) case study 3: delayed logistic map as CPRNG/controlled system with CF_{Simple} ,
- (iv) case study 4: delayed logistic map as CPRNG/controlled system with CF_{UNI} ,
- (v) case study 5: Lozi map as CPRNG/controlled system with CF_{Simple} ,
- (vi) case study 6: Lozi map as CPRNG/controlled system with CF_{UNI} .

This work is focused on the utilization of the chaos driven DE for tuning of parameters for ET DAS control method

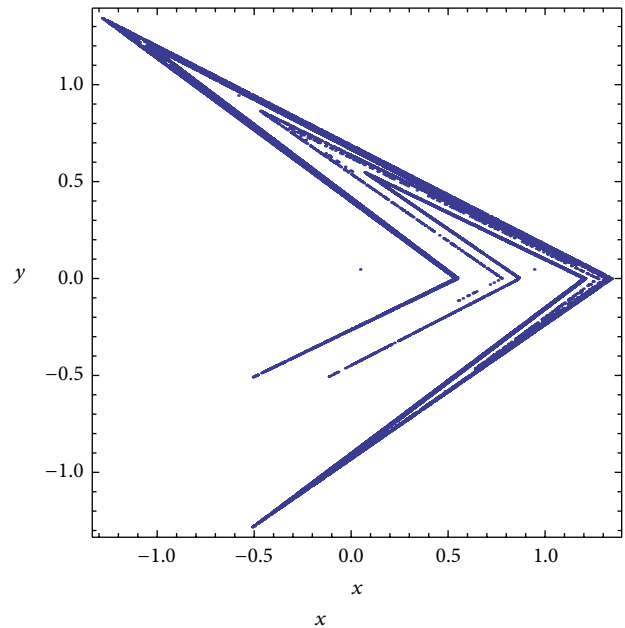


FIGURE 7: x, y plot of the Lozi map.

to stabilize desired unstable periodic orbits (UPO). In the described research, desired UPO was p-1 (stable state). The original control method, ET DAS, in the discrete form suitable for discrete chaotic maps has the following form:

$$F_n = K [(1 - R) S_{n-m} - x_n], \tag{8}$$

$$S_n = x_n + R S_{n-m},$$

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. The perturbation F_n in (8) may have arbitrarily large value, which can cause the divergence of the system. Therefore, F_n should have a value between $-F_{max}$ and F_{max} . The ranges of all evolutionary estimated parameters are given in Table 1.

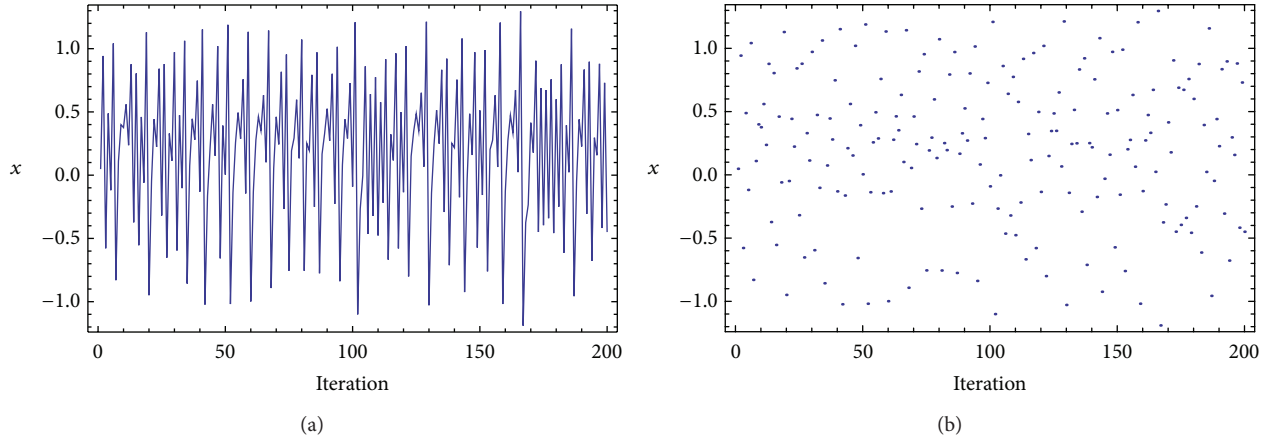


FIGURE 8: Iterations of the Lozi map (variable x) (a), iterations of the Lozi map (variable x), point-plot (b).

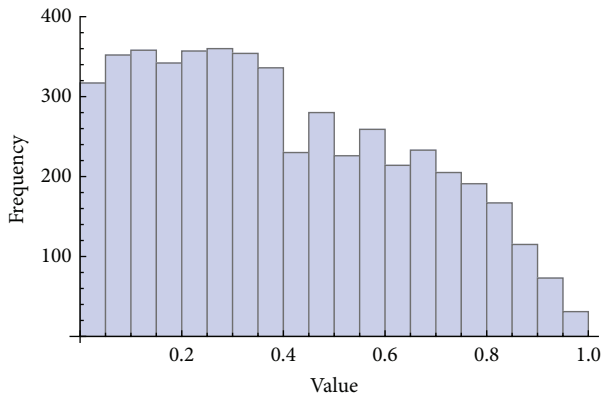


FIGURE 9: Histogram of the distribution of real numbers transferred into the range $\langle 0-1 \rangle$ generated by means of the chaotic Lozi map, 5000 samples.

TABLE 1: Estimated parameters.

Parameter	Min	Max
K	-2	2
R	0	0.99
F_{\max}	0	0.9

TABLE 2: DE settings.

Parameter	Value
Population size	25
F	0.8
Cr	0.8
Generations	250
Max cost function evaluations (CFE)	6250

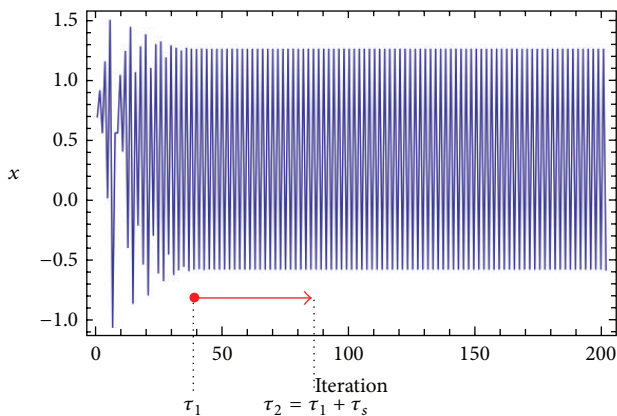


FIGURE 10: "Floating window" for minimization.

Within the research, a total number of 50 simulations with ChaosDE were carried out for each case study. The parameter settings for ChaosDE were obtained analytically based on numerous experiments and simulations (see

Table 2). Experiments were performed in an environment of *Wolfram Mathematica*; PRNG operations therefore used the built-in *Mathematica software* pseudorandom number generator. All experiments used different initialization; that is, different initial population was generated in each run of chaos driven DE.

7. Results

All simulations were successful and have given new optimal settings for ETDAS control method securing the fast stabilization of the chaotic system at required behaviour (p-1 orbit). Tables 4, 5, 6, 7, 8, and 9 contain the simple statistical overview of optimization/simulation results as well as the best founded individual solutions of parameters setup for ETDAS control method, corresponding final CF value, also, the Istab. Value representing the number of iterations required for stabilization on desired UPO, and further the average error between desired output value and real system output from the last 20 iterations.

Graphical simulation outputs of the best individual solutions for particular case studies are depicted in Figures 11, 13, and 15, whereas Figures 12, 14, and 16 show the simulation

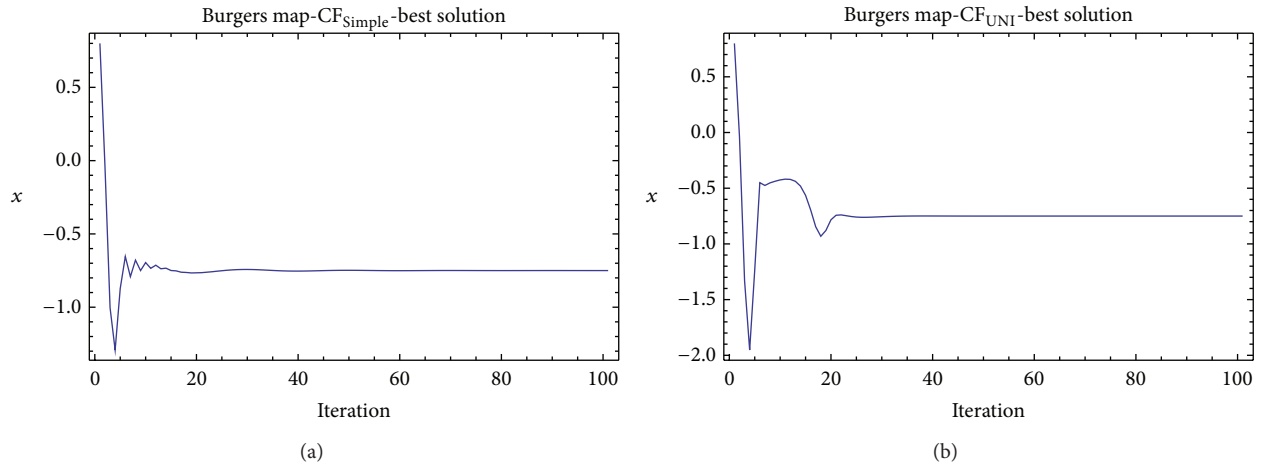


FIGURE 11: Simulation of the best individual solution, ChaosDE and Burgers map: case study 1, CF_{Simple} (a); case study 2, CF_{UNI} (b).

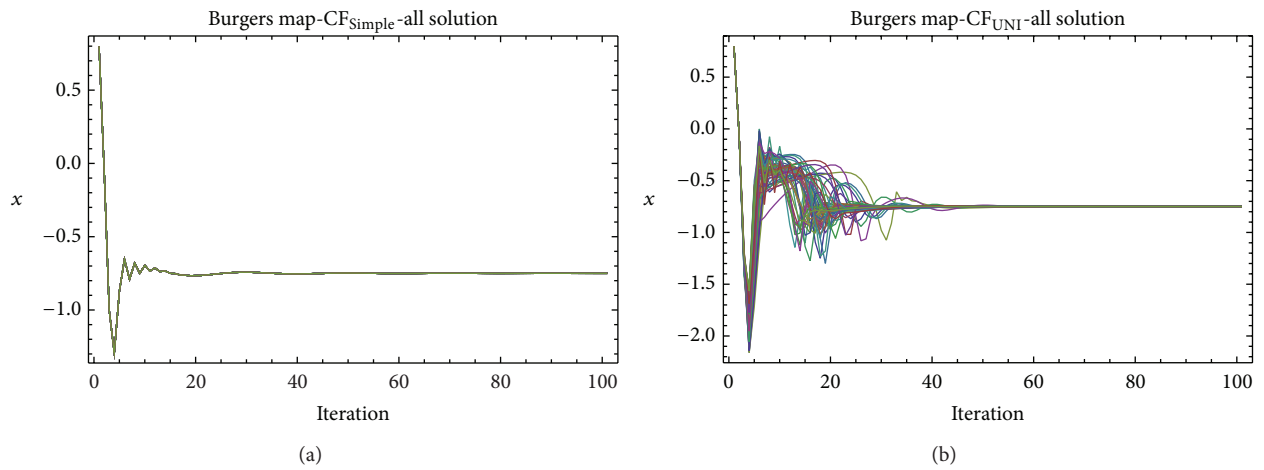


FIGURE 12: Simulation of all the 50 independent runs of EA, ChaosDE and Burgers map: case study 2, CF_{Simple} (a); case study 1, CF_{UNI} (b).

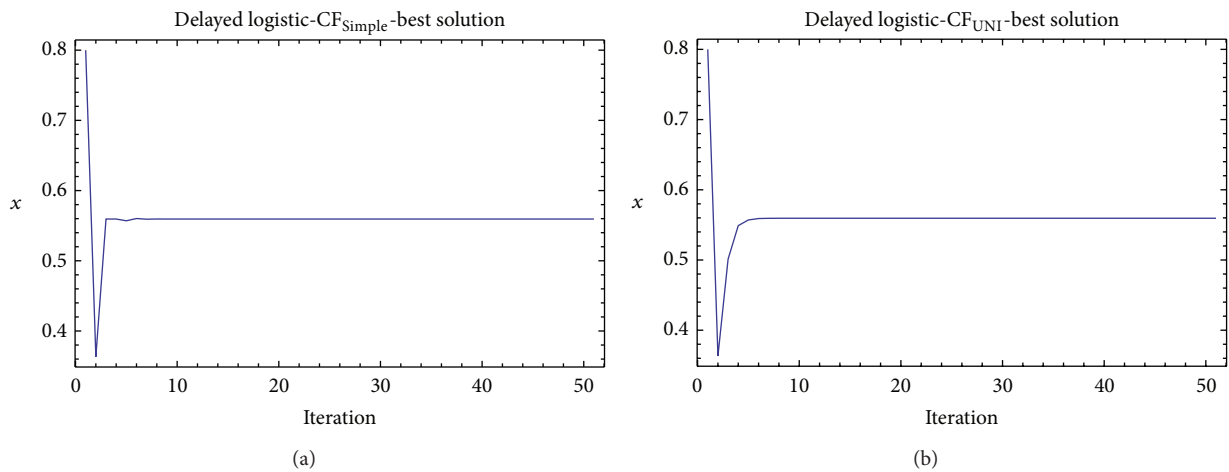


FIGURE 13: Simulation of the best individual solution, ChaosDE and delayed logistic: case study 3, CF_{Simple} (a); case study 4, CF_{UNI} (b).

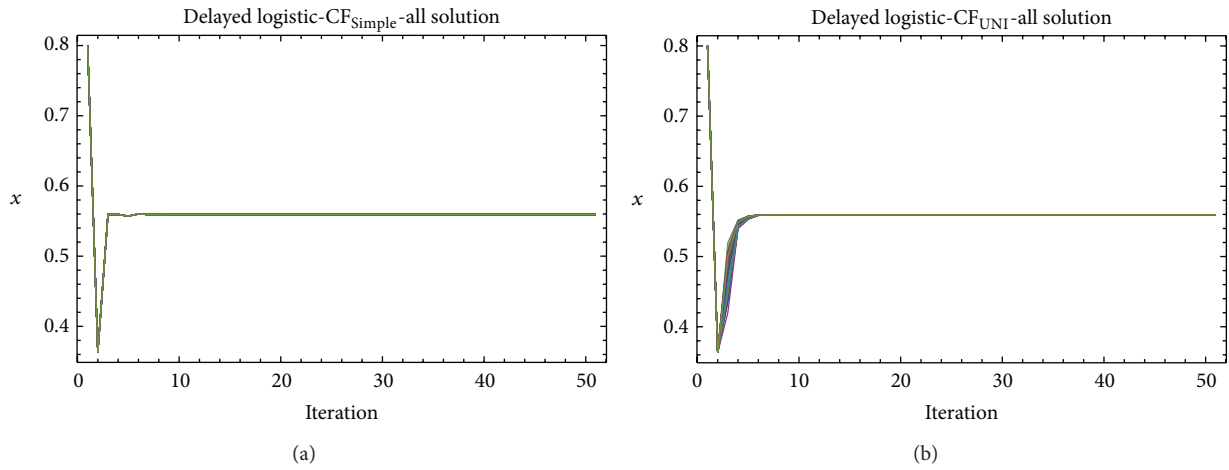


FIGURE 14: Simulation of all the 50 independent runs of EA, ChaosDE and Delayed logistic: case study 3, CF_{Simple} (a); case study 4, CF_{UNI} (b).

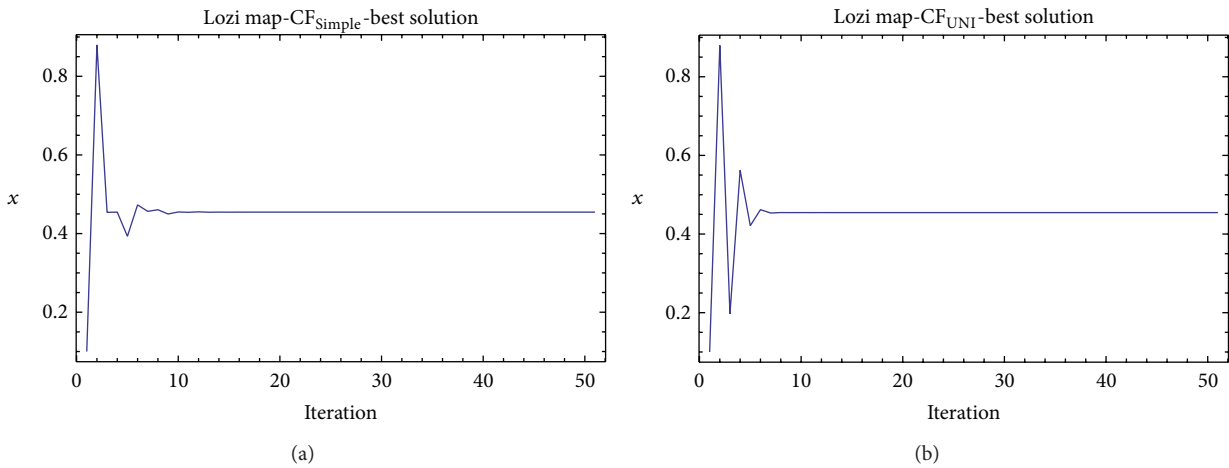


FIGURE 15: Simulation of the best individual solution, ChaosDE and Lozi map: case study 5, CF_{Simple} (a); case study 6, CF_{UNI} (b).

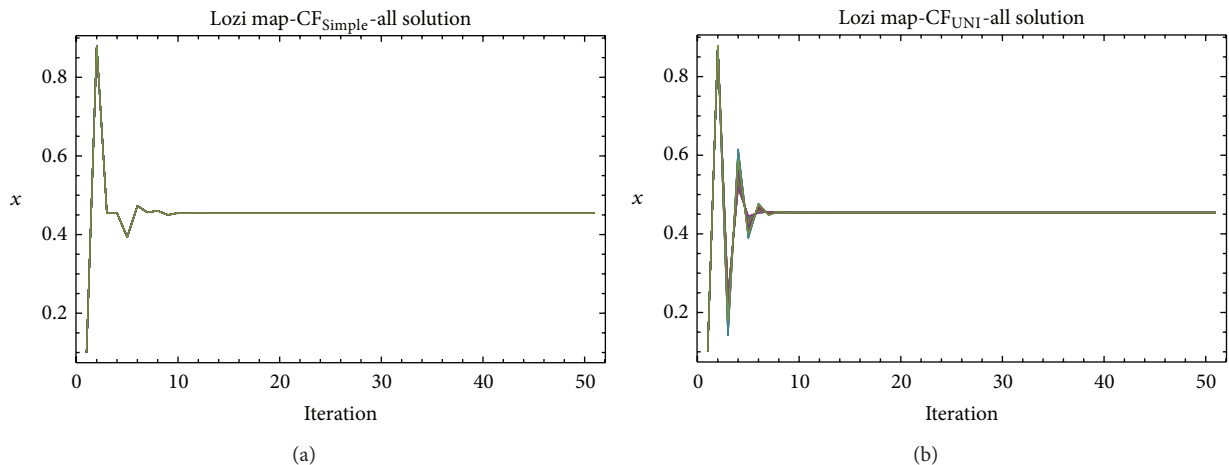


FIGURE 16: Simulation of the all 50 independent solution runs of EA, ChaosDE and Lozi map: case study 5, CF_{Simple} (a); case study 6, CF_{UNI} (b).

TABLE 3: The values for p-1 UPO.

Chaotic system	Values of p-1 UPO of unperturbed system
Burgers map	$x_F = -0.7499$
Delayed logistic map	$x_F = 0.559471$
Lozi map	$x_F = 0.454545$

TABLE 4: Simple CF statistics: joined case studies 1 and 2, Burgers map as CPRNG/controlled system.

Statistical data	Case study 1	Case study 2
	CF _{simple}	CF _{UNI}
	CF value	CF value
Min	2.16199	$1.05 \cdot 10^{-6}$
Max	2.16199	0.0103
Average	2.16199	$6.67 \cdot 10^{-4}$
Median	2.16199	$5.32 \cdot 10^{-7}$
Std. dev.	$5.58 \cdot 10^{-11}$	$1.89 \cdot 10^{-3}$
Average full stab. (iteration)	45	35

output of all 50 runs of ChaosDE, thus confirming the robustness of this approach. For the illustrative purposes, all graphical simulations outputs are depicted only for the variable x of the chaotic systems.

The values for desired p-1 UPOs (fixed points) of unperturbed chaotic systems based on the mathematical analysis of the systems are given in Table 3.

From the results presented in the Tables 4–9, it follows that the CF-simple is very convenient for evolutionary process, which means that repeated runs of EA are giving identical optimal results (i.e., very close to the possible global extreme). This is graphically confirmed in Figures 12, 14, and 16, which show all 50 simulations. All the runs are basically merged into one line.

On the other hand, the disadvantage of including of initial chaotic transient behavior of not stabilized system into the cost function value and subsequent resulting very tiny change of control method setting for extremely sensitive chaotic system is causing suppression of stabilization speed and numerical precision.

Results obtained in the cases utilizing the CF_{UNI} lend weight to the argument that the technique of pure searching for periodic orbits is advantageous for faster and more precise stabilization of chaotic system.

8. Conclusions

Based on obtained results, it may be claimed that the presented ChaosDE driven by selected discrete dissipative chaotic systems has given satisfactory results in the chaos control optimization issue.

The results show that the embedding of the chaotic dynamics in the form of chaotic pseudorandom number generator into the differential evolution algorithm may help to improve the performance and robustness of the DE. ChaosDE is able to obtain optimal solutions securing the very fast and precise stabilization for both convenient CF surface, in case of

TABLE 5: Characteristics of the best solution: joined case studies 1 and 2, Burgers map as CPRNG/controlled system.

Parameter	Case study 1	Case study 2
	CF _{simple}	CF _{UNI}
	Value	Value
K	1.22847	0.732498
F_{max}	0.9	0.48495
R	0.574997	0.811742
CF value	2.16199	$1.05 \cdot 10^{-6}$
Istab. Value	45	25
Average error per iteration	$5.86 \cdot 10^{-5}$	$1.21 \cdot 10^{-8}$

TABLE 6: Simple CF statistics: joined case studies 3 and 4, delayed logistic as CPRNG/controlled system.

Statistical data	Case study 3	Case study 4
	CF _{simple}	CF _{UNI}
	CF value	CF value
Min	0.199798	$2.3 \cdot 10^{-15}$
Max	0.199798	$2.7 \cdot 10^{-15}$
Average	0.199798	$2.44222 \cdot 10^{-15}$
Median	0.199798	$2.40551 \cdot 10^{-15}$
Std. dev.	$5.11 \cdot 10^{-16}$	$9.04629 \cdot 10^{-17}$
Average full stab. (iteration)	10	7.5

TABLE 7: Characteristics of the best solution: joined case studies 3 and 4, delayed logistic as CPRNG/controlled system.

Parameter	Case study 3	Case study 4
	CF _{simple}	CF _{UNI}
	Value	Value
K	1.29837	1.31355
F_{max}	0.394579	0.336294
R	0.01	0.010219
CF value	0.199798	$2.3 \cdot 10^{-15}$
Istab. Value	10	8
Average error per iteration	$2.22 \cdot 10^{-17}$	0

TABLE 8: Simple CF statistics: joined case studies 5 and 6, Lozi map as CPRNG/controlled system.

Statistical data	Case study 5	Case study 6
	CF _{simple}	CF _{UNI}
	CF value	CF value
Min	0.520639	$3.5331 \cdot 10^{-15}$
Max	0.520639	$4.0551 \cdot 10^{-15}$
Average	0.520639	$3.8063 \cdot 10^{-15}$
Median	0.520639	$3.6352 \cdot 10^{-16}$
Std. dev.	$2.41 \cdot 10^{-15}$	$1.19 \cdot 10^{-16}$
Average full stab. (iteration)	32	11

the CF-simple, as well as for the very chaotic and nonlinear CF surface in case of the CF-universal.

When comparing both the CF designs, the CF-simple is very convenient for evolutionary process (i.e., repeated

TABLE 9: Characteristics of the best solution: joined case studies 5 and 6, Lozi map as CPRNG/controlled system.

Parameter	Case study 5	Case study 6
	CF _{simple} Value	CF _{UNI} Value
K	-1.11259	-0.859989
F_{\max}	0.9	0.643099
R	0.289232	0.065669
CF value	0.520639	$3.5331 \cdot 10^{-15}$
Istab. Value	21	9
Average error per iteration	$7.21 \cdot 10^{-15}$	0

runs are giving identical optimal results), but it has many limitations.

The second universal CF design brings the possibility of using it problem free for any desired behavior of arbitrary chaotic systems but at the cost of the highly chaotic CF surface. Nevertheless, the embedding of the chaotic dynamics into the evolutionary algorithms helped to deal with such an issue.

The primary aim of this work was not to develop any new pseudorandom number generator, which should normally pass many statistical tests but to show that through embedding the hidden chaotic dynamics into the evolutionary process in the form of chaotic pseudorandom number generators may help to obtain better results and avoid problems connected with evolutionary computation such as premature convergence and stagnation in local extremes.

Future plans include testing of different chaotic systems, either manually or evolutionary tuning of parameters of chaotic maps, further complex comparisons with different heuristics and obtaining a large number of results to perform statistical tests.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by Grant Agency of the Czech Republic, GACR P103/13/08195S, partially supported by Grant of SGS nos. SP2014/159 and SP2014/170, VŠB-Technical University of Ostrava, Czech Republic and by the Development of Human Resources in Research and Development in Practice Project, Registry no. CZ.1.07/2.3.00/20.0072, funded by Operational Programme Education for Competitiveness, and cofinanced by ESF and state budget of the Czech Republic; it was further supported by the European Regional Development Fund under the Project CEBIA-Tech no. CZ.1.05/2.1.00/03.0089 and by the Internal Grant Agency of Tomas Bata University under the Project no. IGA/FAI/2014/010.

References

- [1] E. Ott, C. Grebogi, and J. A. Yorke, "Controlling chaos," *Physical Review Letters*, vol. 64, no. 11, pp. 1196–1199, 1990.
- [2] I. Zelinka, "Real-time deterministic chaos control by means of selected evolutionary techniques," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 2, pp. 283–297, 2009.
- [3] I. Zelinka, R. Senkerik, and E. Navratil, "Investigation on evolutionary optimization of chaos control," *Chaos, Solitons and Fractals*, vol. 40, no. 1, pp. 111–129, 2009.
- [4] R. Senkerik, I. Zelinka, D. Davendra, and Z. Oplatkova, "Utilization of SOMA and differential evolution for robust stabilization of chaotic Logistic equation," *Computers & Mathematics with Applications*, vol. 60, no. 4, pp. 1026–1037, 2010.
- [5] K. Pyragas, "Control of chaos via extended delay feedback," *Physics Letters A*, vol. 206, no. 5-6, pp. 323–330, 1995.
- [6] W. Just, "Principles of time delayed feedback control," in *Handbook of Chaos Control*, H. G. Schuster, Ed., Wiley- Vch, 1999.
- [7] R. Senkerik, Z. Oplatkova, I. Zelinka, and D. Davendra, "Synthesis of feedback controller for three selected chaotic systems by means of evolutionary techniques: analytic programming," *Mathematical and Computer Modelling*, vol. 57, no. 1-2, pp. 57–67, 2013.
- [8] Z. Kominkova Oplatkova, R. Senkerik, I. Zelinka, and M. Pluhacek, "Analytic programming in the task of evolutionary synthesis of a controller for high order oscillations stabilization of discrete chaotic systems," *Computers and Mathematics with Applications*, vol. 66, no. 2, pp. 177–189, 2013.
- [9] K. Pyragas, "Continuous control of chaos by self-controlling feedback," *Physics Letters A*, vol. 170, no. 6, pp. 421–428, 1992.
- [10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Wash, USA, December 1995.
- [11] M. Abedini, R. Vatankhah, and N. Assadian, "Stabilizing chaotic system on periodic orbits using multi-interval and modern optimal control strategies," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 10, pp. 3832–3842, 2012.
- [12] M. Sadeghpour, H. Salarieh, G. Vossoughi, and A. Alasty, "Multi-variable control of chaos using PSO-based minimum entropy control," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 6, pp. 2397–2404, 2011.
- [13] L. D. S. Coelho and R. B. Grebogi, "Chaotic synchronization using PID control combined with population based incremental learning algorithm," *Expert Systems with Applications*, vol. 37, no. 7, pp. 5347–5352, 2010.
- [14] M. J. Shirazi, R. Vatankhah, M. Boroushaki, H. Salarieh, and A. Alasty, "Application of particle swarm optimization in chaos synchronization in noisy environment in presence of unknown parameter uncertainty," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 2, pp. 742–753, 2012.
- [15] I. Zelinka and A. Raidl, "Evolutionary synchronization of chaotic systems," in *Evolutionary Algorithms and Chaotic Systems*, I. Zelinka, S. Celikovskiy, H. Richter, and G. Chen, Eds., vol. 267, pp. 385–407, Springer, Berlin, Germany, 2010.
- [16] H. Richter and K. J. Reinschke, "Optimization of local control of chaos by an evolutionary algorithm," *Physica D: Nonlinear Phenomena*, vol. 144, no. 3-4, pp. 309–334, 2000.
- [17] I. Zelinka, G. Chen, and S. Celikovskiy, "Chaos synthesis by means of evolutionary algorithms," *International Journal of*

- Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 18, no. 4, pp. 911–942, 2008.
- [18] I. Zelinka, M. Chadli, D. Davendra, R. Senkerik, and R. Jasek, “An investigation on evolutionary reconstruction of continuous chaotic systems,” *Mathematical and Computer Modelling*, vol. 57, no. 1–2, pp. 2–15, 2013.
- [19] I. Aydin, M. Karakose, and E. Akin, “Chaotic-based hybrid negative selection algorithm and its applications in fault and anomaly detection,” *Expert Systems with Applications*, vol. 37, no. 7, pp. 5285–5294, 2010.
- [20] R. Caponetto, L. Fortuna, S. Fazzino, and M. G. Xibilia, “Chaotic sequences to improve the performance of evolutionary algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 3, pp. 289–304, 2003.
- [21] D. Davendra, I. Zelinka, and R. Senkerik, “Chaos driven evolutionary algorithms for the task of PID control,” *Computers and Mathematics with Applications*, vol. 60, no. 4, pp. 1088–1104, 2010.
- [22] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution—a Practical Approach to Global Optimization*, Natural Computing Series, Springer, Berlin, Germany, 2005.
- [23] I. Zelinka, “SOMA—self-organizing migrating algorithm,” in *New Optimization Techniques in Engineering*, vol. 141 of *Studies in Fuzziness and Soft Computing*, pp. 167–217, Springer, Berlin, Germany, 2004.
- [24] L. D. S. Coelho and V. C. Mariani, “A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch,” *Chaos, Solitons & Fractals*, vol. 39, no. 2, pp. 510–518, 2009.
- [25] R. Senkerik, D. Davendra, I. Zelinka, M. Pluhacek, and Z. Oplatkova, “An investigation on the chaos driven differential evolution: an initial study,” in *Proceedings of the 5th International Conference on Bioinspired Optimization Methods and Their Applications (BIOMA '12)*, pp. 185–194, 2012.
- [26] D. Davendra, I. Zelinka, R. Senkerik, and M. Bialic-Davendra, “Chaos driven evolutionary algorithm for the traveling salesman problem,” in *Traveling Salesman Problem, Theory and Applications*, D. Davendra, Ed., InTech, 2010.
- [27] D. Davendra, M. Bialic-Davendra, and R. Senkerik, “Scheduling the lot-streaming flowshop scheduling problem with setup time with the chaos-induced enhanced differential evolution,” in *Proceedings of the IEEE Symposium on Differential Evolution (SDE '13)*, pp. 119–126, Singapore, April 2013.
- [28] M. Pluhacek, R. Senkerik, D. Davendra, Z. Kominkova Oplatkova, and I. Zelinka, “On the behavior and performance of chaos driven PSO algorithm with inertia weight,” *Computers and Mathematics with Applications*, vol. 66, no. 2, pp. 122–134, 2013.
- [29] M. Pluhacek, R. Senkerik, I. Zelinka, and D. Davendra, “Chaos PSO algorithm driven alternately by two different chaotic maps—an initial study,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '13)*, pp. 2444–2449, June 2013.
- [30] M. Pluhacek, R. Senkerik, and I. Zelinka, “Multiple choice strategy based PSO algorithm with chaotic decision making—a preliminary study,” in *International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*, Á. Herrero, B. Baroque, F. Klett et al., Eds., vol. 239 of *Advances in Intelligent Systems and Computing*, pp. 21–30, 2014.
- [31] L. D. S. Coelho and V. C. Mariani, “Firefly algorithm approach based on chaotic Tinkerbell map applied to multivariable PID controller tuning,” *Computers and Mathematics with Applications*, vol. 64, no. 8, pp. 2371–2382, 2012.
- [32] D. Davendra, “Evolutionary algorithms and the edge of chaos,” in *Evolutionary Algorithms and Chaotic Systems*, I. Zelinka, S. Celikovskiy, H. Richter, and G. Chen, Eds., vol. 267, pp. 145–161, Springer, Berlin, Germany, 2010.
- [33] R. Senkerik, M. Pluhacek, I. Zelinka, D. Davendra, Z. Oplatkova, and R. Jasek, “Evolutionary control of chaotic Lozi map by means of chaos driven differential evolution,” in *AETA 2013: Recent Advances in Electrical Engineering and Related Sciences*, I. Zelinka, V. H. Duy, and J. Cha, Eds., vol. 282 of *Lecture Notes in Electrical Engineering*, pp. 371–380, Springer, Berlin, Germany, 2014.
- [34] R. Senkerik, I. Zelinka, M. Pluhacek, and Z. Kominkova Oplatkova, “Evolutionary control of chaotic burgers map by means of chaos enhanced differential evolution,” *International Journal of Mathematics and Computers in Simulation*, vol. 8, pp. 39–45, 2014.
- [35] R. Senkerik, M. Pluhacek, I. Zelinka, D. Davendra, Z. Oplatkova, and R. Jasek, “Chaos driven differential evolution in the task of evolutionary control of delayed logistic chaotic system,” in *Proceedings of the 7th Global Conference on Power Control and Optimization (PCO '13)*, 2013.
- [36] M. F. Tasgetiren, P. N. Suganthan, and Q. K. Pan, “An ensemble of discrete differential evolution algorithms for solving the generalized traveling salesman problem,” *Applied Mathematics and Computation*, vol. 215, no. 9, pp. 3356–3368, 2010.
- [37] G. C. Onwubolu and D. Davendra, Eds., *Differential Evolution: A handbook for Permutation-based Combinatorial Optimization*, vol. 175 of *Studies in Computational Intelligence*, Springer, Berlin, Germany, 2009.
- [38] L. Wang, H. Qu, T. Chen, and F. P. Yan, “An effective hybrid self-adapting differential evolution algorithm for the joint replenishment and location-inventory problem in a three-level supply chain,” *The Scientific World Journal*, vol. 2013, Article ID 270249, 11 pages, 2013.
- [39] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, “Differential evolution using a neighborhood-based mutation operator,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 526–553, 2009.
- [40] T. J. Choi, C. W. Ahn, and J. An, “An adaptive cauchy differential evolution algorithm for global numerical optimization,” *The Scientific World Journal*, vol. 2013, Article ID 969734, 12 pages, 2013.
- [41] A. K. Qin, V. L. Huang, and P. N. Suganthan, “Differential evolution algorithm with strategy adaptation for global numerical optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
- [42] J. Zhang and A. C. Sanderson, “JADE: self-adaptive differential evolution with fast and reliable convergence performance,” in *Proceeding of the 2007 IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 2251–2258, Singapore, September 2007.
- [43] J. Zhang and A. C. Sanderson, “Self-adaptive multi-objective differential evolution with direction information provided by archived inferior solutions,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 2801–2810, Hong Kong, China, June 2008.
- [44] R. Senkerik, D. Davendra, I. Zelinka, and Z. Oplatkova, “Chaos driven differential evolution in the task of chaos control optimization,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '10)*, July 2010.
- [45] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous

- spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [46] K. V. Price, “An introduction to differential evolution,” in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds., McGraw-Hill, 1999.
- [47] J. C. Sprott, *Chaos and Time-Series Analysis*, Oxford University Press, New York, NY, USA, 2003.
- [48] M. A. Aziz-Alaoui, C. Robert, and C. Grebogi, “Dynamics of a Hénon-Lozi-type map,” *Chaos, Solitons & Fractals*, vol. 12, no. 12, pp. 2323–2341, 2001.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

