SIMULATION OF TIME-CONTINUOUS CHAOTIC UEDA OSCILLATOR AS THE GENERATOR OF RANDOM NUMBERS FOR HEURISTIC

[1]Roman Senkerik, [1]Michal Pluhacek, [2]Donald Davendra, [2]Ivan Zelinka, [1]Zuzana Kominkova Oplatkova

[1]Tomas Bata University in Zlin , Faculty of Applied Informatics
Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
{senkerik , oplatkova , pluhacek}@fai.utb.cz

[2]Department of Computer Science, Faculty of Electrical Engineering and Computer Science
VB-TUO, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
{donald.davendra , ivan.zelinka}@vsb.cz

KEYWORDS
Deterministic chaos; Chaotic oscillators; Heuristic; Differential Evolution; Chaotic Pseudo Random Number Generators

ABSTRACT

This paper investigates the utilization of the time-continuous chaotic system, which is UEDA oscillator, as the chaotic pseudo random number generator. (CPRNG). Repeated simulations were performed investigating the influence of the oscillator sampling time to the selected heuristic, which is differential evolution algorithm (DE). Initial experiments were performed on the selected test function in higher dimensions.

INTRODUCTION

Generally speaking, the term “chaos” can denote anything that cannot be predicted deterministically. In the case that the word “chaos” is combined with an attribute such as “deterministic”, then a specific type of chaotic phenomena is involved, having their specific laws, mathematical apparatus and a physical origin. The deterministic chaos is a phenomenon that - as its name suggests - is not based on the presence of a random or any stochastic effects. It is clear from the structure of the equations (see the section 4), that no mathematical term expressing randomness is present. The seeming randomness in deterministic chaos is related to the extreme sensitivity to the initial conditions [1].

Till now, the chaos has been observed in many of various systems (including evolutionary one). Systems exhibiting deterministic chaos include, for instance, weather, biological systems, many electronic circuits (Chua’s circuit), mechanical systems, such as double pendulum, magnetic pendulum, or so called billiard problem.

The idea of using chaotic systems instead of random processes (pseudo-number generators - PRNGs) has been presented in several research fields and in many applications with promising results [2], [3].

Another research joining deterministic chaos and pseudorandom number generator has been done for example in [4]. Possibility of generation of random or pseudorandom numbers by use of the ultra weak multidimensional coupling of p 1-dimensional dynamical systems is discussed there.

Another paper [5] deeply investigate logistic map as a possible pseudorandom number generator and is compared with contemporary pseudo-random number generators. A comparison of logistic map results is made with conventional methods of generating pseudorandom numbers. The approach used to determine the number, delay, and period of the orbits of the logistic map at varying degrees of precision (3 to 23 bits). Another paper [6] proposed an algorithm of generating pseudorandom number generator, which is called (couple map lattice based on discrete chaotic iteration) and combine the couple map lattice and chaotic iteration. Authors also tested this algorithm in NIST 800-22 statistical test suits and for future utilization in image encryption. In [7] authors exploit interesting properties of chaotic systems to design a random bit generator, called CCCBG, in which two chaotic systems are cross-coupled with each other. A new binary stream-cipher algorithm based on dual one-dimensional chaotic maps is proposed in [8] with statistic proprieties showing that the sequence is of high randomness. Similar studies are also done in [9], [10] and [11].

MOTIVATION

Recently the chaos has been used also to replace pseudo-number generators (PRGNs) in evolutionary algorithms (EAs). An evolutionary chaotic approach generally uses the chaotic system in the place of a pseudo random number generator [12]. This causes the heuristic to map unique regions, since the chaotic system iterates to new regions. The task is then to select a very good chaotic system (either discrete or time-continuous) as the pseudo random number generator.

The initial concept of embedding chaotic dynamics into the evolutionary algorithms is given in [13]. Later, the initial study [14] was focused on the simple embedding of chaotic systems in the form of chaos pseudo random number generator (CPRNG) for DE (Differential

Proceedings 29th European Conference on Modelling and Simulation ©ECMS Valeri M. Mladenov, Petia Georgieva, Grisha Spasov, Galidiya Petrova (Editors)
ISBN: 978-0-9932440-0-1 / ISBN: 978-0-9932440-1-8 (CD)

Evolution) and SOMA [15] in the task of optimal PID tuning

Several papers have been recently focused on the connection of heuristic and chaotic dynamics either in the form of hybridizing of DE with chaotic searching algorithm [16] or in the form of chaotic mutation factor and dynamically changing weighting and crossover factor in self-adaptive chaos differential evolution (SACDE) [17]. Also the PSO (Particle Swarm Optimization) algorithm with elements of chaos was introduced as CPSO [18] or CPSO combined with chaotic local search [19].

This idea was later extended with the successful experiments with chaos driven DE (ChaosDE) [20], [21] with both and complex simple test functions and in the task of chemical reactor geometry optimization [22].

The concept of Chaos DE has proved itself to be a powerful heuristic also in combinatorial problems domain [23].

At the same time the chaos embedded PSO with inertia weigh strategy was closely investigated [24], followed by the introduction of a PSO strategy driven alternately by two chaotic systems [25] and novel chaotic Multiple Choice PSO strategy (Chaos MC-PSO) [26].

The primary aim of this work is to try, test, analyze and compare the implementation of different natural chaotic dynamic as the CPRNG, thus to analyze and highlight the different influences to the system, which utilizes the selected CPRNG (including the evolutionary computational techniques).

CPRNG Concept

The general idea of CPRNG is to replace the default PRNG with the chaotic system. As the chaotic system is a set of equations with a static start position, we created a random start position of the system, in order to have different start position for different experiments. This random position is initialized with the default PRNG, as a one-off randomizer. Once the start position of the chaotic system has been obtained, the system generates the next sequence using its current position.

Generally there exist many other approaches as to how to deal with the negative numbers as well as with the scaling of the wide range of the numbers given by the chaotic systems into the typical range 0 – 1:

- Finding of the maximum value of the pre-generated long discrete sequence and dividing of all the values in the sequence with such a maxval number.
- Shifting of all values to the positive numbers (avoiding of ABS command) and scaling.

UEDA Oscillator

UEDA oscillator is the simple example of driven pendulums, which represent some of the most significant examples of chaos and regularity.

The UEDA system can be simply considered as a special case of intensively studied Duffing oscillator that has both a linear and cubic restoring force. Ueda oscillator represents the both biologically and physically important dynamical model exhibiting chaotic motion. It can be used to explore much physical behavior in biological systems. [27]

The UEDA chaotic system equations are given in (1). The parameters are: a = 1.0 b = 0.05, c = 7.5 and ω = 1.0 as suggested in [28]. The x, y parametric plot of the chaotic system is depicted in Fig. 1.

$$\begin{aligned} \frac{dx}{dt} &= y \\ \frac{dy}{dt} &= -ax^3 - by + c\sin\omega t \end{aligned} \quad . \qquad (1)$$

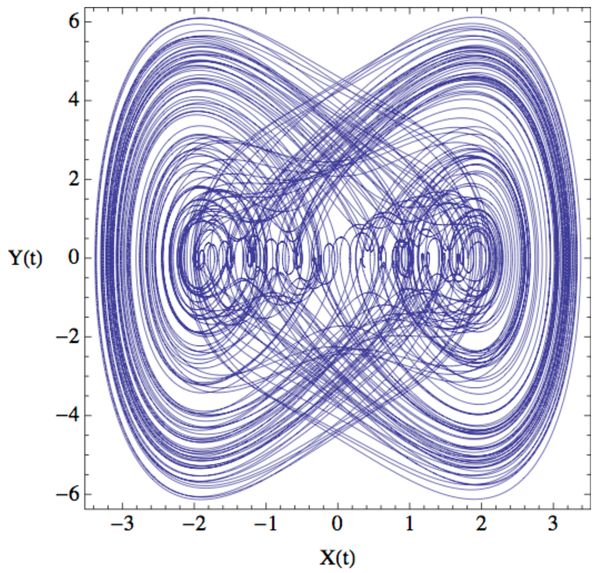

Fig. 1 x, y parametric plot of the UEDA oscillator

DIFFERENTIAL EVOLUTION

This section briefly describes the used heuristic within all experiments.

DE is a population-based optimization method that works on real-number-coded individuals [29]. 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 $\vec{x}_{i,G+1}$. DE is quite robust, fast, and effective, with global optimization ability. It does not require the objective function to be differentiable, and it works well even with noisy and time-dependent objective functions. Please refer to [29], [30] for the detailed description of the used DERand1Bin strategy (2) (both for Chaos DE and Canonical DE) as well as for the complete description of all other strategies.

$$u_{i,G+1} = x_{r1,G} + F \cdot \left(x_{r2,G} - x_{r3,G}\right) \quad (2)$$

EXPERIMENT DESIGN

For the purpose of evolutionary algorithm performance comparison within this initial research, the multimodal Schwefel's test function (3) was selected.

$$f(x) = \sum_{i=1}^{D} -x_i \sin\left(\sqrt{|x_i|}\right) \quad (3)$$

Function minimum:
Position for E_n:
$(x_1, x_2 \ldots x_n) = (420.969, 420.969, \ldots, 420.969)$
Value for E_n: $y = -418.983 \cdot Dimension$

The novelty of this research represents investigating the influence of the oscillator sampling time to the selected heuristic, which is DE.
In this paper, the canonical DE strategy DERand1Bin and the Chaos DERand1Bin (ChaosDE) strategy driven by discretized UEDA oscillators (ChaosDE) were used.
The parameter settings for both canonical DE and ChaosDE were obtained analytically based on numerous experiments and simulations (see Table 1). Experiments were performed in the environment of *Wolfram Mathematica*; canonical DE therefore has used the built-in *Wolfram Mathematica* pseudo random number generator *Wolfram Cellular Automata* representing traditional pseudorandom number generator in comparisons. All experiments used different initialization, i.e. different initial population was generated within the each run of Canonical or Chaos driven DE. The maximum number of generations was fixed at 3000 generations. This allowed the possibility to analyze the progress of DE within a limited number of generations and cost function evaluations.

Table 1: DE settings

DE Parameter	Value
Dimension	20
PopSize	50
F	0.8
CR	0.8
Generations	1000
Max. CF Evaluations (CFE)	50000

EXPERIMENT RESULTS

The statistical results of the experiments are shown in Table 2, which represent the simple statistics for cost function (CF) values, e.g. average, median, maximum values, standard deviations and minimum values representing the best individual solution for all 50 repeated runs of canonical DE and several versions of ChaosDE (with several sampling times of UEDA oscillator).
Table 3 compares the progress of several versions of ChaosDE, and Canonical DE. This table contains the average CF values for the generation No. 250, 500, 750 and 1000 from all 50 runs. The bold values within the both Tables 2 and 3 depict the best obtained results.
Following versions of Multi-ChaosDE were studied:
Figures 2 a) – 2f) show the influence of sampling rate to the distribution of numbers given by particular CPRNG.
Finally the graphical comparison of the time evolution of average CF values for all 50 runs of ChaosDE and canonical DERand1Bin strategy is depicted in Fig. 3.

Table 2: Simple results statistics for the Schwefel's function – 20*D*

DE Version	Avg CF	Median CF	Max CF	Min CF	StdDev
Canonical DE	-4583.05	-4531.29	-4265.43	-5238.75	**268.4226**
ChaosDE - Sampling 0.1s	**-6250.01**	**-6346.65**	**-4947.92**	**-7535.51**	952.9218
ChaosDE - Sampling 0.5s	-4246.65	-4004.84	-3668.64	-5504.73	562.4262
ChaosDE - Sampling 1.0s	-4508.7	-4161.67	-3912.97	-5840.88	699.4066
ChaosDE - Sampling 2.0s	-4224.91	-3967.19	-3765.41	-5159.19	558.5293

Table 3: Comparison of progress towards the minimum for the Schwefel's function

DE Version	Generation No.: 250	Generation No.: 500	Generation No.: 750	Generation No.: 1000
Canonical DE	**-3955.27**	-4287.11	-4422.72	-4583.05
ChaosDE - Sampling 0.1s	-3781.28	**-4618.32**	**-5431.47**	**-6250.01**
ChaosDE - Sampling 0.5s	-3536.42	-3740.76	-3958.08	-4246.65
ChaosDE - Sampling 1.0s	-3443.07	-3758.27	-4081.61	-4508.7
ChaosDE - Sampling 2.0s	-3505.97	-3819.35	-4063.07	-4224.91

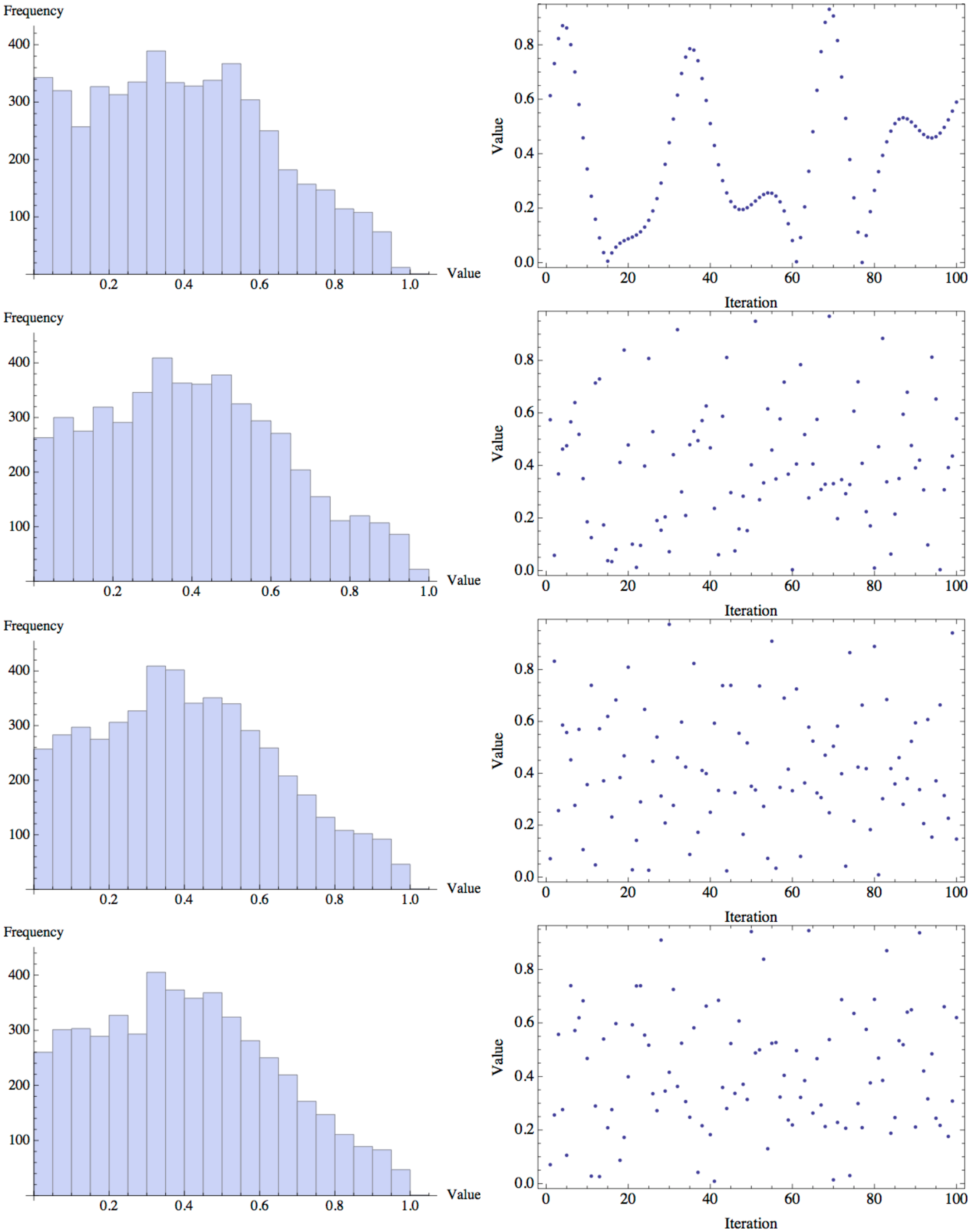

Figure 2: Comparison of the influence of sampling rate to the distribution of numbers given by UEDA CPRNG; Left: Histogram of the distribution of real numbers transferred into the range <0 - 1>; Right: Example of the chaotic dynamics: range <0 - 1> generated by means of UEDA oscillator sampled with the particular sampling rate – variable x ; Sampling rates from up to down: 0.1s, 0.5s, 1.0s, 2.0s.

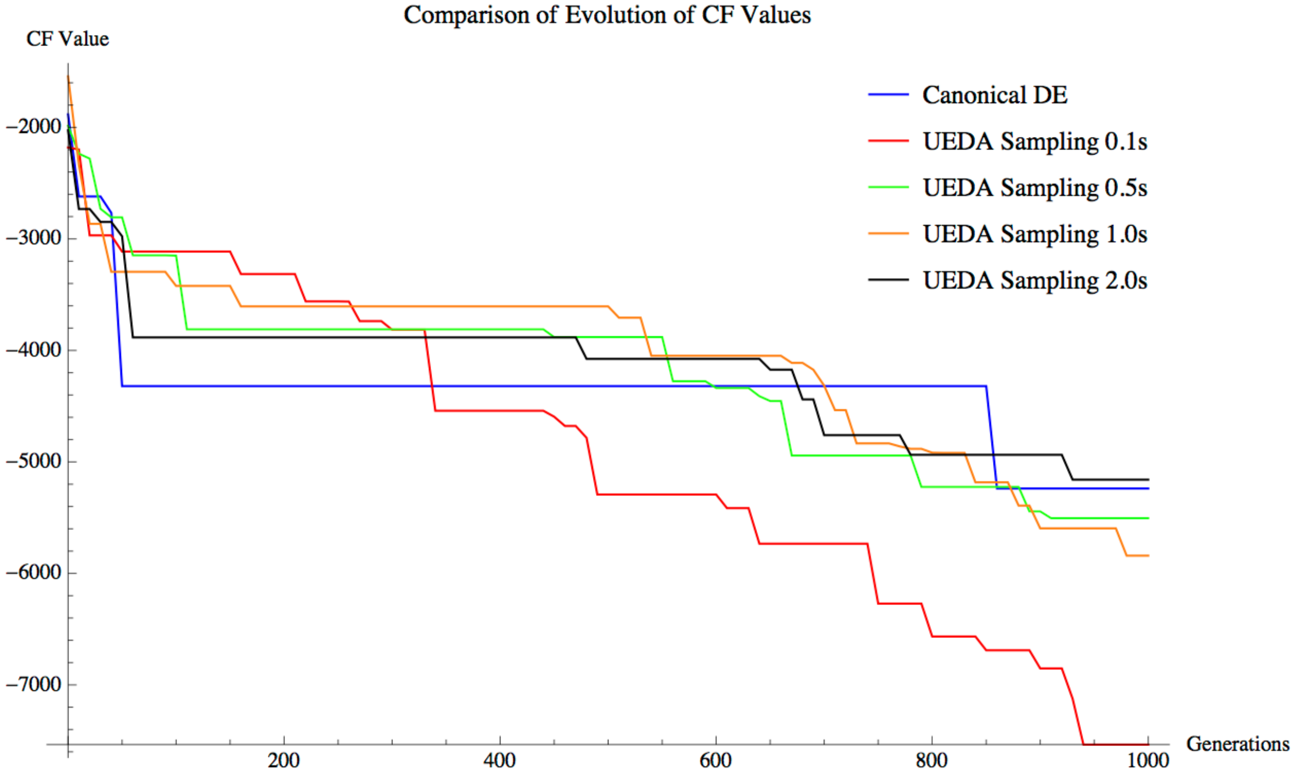

Figure 3: Comparison of the time evolution of avg. CF values for the all 50 runs of Canonical DE and several versions of ChaosDE driven by UEDA oscillators with different settings of sampling time; Schwefel's function, *Dimension* = 20.

CONCLUSION

The novelty of this research represents investigating the influence of the chaotic oscillator sampling time used as the CPRNG to the selected heuristic, which is DE.

In this paper, the concept of chaos driven DERand1Bin strategy was more experimentally analyzed and compared with the canonical DERand1Bin strategy on the selected benchmark function with four different settings of sampling time for the UEDA chaotic oscillator.

Obtained numerical results given in Tables 2 and 3 and graphical comparisons in Figures 2 and 3 support the claim that chaos driven heuristic is very sensitive to the hidden chaotic dynamics driving the CPRNG. Such a chaotic dynamics can be significantly changed by the selection of sampling time in the case of the time-continuous systems.

Another important phenomenon was discovered – Only sampling rate of 0.5s keeps the information about the chaotic dynamics (as in Fig 2b) and by using such chaotic dynamics driving the heuristic, its performance is significantly better. Other settings of sampling rates have given comparable (or even worse) performance with canonical version of heuristic.

Future plans are including the testing of combination of different time-continuous chaotic systems as well as the adaptive switching and obtaining a large number of results to perform statistical tests.

Furthermore chaotic systems have additional parameters, which can by tuned. This issue opens up the possibility of examining the impact of these parameters to generation of random numbers, and thus influence on the results obtained by means of ChaosDE.

ACKNOWLEDGEMENT

This work was supported by Grant Agency of the Czech Republic - GACR P103/15/06700S, further by financial support of research project NPU I No. MSMT-7778/2014 by the Ministry of Education of the Czech Republic and also by the European Regional Development Fund under the Project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089, partially supported by Grant of SGS No. SP2015/142 and SP2015/141 of VSB - Technical University of Ostrava, Czech Republic and by Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2015/057.

REFERENCES

[1] S. Celikovsky and I. Zelinka, "Chaos Theory for Evolutionary Algorithms Researchers," in *Evolutionary Algorithms and Chaotic Systems*. vol. 267, I. Zelinka, S. Celikovsky, H. Richter, and G. Chen, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 89-143.

[2] J. S. Lee and K. S. Chang, "Applications of chaos and fractals in process systems engineering," *Journal of Process Control,* vol. 6, pp. 71-87, 1996.

[3] J. Wu, J. Lu, and J. Wang, "Application of chaos and fractal models to water quality time series prediction," *Environmental Modelling & Software,* vol. 24, pp. 632-636, 2009.

[4] R. Lozi, "Emergence of Randomness from Chaos," *International Journal of Bifurcation and Chaos,* vol. 22, p. 1250021, 2012.

[5] K. J. Persohn and R. J. Povinelli, "Analyzing logistic map pseudorandom number generators for periodicity induced by finite precision floating-point representation," *Chaos, Solitons & Fractals,* vol. 45, pp. 238-245, 2012.

[6] X.-y. Wang and X. Qin, "A new pseudo-random number generator based on CML and chaotic iteration," *Nonlinear Dynamics,* vol. 70, pp. 1589-1592, 2012/10/01 2012.

[7] K. P. Narendra, P. Vinod, and K. S. Krishan, "A Random Bit Generator Using Chaotic Maps," *International Journal of Network Security,* vol. 10, pp. 32 - 38, 2010.

[8] L. Yang and X.-Y. Wang, "Design of Pseudo-random Bit Generator Based on Chaotic Maps," *International Journal of Modern Physics B,* vol. 26, p. 1250208, 2012.

[9] M. Bucolo, R. Caponetto, L. Fortuna, M. Frasca, and A. Rizzo, "Does chaos work better than noise?," *Circuits and Systems Magazine, IEEE,* vol. 2, pp. 4-19, 2002.

[10] H. Hu, L. Liu, and N. Ding, "Pseudorandom sequence generator based on the Chen chaotic system," *Computer Physics Communications,* vol. 184, pp. 765-768, 2013.

[11] A. Pluchino, A. Rapisarda, and C. Tsallis, "Noise, synchrony, and correlations at the edge of chaos," *Physical Review E,* vol. 87, p. 022910, 2013.

[12] 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, pp. 5285-5294, 2010.

[13] 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, pp. 289-304, 2003.

[14] D. Davendra, I. Zelinka, and R. Senkerik, "Chaos driven evolutionary algorithms for the task of PID control," *Computers & Mathematics with Applications,* vol. 60, pp. 1088-1104, 2010.

[15] I. Zelinka, "SOMA — Self-Organizing Migrating Algorithm," in *New Optimization Techniques in Engineering*. vol. 141, ed: Springer Berlin Heidelberg, 2004, pp. 167-217.

[16] W. Liang, L. Zhang, and M. Wang, "The chaos differential evolution optimization algorithm and its application to support vector regression machine," *Journal of Software,* vol. 6, pp. 1297- 1304, 2011.

[17] G. Zhenyu, C. Bo, Y. Min, and C. Binggang, "Self-Adaptive Chaos Differential Evolution," in *Advances in Natural Computation*. vol. 4221, L. Jiao, L. Wang, X.-b. Gao, J. Liu, and F. Wu, Eds., ed: Springer Berlin Heidelberg, 2006, pp. 972-975.

[18] 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, pp. 510-518, 2009.

[19] W.-C. Hong, "Chaotic particle swarm optimization algorithm in a support vector regression electric load forecasting model," *Energy Conversion and Management,* vol. 50, pp. 105-117, 2009.

[20] R. Senkerik, M. Pluhacek, I. Zelinka, Z. Oplatkova, R. Vala, and R. Jasek, "Performance of Chaos Driven Differential Evolution on Shifted Benchmark Functions Set," in *Proc. International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*. vol. 239, Á. Herrero, B. Baruque, F. Klett, A. Abraham, V. Snášel, A. C. P. L. F. Carvalho*, et al.*, Eds., ed: Springer International Publishing, 2014, pp. 41-50.

[21] R. Senkerik, D. Davendra, I. Zelinka, M. Pluhacek, and Z. Kominkova Oplatkova, "On the Differential Evolution Drivan by Selected Discrete Chaotic Systems: Extended Study," in *Proc. 19th International Conference on Soft Computing, MENDEL 2013*, 2013, pp. 137-144.

[22] R. Senkerik, M. Pluhacek, Z. K. Oplatkova, D. Davendra, and I. Zelinka, "Investigation on the Differential Evolution driven by selected six chaotic systems in the task of reactor geometry optimization," in *Proc. 2013 IEEE Congress on Evolutionary Computation (CEC)*, 2013, pp. 3087-3094.

[23] 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 *Proc. 2013 IEEE Symposium on Differential Evolution (SDE)*, 2013, pp. 119-126.

[24] 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 & Mathematics with Applications,* vol. 66, pp. 122-134, 2013.

[25] M. Pluhacek, R. Senkerik, I. Zelinka, and D. Davendra, "Chaos PSO algorithm driven alternately by two different chaotic maps - An initial study," in *Proc. 2013 IEEE Congress on Evolutionary Computation (CEC)*, 2013, pp. 2444-2449.

[26] M. Pluhacek, R. Senkerik, and I. Zelinka, "Multiple Choice Strategy Based PSO Algorithm with Chaotic Decision Making – A Preliminary Study," in *Proc. International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*. vol. 239, Á. Herrero, B. Baruque, F. Klett, A. Abraham, V. Snášel, A. C. P. L. F. Carvalho*, et al.*, Eds., ed: Springer International Publishing, 2014, pp. 21-30.

[27] L. Bharti and M. Yuasa. Energy Variability and Chaos in Ueda Oscillator.
Available: http://www.rist.kindai.ac.jp/no.23/yuasa-EVCUO.pdf

[28] J. C. Sprott, *Chaos and Time-Series Analysis*: Oxford University Press, 2003.

[29] K. Price, "An Introduction to Differential Evolution", *In: New Ideas in Optimization*, (D. Corne, M. Dorigo and F. Glover, Eds.), pp. 79–108, McGraw-Hill, London, UK, ISBN 007-709506-5, 1999.

[30] K. Price, R. Storn and J. Lampinen, "*Differential Evolution - A Practical Approach to Global Optimization*", 2005, Springer, ISBN: 3-540-20950-6.