On the Randomization of Indices Selection for Differential Evolution

Roman Senkerik(✉), Michal Pluhacek, Adam Viktorin, and Tomas Kadavy

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

Abstract. This research deals with the hybridization of two softcomputing fields, which are the chaos theory and evolutionary algorithms. This paper investigates the utilization of the two-dimensional discrete chaotic systems, which are Burgers and Lozi maps, as the chaotic pseudo random number generators (CPRNGs) embedded into the selected heuristics, which is differential evolution algorithm (DE). Through the utilization of either chaotic systems or identical identified pseudo random number distribution, it is possible to fully keep or remove the hidden complex chaotic dynamics from the generated pseudo random data series. Experiments are focused on the extended investigation, whether the different randomization types with different pseudo random numbers distribution or hidden complex chaotic dynamics providing the unique sequencing are more beneficial to the heuristic performance. This research utilizes set of 4 selected benchmark functions, and totally four different randomizations; further results are compared against canonical DE.

Keywords: Differential evolution · Complex dynamics · Deterministic chaos · Randomization · Burgers map · Lozi map

1 Introduction

This research deals with the mutual intersection of the two softcomputing fields, which are the complex dynamics given by the chaotic systems driving the selection of indices in Differential Evolution (DE) algorithm and evolutionary computation techniques (ECT's). Currently the DE [1] is known as powerful heuristic for many difficult and complex optimization problems.

A number of DE variants have been recently developed with the emphasis on adaptivity/selfadaptivity [2], ensemble approach [3] or other modern approaches [4, 5]. The importance of randomization within heuristics as a compensation of limited amount of search moves is stated in the survey paper [6]. This idea has been carried out in subsequent studies describing different techniques to modify the randomization process [7, 8] and especially in [9], where the sampling of the points is tested from modified distribution. The importance and influence of randomization operations was also deeply experimentally tested in simple control parameter adjustment jDE strategy [10]. Together with this persistent development in such mainstream research topics, the

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_53

basic concept of chaos driven DE have been introduced. Recent research in chaotic approach for heuristics generally uses various chaotic maps in the place of pseudo random number generators (PRNG). The focus of this research is the direct embedding of chaotic dynamics in the form of chaos pseudo random number generator (CPRNG) for heuristic. The initial concept of embedding chaotic dynamics into the evolutionary/swarm algorithms is given in [11]. Later, the initial study [12] was focused on the simple embedding of chaotic systems for DE and Self Organizing Migration Algorithm (SOMA) [13]. Also the PSO (Particle Swarm Optimization) algorithm with elements of chaos was introduced as CPSO [14] followed by the introduction of chaos embedded PSO with inertia weigh strategy [15], further PSO strategy driven alternately by two chaotic systems [16] and finally PSO with ensemble of chaotic systems [17]. Recently the chaos driven heuristic concept has been utilized in ABC algorithm [18] and applications with DE [19].

The organization of this paper is following: Firstly, the motivation and novality for this research is proposed. The next sections are focused on the description of the concept of chaos driven DE, identification of chaotic series distribution and the experiment background. Results and conclusion follow afterwards.

2 Motivation

This research is an extension and continuation of the previous successful initial experiment with the single/multi-chaos driven DE (ChaosDE), where the positive influence of hidden complex dynamics for the heuristic performance has been experimentally shown. This research is also a follow up to previous initial experiments with time continuous chaotic systems and different sampling rates used [20].

Nevertheless, the questions remain, as to why it works, why it may be beneficial to use the correlated chaotic time series for generating pseudo random numbers driving the selection, mutation, crossover or other processes in particular heuristics.

The novality of the research is given by the experiment investigationg whether the chaos embedded heuristics concept belongs to the group of either "utilization of different PRNG with different distribution" or the unique chaos dynamics providing unique sequencing of pseudo random numbers is the key of performance improvements. The last point was also inspired by recent advances in connection of complexity and heuristic [21] together with the research focused on selection of indices in DE [22] where the indices (solutions) for mutation process were not selected randomly, but based on the complex behavior and neighborhood mechanisms.

To confirm or disprove the aforementioned hypothesis, a simple experiment was performed and presented here. Through the utilization of either chaotic systems or identical identified pseudo random number distribution, it is possible to fully keep or remove the hidden complex chaotic dynamics from the generated pseudo random data series for obtaining the pseudo random numbers for indices selection inside DE.

3 Differential Evolution

DE is a population-based optimization method that works on real-number-coded individuals [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. There are essentially five inputs to the heuristic. *Dim* is the size of the problem, *Gmax* is the maximum number of generations, *NP* is the total number of solutions, *F* is the scaling factor of the solution and *CR* is the factor for crossover. *F* and *CR* together make the internal tuning parameters for the heuristic. Due to a limited space and the aims of this paper, the detailed description of well known canonical strategy of differential evolution algorithm basic principles is insignificant and hence omitted. Please refer to [1, 23] for the detailed description of the used *DE/Rand/1/Bin* strategy (both for ChaosDE and Canonical DE) as well as for the complete description of all other strategies.

4 Chaotic Systems and Identification of CPRNGs Distributions

Following two well known and frequently utilized discrete dissipative chaotic maps were used as the CPRNGS for DE: Burgers (1), and Lozi map (2).

The Burgers mapping is a discretization of a pair of coupled differential equations to illustrate the relevance of the concept of bifurcation to the study of hydrodynamics flows. The Lozi map is a simple discrete two-dimensional chaotic map. With the typical settings as in Table 1, systems exhibits typical chaotic behavior [24].

Table 1. Definition of chaotic systems used as CPRNGs

Chaotic maps equations	Parameter settings
$X_{n+1} = aX_n - Y_n^2$, $Y_{n+1} = bY_n + X_nY_n$ (1)	$a = 0.75$ and $b = 1.75$
$X_{n+1} = 1 - a\|X_n\| + bY_n$, $Y_{n+1} = X_n$ (2)	$a = 1.7$ and $b = 0.5$

For the comparisons of DE performance with indices selection driven either by CPRNG or identical PRNG distribution without chaotic dynamics, it was necessary to perform the CPRNGs distributions identification with 10000 samples and statistical distribution fit tests. *Statistica* and *Wolfram Mathematica* software were used for this task with following results (See also Figs. 1 and 2):

- Burgers map based CPRNG was identified as Beta distribution (α, β) with $\alpha = 0.63$ and $\beta = 3.54$.
- Lozi map based CPRNG was identified as Beta distribution (α, β) with $\alpha = 1.05$ and $\beta = 1.57$.

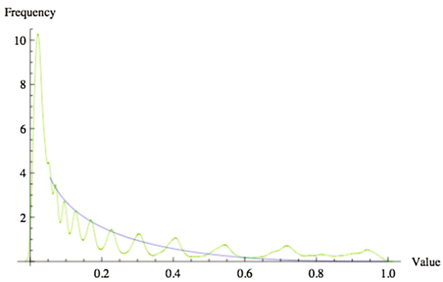

Fig. 1. Identification (blue line) of Burgers map based CPRNG (green line – smooth histogram)

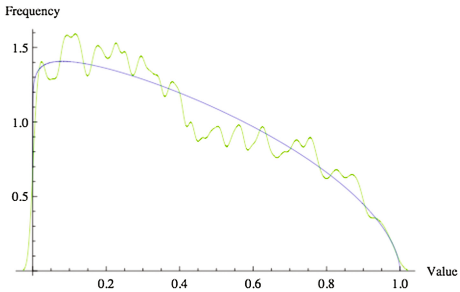

Fig. 2. Identification (blue line) of Lozi map based CPRNG (green line – smooth histogram)

5 The Concept of ChaosDE with Discrete Chaotic System as Driving CPRNG

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. Thus we are utilizing the typical feature of chaotic systems, which is extreme sensitivity to the initial conditions, popularly known as "butterfly effect". 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. Used approach is based on the following definition (3):

$$rndreal = \mathrm{mod}(\mathrm{abs}(rndChaos), 1.0) \tag{3}$$

6 Experiment Design

For the purpose of ChaosDE performance comparison within this research, the Schwefel's test function (4), shifted Grienwang function (5), and shifted Ackley's original function in the form (6) and shifted Rastrigin's function (7) were selected.

$$f(x) = \sum_{i=1}^{\dim} -x_i \sin\left(\sqrt{|x_i|}\right) \tag{4}$$

Function minimum:
Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = (420.969, 420.969, \ldots, 420.969)$
Value for $\mathrm{E_n}$: $y = -418.983 \cdot dim$; Function interval: $< -500, 500 >$.

$$f(x) = \sum_{i=1}^{\dim} \frac{(x_i - s_i)^2}{4000} - \prod_{i=1}^{\dim} \cos\left(\frac{x_i - s_i}{\sqrt{i}}\right) + 1 \tag{5}$$

Function minimum: Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$; Value for $\mathrm{E_n}$: $y = 0$
Function interval: $< -50, 50 >$.

$$f(x) = -20\exp\left(-0.02\sqrt{\frac{1}{D}\sum_{i=1}^{\dim}(x_i - s_i)^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{\dim}\cos 2\pi(x_i - s_i)\right) + 20 + \exp(1) \tag{6}$$

Function minimum: Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$; Value for $\mathrm{E_n}$: $y = 0$
Function interval: $< -30, 30 >$.

$$f(x) = 10\dim\sum_{i=1}^{\dim}(x_i - s_i)^2 - 10\cos(2\pi x_i - s_i) \tag{7}$$

Function minimum: Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$, Value for $\mathrm{E_n}$: $y = -90000(\dim 30)$
Function interval: $< -5.12, 5.12 >$.

Where s_i is a random number from the 90% range of function interval; $\boldsymbol{s}$ vector is randomly generated before each run of the optimization process.

The parameter settings for both canonical DE and ChaosDE were obtained based on numerous experiments and simulations (see Table 2). It was experimentally determined, that ChaosDE requires lower values of *Cr* parameter [25] for any type of used CPRNG. Canonical DE is using the recommended settings [1]. The maximum number of generations was fixed at 1500 generations. This allowed the possibility to analyze the progress of DE within a limited number of generations and cost function evaluations. 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

Table 2. Parameter set up for ChaosDE and Canonical DE

DE parameter	Value
Popsize	75
F (for ChaosDE)	0.4
CR (for ChaosDE)	0.4
F (for Canonical DE)	0.5
CR (for Canonical DE)	0.9
Dim	30
Max. Generations	1500

number generator in comparisons. All experiments used different initialization, i.e. different initial population was generated within the each run of Canonical or ChaosDE.

7 Results

Statistical results for the Cost Function (CF) values are shown in comprehensive Tables 3–6 for all 50 repeated runs of DE/ChaosDE, four different benchmark functions and five randomization schemes.

Table 3. Simple results statistics for the Canonical DE and ChaosDE; Schwefel's function

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	−5493.26	−5339.34	−4944.96	−6628.4	**440.8144**	–
Burgers dist	−10375.9	−10360	**−9245.86**	−11722.9	518.8032	0.010864
Burgers map	**−10793.5**	**−11413.9**	−6787.51	−12328.1	1387.362	
Lozi dist	−8709.74	−8530.74	−7814.36	−11042.5	661.7437	0.000112
Lozi map	−9932.46	−9922.25	−8200.06	**−12530.9**	1043.777	

The bold values within the all Tables 3–6 depict the best obtained results, italic values are considered to be similar. Statistical comparisons are based on the *Wilcoxon signed-rank test* with significance level 0.05; and performed for the pairs of ChaosDE with CPRNG and identified similar PRNG distribution. The graphical comparisons of the time evolution of average CF values for all 50 runs of five versions of DE/ChaosDE with different randomizations and two selected benchmark functions are depicted in Figs. 3 and 4. The notation in Tables and Figures is following: *Burgers/Lozi Map* represents the chaotic based CPRNG, whereas *Burgers/Lozi Dist* represents identified distribution PRNG.

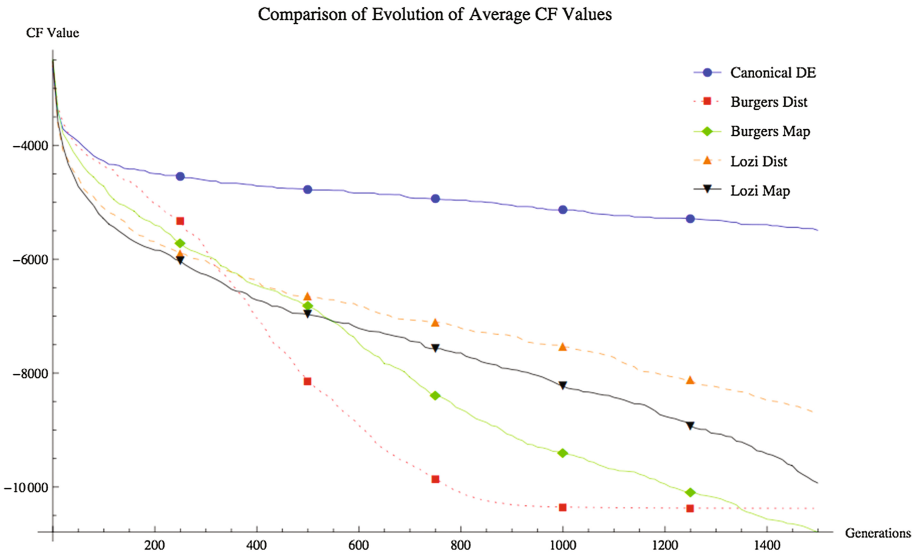

Fig. 3. Comparison of the time evolution of avg. CF values for the all 50 runs of Canonical DE, and four versions of ChaosDE with different randomization. Schwefel's function.

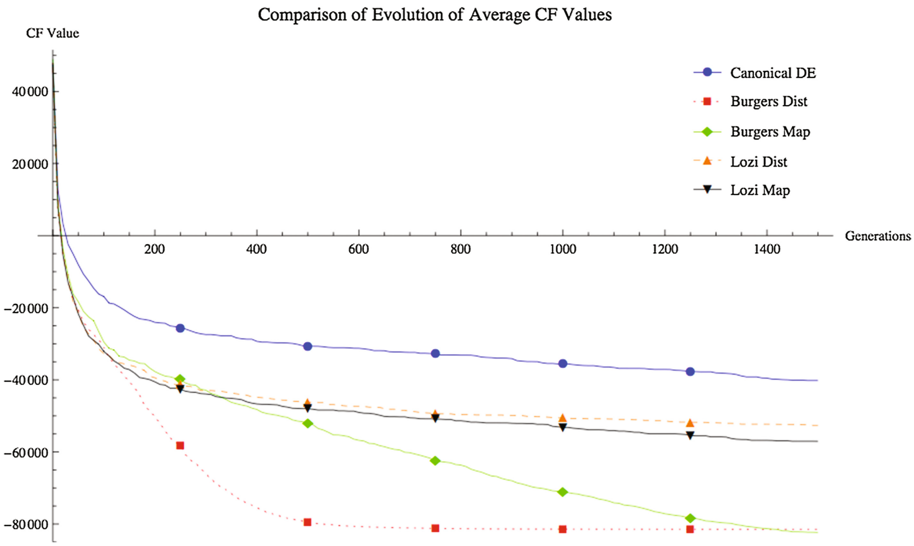

Fig. 4. Comparison of the time evolution of avg. CF values for the all 50 runs of Canonical DE, and four versions of ChaosDE with different randomizations. Shifted Rastrigin's function.

Table 4. Simple results statistics for the Canonical DE and ChaosDE; shifted Rastrigin's func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	−40188.49	−39264.95	−33629.64	−49994.34	3983.79	–
Burgers dist	−81465.42	−82256.88	**−74959.67**	−85542.21	2521.65	0.02812
Burgers map	**−82339.03**	**−83552.39**	−63945.49	**−87977.11**	5262.81	
Lozi dist	−52641.81	−52722.69	−49731.38	−57271.93	**1851.70**	$8.0695.10^{-6}$
Lozi map	−57054.66	−56667.65	−52235.56	−62969.12	2927.69	

Table 5. Simple results statistics for the Canonical DE and ChaosDE; shifted Ackley's func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	3.38E-09	2.68E-09	7.55E-09	9.48E-10	1.74E-09	–
Burgers dist	4.333288	4.554203	7.464985	2.013873	1.328967	$1.8253.10^{-6}$
Burgers map	1.43E-06	1.25E-12	1.775137	1.47E-14	0.391209	
Lozi dist	1.64E-14	***1.47E-14***	3.6E-14	***7.55E-15***	5.26E-15	0.5231
Lozi map	**1.54E-14**	***1.47E-14***	**2.89E-14**	***7.55E-15***	**4.11E-15**	

Table 6. Simple results statistics for the Canonical DE and ChaosDE; shifted Grienwang func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	***0***	***0***	***0***	***0***	***0***	–
Burgers dist	0.525982	0.514041	0.998373	0.098319	0.26012	$1.8626.10^{-6}$
Burgers map	6.89E-07	1.47E-09	0.15187	0	0.00323	
Lozi dist	***0***	***0***	***0***	***0***	***0***	1
Lozi map	***0***	***0***	***0***	***0***	***0***	

8 Conclusions

The primary aim of this work is to experimentally investigate the utilization of the various discrete chaotic systems, as the chaotic pseudo random number generator embedded into DE. Experiments are focused on the extended investigation, whether the different randomization and pseudo random numbers distribution given by particular PRNG or hidden complex chaotic dynamics providing the unique sequencing are more beneficial to the heuristic performance. The findings can be summarized as:

- Obtained graphical comparisons and data in Tables 3–6 and Figs. 3 and 4 support the claim that chaos driven heuristic is more sensitive to the hidden chaotic dynamics driving the selection, mutation, crossover or other processes through CPRNG. The influence of different PRNG randomization (distribution) type is strengthened by the presence of chaotic dynamics and sequencing in the pseudo random series given by the dynamics of discretized chaotic attractor/flow.
- It is clear that (selection of) the best CPRNGs are problem-dependent. By keeping the information about the chaotic dynamics driving the selection/mutation processes

inside heuristic, its performance is significantly different: either better or worse against other compared versions.

- In the first two cases (Schwefel and shifted Rastrigin function – Tables 3 and 4), the performance of ChaosDE was significantly better in comparison with canonical DE. Furthermore the effect of different PRNG distribution became even stronger with the chaotic dynamics kept inside CPRNG sequences. Lozi map based CPRNG has given stable better performance than similar identified PRNG. Both Lozi map based PRNG/CPRNG have been outperformed by the utilization of Burgers map based PRNG/CPRNG. An interesting phenomenon has been revealed. The Burgers map based not-chaotic PRNG drives DE to the strong and fast progress towards function extreme (local) followed by premature population stagnation phase. Whereas Burgers map CPRNG with chaotic dynamics secured the continuous development of population towards global best solution without stagnation.
- The third and the fourth case study (Tables 5 and 6) have given absolutely reversed character of results. Performance of Lozi based CPRNG/PRNG is comparable even with canonical DE (slightly better results for Lozi map CPRNG and Ackley function), whereas the Burgers map based randomization has given worse results. As aforementioned in the previous point, the premature stagnation for PRNG has occurred also here (more considerable), whereas the Burgers map based CPRNG with chaotic dynamics has driven the DE more or less towards the function extreme.
- Since the aim was to investigate the randomization/sequencing of indices selection inside DE, only the simplest canonical *DE/Rand/1/Bin* strategy has been utilized in this research. The parameter adjustment/strategy adaptation or ensembles techniques in jDE, EPSDE, SHADE may significantly interact with the dynamics of sequencing (selection) of indices driven by particular not-uniform PRNG/CPRNG.
- Sequencing of pseudo random numbers and chaotic dynamics hidden inside pseudo random series can be significantly changed by the selection of chaotic systems, thus to avoid the CF landscape dependency. The simplest way for changing the influence to the heuristic during the run is to swap currently used chaotic system for different one, or to change the internal parameters of chaotic systems (Table 1).
- Furthermore many previous implementations of chaotic dynamics into the evolutionary/swarm based algorithms (not-adaptive/adaptive/ensemble based) showed that it is advantageous, since it can be easily implemented into any existing algorithm as a plug-in module.

Acknowledgements. 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, and by Internal Grant Agency of Tomas Bata University under the projects No. IGA/CEBIA-Tech/2017/004.

References

1. Price, K.V., An introduction to differential evolution. In: Corne, D., Dorigo, M., Glover, F., (eds.) New Ideas in Optimization, pp. 79–108. McGraw-Hill Ltd., London (1999)
2. Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. **10**(6), 646–657 (2006)
3. Mallipeddi, R., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F.: Differential evolution algorithm with ensemble of parameters and mutation strategies. Appl. Soft Comput. **11**(2), 1679–1696 (2011)
4. Das, S., Suganthan, P.N.: Differential evolution: a survey of the state-of-the-art. IEEE Trans. Evol. Comput. **15**(1), 4–31 (2011)
5. Das, S., Mullick, S.S., Suganthan, P.N.: Recent advances in differential evolution – an updated survey. Swarm Evol. Comput. **27**, 1–30 (2016)
6. Neri, F., Tirronen, V.: Recent advances in differential evolution: a survey and experimental analysis. Artif. Intell. Rev. **33**(1–2), 61–106 (2010)
7. Weber, M., Neri, F., Tirronen, V.: A study on scale factor in distributed differential evolution. Inf. Sci. **181**(12), 2488–2511 (2011)
8. Neri, F., Iacca, G., Mininno, E.: Disturbed exploitation compact differential evolution for limited memory optimization problems. Inf. Sci. **181**(12), 2469–2487 (2011)
9. Iacca, G., Caraffini, F., Neri, F.: Compact differential evolution light: high performance despite limited memory requirement and modest computational overhead. J. Comput. Sci. Technol. **27**(5), 1056–1076 (2012)
10. Zamuda, A., Brest, J.: Self-adaptive control parameters' randomization frequency and propagations in differential evolution. Swarm Evol. Comput. **25**, 72–99 (2015)
11. Caponetto, R., Fortuna, L., Fazzino, S., Xibilia, M.G.: Chaotic sequences to improve the performance of evolutionary algorithms. IEEE Trans. Evol. Comput. **7**(3), 289–304 (2003)
12. Davendra, D., Zelinka, I., Senkerik, R.: Chaos driven evolutionary algorithms for the task of PID control. Comput. Math. Appl. **60**(4), 1088–1104 (2010)
13. Zelinka, I.: SOMA — self-organizing migrating algorithm. In: Zelinka, I. (ed.) New Optimization Techniques in Engineering. Studies in Fuzziness and Soft Computing, vol. 141, pp. 167–217. Springer, Heidelberg (2004)
14. dos Santos Coelho, L., Mariani, V.C.: A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch. Chaos, Solitons Fractals **39**(2), 510–518 (2009)
15. Pluhacek, M., Senkerik, R., Davendra, D., Kominkova Oplatkova, Z., Zelinka, I.: On the behavior and performance of chaos driven PSO algorithm with inertia weight. Comput. Math. Appl. **66**(2), 122–134 (2013)
16. Pluhacek, M., Senkerik, R., Zelinka, I., Davendra, D.: Chaos PSO algorithm driven alternately by two different chaotic maps – an initial study. In: 2013 IEEE Congress on Evolutionary Computation (CEC), 20–23 June 2013, pp. 2444–2449 (2013)
17. Pluhacek, M., Senkerik, R., Davendra, D.: Chaos particle swarm optimization with Eensemble of chaotic systems. Swarm Evol. Comput. **25**, 29–35 (2015)
18. Metlicka, M., Davendra, D.: Chaos driven discrete artificial bee algorithm for location and assignment optimisation problems. Swarm Evol. Comput. **25**, 15–28 (2015)
19. dos Santos Coelho, L., Ayala, H.V.H., Mariani, V.C.: A self-adaptive chaotic differential evolution algorithm using gamma distribution for unconstrained global optimization. Appl. Math. Comput. **234**, 452–459 (2014)

20. Senkerik, R., Pluhacek, M., Zelinka, I., Davendra, D., Janostik, J.: Preliminary study on the randomization and sequencing for the chaos embedded heuristic. In: Abraham, A., Wegrzyn-Wolska, K., Hassanien, A.E., Snasel, V., Alimi, A.M. (eds.) Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015. AISC, vol. 427, pp. 591–601. Springer, Cham (2016). doi:10.1007/978-3-319-29504-6_55
21. Zelinka, I.: A survey on evolutionary algorithms dynamics and its complexity – mutual relations, past, present and future. Swarm Evol. Comput. **25**, 2–14 (2015)
22. Das, S., Abraham, A., Chakraborty, U.K., Konar, A.: Differential evolution using a neighborhood-based mutation operator. IEEE Trans. Evol. Comput. **13**(3), 526–553 (2009)
23. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution – A Practical Approach to Global Optimization. Natural Computing Series. Springer, Heidelberg (2005)
24. Sprott, J.C.: Chaos and Time-Series Analysis. Oxford University Press, New York (2003)
25. Senkerik, R., Pluhacek, M., Kominkova Oplatkova, Z., Davendra, D.: On the parameter settings for the chaotic dynamics embedded differential evolution. In: 2015 IEEE Congress on Evolutionary Computation (CEC), 25–28 May 2015, pp. 1410–1417 (2015)