On the Adaptivity and Complexity Embedded into Differential Evolution

Roman Senkerik[a], Michal Pluhacek[a], Ivan Zelinka[b] and Roman Jasek[a]

[a]*Tomas Bata University in Zlin , Faculty of Applied Informatics, Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic, {senkerik,pluhacek}@fai.utb.cz*
[b]*Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, 17. listopadu 15,708 33 Ostrava-Poruba, Czech Republic, ivan.zelinka@vsb.cz*

Abstract. This research deals with the comparison of the two modern approaches for evolutionary algorithms, which are the adaptivity and complex chaotic dynamics. This paper aims on the investigations on the chaos-driven Differential Evolution (DE) concept. This paper is aimed at the embedding of discrete dissipative chaotic systems in the form of chaotic pseudo random number generators for the DE and comparing the influence to the performance with the state of the art adaptive representative jDE. This research is focused mainly on the possible disadvantages and advantages of both compared approaches. Repeated simulations for Lozi map driving chaotic systems were performed on the simple benchmark functions set, which are more close to the real optimization problems. Obtained results are compared with the canonical not-chaotic and not adaptive DE. Results show that with used simple test functions, the performance of ChaosDE is better in the most cases than jDE and Canonical DE, furthermore due to the unique sequencing in CPRNG given by the hidden chaotic dynamics, thus better and faster selection of unique individuals from population, ChaosDE is faster.

Keywords: Heuristic, chaotic dynamics, discrete chaotic maps, evolutionary algorithms.
PACS: 05.45.Gg, 07.05.Mh

INTRODUCTION

This research deals with the comparison of the two modern approaches for evolutionary algorithms, which are the adaptivity and embedding of complex chaotic dynamics. This paper is aimed at investigating the influence of chaotic dynamics to the performance of Differential Evolution (DE) algorithm [1] and comparing the advantages and disadvantages of chaotic approach with the adaptive techniques for used heuristic. The adaptive strategy of interest within this paper is the state of the art representative jDE. [2]. A chaotic approach generally uses the chaotic map in the place of a pseudo random number generator [3] (CPRNG).

Recently the concepts of chaos driven heuristic have been more intensively studied. Several papers have been recently focused on the connection of DE and chaotic dynamics either in the form of hybridizing of DE with chaotic searching algorithm [4] or in the form of chaotic mutation factor and dynamically changing weighting and crossover factor in self-adaptive chaos differential evolution (SACDE) [5]. The work [6] uses chaos for the initialization of DE (CIDE algorithm). The focus of our research is the direct embedding of chaotic systems in the form of chaos pseudo random number generator (CPRNG) into the DE (ChaosDE) as introduced firstly in [7].

Also the PSO (Particle Swarm Optimization) algorithm with elements of chaos was introduced as CPSO [8]. Later on, the chaos embedded PSO with inertia weigh strategy was closely investigated [9], followed by the introduction of a PSO strategy driven alternately by two chaotic systems [10]. Recently the chaos driven firefly algorithm has been introduced [11], chaotic differential bee colony[12]and the concept of chaos driven DE has become more intensively studied [13], [14].

USED HEURISTIC

Differential Evolution is a population-based optimization method that works on real-number-coded individuals [1], [15]. DE is quite robust, fast, and effective, with global optimization ability. A simple and very efficient adaptive DE strategy, known as jDE, was introduced by Brest et al. [2]. This adaptive strategy utilizes basic DE/rand/1/bin scheme[1] with a simple adaptive mechanism for mutation and crossover control parameters (F and Cr). The ensemble of these two control parameters is assigned to each individual of the population and survives if an individual is successful. The initialization is fully random with uniform distribution for each solution in population.

International Conference of Numerical Analysis and Applied Mathematics 2015 (ICNAAM 2015)
AIP Conf. Proc. 1738, 120030-1–120030-4; doi: 10.1063/1.4951913
Published by AIP Publishing. 978-0-7354-1392-4/$30.00

If the new generated solution is not successful; the new (possibly) mutated control parameters disappear together with not successful solution [16].

EXPERIMENT DESIGN

The general idea of basic ChaosDE and CPRNG is to replace the default pseudorandom number generator (PRNG) with the discrete chaotic map.In this research, direct output iterations of the chaotic maps were used for the generation of real numbers in the process of crossover inside DE and for the generation of the integer values used for selection of individuals from the population.

Previous successful experiments with chaos driven PSO and DE algorithms have manifested that very promising experimental results were obtained through the utilization of Lozi map. Mathematical description of aforementioned chaotic map is given in [17].

For the purpose of ChaosDE, Canonical DE and jDE investigation on DE performance, Schwefel's test function (1), shifted 1st De Jong's function (2), shifted Ackley's original function (3), shifted Rastrigin`s function (4) were selected. Dimension was set to 30.

$$f(x) = 418.9829 \cdot \mathrm{D} - \sum_{i=1}^{D} -x_i \sin(\sqrt{|x|}). \tag{1}$$

Function minimum:
Position for E_n: $(x_1, x_2 \ldots x_n) = (420.969, 420.969, \ldots, 420.969)$; Value for E_n: $y = 0$, $x_i \in \langle -500,500 \rangle$

$$f(x) = \sum_{i=1}^{\dim} (x_i - s_i)^2 . \tag{2}$$

Function minimum: Position for E_n: $(x_1, x_2 \ldots x_n) = \boldsymbol{s}$; Value for E_n: $y = 0$, $x_i \in <-5.12, 5.12>$.

$$f(x) = -20\exp\left(-0.02\sqrt{\frac{1}{D}\sum_{i=1}^{D}(x_i)^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi(x_i)\right) + 20 + \exp(1). \tag{3}$$

Function minimum: Position for E_n: $(x_1, x_2 \ldots x_n) = (0, 0, \ldots, 0)$; Value for E_n: $y = 0$, $x_i \in \langle -30,30 \rangle$

$$f(x) = 10\dim + \sum_{i=1}^{\dim} (x_i - s_i)^2 - 10\cos(2\pi x_i - s_i) . \tag{4}$$

Function minimum: Position for E_n: $(x_1, x_2 \ldots x_n) = \mathbf{s}$; Value for E_n: $y = 0$, $x_i \in <-5.12, 5.12>$.

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

Experiments were performed in the combined environments of *Wolfram Mathematica* and *C language*, canonical DE and jDE therefore used the built-in *C language* pseudo random number generator *Mersenne Twister C* representing traditional pseudorandom number generators in comparisons. All experiments used different initialization, i.e. different initial population was generated in each run.

Within this research, one type of experiment was performed. It utilizes the maximum number of generations fixed at 1500 generations. This allowed the possibility to analyze the progress of all studied DE variants within a limited number of generations and cost function evaluations. Parameter setting was following: $F = Cr = 0.4$ for ChaosDE, $F = 0.5$, $Cr = 0.9$ for canonical DE, parameters $a = 1.4$, $b = 0.3$ for Lozi map (See[17]), jDE has utilized the recommended settings as in [16].

RESULTS

Statistical results of the selected experiments are shown in Tables1-4, which represents the simple statistics for Cost Function (CF) values, e.g. average, minimum values representing the best individual solution, standard deviations and execution time for all 50 repeated runs of DE/ChaosDE/jDE.The bold values within the all Tables 1-4 depict the best obtained results.

TABLE 1. Statistical results for Schwefel's test function.

Algorithm	Avg. CF	Min CF	Std. dev	Time
Canonical DE	-5384.8	-6286.55	337.628	23.7968
Chaos DE Lozi map	-12412.9	**-12569.5**	147.985	**20.8750**
jDE	**-12569.5**	**-12569.5**	**0.0035**	24.4531

TABLE 2. Statistical results for shifted 1st De Jong's function.

Algorithm	Avg. CF	Min CF	Std. dev	Time
Canonical DE	2.4E-21	1.73E-22	3.17E-21	16.1093
Chaos DE Lozi map	**0**	**0**	**1.67E-27**	**14.5312**
jDE	3.18E-15	6.95E-16	3.83E-15	18.8125

TABLE 3. Statistical results for shifted Ackley's original function.

Algorithm	Avg. CF	Min CF	Std. dev	Time
Canonical DE	**1.78E-10**	3.40E-11	**1.03E-10**	19.2031
Chaos DE Lozi map	0.0117	**3.99E-15**	0.0642	**17.6250**
jDE	2.48E-07	1.26E-07	7.94E-08	22.2187

TABLE 4. Statistical results for shifted Rastrigin`s function.

Algorithm	Avg. CF	Min CF	Std. dev	Time
Canonical DE	168.5168	126.1883	16.2593	19.2968
Chaos DE Lozi map	**15.8374**	**4.9521**	14.7125	**16.8125**
jDE	32.4262	24.2600	**3.6793**	21.0625

CONCLUSION

This work was aimed at the deeper analysis of the chaotic dynamics directly injected into the DE. This paper compared the ChaosDE with state-of-the art adaptive representative, which is simple adaptive jDE.This research is focused mainly on the possible disadvantages and advantages of both compared approaches.

Results lend weigh to the argument that through utilization of simple discrete chaotic map in the place of pseudo random number generator inside heuristic, the chaos embedded heuristic may in some results attributes outperform even the adaptive strategies. The findings can be summarized as follows:

- The high sensitivity of the DE to the internal dynamics of the CPRNG is fully manifested within all four case studies.
- With used simple test functions, the performance of ChaosDE is better in the most cases than jDE and Canonical DE. This makes the ChaosDe concept easy tu use (plug-in) for (simple) real optimization tasks.
- Due to the unique sequencing in CPRNG given by the hidden chaotic dynamics, thus better and faster selection of unique individuals from population, ChaosDE is faster.
- jDE is advantageous from the point of view of stable performance (very low std. dev value) and easy parameter settings thanks to their adaptivity.

ACKNOWLEDGMENTS

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 projects No. IGA/FAI/2015/057.

REFERENCES

1. Price K.V. An Introduction to Differential Evolution. In: Corne D, Dorigo M, Glover F (eds) New Ideas in Optimization. McGraw-Hill Ltd., pp 79-108, 1999.
2. Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V.: Self-adapting control parame-ters in differential evolution: A comparative study on numerical benchmark problems. Evolutionary Computation, IEEE Transactions on 10(6), 646-657 2006.
3. Caponetto R., Fortuna L., Fazzino S., Xibilia M.G. Chaotic sequences to improve the performance of evolutionary algorithms. IEEE Transactions on Evolutionary Computation 7 (3):289-304, 2003.
4. 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.
5. 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.
6. A. Bedri Ozer, “CIDE: Chaotically Initialized Differential Evolution,” Expert Systems with Applications, vol. 37, no. 6, 2010, pp. 4632-4641.
7. 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.
8. Coelho L.d.S., 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.
9. Pluhacek M., Senkerik R., Davendra D., Kominkova Oplatkova Z., Zelinka I. On the behavior and performance of chaos driven PSO algorithm with inertia weight. Computers & Mathematics with Applications 66 (2):122-134, 2013.
10. 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 2013. pp 2444-2449.
11. A. H. Gandomi, X. S. Yang, S. Talatahari, and A. H. Alavi, "Fire fly algorithm with chaos," *Communications in Nonlinear Science and Numerical Simulation,* vol. 18, pp. 89-98, 2013.
12. P. Lu, J. Zhou, H. Zhang, R. Zhang, and C. Wang, "Chaotic differential bee colony optimization algorithm for dynamic economic dispatch problem with valve-point effects," *International Journal of Electrical Power & Energy Systems,* vol. 62, pp. 130-143, 2014.
13. L. d. S. Coelho and M. W. Pessôa, "A tuning strategy for multivariable PI and PID controllers using differential evolution combined with chaotic Zaslavskii map," *Expert Systems with Applications,* vol. 38, pp. 13694-13701, 2011.
14. L. d. S. Coelho, H. V. H. Ayala, and V. C. Mariani, "A self-adaptive chaotic differential evolution algorithm using gamma distribution for unconstrained global optimization," *Applied Mathematics and Computation,* vol. 234, pp. 452-459, 2014.
15. Price K.V., Storn RM, Lampinen JA. Differential Evolution - A Practical Approach to Global Optimization. Natural Computing Series.Springer Berlin Heidelberg, 2005.
16. Tvrdík, J., Poláková, R., Veselský, J., Bujok, P.: Adaptive Variants of Differential Evolution: Towards Control-Parameter-Free Optimizers. In: Zelinka, I., Snášel, V., Abraham, A. (eds.) Handbook of Optimization, vol. 38. Intelligent Systems Reference Library, pp. 423-449. Springer Berlin Heidelberg, (2013).
17. Sprott J.C. Chaos and Time-Series Analysis. Oxford University Press, 2003.