COMPARISON OF PSO AND DE IN THE TASK OF OPTIMAL CONTROL OF CHAOTIC LOZI MAP

Roman Senkerik[(a)], Michal Pluhacek[(b)], Ivan Zelinka[(c)], Adam Viktorin[(d)]

[(a),(b),(d)]Tomas Bata University in Zlin , Faculty of Applied Informatics, Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic

[(c)] Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic

[(a)]senkerik@fai.utb.cz, [(c)]ivan.zelinka@vsb.cz

ABSTRACT
In this paper, evolutionary algorithm Differential Evolution (DE) is compared with swarm based technique PSO in the task of optimal evolutionary tuning of controller parameters for the stabilization of selected discrete chaotic system, which is the two-dimensional Lozi map. The novelty of the approach is that the most utilized examples of evolutionary/swarm based algorithms are compared directly on the highly nonlinear and complex multimodal optimization and simulation task. The simulations were performed for three different required final behavior of the chaotic system.

Keywords: Chaos control, Differential Evolution, PSO, optimization, evolutionary algorithms, swarm algorithms, soft computing.

1. INTRODUCTION
In many applications, one of the most challenging tasks is the controlling of highly nonlinear dynamical systems in order to either eliminate or synchronize the chaos. The first successful approach to control chaotic dynamics by means of a simple linearization technique was introduced in 1990s by Ott, Grebogy and Yorke (i.e. OGY method) (Ott, Grebogi, and York 1990). Later, the rapid development of methods for stabilizing of chaotic dynamics has arisen and more advanced modern techniques have been applied for chaos control and chaos synchronization including unconvential methods from the soft computing field.
The most current intelligent methods are mostly based on soft computing, 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 (EA's) and fuzzy logic. Currently, EA's are known as a powerful set of tools for almost any difficult and complex optimization problem.
The interest about the connection between evolutionary techniques and (not only) control of chaotic systems is rapidly spreading. The initial research was conducted in (Zelinka 2009), whereas (Zelinka, Senkerik, and Navratil 2009) and (Senkerik et. al 2010) was more concerned with the tuning of parameters inside the existing chaos control technique based on the Pyragas extended delay feedback control (ETDAS) (Pyragas 1995). Later works (Senkerik et. al 2013; Kominkova-Oplatkova et al. 2013; Senkerik et al. 2014) show a novel approach as to how to generate the entire control law (control method) for the purpose of stabilization of any chaotic system.
Other approaches utilizing the EA's for the stabilizing of chaotic dynamics have mostly applied the Particle Swarm Optimization algorithm (PSO) (Kennedy and Eberhart 1995) and multi-interval gradient-method (Abedini, Vatankhaha and Assadian 2012) or minimum entropy control technique (Sadeghpour et al. 2011). EA's have been also frequently used in the task of synchronization of chaos (Coleho and Grebogi 2012). In (Richter 2000) an EA for optimizing local control of chaos based on a Lyapunov approach is presented.
This work is an extension of previous aforementioned research focused on optimal stabilization of chaotic systems by means of evolutionary algorithms. The novelty of this work and motivation was to perform a direct comparison between swarm based algorithms and evolutionary algorithms in such a highly nonlinear and complex multimodal simulation and optimization task.
Firstly, a problem design is proposed. The next sections are focused on the description of used metaheuristics, experiment workflow, results and conclusion.

2. PROBLEM DESIGN
The brief description of used chaotic system and original feedback chaos control method, ETDAS is given.

2.1. Lozi Map Chaotic System
The chosen example of chaotic system was the two-dimensional Lozi map in form (1).

$$\begin{aligned} X_{n+1} &= 1 \quad a\left|X_n\right| + bY_n \\ Y_{n+1} &= X_n \end{aligned} \tag{1}$$

The Lozi map is a simple discrete two-dimensional chaotic map. The map equations are given in (1). The parameters used in this work are: a = 1.7 and b = 0.5 as suggested in (Sprott 2003). For these values, the system exhibits typical chaotic behavior and with this parameter

setting it is used in the most research papers and other literature sources (Aziz-Alaoui and Grebogi 2001).
The *x*, *y* plot of the selected map is depicted in Figure 1. The chaotic behavior of the map, represented by the example of output iterations is depicted in Figure 2.

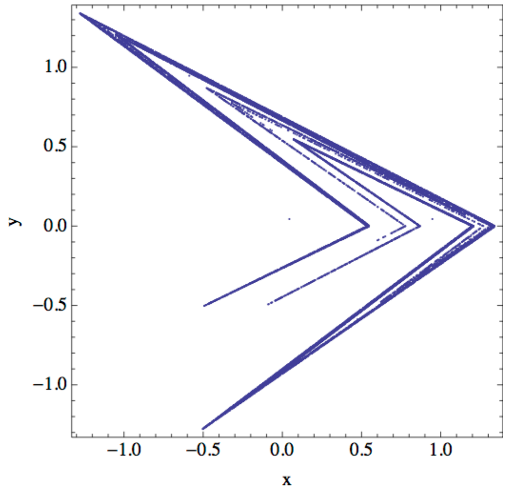

Figure 1: *x*, *y* plot of chaotic Lozi map

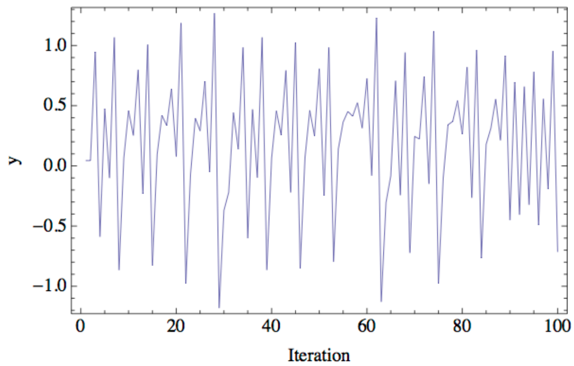

Figure 2: Iterations of the uncontrolled Lozi map.

2.2. ETDAS Control Method

This work is focused on the direct performance comparisons of DE and PSO algorithms in the task of tuning of parameters for ETDAS control method to stabilize desired Unstable Periodic Orbits (UPO). The original control method – ETDAS has form (2).

$$\begin{aligned} F(t) &= K[(1-R)S(t-\tau_d) - x(t)] \\ S(t) &= x(t) + RS(t-\tau_d) \end{aligned} \quad (2)$$

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 τ_d is a time delay.

The original control method – ETDAS in the discrete form suitable for optimizations and connection with discrete maps has form (3).

$$\begin{aligned} F_n &= K[(1-R)S_{n-m} - x_n] \\ S_n &= x_n + RS_{n-m} \end{aligned} \quad (3)$$

Where: m is the period of m-periodic orbit to be stabilized. The perturbation F_n in equations (3) may have arbitrarily large value, which can cause diverging of the system outside the interval $\{0, 1.0\}$. Therefore, F_n should have a value between $F_{\max}$, $F_{\max}$. To find the optimal value, it is a task of metaheuristics.

2.3. Cost Function

The idea of the basic cost function (CF_{Simple}), which could be used problem-free only for the stabilization of p-1 orbit (stable state of chaotic system – no oscillations), 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. (4). This CF design is very convenient for the evolutionary searching process due to the relatively favorable CF surface and it is convenient for simple simulation based performance comparisons. Nevertheless, this simple approach has one big disadvantage, which is the including of initial chaotic transient behavior of not stabilized system into the CF value. As a result of this, the very tiny change in control method setting for extremely sensitive chaotic system (given by the very small change of CF value), can be suppressed by the above-mentioned including of initial chaotic transient.

$$CF_{Simple} = \sum_{t=0}^{\tau_i} |TS_t - AS_t|, \quad (4)$$

where: TS - target state, AS - actual state

3. USED METAHEURISTICS

This work has utilized two metaheuristics as an examples of swarm based algorithm, which is PSO and population (evolution) based algorithm DE.

3.1. Particle Swarm Optimizer – PSO

Original PSO algorithms take their inspiration from behaviour of fish and birds (Kennedy and Eberhart 1995). The knowledge of the global best-found solution (typically denoted as *gBest*) is shared among the particles in the swarm. Furthermore, each particle has the knowledge of its own (personal) best-found solution (designated *pBest*). The last important part of the algorithm is the velocity of each particle, which is taken into account during the calculation of the particle´s movement. The new position of each particle is then given by (5), where x_i^{t+1} is the new particle position; x_i^t refers to the current particle position and v_i^{t+1} is the new velocity of the particle.

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (5)$$

To calculate the new velocity, the distance from *pBest* and *gBest* is taken into account along with its current velocity (6).

$$v_{ij}^{t+1} = v_{ij}^t + c_1 \cdot Rand \cdot (pBest_{ij} - x_{ij}^t) + c_2 \cdot Rand \cdot (gBest_j - x_{ij}^t) \quad (6)$$

Where:
v_{ij}^{t+1} - New velocity of the i^{th} particle in iteration t+1.; (component j of the dimension D).
v_{ij}^{t} - Current velocity of the i^{th} particle in iteration t.; (component j of the dimension D).
c_1, c_2 = 2 - Acceleration constants.
$pBest_{ij}$ – Local (personal) best solution found by the i^{th} particle; (component j of the dimension D).
$gBest_j$ - Best solution found in a population; (component j of the dimension D).
x_{ij}^{t} - Current position of the i^{th} particle; (component j of the dimension D) in iteration t.
Rand – Pseudo-random number, interval (0, 1).

3.2. Differential Evolution

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 (Price, Storn and Lampinen 2005), (Storn and Price 1997), (Price 1999). 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 following.

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 (7) for notation of crossover, and to (Price, Storn and Lampinen 2005) for the detailed description of used DERand1Bin strategy and all other DE strategies:

$$u_{i,G+1} = x_{r1,G} + F\ \left(x_{r2,G} \quad x_{r3,G}\right) \tag{7}$$

4. SIMULATION RESULTS

This research encompasses three case studies. Three different required behavior of the chaotic system were simulated in the following form:

- Case study 1: p-1 UPO, Lozi map as controlled system with CF_{Simple}.
- Case study 2: p-2 UPO, Lozi map as controlled system with CF_{Simple}.
- Case study 3: higher order p-4 UPO, Lozi map as controlled system with CF_{Simple}.

The ranges of all evolutionary estimated parameters are given in Tab. 1.

Table 1: Estimated parameters and ranges

Parameter	Min	Max
K	-2	2
R	0	0.99
Fmax	0	0.9

Within the research a total number of 50 simulations for each experiments were performed in an environment of *Wolfram Mathematica*. All experiments used different initialization, i.e. different initial population was generated in each run of DE/PSO.
The parameter settings for both DE and PSO were given following way (see Table 2 and 3)

Table2: DE settings

Parameter	Value
PopSize	25
F	0.5
Cr	0.9
Generations	300
Max. CF Evaluations (CFE)	7500

Table 3: PSO settings

Parameter	Value
PopSize	25
c_1 , c_2	2.0
Iterations	300
Max. CF Evaluations (CFE)	7500

All simulations were successful and have given new optimal settings for ETDAS control method securing the fast stabilization of the chaotic system at required behaviors, which were p-1 UPO (stable state), p-2 UPO (oscillation between two values) and finally p-4 UPO. The organization of the results is following:
Tables 5 -7 are focused on the performance comparisons between DE and swarm based PSO algorithm. These tables contain simple statistical overview of evolutionary optimization/simulation results i.e. average, median max, min (the best solution), std. dev. values for the particular cost function and for all 50 runs of both compared heuristics. Italic numbers represent the best result.

The chaos stabilization properties for the particular case studies are given in the description (caption) of the figures with the simulation of the best individual solutions. These chaos stabilization properties contain parameters set up for ETDAS control method, 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 3, 5 and 7 whereas the Figures 4, 6 and 8 show the simulation output of all 50 runs of the best performing metaheuristic, thus confirm the robustness of this approach. For the illustrative purposes, all graphical simulations outputs are depicted only for the variable *x* of the stabilized chaotic system.

The graphical comparisons for the performance analysis of DE and PSO within all 3 case studies are given in complex Figure 9. It shows the comparisons of time evolution of average CF values for all 50 runs of DE/PSO, thus it confirms the robustness of both used metaheuristic strategies within many repeated runs.

The values for desired UPOs of unperturbed chaotic Lozi map based on the mathematical analysis of the systems are given in Table 4.

Table 4: The values for desiredUPOs.

UPO	Values of UPO of unperturbed system
p-1	$x_F = 0.454545$
p-2	$x_1 = -0.382166$; $x_2 = 0.700637$
p-4	$x_1 = -0.691899$; $x_2 = 0.334059$ $x_3 = 0.086151$; $x_4 = 1.020573$

4.1. Case study 1

In the simplest case, the performance of both heuristic is similar from the statistical point of view. Nevertheless, the DE has converged very fast towards optimal solution in 20 generations, as in Fig 9, (multiplied by population size, it means only 50 CF evaluations were required). All 50 runs of heuristic have given identical results (Fig 4), i.e. confirmed the robustness of the DE algorithm while searching in very nonlinear solution space. Convergence of PSO is much slower, DE is therefore suitable also for on-line control of nonlinear chaotic dynamics.

Table 5: Comparison for DE and PSO case study 1.

Statistical data	DE CF Value	PSO CF Value
Min	*0.520639*	0.530679
Max	*0.527132*	0.573742
Average	*0.520769*	0.548688
Median	*0.520639*	0.549588
Std.Dev.	$9.18 \cdot 10^{-4}$	$1.07 \cdot 10^{-2}$

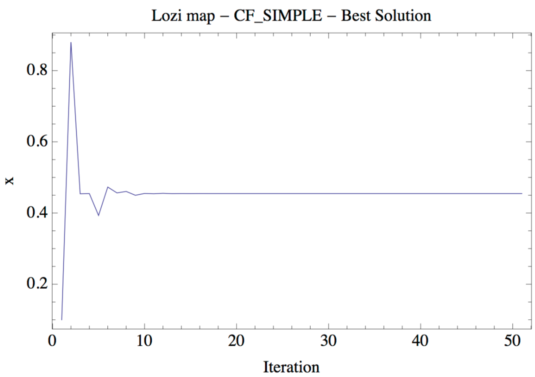

Figure 3: Simulation of the best individual solution – DE and Lozi map: Case study 1, $K = -1.11259$, $F_{max} = 0.9$, $R = 0.289232$, Istab. Value = 21, Avg. err. = $7.21 \cdot 10^{-15}$

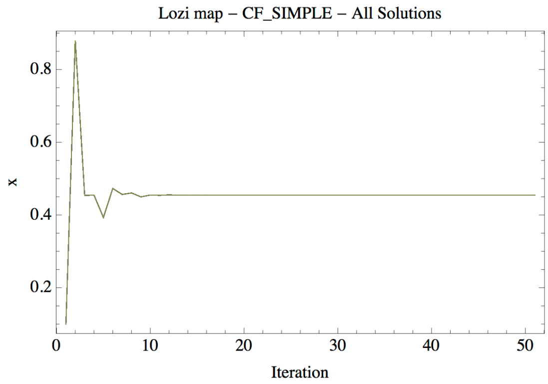

Figure 4: All 50 runs of EA – DE and Lozi map: Case study 1.

4.2. Case study 2

As in the previous case, results structure, as well as simulation outputs show similar features, even with a larger difference in favor of the algorithm DE.

Table 6: Comparison for DE and PSO case study 2.

Statistical data	DE CF Value	PSO CF Value
Min	*6.99829*	7.29818
Max	*7.33379*	8.05143
Average	*7.2827*	7.67487
Median	*7.33379*	7.6944
Std.Dev.	$9.29 \cdot 10^{-2}$	0.216771

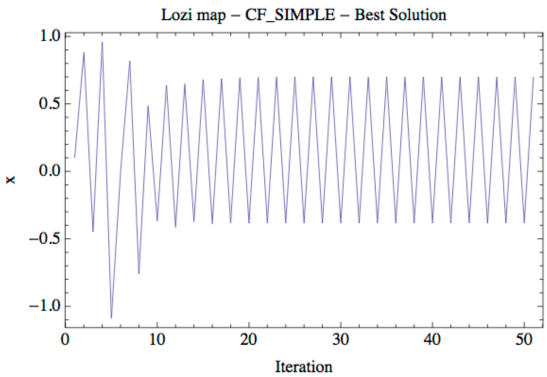

Figure 5: Simulation of the best individual solution – DE and Lozi map: Case study 2, $K = 0.5740$, $F_{max} = 0.4308$, $R = 0.4454$, Istab. Value = 22, Avg. err. = $2.98 \cdot 10^{-8}$

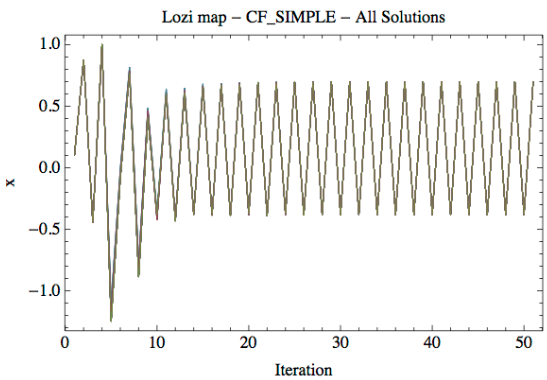

Figure 6: All 50 runs of EA – DE and Lozi map: Case study 2.

4.3. Case study 3

Table 7: Comparison for DE and PSO case study 3.

Statistical data	DE	PSO
	CF Value	CF Value
Min	*13.7305*	14.9151
Max	51.9391	*29.6333*
Average	*15.4354*	21.794
Median	*14.2548*	22.4428
Std.Dev.	5.4219	*3.5719*

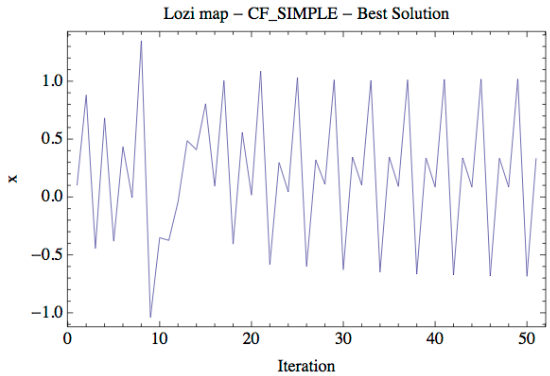

Figure 7: Simulation of the best individual solution – DE and Lozi map: Case study 1, K = -0.8693, F_{max} = 0.25556, R = 0.4093, Istab. Value = 46, Avg. err. = $2.96 \cdot 10^{-4}$

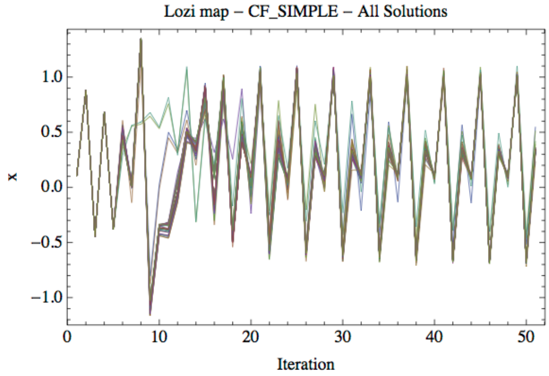

Figure 8: All 50 runs of EA – DE and Lozi map: Case study 3.

The last case study dealing with the most complex and highly nonlinear dynamics shows interesting features. Classical geometrical (vector crossover) based evolutionary algorithm has been stacked in many suboptimal solutions, with small chance to leave this area of solution space. While PSO (swarm based) algorithm was statistically better in searching process (lower standard deviation, range of CF). Even though the DE has found lower final CF value, the PSO seems to be a better and more robust choice for optimization in extremely nonlinear solution space due to its natural better exploration ability.

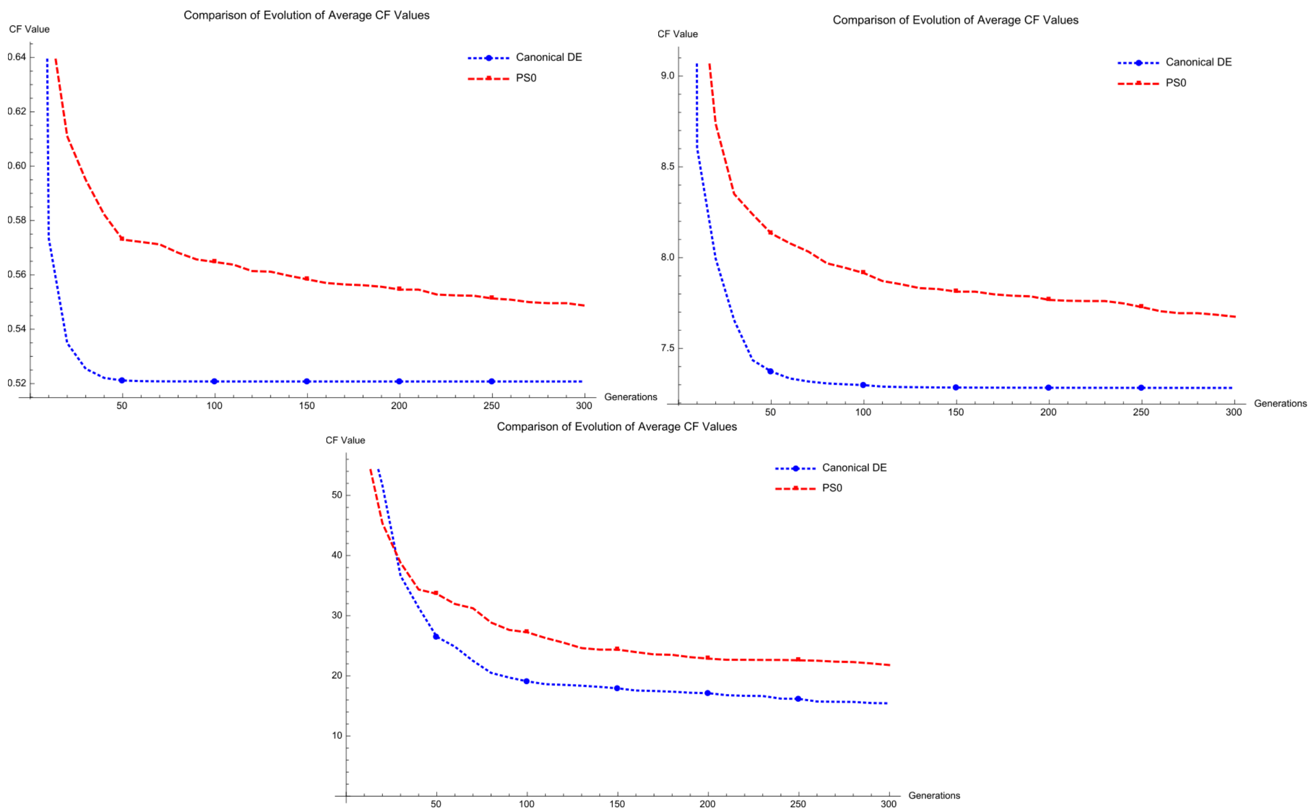

Figure 9: Comparisons of time evolution of average cost function values of the best solutions from all 50 runs of DE and PSO; from left to right and from above to bottom: Case study 1 – Case study 3.

5. CONCLUSION

In this paper, evolutionary algorithm Differential Evolution and swarm algorithm PSO were used for the evolutionary tuning of controller parameters for the stabilization of selected discrete chaotic system, which was the two-dimensional Lozi map.

The originality of the presented approach is that the examples of swarm based algorithms and evolutionary algorithms are compared in such a highly nonlinear and complex multimodal optimization task, which is optimal control of chaotic systems.

Comparisons between both DE versions and swarm based PSO algorithm show, that PSO is not good choice in the task of nonlinear/chaos control optimization.

Future research will be aimed at energy costs and more precise and faster stabilization and at the time-continuous systems, not only discrete chaotic maps.

ACKNOWLEDGMENTS

This work was supported by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089 and project IT4Innovations Centre of Excellence No. CZ.1.05/1.1.00/02.0070 and and by Internal Grant Agency of Tomas Bata University under the project No.IGA/FAI/2012/037.

REFERENCES

Abedini, M., R. Vatankhah, N. Assadian Stabilizing chaotic system on periodic orbits using multi-interval and modern optimal control strategies. Communications in Nonlinear Science and Numerical Simulation, 2012, 17(10), 3832-3842.

Aziz-Alaoui, M. A., C. Robert, C. Grebogi Dynamics of a Hénon–Lozi-type map. Chaos, Solitons & Fractals, 2001, 12(12), 2323-2341.

Coelho, L. D. S., R. B. Grebogi Chaotic synchronization using PID control combined with population based incremental learning algorithm. Expert Systems with Applications, 2010, 37(7), 5347-5352.

Kennedy, J., R. Eberhart. Particle swarm optimization. In *IEEE International Conference on Neural Networks.* 1995, vol. 4, p. 1942-1948.

Kominkova Oplatkova, Z., R. Senkerik, I. Zelinka, M. Pluhacek Analytic programming in the task of evolutionary synthesis of a controller for high order oscillations stabilization of discrete chaotic systems. Computers & Mathematics with Applications, 2013, 66(2), 177-189.

Ott, E., C. Grebogi, J. A. Yorke Controlling chaos. Physical Review Letters, 1990, 64(11), 1196-1199.

Price K., Storn R. M., Lampinen J. A., 2005, “Differential Evolution : A Practical Approach to Global Optimization”, (Natural Computing Series), Springer; 1 edition.

Price, K. V. An Introduction to Differential Evolution. In D. CORNE, M. DORIGO AND F. GLOVER eds. *New Ideas in Optimization.* McGraw-Hill Ltd., 1999, p. 79-108.

Pyragas K., 1995. “Control of chaos via extended delay feedback”, Physics Letters A, vol. 206, 1995, pp. 323-330.

Richter, H., K. J. Reinschke Optimization of local control of chaos by an evolutionary algorithm. Physica D: Nonlinear Phenomena, 2000, 144(3–4), 309-334.

Sadeghpour, M., H. Salarieh, G. Vossoughi, A. Alasty Multi-variable control of chaos using PSO-based minimum entropy control. Communications in Nonlinear Science and Numerical Simulation, 2011, 16(6), 2397-2404.

Senkerik, R., I. Zelinka, D. Davendra, Z. Oplatkova Utilization of SOMA and differential evolution for robust stabilization of chaotic Logistic equation. Computers & Mathematics with Applications, 2010, 60(4), 1026-1037.

Senkerik, R., Z. Kominkova Oplatkova, I. Zelinka, B. Chramcov, et al. Utilization of analytic programming for the evolutionary synthesis of the robust multi-chaotic controller for selected sets of discrete chaotic systems. Soft Computing, 2014, 18(4), 651-668.

Senkerik, R., Z. Oplatkova, I. Zelinka, D. Davendra Synthesis of feedback controller for three selected chaotic systems by means of evolutionary techniques: Analytic programming. Mathematical and Computer Modelling, 2013, 57(1–2), 57-67.

Sprott, J. C. *Chaos and Time-Series Analysis*. Edtion ed.: Oxford University Press, 2003. 528 p.

Storn, R., K. Price Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization, 1997, 11(4), 341-359.

Zelinka, I. Real-time deterministic chaos control by means of selected evolutionary techniques. Engineering Applications of Artificial Intelligence, 2009, 22(2), 283-297.

Zelinka, I., R. Senkerik, E. Navratil Investigation on evolutionary optimization of chaos control. Chaos, Solitons & Fractals, 2009, 40(1), 111-129.