Preliminary Investigation on Relations Between Complex Networks and Evolutionary Algorithms Dynamics

Ivan Zelinka, Donald Davendra, Václav Snášel
Department of Informatics
Faculty of Electrical Engineering and Computing Science, Technical University of Ostrava, Tr. 17. Listopadu 15, Ostrava, Czech Republic
ivan.zelinka@vsb.cz, vaclav.snasel@vsb.cz

Roman Jašek, Roman Šenkeřík, Zuzana Oplatková
Department of Artificial Intelligence and Informatics
Faculty of Applied Informatics, Nad Stranemi 4511, Tomas Bata University in Zlín, Czech Republic
{jasek, senkerik, oplatkova}@fai.utb.cz

Abstract— **In this article we discuss relations between the so-called complex networks and dynamics of evolutionary algorithms. The main aim of this article is to investigate whether it is possible to model (or vizualize) evolutionary dynamics as complex networks, whose connections will represent interactions amongst the individuals during all generations. Our simulations are based on selected evolutionary algorithms (2 algorithms in 6 versions) and test functions (4 out of 17). Data obtained through the simulations were processed graphically as well as statistically.**

Keywords – evolutionary algorithms, dynamics, complex networks

I. Introduction

In this article, we try to merge two completely different (at first glance) areas of research: complex networks and evolutionary computation.

Large-scale networks, exhibiting complex patterns of interaction amongst vertices exist in both nature and in man-made systems (i.e., communication networks, genetic pathways, ecological or economical networks, social networks, networks of various scientific collaboration, Internet, World Wide Web, power grid etc.). The structure of complex networks thus can be observed in many systems.

The word "complex" networks [1], [2] comes from the fact that they exhibit substantial and non-trivial topological features, with patterns of connection between vertices that are neither purely regular nor purely random. Such features include a **heavy tail** in the degree distribution, a **high clustering coefficient**, **hierarchical structure**, amongst other features. In the case of directed networks, these features also include reciprocity, triad significance profile and other features.

Amongst many studies, two well-known and much studied classes of complex networks are the **scale-free networks** and **small-world networks** (see examples in Figures 1 and 2), whose discovery and definition are vitally important in the scope of this research. Specific structural features can be observed in both classes i.e. so called power-law degree distributions for the scale-free networks and short path lengths with high clustering for the smal-world networks. Research in the field of complex networks has joined together researchers from many areas, which were outside of this interdisciplinary research in the past like mathematics, physics, biology, chemistry computer science, epidemiolog etc..

Evolutionary computation is a sub-disipline of computer science belonging to the "bioinspired" computing area. Since the end of the second world war, the main ideas of evolutionary computation has been published [3] and widely introduced to the scientific community [4]. Hence, the "golden era" of evolutionary techniques began, when Genetic Algorithms (GA) by J. Holland [4], Evolutionary Strategies (ES), by Schwefel [5] and Rechenberg [6] and Evolutionary Programming (EP) by Fogel [7] had been introduced. All these designs were favored by the forthcoming of more powerful and more easily programmable computers, so that for the first time interesting problems could be tackled and evolutionary computation started to compete with and became a serious alternative to other optimization methods.

II. Motivation and Preliminaries

Motivation of this research is quite simple. As mentioned in the introduction, evolutionary algorithms are capable of hard problem solving. A number of examples on evolutionary algorithms can be easily found. Evolutionary algorithms (EA) use with chaotic systems is done for example in [8] where EAs has been used on local optimization of chaos, [9] for chaos control with use of the multi-objective cost function or in [10, 11] where evolutionary algorithms have been studied on chaotic landscapes. Slightly different approach with evolutionary algorithms is presented in [12] where selected algorithms were used to synthesize artificial chaotic systems. In [13], [14] EAs has been successfully used for real-time chaos control and in [15] EAs was used for optimization of Chaos Control.

Other examples of evolutionary algorithms application can be found in [16], which developed statistically robust evolutionary algorithms, alongside research conducted by [17]. Parameters of permanent magnet synchronous motors has been optimized by PSO and experimentally validated on the servomotor. Another research was focused on swarm intelligence, which has been used for IIR filter synthesis, co-evolutionary particle swarm oprimization (CoPSO) approach for the design of constrained engineering problems, particularly for pressure vessel, compression spring and welded beam, etc.

On the other side, complex networks, widely studied across many branches of science are promising and is a modern interdisciplinary research. Evolutionary algorithms, based on its cannonical central dogma (following darwinian ideas) clearly demonstrate intensive interaction amongst individual in the population, which is, in general, one of the important attributes of complex networks (intensive interaction amongst the vertices).

The main motivation (as well as a question) is whether it is possible to visualize and simulate underlying dynamics of evoutionary process like complex network. Reason for this is such that today various techniques for analysis and control of complex networks exists and if complex network structure would be hidden behind EA dynamics, then we believe, that for example above mentioned control techniques could be used to improve dynamics of EAs. All experiments here were designed to analyse and either confirm or reject this idea.

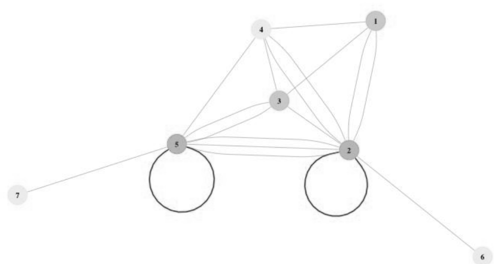


Figure 1. Example of a small network.

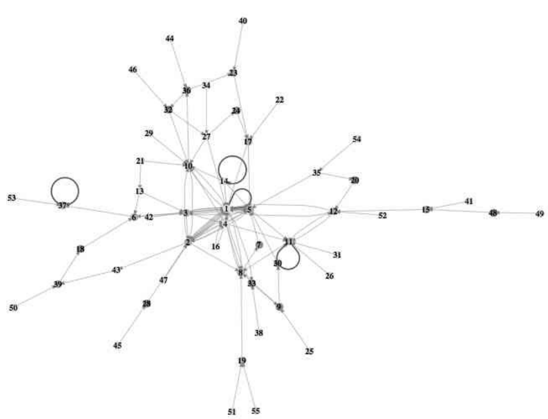


Figure 2. Example of a more complex network with multiple edges and selfloops.

III. Experiment design

A. Selected algorithms and its settings

For the experiments described here, stochastic optimization algorithms, such as DE [18] and Self Organizing Migrating Algorithm (SOMA) [19], have been used. Application of alternative algorithms like GA and Simulated Annealing (SA), ES and/or Swarm Intelligence are now in process.

All experiments have been done on a special server consisting of 16 Apple XServer (2 x 2 GHz Intel Xeon, 1 GB RAM,), each with 4 CPU, so in total 64 CPUs were available for calculations. It is important to note here, that such technology was used to save time due to a large number of calculations, however it must be stated that evolutionary identification described here, is also solvable on a single PC (with longer execution time). For all calculations and data processing, *Mathematica* version 7.0.1.0 was used.

Four versions of SOMA and two versions of DE have been applied for all simulations in this paper. See Table I – Table IV for relation between each version and index corresponding to other Tables. Parameters for the optimizing algorithm were set up in such a way as to reach similar value of maximal cost function evaluations for all used versions. Each version of EAs has been applied 50 times in order to get less or more valuable statistical data.

The primary aim here is not to show which version of EA is better or worse, but to show that dynamics of the EAs can in reality be modelled like complex network.

TABLE I. Used Versions of SOMA

Algorithm	Index
AllToOne	S1
AllToRandom	S2
AllToAll	S3
AllToAllAdaptive	S4

TABLE II. Used Versions of SOMA

Algorithm	Index
DERand1Bin	D1
DELocalToBest	D2

TABLE III. Used Versions of SOMA

	S1	S2	S3	S4
PathLength	3	3	3	3
Step	0.11	0.11	0.11	0.11
PRT	0.1	0.1	0.1	0.1
PopSize	[5-100]	[5-100]	[5-100]	[5-100]
Migrations	[10-100]	[10-100]	[10-100]	[10-100]
MinDiv	-1	-1	-1	-1
Individual Length	[5-50]	[5-50]	[5-50]	[5-50]

TABLE IV. USED VERSIONS OF SOMA

	D1 – D2
NP	[5-100]
F	0.9
CR	0.3
Generations	3000
Individual Length	[5-50]

B. *Selected test functions and its dimensionality*

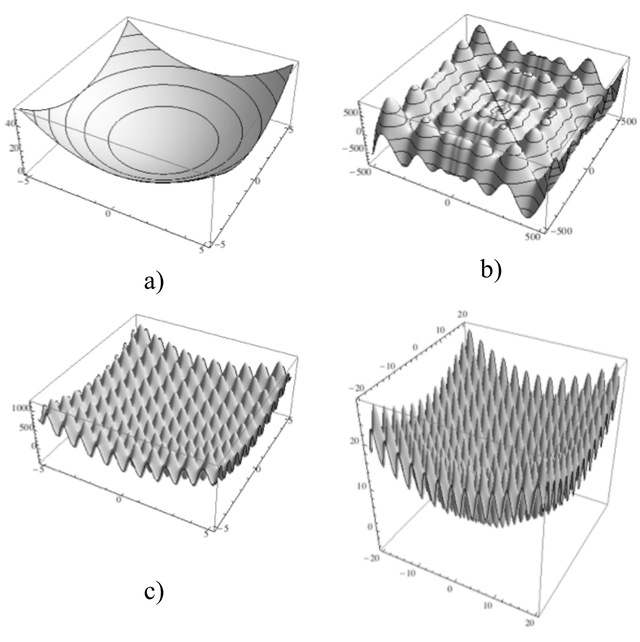
The test functions applied in this experimentation were selected from the test bed of 17 test functions. In total 4 test function were selected as a representative subset of functions which shows geometrical simplicity and low complexity as well as functions from the "opposite side of spectra". Selected functions (see Figure 3) were: 1st DeJong (1), Schwefel's function (2), Rastrigin's function (3) and Ackley's function (4). Each of them has been used for identification of complex networks dynamics and structure in 5, 20 and 50 dimensions (individual length was thus also 5, 20 and 50).

$$\sum_{i=1}^{D} x_i^2 \tag{1}$$

$$\sum_{i=1}^{D} -x_i \sin(\sqrt{|x_i|}) \tag{2}$$

$$2D \sum_{i=1}^{D} x_i^2 - 10\cos(2\pi x_i) \tag{3}$$

$$\sum_{i=1}^{D-1} \frac{1}{e^5}\sqrt{(x_i^2 + x_{i+1}^2)} + 3(\cos(2x_i) + \sin(2x_{i+1})) \tag{4}$$



a) b) c) d)

Figure 3. Selected test functions: 1st DeJong (a), Schwefel's function (b) Rastrigin's function (c) and Ackley's function (d)

C. *Data for complex network vizualization*

The most critical point of this research and related simulations was as to which data and relations should be selected and consequently vizualized. Based on investigated algorithms, we believe that there is no universal aproach, but rather a "personal" one, based on the knowledge of algorithm principle. Of course, some conclusions (see section Conclusion) can be generalized over a class or family of algorithms. As mentioned in the previous sections, algorithms like DE and SOMA were used. Each class of algorithm is based on a different principle. The main idea was such that **each individual is represented by vertex and edges between vertices should reflect dynamics in population, i.e. interactions between individuals** (which individual has been used for offspring creation,…). The SOMA algorithm, as described in [19], consists of a Leader attracting the entire population in each migration loop (equivalent of generation), so in that class of swarm like algorithm, it is clear that the position in the population of activated Leaders shall be recorded like vertex and used (with remaining part of population) for vizualization and statistical data processing. The other case is DE, e.g. DERand1Bin in which each individual is selected in each generation to be a parent. Thus in DE, we have recorded ony those individuals-parents, that has been replaced by better offspring (like vertex with added connections). In the DE class of algorithms we have omitted the philosophy that a bad parent is replaced by a better offspring, but accepted philosophical interpretation, that individual (worse parent) is moving to the better possition (better offspring). Thus no vertex (individual) has to be either destroyed or replaced in the philosophical point of view. If, for example, DERand1Bin has a parent been replaced by offspring, then it was considered as an activation (new additional links, edges) of vertex-worse parent from three another vertices (randomly selected individuals, see [18]).

D. *Vizualization methods*

Experimental data can be vizualized in a few different ways and as an example, a few typical vizualizations is depicted here. For example in Figure 4 interactions between individuals in the population during entire evolution is described. As mentioned in the previous section, vertices in complex graph are individuals that are activated by other individuals, incrementally from generation to generation. This can be vizualized as in Figure 4, where DE and SOMA example populations are depicted. Different gray levels represent different number of inputs to vertex (different activations of selected individual). White color represent no relations (activations) between individuals, e.g. white square betweeen individual 6 (y axe) and 3 (x axe) means that individual 6 was never used to compete for the position in the new population or to create new offsprings with individual 2. Philosophy of competition or offspring creation is based on the principles of used algorithm (see previous subsection C).

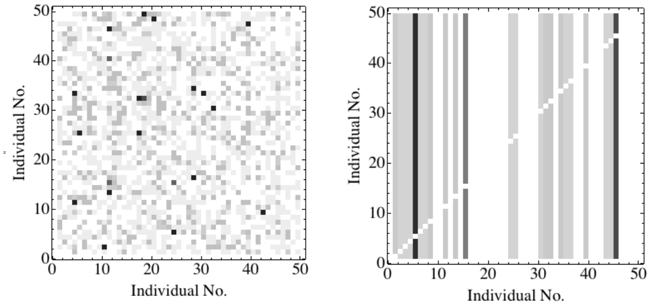


Figure 4. Table of interacion, DE (left) and SOMA (right). On both axes are individuals. White color means no interaction between individuals. See SOMA (right) where diagonal white line means that Leader individual cannot select himself. From SOMA interactions is clear that some individuals has newer been selected (white columns).

Another kind of vizualization is depicted in Figure 5, in which one can see which individual (out of 50) has been activated for offspring creation (in this case selected like Leader in SOMA). Information from Figure 5 is in close relation with Figure 4 – white columns from Figure 4. (unused individuals – never selected for Leaders) are empty rows (without dots) as shown in Figure 5.

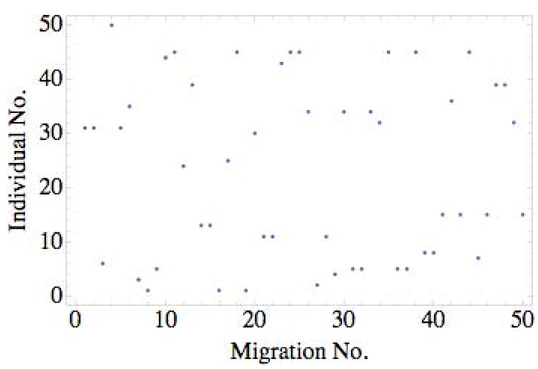


Figure 5. An example of activated leaders (y axe) with dependance on Migrations (x axe) SOMA. For example 31th individual (y axe) has been select for 33 times during 50 Migrations (x axe).

Figures 4 and 5 are sort of auxiliary visualizations, which does not give total view on complex network structure behind evolutionary dynamics. Better vizualization that can be used is as in Figure 6, 7 or 8, which shows, that interactions between individuls create (at the first glance) structures, which looks like complex networks. However, it has to be said, that we have met results whose visualizations (see section results) looks like net and resemble complex networks but after closer complex network characteristics calculations, those networks did not belong to the class of complex networks with small world phenomenon. Meaning of vertices in the above mentioned figures is given by ratio of incoming and outgoing edges and and inplies that: small vertex (small gray with dashed edges) has less incoming edges than outgoing. White (middle-sized) vertex is balanced (i.e. has the same incoming number of edges as outgoing) and dark gray, the biggest, are vertices with more incoming edges than outgoing. The light gray vertex is the most activated individual – vertex with the maximum of incoming edges. In EA jargon, small vertex is an individual, which has been used more times for offspring creation rather than as a succesful parent and dark (light) grey vertices reflects the opposite.

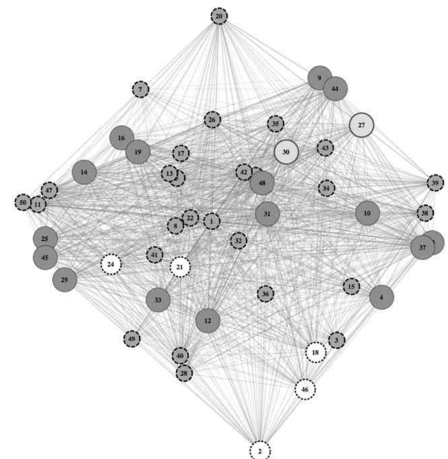


Figure 6. Complex network example of SOMA dynamics in a "natural" format.

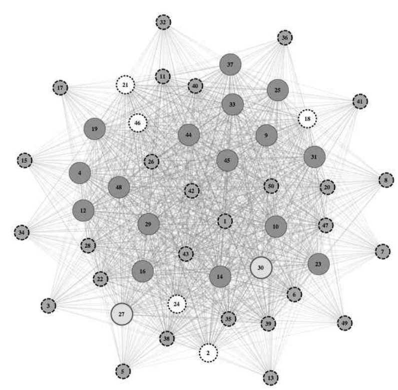


Figure 7. The same network as in Figure 6 in a "symmetrical" format.

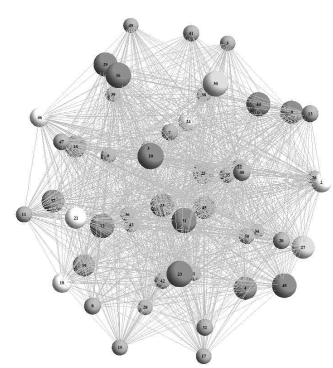


Figure 8. 3D vizualization of complex network depicted in Figure 6.

To ensure that an algorithm and its dynamics actually investigated for complex network phenomenon can be really understood and modelled like a complex network, typical characteristics has been calculated, like for example distributions of vertices degree, see Figure 10.

IV. Results

As reported above, all 6 algorithms has been tested on various test function (to reveal its complex networks dynamics) with different level of test function dimensionality (i.e. individual length) and different number of generations (migrations) in all used algorithms. All data has been processed graphically (e.g. Figure 10-12, etc.) alongside calculations of basic statistical properties. Emergence of complex network structure behind evolutionary dynamics depend on many factors, however some special versions of used algorithms did not show complex network structure despite the fact that the number of generations was quite large. All main ideas coming from the results are discussed in the next subsection.

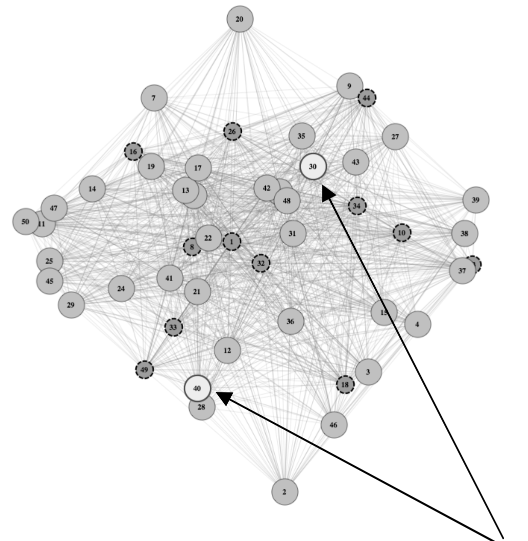


Figure 11. Complex network of the DELocalToBest with two the most intensively connected vertices (individuals)…

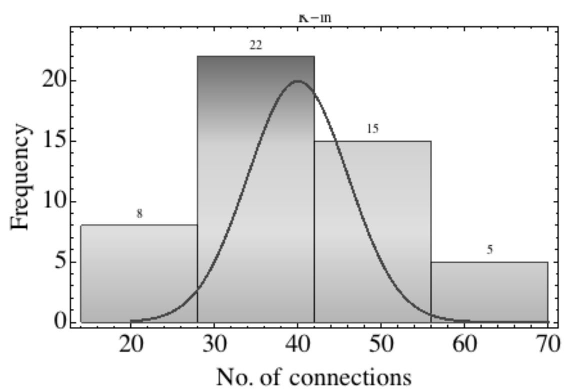


Figure 9. An example of DERand1Bin exhibiting normal distribution of vertices degree, no complex networks has been observed behind the evolutionary dynamics.

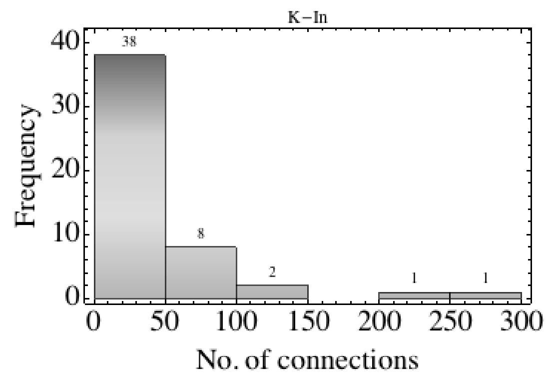


Figure 12. … and its histogram of the vertices connections (note that two vertices are with almost 300 connections each).

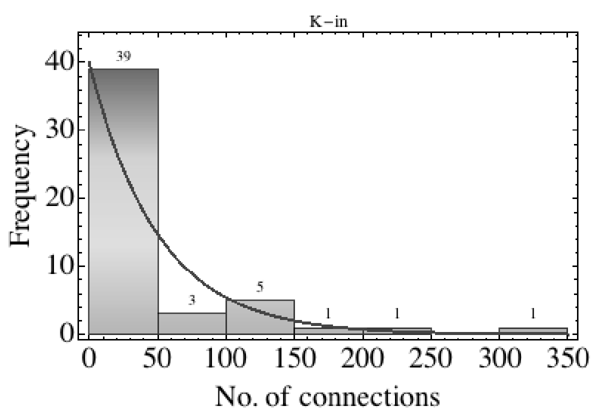


Figure 10. An example of histogram exhibiting long tail distribution of vertices degree, typical result for SOMA swarmlike algorithm.

V. Conclusion

The main motivation of this research is whether it is possible to visualize and simulate underlying dynamics of an evolutionary process as a complex network. Based on **preliminary** results (based only on 2 algorithms in 6 versions and 4 test function out of 17) it can be stated that:

- **No. of generations:** occurrence of the complex network structure (CNS) sensitively depends on the number of generations. If the number of generations was small, then no CNS was established. This is quite a logical observation in complex network dynamic when CNS is not observable at the beginning of linking process.
- **Dimensionality:** impact on CNS forming has been observed only when the number of generations was too low and the selected EA was not able to finish succesfully the global extreme search – not all connections had been properly established.

- **Test functions:** dependence of CNS forming on the test function was not strictly observed, the general consensus being that for more complex test functions, like Schwefel, etc, the algorithm needs more generations to establish CNS, i.e. more complex function requires more generations and/or bigger population size.
- **Population size:** CNS forming was observed usually from population size of 20 and more individuals; see previous point.
- **Used algorithm:** CNS forming has also been clearly observed with algorithms, that are more or less based on swarm philosophy or partly asociated with it. For example DERand1Bin did not show any CNS formating (in principle each individual is selected to be parent), see Figure 9, while in the case of the DELocalToBest (Figure 12) in which the best solution in the population play an important role, CNS has been observed, as well as in the SOMA strategies, see Figure 11. The conclusion reached is that CNS formating is more likely observable with swarm like algorithms rather than "randomly remoted" algorithms. We think that this is quite logical and close to the idea of prefered linking in the complex networks modelling social behavior (citation networks, etc…)

Results presented here shall be taken into consideration as "**preliminary**" findings, because more robust and statistically massive simulations are needed for more solid conclusions, which are currently in process. However, based on the results reported here, it can be stated, in general, that the dynamics of evolutionary algorithms can be vizualized as complex network structures and therefore the features of complex networks can probably be used to measure the efficiency of the used algorithm, or to control its dynamics in the future research.

Acknowledgment

This work was supported by grant No. MSM 7088352101 of the Ministry of Education of the Czech Republic and by grants of the Grant Agency of the Czech Republic GACR 102/09/1680.

References

[1] S. N. Dorogovtsev and J.F.F. Mendes, Evolution of Networks, Adv. Phys. 51, 1079 (2002)

[2] S. Boccaletti et al., Complex Networks: Structure and Dynamics, Phys. Rep., 424 (2006), 175-308.

[3] Turing, A.: Intelligent machinery, unpublished report for National Physical Laboratory. In: Michie, D. (ed.) Machine Intelligence, vol. 7 (1969); Turing, A.M. (ed.): The Collected Works, vol. 3, Ince D. North-Holland, Amsterdam (1992)

[4] Holland, J.: Adaptation in natural and artificial systems. Univ. of Michigan Press, Ann Arbor (1975)

[5] Schwefel, H.: Numerische Optimierung von Computer-Modellen, PhD thesis (1974); Reprinted by Birkhauser (1977)

[6] Rechenberg, I.: (1971) Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution (PhD thesis), Printed in Fromman-Holzboog (1973)

[7] Fogel,D.B.:UnearthingaFossilfromtheHistoryofEvolutionaryComputation.Fundamenta Informaticae 35(1-4), 1–16 (1998)

[8] Richter H. and Reinschke K. J., (2000), Optimization of local control of chaos by an evolutionary algorithm, Physica D, 144, pp. 309-334

[9] Richter H., An evolutionary algorithm for controlling chaos: The use of multi-objective fitness functions, in Parallel Problem Solving from Nature-PPSN VII (Eds.: Guervós M., Panagiotis J. J., Beyer A.,Villacanas F. H. G., J. L. & Schwefel H. P.), Lecture Notes in Computer Science, Vol. 2439 (Springer-Verlag, Berlin), (2002), pp. 308-317

[10] Richter H., Evolutionary Optimization in Spatio- temporal Fitness Landscapes, Lecture Notes In Computer Science, Springer-Verlag, NUMB 4193, pp. 1-10, (2006) ISSN 0302-9743

[11] Richter H., A study of dynamic severity in chaotic fitness landscapes, Evolutionary Computation, 2005. The IEEE Congress, Volume 3, Issue 2-5 Sept. 2005, (2005), pp. 2824 – 2831, Vol. 3

[12] Zelinka I., Chen G., Celikovsky S., Chaos Synthesis by Means of Evolutionary Algorithms, International Journal of Bifurcation and Chaos, University of California, Berkeley USA, Vol 18, No 4, (2008), p. 911 – 942

[13] Zelinka, I., Real-time deterministic chaos control by means of selected evolutionary algorithms, Engineering Applications of Artificial Intelligence, (2008), doi:10.1016/j.engappai.2008.07.008

[14] Zelinka I, Investigation on Realtime Deterministic Chaos Control by Means of Evolutionary Algorithms, In: 1st IFAC Conference on Analysis and Control of Chaotic Systems, Reims, France, (2006)

[15] Senkerik R., Zelinka I., and Navratil E., Optimization of feedback control of chaos by evolutionary algorithms, In: : 1st IFAC Conference on Analysis and Control of Chaotic Systems, Reims, France, (2006)

[16] Dashora, Y., et al., Improved and generalized learning strategies for dynamically fast and statistically robust evolutionary algorithms. Engineering Applications of Artificial Intelligence, (2007), doi:10.1016/j.engappai.2007.06.005

[17] Li L., Wenxin L., and David A. C.,, (2007), Particle swarm optimization-based parameter identification applied to permanent magnet synchronous motors, Engineering Applications of Artificial Intelligence, doi:10.1016/j.engappai.2007.10.002

[18] Price, K. An Introduction to Differential Evolution, New Ideas in Optimization, eds. Corne, D., Dorigo, M. & Glover, F. (McGraw-Hill, London, UK), pp. 79–108. (1999)

[19] Zelinka I., (2004), SOMA – Self Organizing Migrating Algorithm",Chapter 7, 33 p. In: B.V. Babu, G. Onwubolu (eds), New Optimization Techniques in Engineering, Springer-Verlag, ISBN 3-540-20167X