Differential Evolution Dynamics Analysis Using Aggregated Networks

Lenka SKANDEROVA[1, *], *Ivan ZELINKA*[1,2]

[1]Department of Computer Science, Faculty of Electrical Engineering and Computer Science, VSB Technical University of Ostrava, 17. listopadu 15/2172, 70800 Ostrava, Czech Republic
[2]Modeling Evolutionary Algorithms Simulation and Articial Intelligence, Faculty of Electrical an Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam

*lenka.skanderova@vsb.cz

(Received: 21-February-2017; accepted: 28-April-2017; published: 8-June-2017)
DOI: http://dx.doi.org/10.25073/jaec.201711.49

Abstract. *In this paper, the dynamics of the selected variants of the differential evolution is modelled by aggregated network capturing the relationships between individuals established during the population evolution. The motivation of this research is to better understand the relationships between individuals of the selected variants of the differential evolution. Thanks to the analysis of the aggregated networks, the advantages as well as bottlenecks of the selected algorithms can be specified more precisely and the results of the analysis can be used to develop novel algorithms.*

Keywords

Differential evolution, complex network, aggregated network, weighted clustering coefficient, node strength.

1. Introduction

Complex networks have become an effective tool to model real-world systems such as computer, social, epidemiology, telecommunication, biology, genetic regulatory, and metabolic networks, Internet, World Wide Web, Airline Routes etc. All of these systems consist of the huge number of highly interconnected dynamic units, where the connection between two units can be considered to be the certain form of communication. For example, in social networks, a connection between two users might express that these users are in relationship. In the food web, the connection between two organisms can be interpreted such that one organism is food of another one etc.

The evolutionary algorithms are the population-based meta-heuristics which belong to a great family of the artificial intelligence. They are inspired by the biological evolution and they are based on principles as reproduction, crossover, mutation and natural selection. The population consists of individuals. They are crossed, worse individuals die, better offspring takes place of their parents etc. We can say that there are tights between individuals – each offspring has its parents; to create a new offspring, several parents are selected (they are in relationship). Some individuals are used to improve the population, the others are useless from the perspective of a new offspring creation etc. The parameters of the individuals are changing, their fitness is improved etc. When we consider the difficult systems mentioned above, we can see some analogy. The population of individuals in the evolutionary algorithm can be considered to be a dynamical system of the interconnected units.

Differential evolution (DE) is well-known population based meta-heuristic introduced by

Storn and Price [1] in 1995 and it has been chosen for this research for several reasons: One of the most important reason is its good performance in comparison with other evolutionary and swarm algorithms [2], [3] and the fact that it has been successfully used in different areas of research, for example, system design [4], cybernetics [5], electromagnetic [6], clustering of large unlabeled data sets [7], neural network training [8] etc.

Various novel schemes of the differential evolution (DE) algorithm have been described in the last years. Among others, each scheme differs from each other in manipulation with individuals or in the control parameters settings, which offers some alternatives for our research. In this paper, the differential evolution is considered to be a system consisting of dynamical units which are highly interconnected. Different variants of the DE algorithm might mean a different form of interconnection between individuals in the population. The studies dealing with comparison of these variants from the perspective of the quality of provided solution, for example [9], indicate connection between the principle of individual selection to create a new offspring and the algorithm performance.

The rest of paper is organized as follows: In Sec. 2, the principle of the DE as well as the selected variants used for out experiments are described briefly. In Sec. 3, the idea to model the evolutionary algorithms dynamics using complex networks as well as related publications are mentioned. In Sec. 4, the idea to consider the network generated on the basis of the DE algorithm dynamics to be artificial social network is clarified. Sec. 5 deals with the aggregated networks and its analysis. In Sec. 6, the experiment is presented and in Sec. 7, the results are discussed.

2. Differential Evolution

The DE was described in 1995 [1] by Storn and Price and since 1996 it was demonstrated at the First International Contest on Evolutionary Optimization, where the DE won the third place for proposed benchmark. In 1997, Price presented DE at the Second International Contest on Evolutionary Optimization [10] and in 1999, the DE was summarized in the compendium New Ideas in Optimization [11], [12]. Price and Storn described various applications of the DE, for example, [13], [14] and [15] and their research has been early followed by many authors from all over the world [12].

The differential evolution belongs to a great family of evolutionary algorithms (EAs) whose principles are inspired by biological evolution such as mutation, crossover and natural selection. As well as the other EAs, the DE works with population of NP individuals which are represented by D-dimensional vectors $\vec{x}_i^G$ of real-valuated parameters. Each parameter is constrained by its search range, hence for each parameter j of vector $\vec{x}_i^G$ upper and lower bounds must be specified. These values can be collected into two D-dimensional vectors denoted as $\vec{b}_U \text{and} \vec{b}_L$, where U and L indicates the upper and lower bound, respectively. The initial population P_x^0 is then composed by vectors generated randomly in the prescribed range such that

$$x_{i,j}^0 = rand_j(0,1) \cdot (b_{j,U} - b_{j,L}) + b_{j,L} \tag{1}$$

where $rand_j(0,1)$ is the uniformly distributed random number within range [0,1] [16].

When a population is initialized, mutation and crossover are used to generate new vectors (trial vectors) and the selection determines which vector will be accepted to the next generation. In this part, the mutation step will be discussed. Unlike the genetic algorithm, DE performs mutation step before crossover operation such that for each target vector $\vec{x}_i^G$ a mutation vector (will be also denoted as a donor vector) is generated according to the following equation

$$\vec{v}_i^G = \vec{x}_{r_1}^G + F \cdot (\vec{x}_{r_2}^G - \vec{x}_{r_3}^G) \tag{2}$$

where $\vec{x}_{r_1}^G$, $\vec{x}_{r_2}^G$, and $\vec{x}_{r_3}^G$ $(r_1 \neq r_2 \neq r_3 \neq i)$ are solution vectors randomly selected from the current population and F denotes the scale factor. Randomly selected solution vectors must be different from each other and from a target vector i, so in this case, a population must consist at least four individuals. If a donor vector $\vec{v}_i^G$ contains parameters out of the predefined range, these parameters will be regenerated randomly in the space of possible solutions.

The main role of the crossover operation is to construct a trial vector $\vec{u}_i^G$using a combination of parameters of donor vector $\vec{v}_i^G$and target vector$\vec{x}_i^G$. There are two kinds of crossover methods – binomial and exponential. In this paper, the binomial crossover will be used.

At the beginning of the binomial crossover, a random integer j_{rn} ensuring that at least one parameter will be taken from a donor vector is generated from the interval$[0, D-1]$. A trial vector is then constructed such that for each from D parameters a random real number $rn(j)$ from the unit interval with the uniform distribution is generated and if the value of $rn(j)$ is not greater than value of the crossover rate CR or the index of the parameter j is equal toj_{rn}, the element from a donor vector is taken as a parameter of a trial vector. Otherwise, a parameter of a target vector is accepted [17]. The binomial crossover can be outlined as follows:

$$\vec{u}_{i,j}^G = \begin{cases} \vec{v}_{i,j}^G, \text{ if } rn(j) \leq CR \text{ or } j = j_{rn} \\ \vec{x}_{i,j}^G, \text{ otherwise} \end{cases}, \tag{3}$$

When a trial vector is constructed, it is evaluated. Then the selection operation is performed such that the objective function value of the trial vector $f(\vec{u}_i^G)$ is compared with the objective function value of the target vector$f(\vec{x}_i^G)$. If the objective function value of the trial vector is not worse than the objective function value of the target vector, the trial vector will survive to the next generation. In opposite case, the target vector will be accepted to the next generation. The selection operation is mathematically defined as follows:

$$\vec{x}_i^{G+1} = \begin{cases} \vec{u}_i^G, \text{ if } f(\vec{u}_i^G) \leq f(\vec{x}_i^G) \\ \vec{x}_i^G, \text{ otherwise} \end{cases}, \tag{4}$$

where $f()$ is the function to be minimized. In other words, if the objective function value of the trial vector is not greater than objective function value of the target vector, the trial vector will replace corresponding target vector at the next generation; otherwise, the target vector will be preserved. The trial vector is accepted to the next generation even in the case that the value of its objective function is equal to the objective function value of the target vector, which enables vectors to overcome flat objective function landscapes [18].

2.1. Selected Variants of Differential Evolution

Differential evolution is powerful and efficient evolutionary algorithm, however, there exist some bottlenecks whose solution is the subject of many researchers from all of the world. In the following text, the prominent DE variants, which have been used for the purpose to investigate the differences between the relationships established during the evolution process will be described briefly.

We have chosen four different variants of the differential evolution. Our choice has been dependent on the principles of the control parameters adaptation and on the mutation strategies used in these algorithms. Therefore, as the first algorithm, the original version of the DE (DE/rand/1/bin) has been selected. There is no adaptation of control parameters. The original DE algorithm uses only one mutation strategy rand/1, which has been described in the previous section, and the binomial crossover.

The second DE algorithm investigated in this work is the Self-adaptive DE denoted as jDE introduced by Brest et al. [19]. In the jDE algorithm, each individual in population has its own crossover rate CR_i^G and scale factorF_i^G. At the beginning, for each individual, the scale factor F_i^G is generated with the uniform distribution from the interval $[0.1, 1.0]$ and CR_i^Gtakes a value from $[0, 1]$. New control parameters F_i^{G+1} and CR_i^{G+1} for the individual $\vec{x}_i^G$ are then recomputed as follows:

$$F_i^{G+1} = \begin{cases} F_l + r_1 F_u, \text{ if } r_2 < \tau_1 \\ F_i^G, \text{ otherwise} \end{cases}, \tag{5}$$

$$CR_i^{G+1} = \begin{cases} r_3 \text{ if } r_4 < \tau_1 \\ CR_i^G, \text{ otherwise} \end{cases}, \tag{6}$$

where F_l and F_u are the lower and upper bounds of F_i , τ_1 and τ_2 denote probabilities to adjust factors F_i andCR_i, andr_j, where $j \in \{1,2,3,4\}$, is a uniform random value from the interval [0,1]. New control parameters F_i^{G+1} and $CR_i^{G+1}$of the individual $\vec{x}_i^{G+1}$ are obtained before the mutation operation is performed and thus they affect the mutation and crossover operations of the actual solution vector$\vec{x}_i^G$. We have chosen this algorithm because of its control parameters adaptation. The jDE algorithm uses the same mutation and crossover strategy as the original DE variant. Therefore we can investigate how the adaptation of the control parameters influences the relationships between individuals.

The third selected algorithm combines two mutation strategies on the basis of the exploration or exploitation requirement. Moreover, it uses the control parameters adaptation, which differs from the principle used in the jDE algorithm. This algorithm has been proposed by Yi et al. [20] and it is denoted as HSDE. At the beginning of the algorithm, the population is generated randomly in the space of possible solutions and for each individual the control parameters F_i and CR_i are generated randomly from within the feasible range. Then for each target vector a donor vector is generated such that five mutually different individuals are chosen randomly and if an objective function value of a target vector is worse than objective function values of the first two randomly selected individuals, the mutation strategy rand/1 will be used to generate a donor vector. Otherwise, mutation strategy current-to-best/1 will be used

The last selected algorithm is the JADE described by Zhang et al. [21] using the different principle of the control parameters adaptation and novel mutation strategy denoted as current-to-pbest/1. In this work, we will work with the JADE version with no external archive. In the strategy "current-to-pbest/1", where the external archive is not used, the donor vector $\vec{v}_i^G$ is generated as follows:

$$\vec{v}_i^G = \vec{x}_i^G + F_i \cdot (\vec{x}_{best}^{p,G} - \vec{x}_i^G) + F_i \cdot (\vec{x}_{r_1}^G - \vec{x}_{r_2}^G) \quad (7)$$

where $\vec{x}_{best}^{p,G}$ is randomly chosen as one of the top $100p\%$ individuals in the current population with $p \in (0,1]$ and F_i is the scale factor associated with $\vec{x}_i^G$. The scale factor is dynamically updated at each generation.

The adaptation of control parameters in the JADE algorithm can be described as follows: At each generation, for each solution vector, the scale factor is recomputed as follows:

$$F_i = randc_i(\mu_F, 0.1) \quad (8)$$

meaning that the scale factor $F_i$of each solution vector is generated according to Cauchy distribution with location parameter μ_F and scale parameter 0.1. If$F_i \geq 1$, it is truncated to be 1. If$F_i < 0$, it is regenerated. At the beginning of the algorithm, the location parameter $\mu_F$is set to be 0.5 and then updated at the end of each generation according to the following equation:

$$\mu_F = (1-c) \cdot \mu_F + c \cdot mean_L(S_F), \quad (9)$$

where $mean_L(\cdot)$is the Lehmer mean defined as

$$mean_L(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F}, \quad (10)$$

where S_F is the set of all successful scale factors F_i 's at the actual generation.

The crossover probability CR_i of each solution vector is generated according to normal distribution of mean μ_{CR} and standard deviation 0.1, which can be outlined as

$$CR_i = randn_i(\mu_{CR}, 0.1). \quad (11)$$

Then it is truncated to [0, 1]. The mean $\mu_{CR}$is recomputed at each generation as follows:

$$\mu_{CR} = (1-c) \cdot \mu_{CR} + c \cdot mean_A(S_{CR}), \quad (12)$$

where c is a positive constant between 0 and 1 and $mean_A(\cdot)$ denotes the arithmetic mean. $S_{CR}$is an archive of the successful crossover probabilities at the actual generation [21].

2.2. Evolutionary Algorithms Dynamics Represented by Complex Networks

Connection of the evolutionary algorithms and graph theory is not new. In 1999, Ashlock et al. [22] described a graph based genetic algorithm, where the combinatorial graphs are used to limit the choice of the partner for the crossover operation with the goal to preserve the diversity of population. In 2007, Mabu et al. [23] proposed a graph-based evolutionary algorithm denoted as the Genetic Network Programming (GNP), where the programs of the GNP consists of nodes, which are capable to execute simple judgment or processing.

On the other hand, in the history, the evolutionary algorithms have been successfully used to solve some problems relating with graphs and networks as the graph coloring [24], [25], community detection [26], [27], maximum clique problem [28], [29] etc.

In 2010, Zelinka et al. [30] introduced a preliminary study investigating relations between complex networks (CNs) and evolutionary algorithms dynamics. In this study, two versions of the DE algorithm and four versions of the Self-organizing migrating algorithm (SOMA) [31] have been selected and on the basis of their dynamics the complex networks have been generated such that the individuals have been modelled by nodes and the relationships between individuals by oriented lines (arcs) of the CN.

Main motivation of this research was to investigate whether it is possible to visualize the dynamics of the selected evolutionary algorithms by complex networks and how the analysis of such complex networks can be used to improve the selected EAs performance. On the basis of the experimental results authors concluded that the occurrence of the CN structure is depended on the number of individuals as well as on the number of generations (migrations). Authors also emphasized that the CN structure forming has been observed in algorithms *"based or partly associated with swarm philosophy rather than randomly remote algorithms"*. This study has been extended in 2011, where Zelinka [32] described, how the CN dynamics are visualized by means of chaos visualization techniques and then controlled by EAs or classical control techniques.

In 2014, Davendra et al. [33] presented results of the analysis of the complex networks generated on the basis of the dynamics of the Discrete Self-organizing Migrating Algorithm (DSOMA), where the CN attributes as minimal cut, degree centrality, closeness centrality, betweenness centrality, Katz centrality, mean neighbor degree, k-clique, k-plan, k-club, and graph of communities are investigated. These results confirmed that the analysis of the CNs generated on the basis of the DSOMA algorithm dynamics can be useful to obtain valuable information about population development. In the same year, Davendra et al. [34] used the complex networks to analyze the attributes of population dynamics of the Enhanced DE algorithm applied to the flow-shop scheduling with no-wait problem.

Recently, Janostik et al. [35] introduced a graph representation of the swarm evolution in the Particle Swarm Optimization (PSO) and the diversity measure based on this graph representation. In [36], the networks are used to capture inner dynamics of Firefly Algorithm and in [37], the networks are used to capture the inner dynamics of the PSO. As mentioned before, the goal of this work is to analyze the dynamics of the differential evolution algorithms using networks. The comparison of the selected DE algorithms from the perspective of the network structure might bring new information about the relationships between individuals, which can be later used to improve the properties of the selected DE variants. In this work, the DE algorithms have been selected on the basis of their mutation strategies and control parameters adaptation principles. Thanks to the various number of the DE algorithms, we can investigate how only the simple mechanism of the control parameters adaptation or the different mutation strategy affects the relationships between individuals in the population.

3. Differential Evolution Dynamics and Social Networks

Modeling of the evolutionary algorithms dynamics using complex networks has been described by Zelinka et al. [38]. As mentioned, main motivation was to better understand the relationships between individuals and apply this knowledge to improve the performance of the selected evolutionary algorithms. Differential evolution and Self-organizing migrating algorithm have become the first investigated algorithms.

In [38] and [39], Zelinka et al. proposed a new perspective on the new offspring creation in the DE algorithm, where a new solution vector generation is considered to be just an *activation* of a target vector to move to a better position. More precisely, if a trial vector $\vec{u}_i^G$ replaces a target vector $\vec{x}_i^G$at the next generation such that $x_i^{G+1} = u_i^G$, it will be considered to be an activation of a target vector $\vec{x}_i^G$by three solution vectors $\vec{x}_{r_1}^G$, $\vec{x}_{r_2}^G$, and $\vec{x}_{r_3}^G$ (selected randomly in the mutation operation to create a donor vector v_i^G) to move to a better position.

The activation of target vectors is modelled by a directed graph such that a target vector $\vec{x}_i^G$ and all three solution vectors $\vec{x}_{r_1}^G$, $\vec{x}_{r_2}^G$, and $\vec{x}_{r_3}^G$ are represented by nodes and there is a directed arc leading from each node representing a solution vector $x_{r_j}^G$, where $j = 1, 2, 3$ to a node representing a target vector $\vec{x}_i^G$, see Fig. 1. For each generation, one directed graph is created.

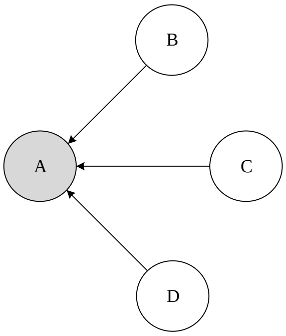


Fig. 1: Activation of target vector $\vec{x}_i^G$(represented by node$A$) by the solution vectors$\vec{x}_{r_1}^G$,$\vec{x}_{r_2}^G$ and $\vec{x}_{r_3}^G$(represented by nodes B, $C$and$D$) to move to a better position

When we consider the process of new offspring creation, we can say that there is a relationship between target and donor vector, or between a target vector and trial vector. However, we can also take into account the relationship between target vector and solution vectors, which have been chosen randomly to create donor vector etc. Moreover, there can be more than one type of the relationship between two individuals. When we look at the individuals in the population as at the artificial social units, we can try to use the social networks to model the dynamics of the DE algorithms and the social network analysis methods to analyze such networks.

The main idea of this section is that the individual in a population can be considered to be a *social unit.* Social unit can be anything – pupil in a class, a class at a school, or a school in a city. As well as other social units, the individuals in a population have their own properties – parameters and fitness. These properties are in the context of the social network analysis very important. For example we can investigate if there is some connection between the individual with the best fitness value and the node with the highest out-degree.

Formally, a social network can be represented by a directed or undirected graph $G = \{N, L\}$, where N denotes the set of nodes and L the set of lines between pairs of nodes. The set of nodes represents the set of actors $N = (n_1, n_2, \ldots n_g)$. Relation between two actors in a social network is usually drawn by a line connecting two nodes in a graph. This line can be directed, in such case we talk about *arc,* or undirected, then we talk about *edge* [40]. In the case of the directed graph, the maximum number of arcs is $g \cdot (g-1)$, where g denotes the number of nodes. In the undirected graph, the set L contains at most $g \cdot (g-1)/2$ lines.

In this work, two possibilities how to represent the evolution process by a social network have been taken into account. Considering the fact that the DE algorithms work in generations, it is possible to create one directed network for each generation. These networks will be in the following text denoted as the *short-interval networks.* Thanks to this mechanism, the influence of the solution vectors (selected to create a donor vectors) on the evolution of target vectors can be investigated. However, this principle does not enable to analyze the influence of the individual on the evolution of the whole population during the algorithm. This problem can be solved by using so called *aggregated network*, which is the accumulation of the short-interval networks, see Fig. 2.

Beside the directed graphs, the undirected graphs, which would represent the short-interval networks, have been under consideration such that in the case of the successful trial vector creation, the undirected edge would be created between each pair of nodes representing the participating individuals. We can say that the individuals participating in the successful trial vector generation would be considered to be the "authors" of the offspring. The networks generated on the basis of this principle could be analyzed as the co-authorship network. However, in the case of the network generated in this way, we could easily loose the information about the "role" of the individual represented by a node. In other words, we would not be capable to visually distinguish between nodes representing solution vectors selected in the mutation operation to create a donor vector (activators) and nodes representing target vectors, which have been replaced by better offspring (activated individuals). For this reason, we have decided to model the dynamics of the DE algorithms as suggested in [30].

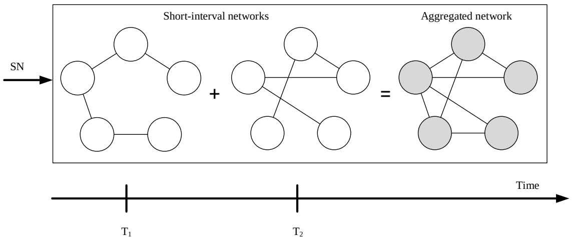


Fig. 2: Illustration of the short-interval networks and aggregated network [41].

4. Aggregated Networks

In this work, the relationships between individuals, which have been established during the population evolution from the beginning till the end of the DE algorithm will be analyzed using aggregated networks. As mentioned before, the aggregated network is the accumulation of the SINs, which means that the analysis of the sequence of the SINs is replaced by the analysis of the one network.

Consider the principle of the DE algorithm and the settings of the parameters, where the number of generations is set to $G = 3000$ and the number of individuals to $NP = 100$. Because there is a relatively small number of individuals and a large number of generations, there is a high probability that during the population evolution, two individuals i and j will be in relation more than once. Therefore the intensity of the relation between two individuals will be expressed by the weight of the arc. The networks will be created by the individuals participating in the evolution of the population (activators and activated individuals) as suggested in [38], [39], for better illustration see Fig. 1 and Fig. 3. This means that individuals, which have not been replaced by better offspring or they have never been used to create better offspring will not be taken into consideration.

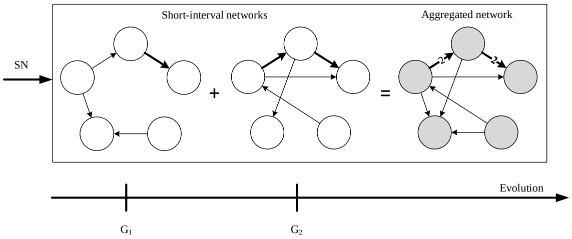


Fig. 3: Illustration of the aggregated network creation on the basis of the DE algorithms [41].

5. Experiment

The motivation of this experiment is to investigate the relationships between individuals established from the beginning till the end of the DE algorithm. We are especially interested in the differences between the selected characteristics of the networks generated on the basis of the different DE algorithms dynamics.

The aggregated networks are generated on the basis of the four state-of-the-art DE algorithms, which have been used to optimize the difficult test functions from the benchmark set CEC'2013 Special session on real-parameter optimization [42]. For this experiment, two unimodal (f_1, f_5) one multimodal (f_6), and one composition (f_{21}) function has been chosen.

The detailed parameters settings for the DE variants is given as follows: The population size has been set to $NP = 100$ for all algorithms and benchmark function, the dimension has been set to $D = 30$ and the number of generation to $G = 3000$ which corresponds with the 300,000 objective function evaluations. This is in compliance with the suggested values of the number of the objective function evaluations mentioned in [42].

In the DE/rand/1/bin, the scale factor has been set to $F = 0.5$ and crossover rate to$CR = 0.9$. In the case of the jDE algorithm,$\tau_1 = \tau_2 = 0.1$ and$F \in [0.1, 1.0]$. The same settings of the lower and upper bound of the scale parameter has been chosen for the HSDE algorithm. For the JADE algorithm, the number of the best individuals, which are selected in the mutation operation has been set to$p = 5$. We do not use any archive, so the number of individuals in the archive$A = 0$. The second parent is selected only from the population, not from the union of the population and archive. At the beginning of the algorithm$\mu_F = \mu_{CR} = 0.5$. The parameter c has been set to$c = 0.1$.

The algorithms have been implemented in C#, Microsoft .NET Framework 4.5.1 and run on a computer with CPU Intel Xeon 2.83~GHz. To analyze the networks generated on the basis of the DE algorithms dynamics, the software UCINET 6.0 and Wolfram Mathematica 10.0 have been used.

5.1. Investigated Characteristics

Due to the high number of generations ($G = 3000$), the aggregated networks generated on the basis of the dynamics of the DE algorithms are represented by complete graphs (where each node is connected with each other). Therefore, in this experiment, we are specialized in two characteristics of the aggregated networks providing valuable information about the relationships between individuals, which have been established during the population evolution, namely the weighted clustering coefficient and node out-strength distribution.

The weighted clustering coefficient reflects the strength of the relationships between individuals as well as the tendency of the DE algorithm to create tightly interconnected groups of individuals and in this work, it is calculated on the basis of the work of Fagiolo [43].

The node out-strength distribution indicates how often the individuals are used to create the successful offspring. In this part, the node out-strength distribution is analyzed to find out, whether in the case of the DE algorithms, where the solution vectors are selected randomly (DE/rand/1/bin, jDE), some individuals are used more often to create a successful trial vector than the others. On the other hand, in the networks generated on the basis of the HSDE and JADE algorithms, where the best individuals are crucial for the new offspring creation, we are investigated, whether the phenomenon "rich becomes to be richer" is occurred and when.

5.2. Experimental Results

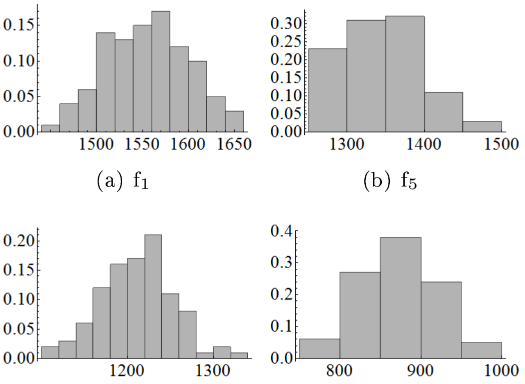
In Tab. 1, the weighted clustering coefficients of the aggregated networks generated on the basis of the four different variants of the DE algorithm dynamics are mentioned. In Figs. 4 a), b), c) and d), the out-strength distribution in the aggregated networks generated on the basis of the DE/rand/1/bin algorithm dynamics is depicted. Figs. 5 a), b), c) and d) present the out-strength distribution in the aggregated networks generated on the basis of the dynamics of the jDE algorithm. In Figs. 6 a), b), c) and d), the node out-strength distribution in the aggregated networks generated by the basis of the dynamics of the HSDE algorithm are depicted. And in Fig. 7 a), b), c) and d), the out-strength distribution in the aggregated networks generated on the basis of the dynamics of the JADE algorithm is shown. In Tab. 2, the results of the Shapiro-Wilk's test of normality of the out-strength distribution at the significance level $\alpha = 0.05$ are mentioned.

Tab. 1: Weighted clustering coefficients of the aggregated networks generated on the basis of the DE algorithms.

	DE	jDE	HSDE	JADE
f_1	15.716	59.563	78.074	79.127
f_5	13.589	50.848	71.939	72.121
f_6	12.232	10.483	34.282	42.992
f_{21}	8.810	51.091	78.045	69.766

Tab. 2: The results of the Shapiro-Wilk's test (p-values) of the out-strength distribution at significance level$\alpha = 0.05$.

	DE	jDE	HSDE	JADE
f_1	0.774	0.886	0.000	0.000
f_5	0.037	0.943	0.000	0.000
f_6	0.996	0.000	0.455	0.712
f_{21}	0.679	0.429	0.000	0.000



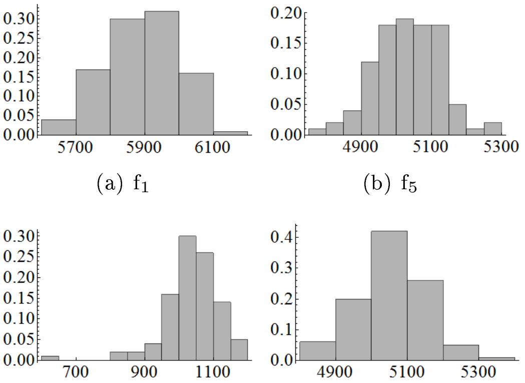
(a) f_1

(b) f_5

(c) f_6

(d) f_{21}

Fig. 4: The node out-strength distribution in the four aggregated networks generated on the basis of the DE/rand/1/bin dynamics.



(a) f_1

(b) f_5

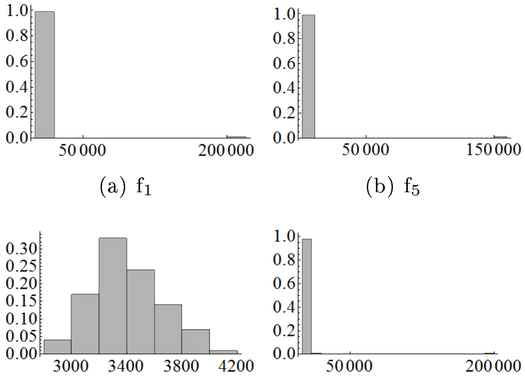
(c) f_6

(d) f_{21}

Fig. 5: The node out-strength distribution in the four aggregated networks generated on the basis of the jDE dynamics.

6. Discussion and Conclusion

In Tab. 1, the weighted clustering coefficients of the aggregated networks generated on the basis of the four DE variants are mentioned. As we can see, there is a significant difference between the weighted clustering coefficients of aggregated networks generated on the basis of the dynamics of the DE/rand/1/bin and jDE algorithms and the weighted clustering coefficients of aggregated networks generated on the basis of the dynamics of the HSDE and JADE algorithms.



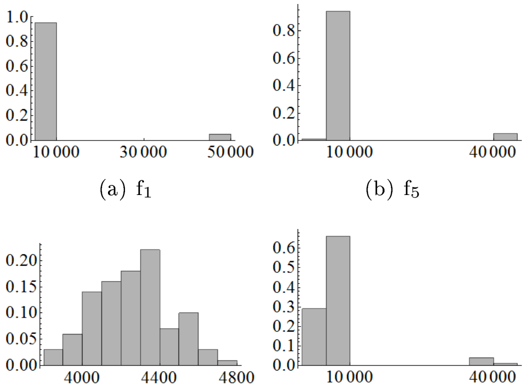
(a) f_1

(b) f_5

(c) f_6

(d) f_{21}

Fig. 6: The node out-strength distribution in the four aggregated networks generated on the basis of the HSDE dynamics.



(a) f_1

(b) f_5

(c) f_6

(d) f_{21}

Fig. 7: The node out-strength distribution in the four aggregated networks generated on the basis of the JADE dynamics

In the case of the aggregated networks generated on the basis of the HSDE and JADE algorithms dynamics, the weighted clustering coefficients are significantly higher in comparison with the weighted clustering coefficients of aggregated networks generated on the basis of the DE/rand/1/bin and jDE algorithms dynamics. This is given by mutation strategies used in the HSDE and JADE algorithms as well as by relatively large number of successful trial vectors generated by these DE algorithms.

When we look at the results of the weighted clustering coefficients of the networks generated on the basis of the DE/rand/1/bin and jDE algorithms, we will see that except the test function f_6, the jDE algorithm has generated networks with the significantly higher weighted clustering coefficient that the DE/rand/1/bin. Because the jDE algorithm uses the same mutation strategy as the DE/rand/1/bin, we can conclude that the principle of the control parameters adaptation used in the jDE affects the relationships between individuals such that there is higher tendency to create tightly interconnected groups. However, the comparison of the weighted clustering coefficients of the aggregated networks generated on the basis of the jDE and the HSDE and JADE algorithms dynamics indicates the limited impact of the control parameters adaptation used in the jDE algorithm. The results of the weighted clustering coefficients analysis have shown negative consequences of the mutation strategy rand/1 to the establishment of the relationships between individuals in the population. On the other hand, the positive impact of the mutation strategies using the best individuals, control parameters adaptation as well as combination of more mutation strategies have been confirmed.

In Figs. 4–7, the out-strength distributions in the aggregated networks generated on the basis of the four DE variants are presented. In Tab. 2, the results of the Shapiro-Wilk's test of normality are mentioned. The node out-strength of the aggregated networks generated on the basis of the dynamics of the DE/rand/1/bin follows normal distribution in the case of the test functions f_1, f_6, and f_{21}. The node out-strength of the aggregated networks generated on the basis of the jDE algorithm dynamics follows normal distribution in the case of the test functions f_1, f_5, and f_{21}. As we can see, the out-strength of the aggregated networks generated on the basis of the DE/rand/1/bin and jDE algorithm is normally distributed in three from four test functions.

On the other hand, the node out-strength of the aggregated networks generated on the basis of the dynamics of the HSDE and JADE algorithms follows normal distribution only in the case of the multimodal test function f_6. In the case of the rest of test functions, the phenomenon "rich becomes to be richer" can be observed, which means that several individuals are dominant from the perspective of the new offspring creation. They are used to create the successful trial vectors most often, however, they are also replaced by better offspring (they are activated to move to better position). In the aggregated networks, these individuals are represented by so called *hubs* – nodes having significantly larger number of connections than the others. On the other hand, the node out-strength distribution in the aggregated networks generated on the basis of the HSDE and JADE algorithms to optimize the test function f_6 indicates that there are more individuals selected as the best individuals during the evolution.

Based on the results discussed above, we have concluded that the mutation strategy rand/1 and the absence of the control parameters adaptation negatively influence the relationships between individuals in the population. We have shown that the simple principle of the control parameters adaptation can positively influence the tights between individuals. Moreover, the results of the analysis of the aggregated networks indicate the positive influence of the mutation strategy current-to-best/1 as well as current-to-pbest/1, where the best individuals are crucial for the successful trial vector creation. These individuals are responsible for the generation of the larger number of successful trial vectors.

These results have become the stepping stone for our work, where the weighted clustering coefficient and the strength of nodes representing the individuals in the population are used to influence the selection of the individuals in the mutation step.

Acknowledgment

The following grants are acknowledged for the financial support provided for this research:

Grant Agency of the Czech Republic - GACR P103/15/06700S, Grant of SGS No. SGS 2017/134, VSB-Technical University of Ostrava. The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPUII) project "IT4 Innovations excellence in science - LQ1602"

References

[1] STORN, R. and K. Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, vol. 11, no. 4: 341-359, 1997.

[2] VESTERSTROM, J. and R. THOMSEN. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In: *Evolutionary Computation, 2004. CEC2004. Congress on.* IEEE, p. 1980-1987, 2004.

[3] PANDURO, M., et al. A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scannable circular antenna arrays. *Progress In Electromagnetics Research B*, 13: 171-186, 2009

[4] STORN, R. System design by constraint adaptation and differential evolution. *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, 22-34, 1999.

[5] JOSHI, R., A. C. SANDERSON. Minimal representation multisensor fusion using differential evolution. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 29, no. 1, 63-76, 1999.

[6] ROCCA, P., G. OLIVERI and A. MASSA. Differential evolution as applied to electromagnetics. *IEEE Antennas and Propagation Magazine*, vol. 53, no. 1, 38-49, 2011.

[7] DAS, D., A. ABRAHAM and A. KONAR. Automatic clustering using an improved differential evolution algorithm. *IEEE Transactions on systems, man, and cybernetics-Part A*, *Systems and Humans*, vol. 38, no. 1, 218-237, 2008.

[8] CHAUHAN, N. and V. RAVI, D. K. CHANDRA. Differential evolution trained wavelet neural networks: Application to bankruptcy prediction in banks. *Expert Systems with Applications*, vol. 36, no. 4, 7659-7665, 2009.

[9] MEZURA, M. E., J. V. REYES and C. A. COELLO. A comparative study of differential evolution variants for global optimization. *In: Proceedings of the 8th annual conference on Genetic and evolutionary computation.* ACM, 485-492, 2006.

[10] STORN, R. and K. PRICE. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, vol. 11, no. 4: 341-359, 1997.

[11] PRICE, K. An introduction to differential evolution. *New ideas in optimization.* 1999.

[12] VITALITY, F. Differential evolution–in search of solutions. 2006.

[13] STORN, R. Differential evolution design of an IIR-filter. In: Evolutionary Computation. *Proceedings of IEEE International Conference on. IEEE*, 268-273, 1996.

[14] STORN, R. System design by constraint adaptation and differential evolution. *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, 22-34., 1999

[15] STORN, R. Designing digital filters with differential evolution. *In: New ideas in optimization.* McGraw-Hill Ltd., UK, p. 109-126, 1999.

[16] PRICE, K., R. STORN and J. A. LAMPINEN. Differential evolution: a practical approach to global optimization. Springer Science & Business Media, 2006.

[17] A. QIN, V. L. HUANG and P. N. SUGANTHAN. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on*

Evolutionary Computation, vol. 13, no. 2, 398-417, 2009.

[18] DAS, S. and P. N. SUGANTHAN. Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, vol. 15, no. 1, 4-31, 2011.

[19] BREST, J. et al. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation*, vol. 10, no. 6, 646-657, 2006.

[20] YI, W. et al. A new differential evolution algorithm with a hybrid mutation operator and self-adapting control parameters for global optimization problems. *Applied Intelligence*, vol. 42, no. 4, 642-660, 2015.

[21] J. ZHANG and A. C. SANDERSON. JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on evolutionary computation*, vol. 13, no. 5, 945-958, 2009.

[22] ASHLOCK, D., M. SMICKER and J. WALKER. Graph based genetic algorithms. In: Evolutionary Computation, 1999. CEC 99. *Proceedings of the 1999 Congress on. IEEE*, p. 1362-1368, 1999.

[23] MABU, S., K. HIRASAWA and J. HU. A graph-based evolutionary algorithm: Genetic network programming (GNP) and its extension using reinforcement learning. *Evolutionary Computation*, vol. 15, no. 3, 369-398, 2007.

[24] EIBEN, A. E., J. K. HAUW and J. I. HEMERT. Graph coloring with adaptive evolutionary algorithms. *Journal of Heuristics*, vol. 4, no. 1, 25-46, 1998.

[25] FLEURENT, C. and J. A. FERLAND. Genetic and hybrid algorithms for graph coloring. *Annals of Operations Research*, vol. 63, no. 3, 437-461, 1996.

[26] PIZZUTI, C. Ga-net: A genetic algorithm for community detection in social networks. *In: International Conference on Parallel Problem Solving from Nature*. Springer Berlin Heidelberg, 1081-1090., 2008

[27] GONG, M., et al. Community detection in networks by using multiobjective evolutionary algorithm with decomposition. *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 15, 4050-4060, 2012.

[28] MARCHIORI, E. A simple heuristic based genetic algorithm for the maximum clique problem. *In: Symposium on Applied Computing: Proceedings of the 1998 ACM symposium on Applied Computing*, 366-373, 1998

[29] ZHANG, Q., J. SUN and E. TSANG. An evolutionary algorithm with guided mutation for the maximum clique problem. *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 2: 192-200, 2005.

[30] ZELINKA, I., et al. Preliminary investigation on relations between complex networks and evolutionary algorithms dynamics. *In: Computer Information Systems and Industrial Management Applications (CISIM), 2010 International Conference on. IEEE*, 148-153, 2010.

[31] ZELINKA, I. SOMA—self-organizing migrating algorithm. *In: New optimization techniques in engineering*. Springer Berlin Heidelberg, 167-217, 2004.

[32] ZELINKA, I. Investigation on relationship between complex networks and evolutionary algorithms dynamics. *In: SIMOS, Theodore E., et al. (ed.). AIP Conference Proceedings. AIP*, 1011-1014, 2011.

[33] DAVENDRA, D. D., et al. Complex network analysis of evolutionary algorithms applied to combinatorial optimization problem. *In: Proceedings of the Fifth International Conference on Innovations in Bio-Inspired Computing and Applications IBICA 2014*. Springer International Publishing, 141-150, 2014.

[34] DAVENDRA, D. D. and G. ONWUBOLU. Enhanced differential evolution hybrid scatter search for discrete optimization. *In: Evolutionary Computation, 2007. CEC 2007*. IEEE Congress on. IEEE, 1156-1162, 2007.

[35] JANOSTIK, J., et al. Particle swarm optimizer with diversity measure based on swarm representation in complex network. *In: Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015.* Springer International Publishing, 561-569, 2016.

[36] JANOSTIK, J., et al. Capturing inner dynamics of firefly algorithm in complex network—initial study. *In: Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015.* Springer International Publishing, 571-577, 2016.

[37] PLUHACEK, M., et al. PSO as complex network—capturing the inner dynamics—initial study. *In: Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015.* Springer International Publishing, 551-559, 2016.

[38] I. Zelinka et al. Do evolutionary algorithms dynamics create complex network structures? *Complex Systems*, vol. 20, no. 2, 127, 2011.

[39] ZELINKA, I., D. D. DAVENDRA and L. SKANDEROVA. Visualization of complex networks dynamics: Case study. *In: International Conference on Research in Networking.* Springer Berlin Heidelberg, 145-150, 2012.

[40] NOOY, W. D., A. MRVAR and V. BATAGEJL. Exploratory social network analysis with Pajek. Cambridge University Press, 2011.

[41] UDDIN, S., A. KHAN and M. PIRAVEENAN. A set of measures to quantify the dynamicity of longitudinal social networks. Complexity, 2015.

[42] LIANG, J. J., et al. Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report, 201212, 2013.

[43] FAGIOLO, G. Clustering in complex directed networks. *Physical Review E*, vol. 76, no. 2, APS, 2007. DOI: 10.1103/PhysRevE.76.026107

About Authors

Lenka SKANDEROVA is Ph.D. student at the Department of Computer Science, Faculty of Electrical Engineering and Computer Science, VSB – Technical University of Ostrava.

Her research is focused on the analysis of the evolutionary algorithms using complex networks, artificial intelligence and chaos. Currently, she is preparing for the defense of her doctoral thesis.

Ivan ZELINKA is currently associated with the Department of Computer Science of the Technical University of Ostrava. He graduated consequently at the Technical University in Brno (1995, MSc.), UTB in Zlín (2001 Ph.D.) and again at Technical University in Brno (2004, Assoc. Prof.) and VSB-Technical University of Ostrava (2010, Professor). Prof. Zelinka is responsible supervisor of several grant researches of Czech grant agency GAČR as for example *Unconventional Control of Complex Systems*, *Security of Mobile Devices and Communication* (bilateral project between Czech and Vietnam) and co-supervisor of grant FRVŠ - Laboratory of parallel computing amongst the others. He was also working on numerous grants and two EU projects as member of team (FP5 - RESTORM) and supervisor (FP7 - PROMOEVO) of the Czech team. He is also head of research team NAVY http://navy.cs.vsb.cz/.

He is a member of the British Computer Society, Machine Intelligence Research Labs (MIR Labs, http://www.mirlabs.org/czech.php), IEEE (committee of Czech section of Computational Intelligence), a few international program committees of various conferences, and three international journals. He is also the founder and editor-in-chief of a new book series entitled *Emergence, Complexity and Computation* (Springer series 10624, see also www.ecc-book.eu).