STUDY ON SWARM DYNAMICS CONVERTED INTO COMPLEX NETWORK

[1]Michal Pluhacek, [1]Roman Senkerik, [1]Jakub Janostik, [1]Adam Viktorin and [2]Ivan Zelinka

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

[2]Department of Computer Science, Faculty of Electrical Engineering and Computer Science
VSB-TUO, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
ivan.zelinka@vsb.cz

KEYWORDS
Swarm Intelligence, Particle Swarm Optimization, Firefly Algorithm, Complex Network

ABSTRACT

In this study it is presented a summarization of our research of possible ways of creating of complex networks from the inner dynamics of Swarm Intelligence based algorithms. The particle swarm optimization algorithm and the firefly algorithm are studied in this paper. Several methods of complex network creation are proposed and discussed alongside with possibilities for future research and application.

INTRODUCTION

The Particle Swarm Optimization (PSO) (Kennedy, Eberhart 1995, Shi, Eberhart, 1998, Kennedy 1997, Nickabadi et al., 2011) and Firefly algorithm (Yang, 2008, 2009, 2013, Tilahun, 2012) are among the most prominent members of Swarm Intelligence based algorithms. These evolutionary computational techniques (ECTs) are in recent years in the center of interest of the research community. Recently the links between ECTs and complex networks (CNs) has been studied (Zelinka 2011a, 2011b, 2013).
In this study it is presented the possibilities of successful CNs creation from two swarm algorithms. Despite that the algorithms do differ the created networks seem to share similarities and in future various statistical methods may be used in order to gather information about the otherwise hidden inner dynamic of the swarm algorithms. The complex networks have many unique attributes that may help to understand and analyze the inner dynamic of Swarm algorithms. The goal is to use gathered knowledge to improve the performance of the optimization method. The usefulness of such approach was already shown in (Davendra, 2014a, 2014b).

In this study a methodology for complex network creation for PSO and Firefly Algorithm is presented. The rest of the paper is structured as follows: In the next section the PSO algorithm is described. Following is the description of Firefly Algorithm. The experimental details alongside with methodology for CN creation and first visualizations are given in following two sections. Afterwards the conclusions are presented.

PARTICLE SWARM OPTIMIZATION

The Particle Swarm Optimization algorithm (PSO) is the evolutionary optimization algorithm based on the natural behavior of bird and fish swarms and was firstly introduced by R. Eberhart and J. Kennedy in 1995 (Kennedy, Eberhart 1995). PSO proved itself to be able to find better solutions for many optimization problems.
In the PSO algorithm the particles move trough the multidimensional space of possible solutions. The new position of the particle in the next iteration is then obtained as a sum of actual position and velocity. The velocity calculation follows two natural tendencies of the particle: To move to the best solution found so far by the particular particle (known in the literature as personal best: *pBest* or local best: *lBest*). And to move to the overall best solution found in the swarm or defined sub-swarm (known as global best: *gBest*)..

In the original PSO the new position of particle is altered by the velocity given by Eq. 1:

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

Where:
v_i^{t+1} - New velocity of the *i*th particle in iteration *t*+1.
w – Inertia weight value.
v_i^t - Current velocity of the *i*th particle in iteration *t*.
c_1, c_2 - Priority factors (set to the typical value = 2).
$pBest_i$ – Local (personal) best solution found by the *i*th particle.
gBest - Best solution found in a population.
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). The chaotic pseudo-random number generator is applied here.

The maximum velocity of particles in the PSO is typically limited to 0.2 times the range of the optimization problem and this pattern was followed in

Proceedings 30th European Conference on Modelling and Simulation ©ECMS Thorsten Claus, Frank Herrmann, Michael Manitz, Oliver Rose (Editors)
ISBN: 978-0-9932440-2-5 / ISBN: 978-0-9932440-3-2 (CD)

this study. The new position of a particle is then given by Eq. 2, where x_i^{t+1} is the new particle position:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{2}$$

Finally the linear decreasing inertia weight (Nickabadi et al., 2011). is used in the PSO here. Its purpose is to slow the particles over time thus to improve the local search capability in the later phase of the optimization. The inertia weight has two control parameters w_{start} and w_{end}. A new w for each iteration is given by Eq. 3, where t stands for current iteration number and n stands for the total number of iterations.

$$w = w_{start} - \frac{\left(\left(w_{start} - w_{end}\right)\cdot t\right)}{n} \tag{3}$$

FIREFLY ALGORITHM

Firefly algorithm was first presented by Xin-She Yang in at Cambridge University (Yang, 2008, 2009). FA is based on simplified behavior of fireflies in night. Following rules were established to describe mentioned behavior (Yang, 2008, 2009, 2013, Tilahun, 2012):

1. All fireflies are unisex so that fireflies will attract each other regardless of their sex.
2. The attractiveness is proportional to the brightness, and they both decrease as their distance increases. This means that for any two flashing fireflies, the less bright one will move towards the brighter one. Firefly will move randomly if there is no brighter one.
3. The brightness of a firefly is determined by the landscape of the objective function.

Firefly's attractiveness is determined by its light intensity, which is proportional to the encoded objective function. The brightness $I(r)$ varies with the distance r monotonically and exponentially Eq. 4. That is,

$$I(r) = \frac{I_0}{1 + \lambda r^2}, \tag{4}$$

,where I_0 is the initial brightness and λ is the light absorption coefficient. Similarly, the attractiveness of a firefly can be defined using following formula Eq. 5:

$$A(r) = \frac{A_0}{1 + \lambda r^2}, \tag{5}$$

,where A_0 is the initial attractiveness.
If a firefly located at $\acute{x} = (\acute{x}_1, \acute{x}_2, \dots, \acute{x}_n)$ is brighter than firefly located at $x = (x_1, x_2, \dots x_n)$, the firefly located at x will move towards one located at $\acute{x}$.

The algorithm can be summarized as follows:

1. Generate a random solution set $\{x_1, x_2, \dots x_n\}$.
2. Compute intensity $\{I_1, I_2, \dots I_n\}$ for each member of solution set.
3. Move firefly towards other brighter firefly if possible, move it randomly if not.
4. Update solution set.
5. If a termination criterion is fulfilled terminate algorithm. Otherwise go to step 2.

PSO EXPERIMENTS

During the experiments several different ways of complex network creation and visualization were tested. The goal was to capture the inner dynamics of swarm algorithms in a sufficient detail but in a network of appropriate size for further processing.

In the first experiment the PSO with typical defaults setting was used to optimize the commonly used Schwefel's benchmark function for 100 iterations with population size set to 30.
In this experiment the main interest was in the communications that leads to population quality improvement. Therefore only communication leading to improvement of the particles personal best (pBest) was tracked. The link was created between the particle that has improved and particle that triggered the current gBest's update.
In Figure 1 the created complex network is visualized. Nodes of similar color represent particles with same ID during different iterations. All links are from particle that triggered gBest update to particle that has improved based on that gBest.
In Figure 2 a zoomed partial view of the network is presented. It is possible to clearly see the density of the network and links of various lengths.
Close look on a single cluster in the network is presented in Figure. 3. The numbers in nodes represent a code for a particle ID and current iteration. That way it is possible to track exactly the development of the network and the communication that happens within the swarm. On this example cluster it can be observed a single gBest update led to improvement of multiple particles in different iterations.
A different visualization method was used in Figure. 4 where a smaller network is depicted. Both networks share many similarities.

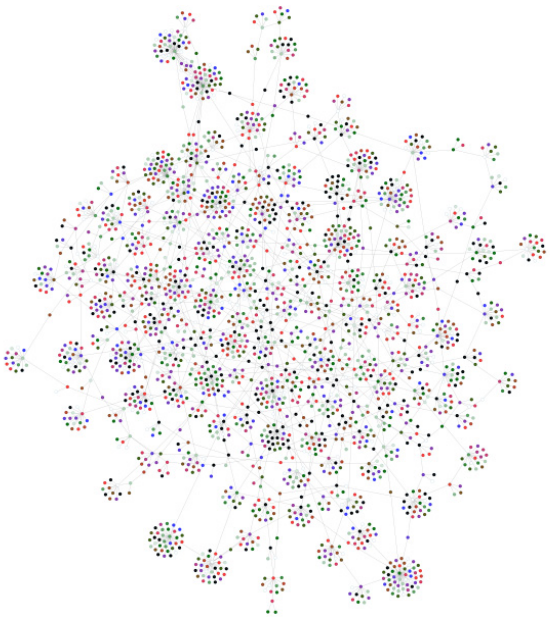

Figure 1: PSO dynamic as complex network – complete view

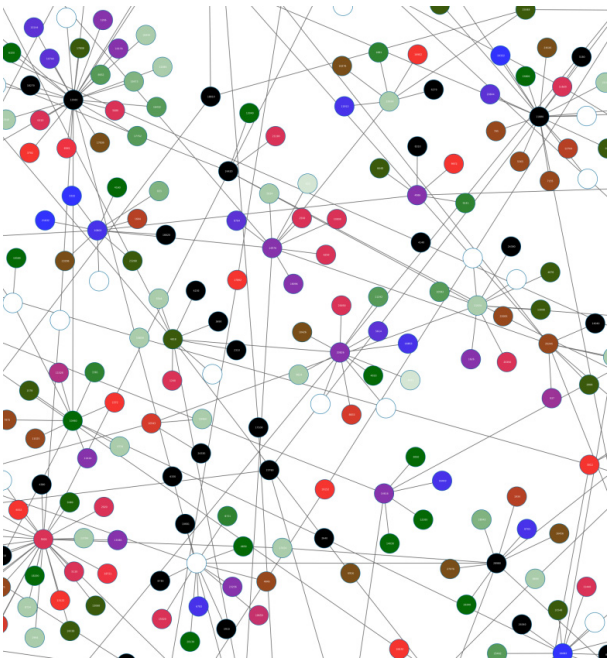

Figure 2: PSO dynamic as complex network – partial view

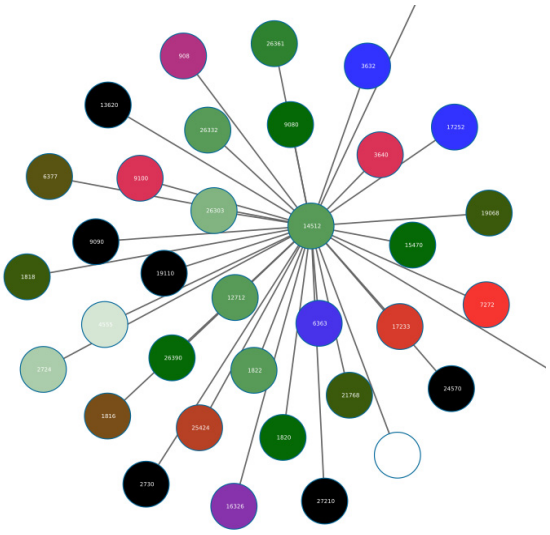

Figure 3: PSO dynamic as complex network– close view

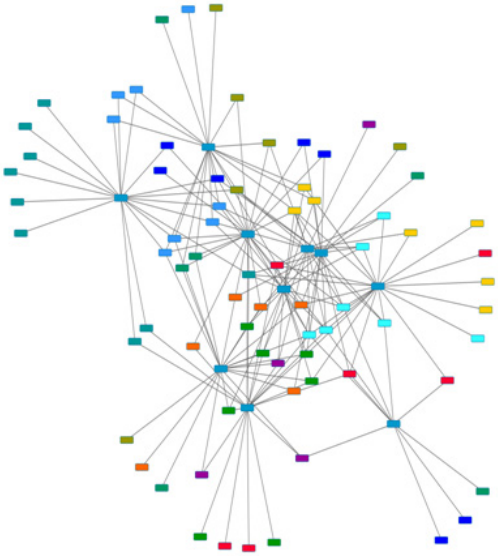

Figure 4: PSO dynamic as complex network – swarm of size 10 running for 10 iterations. Nodes with same color belong to same iteration.

FIREFLY EXPERIMENTS

In the second experiment the firefly algorithm was used. The algorithm optimized Schwefel's benchmark function for 100 iterations with population of size 30.
In the process of creation of the network every firefly was visualized as a node. Connection between nodes is plotted for every successful interaction between fireflies. Successful interaction is defined as such interaction where one of the individuals gets improved. In the case of FA it is when firefly flies towards another and improves own brightness. This leads to network presented in Figure 5 and Figure 6. Duplicate connections were omitted from the network in the sake of clarity.

Because across multiple iterations of algorithm there may be multiple connections between nodes the connections were weighted in this design. If there is connection between the firefly A and B it starts with weight 1. If in another iteration there is another successful interaction between the firefly A and B, new connection is not created but the weight of the existing connection is incremented by 1. At the end of evolution the weight is normalized. If the firefly gets improved in all iterations, at the end of the evolution their connection will have weight 1. If it never gets improved their connection will have weight 0.

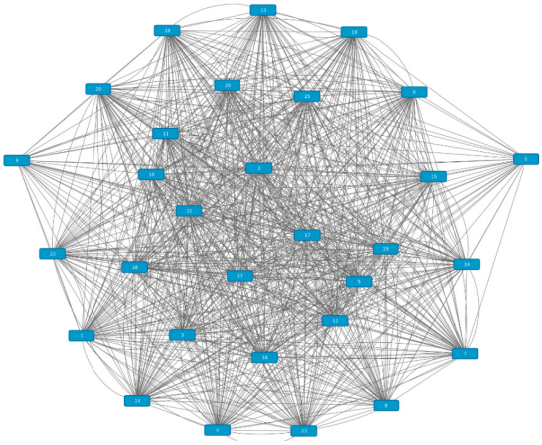

Figure 5: Basic weighted oriented network for population of size 30 after 100 iterations.

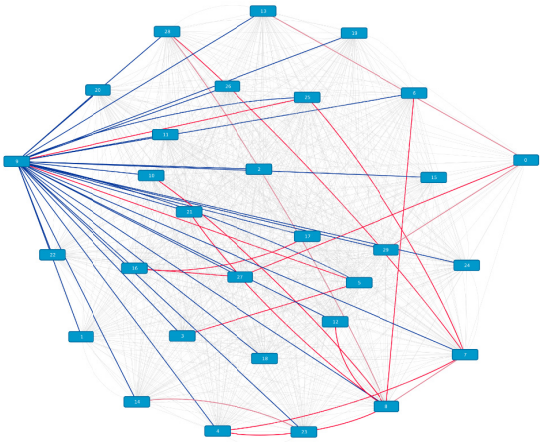

Figure 6: Basic weighted oriented network for population of size 30 after 100 iterations with visually highlighted weights (Blue: 0.7 < weight, Grey: 0.3 < weight < 0.7, Red: weight < 0.3).

Finally, even though previous network yield interesting information, its size is limited by the size of population. This may not be favorable for all tools of network analysis. In the extended model (Figure 7 – 9) nodes are not created by fireflies alone but by fireflies and the iteration in which they were created. This way every firefly will get represented up to number of times that is equal to the number of iterations. Because of this there cannot be multiple connections between same nodes, so the weights become unnecessary.

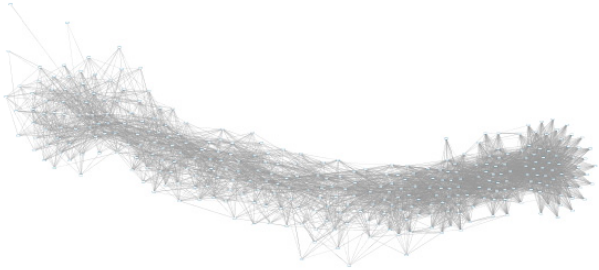

Figure 7: Time capturing oriented network

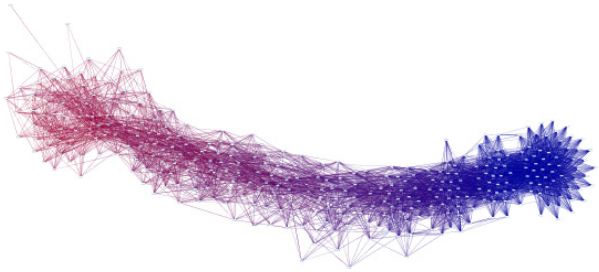

Figure 8: Time capturing oriented network with visually encoded iteration order (Red: iteration 1, Blue: iteration 10)

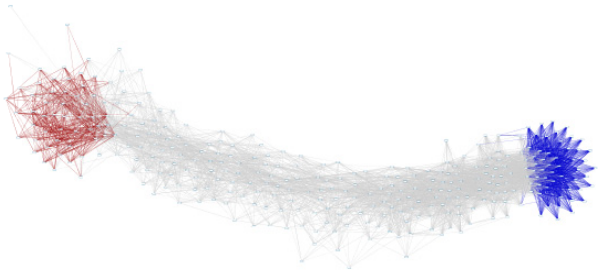

Figure 9: Time capturing oriented network with highlighted iteration 0 and 10.

CONCLUSION

In this study several complex networks were created from Swarm Intelligence based algorithms and analyzed. The goal is to capture the hidden inner dynamics of swarm algorithms. The information based on the complex network analysis may be used in various adaptive approaches. The complex networks may prove a very beneficial tool for capturing the inner dynamics

of swarm algorithms. The future research will shift the focus from creating the networks to implement various analytic and statistical tools and further to impellent adaptive mechanisms in order to improve the performance of the optimization algorithm

ACKNOWLEDGEMENT

This work was supported by Grant Agency of the Czech Republic – GACR P103/15/06700S, further by the Ministry of Education, Youth and Sports of the Czech Republic within the National Sustainability Programme project No. LO1303 (MSMT-7778/2014) and also by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089 and by Internal Grant Agency of Tomas Bata University under the Project no. IGA/CebiaTech/2016/007.

REFERENCES

Davendra, D., Zelinka, I, Metlicka, M., Senkerik, R., Pluhacek, M., "Complex network analysis of differential evolution algorithm applied to flowshop with no-wait problem," Differential Evolution (SDE), 2014 IEEE Symposium on , vol., no., pp.1,8, 9-12 Dec. 2014

Davendra, D., Zelinka, I., Senkerik, R. and Pluhacek, M. Complex Network Analysis of Discrete Self-organising Migrating Algorithm, in: Zelinka, I. and Suganthan, P. and Chen, G. and Snasel, V. and Abraham, A. and Rossler, O. (Eds.) Nostradamus 2014: Prediction, Modeling and Analysis of Complex Systems, Advances in Intelligent Systems and Com-puting, Springer Berlin Heidelberg, pp. 161–174 (2014).

Kennedy J. and Eberhart R., "Particle swarm optimization," in Proceedings of the IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.

Kennedy J., "The particle swarm: social adaptation of knowledge," in Proceedings of the IEEE International Conference on Evolutionary Computation, 1997, pp. 303–308.¨

Nickabadi A., Ebadzadeh M. M., Safabakhsh R., A novel particle swarm optimization al-gorithm with adaptive inertia weight, Applied Soft Computing, Volume 11, Issue 4, June 2011, Pages 3658-3670, ISSN 1568-4946

Shi Y. and Eberhart R., "A modified particle swarm optimizer," in Proceedings of the IEEE International Conference on Evolutionary Computation (IEEE World Congress on Computational Intelligence), 1998, pp. 69–73.I. S.

Tilahun, Surafel Luleseged; ONG, Hong Choon. Modified Firefly Algorithm. Journal of Applied Mathematics. 2012, vol. 2012. ISSN:1110-757X.

Yang X. S., Nature-Inspired Metaheuristic Algorithms, Luniver Press, UK, (2008).

Yang X. S., Firefly algorithms for multimodal optimization, Proc. 5th Symposium on Sto-chastic Algorithms, Foundations and Applications, (Eds. O. Watanabe and T. Zeugmann), Lecture Notes in Computer Science, 5792: 169-178 (2009).

Yang X. S.; Xingshi H. Firefly algorithm: Recent Advances and Applications. 2013.

Zelinka, I. Investigation on relationship between complex network and evolutionary algo-rithms dynamics, AIP Conference Proceedings 1389 (1) 1011–1014 2011a.

Zelinka, I., Davendra, D., Enkek, R., Jaek, R.: Do Evolutionary Algorithm Dynamics Create Complex Network Structures? Complex Systems 2, 0891–2513, 20, 127–140, 2011b

Zelinka, I., Davendra, D.D., Chadli, M., Senkerik, R., Dao, T.T., Skanderova, L.:Evolutionary Dynamics as The Structure of Complex Networks. In: Zelinka, I.,Snasel, V., Abraham, A. (eds.) Handbook of Optimization. ISRL, vol. 38, pp. 215–243. Springer, Heidelberg (2013)

AUTHOR BIOGRAPHIES

MICHAL PLUHACEK was born in the Czech Republic, and went to the Faculty of Applied Informatics at Tomas Bata University in Zlín, where he studied Information Technologies and obtained his MSc degree in 2011 and Ph.D. in 2016 with the dissertation topic: Modern method of development and modifications of evolutionary computational techniques. He now works as a researcher at the same university. His email address is: pluhacek@fai.utb.cz

ROMAN SENKERIK was born in the Czech Republic, and went to the Tomas Bata University in Zlin, where he studied Technical Cybernetics and obtained his MSc degree in 2004, Ph.D. degree in Technical Cybernetics in 2008 and Assoc. prof. in 2013 (Informatics). He is now an Assoc. prof. at the same university (research and courses in: Evolutionary Computation, Applied Informatics, Cryptology, Artificial Intelligence, Mathematical Informatics). His email address is: senkerik@fai.utb.cz

ADAM VIKTORIN was born in the Czech Republic, and went to the Faculty of Applied Informatics at Tomas Bata University in Zlín, where he studied Computer and Communication Systems and obtained his MSc degree in 2015. He is studying his Ph.D. at the same university and the field of his studies are: Artificial intelligence, data mining and evolutionary algorithms. His email address is: aviktorin@fai.utb.cz

IVAN ZELINKA was born in the Czech Republic, and went to the Technical University of Brno, where he studied Technical Cybernetics and obtained his degree in 1995. He obtained Ph.D. degree in Technical Cybernetics in 2001 at Tomas Bata University in Zlin. Now he is a professor at Technical University of Ostrava (research in: Artificial Intelligence, Theory of Information). Email address: ivan.zelinka@vsb.cz .