Assessing Particle Swarm Optimizers Using Network Science Metrics

Marcos A. C. Oliveira-Júnior, Carmelo J. A. Bastos-Filho and Ronaldo Menezes

Abstract Particle Swarm Optimizers (PSOs) have been widely used for optimization problems, but the scientific community still does not have sophisticated mechanisms to analyze the behavior of the swarm during the optimization process. We propose in this paper to use some metrics described in network sciences, specifically the *R*-value, the number of zero eigenvalues of the Laplacian Matrix, and the Spectral Density, in order to assess the behavior of the particles during the search and diagnose stagnation processes. Assessor methods can be very useful for designing novel PSOs or when one needs to evaluate the performance of a PSO variation applied to a specific problem. In order to apply these metrics, we observed that it is not possible to analyze the dynamics of the swarm by using the communication topology because it does not change. Therefore, we propose in this paper the definition of the influence graph of the swarm. We used this novel concept to assess the dynamics of the swarm. We tested our proposed methodology in three different PSOs in a well-known multimodal benchmark function. We observed that one can retrieve interesting information from the swarm by using this methodology.

1 Introduction

Computational Swarm Intelligence (SI) is a set of bio-inspired algorithms based on populations of simple reactive agents. They interact locally among themselves in other to generate global patterns that can be used to solve complex tasks [6]. Among the most famous SI algorithms, we can cite: Particle Swarm Optimization (PSO) [9], Ant Colony Optimization (ACO) [5], Artificial Bee Colony (ABC) [8] and Fish School Search (FSS) [2].

Marcos A. C. Oliveira-Júnior and Carmelo J. A. Bastos-Filho
University of Pernambuco, Brazil e-mail: carmelofilho@ieee.org

Ronaldo Menezes
Florida Institute of Technology, USA e-mail: rmenzes@cs.fit.edu

PSO has been widely used to solve optimization problems in hyper-dimensional search spaces with continuous variables. PSO was first proposed by Kennedy and Eberhart in 1995 [9], inspired by the social behavior of flocks of birds aiming to find food. In PSO, each particle in the swarm represents a candidate solution for the optimization problem. During the algorithm execution, each particle adjusts its position based on the current position, the current velocity, the best position achieved by itself during the search process so far and the best position obtained by the best particle in its neighborhood. This neighborhood is defined by the swarm communication topology, which defines which particles can exchange information among each other. The topology influences on the convergence velocity and on the quality of the solution obtained by the algorithm [3, 10]. Less connected topologies slow down the information flow, since the information is transmitted indirectly through intermediary particles [10]. Conversely, highly connected topologies decrease the average distance between any pair of individuals. As a consequence, there is a tendency for the whole swarm to move quickly toward the first local optimum found by any particle, when the average distance between particles is too short. In order to overcome this trade-off, some dynamic self-adjustable topologies were proposed aiming to manage the information flow during the execution of the PSO [13][12]. Furthermore, Oliveira-Júnior *et al.* [12] proposed one dynamical topology based on the preferential attachment mechanism of scale-free networks [1].

Nevertheless, the analyses on the influence of the communication topology in the algorithm performance generally are performed by using tools that can not provide comprehensive information about the swarm behavior. In general, researchers use metrics that do not assess the flow of information within the swarm; they just evaluate simple metrics, such as the average distance between particles, and the evolution of the fitness of the particles along the iterations. In fact, as we look further into the literature we conclude that there are no tools to appropriately assess the communication processes within swarm.

Network Science is the study of the theoretical foundations of network structure, its dynamic behavior, and the application of networks to many subfields [11]. There are some networks that present some specific characteristics, such as Scale-Free [1] and Small-World Networks [15]. In order to classify networks based on their structural features, many metrics have been developed [7]. In this paper, we propose to use these metrics to analyze the communication behavior of the particles during the search and diagnose stagnation processes. We also present the concept of *influence graph* of the swarm and we use this concept to assess the communication among the particles.

The paper is organized as follows: we briefly review the Particle Swarm Optimization and some Network Science metrics in Section 2. In Section 3, the simulation setup and results are presented. Finally, we provide our conclusions and suggest some future works in Section 4.

2 Background

2.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic, bio-inspired, population-based global optimization technique [9]. In PSO, each particle i has a position at time t within the search space $\mathbf{x_i}(t)$ and each position represents a possible solution for a d-dimensional optimization problem.

The particles "fly" through the search space of the problem seeking best solutions. Each particle updates its position according to the current velocity $\mathbf{v}_i(t)$, the best position found by itself $\mathbf{P}_{best_i}(t)$ and the best position found by the neighborhood of the particle i during the search so far $\mathbf{N}_{best_i}(t)$.

The velocity and the position of every particle are updated iteratively by applying the following update equations:

$$\mathbf{v_i}(t+1) = \mathbf{v_i}(t) + r_1 c_1 [\mathbf{P}_{best_i}(t) - \mathbf{x_i}(t)] + r_2 c_2 [\mathbf{N}_{best_i}(t) - \mathbf{x_i}(t)], \quad (1)$$

$$\mathbf{x_i}(t+1) = \mathbf{x_i}(t) + \mathbf{v_i}(t+1), \quad (2)$$

in which r_1 and r_2 are random numbers generated by an uniform probability density function in the interval [0,1] at each iteration for each particle for every dimension. The learning factors c_1 and c_2 are the cognitive and the social acceleration constants. They are non-negative constants and weight the contribution of the cognitive and social components, *i.e.* the second and the third terms of Equation 1.

Clerc [4] observed that the original PSO can operate in unstable states if the parameters of Equation 1 are not selected properly and determined a relation based on the constriction factor (χ) that avoids the explosion state. χ is defined according to the following equation:

$$\chi = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}, \ \varphi = c_1 + c_2. \quad (3)$$

The mechanism to update the velocity proposed by Clerc is presented in Equation 4.

$$\mathbf{v_i}(t+1) = \chi \cdot \{\mathbf{v_i}(t) + r_1 c_1 [\mathbf{P}_{best_i}(t) - \mathbf{x_i}(t)] + r_2 c_2 [\mathbf{N}_{best_i}(t) - \mathbf{x_i}(t)]\}. \quad (4)$$

The constriction factor was designed to adjust the influence of the previous particle velocities on the optimization process. It also helps to switch the search mode of the swarm from exploration to exploitation during the search process.

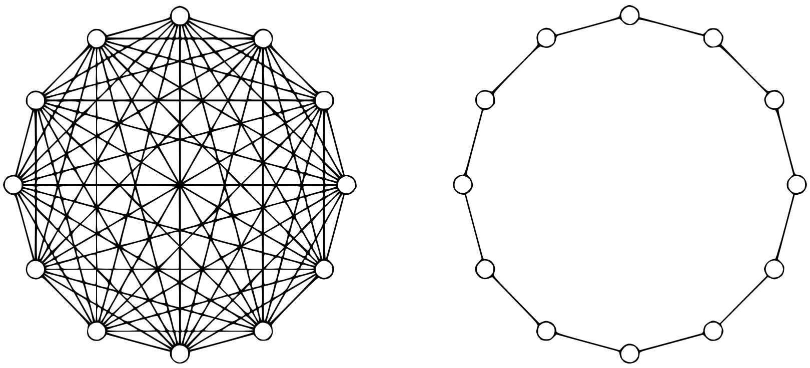
2.1.1 Particle Swarm Optimization Topologies

The way the information flows through the particles is determined by the communication topology used by the swarm. The topology defines the neighborhood of each particle, *i.e.* the subset of particles which the particle is able to communicate with.

There are some factors on the topology structure that influence on the flow of information between the particles. Kennedy and Mendes have shown that when the average distance between nodes are too short, there is a tendency for the population to move quickly toward the best solution found in earlier iterations [10]. For simple unimodal problems, it usually implies in a faster convergence to the global optimum. However, this fast convergence might be premature in a local optimum, specially in multimodal problems [3]. In this case, communication topologies with lower number of connections may reach better results [3].

The Global topology, as known as $\mathbf{G}_{best}$, was the first topology proposed for the PSO [9]. In the $\mathbf{G}_{best}$, all the particles of the swarm are neighbors, as shown in Figure 1(a). Thus, the social memory of the particles is shared by the entire swarm.

On the other hand, in local topologies, each particle only shares information with a subset of the swarm. Therefore, the social memory is not the same for the whole swarm. The most used local topology is the Ring topology [3], where each particle has only two neighbors, as depicted in the Figure 1(b). This structure helps to avoid a premature attraction of all particles to a single spot of the search space, once the information is spread slowly, but with the caveat of a slow convergence [3].



(a) Global topology (b) Ring topology

Fig. 1 Most-used particle swarm optimization communication strategies are based on global and local topologies.

These two topologies lead to extreme behaviors in the swarm, therefore many efforts have been made to propose approaches that present fast convergence while avoiding local minima [3, 10]. Some topologies that can self-adapt dynamically were proposed recently [13, 12], including the approach proposed by Oliveira-Júnior *et al.* [12], which is based on the preferential attachment mechanism present on the Barabási-Albert model [1]. In this case, each particle tries to find better particles to create connections with. We will call this topology as dynamic topology.

The Dynamic topology is based on the state of the swarm and it tries to change the information flow only when it is necessary. Because of this, a new attribute, called P_k*failures*, was included in each particle to determine the number of times a particle k does not improve its fitness. If it reaches a pre-determined threshold, then the particle is considered stagnated. If the particle k does not improve its position in the current iteration, P_k*failures* is incremented, otherwise P_k*failures* is set to zero.

The Dynamic topology is initialized as a ring topology. At each iteration of the PSO, all particles update their P_k*failures* and when a preset threshold of failures is reached, the particle searches for better particles to follow, and to stop following as well. This selection of new neighbors is based on a roulette wheel with a fitness-based rank. More details about the dynamic topology can be found in [12].

2.2 Network Science Metrics

The networks discussed in this paper are modeled as graphs. A graph G consists of a pair $[V(G), E(G)]$ where set $V(G)$ set of vertices and $E(G)$ is a set of edges. Any undirected unweighted graph G can be represented by its adjacency matrix $A(G)$, in which the non-diagonal entries (i, j) are equal to "1" if the nodes i and j are adjacent (connected), or "0" otherwise. In A, the entries (i,i) are always equal to "0", because a node cannot be connected to itself. A diagonal matrix, which contains information about the degrees of the nodes, is named Node Degree matrix, $D(G)$. The diagonal entries (i,i) are equal to the degree of the nodes D_i. The Laplacian matrix of a graph G, represented as $L(G)$, is $L(G) = D(G) - A(G)$.

Many properties of a graph can be inferred by using the Laplacian matrix. The eigenvalues $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ of L are important because they relate to many of these graph properties. Given that L is symmetric, all eigenvalues are real and non-negative. One of the first properties that arises from the Laplacian eigenvalues is the number of components in a graph. The number of zero eigenvalues corresponds exactly to the number of independent sub-graphs in the graph.

The second-smallest eigenvalue λ_2 of L is called the algebraic connectivity (or Fiedler value) of G. The magnitude of this value shows how well connected the graph is. Moreover, the value is greater than 0, if and only if, G is a connected graph.

The Adjacency matrix can also be used to provide information about the structure of the graph. The spectrum of a graph can be defined as the set of eigenvalues of its Adjacency matrix. Assuming this, the spectral density of a graph can be defined as the density of these eigenvalues and can be stated as a probability density function shown in Equation 5.

$$\rho(\lambda) = \frac{1}{N} \sum_{j=1}^{N} \delta(\lambda - \lambda_j). \tag{5}$$

Farkas *et al.* [7] showed that topological features of some kinds of graphs (uncorrelated random networks, the small-world networks, and the scale-free networks) can be identified by its graph spectral density. They also have presented some practical tools for the identification of basic types of random graphs and for classification of real-world graphs [7]. These tools are based on the extremal eigenvalues of the Adjacency matrix. The extremal eigenvalues contain useful information of the structure of the graph. The principal eigenvalue is detached from the rest of the spectrum depending on the periodicity of the graph structure. Thus, they proposed a quantity named R, defined as:

$$R = \frac{\lambda_1 - \lambda_2}{\lambda_2 - \lambda_N}, \tag{6}$$

that measures the distance of the first eigenvalue from the main part of $\rho(\lambda)$.

The R-value can be used in order to distinguish between some graph-structure features: (i) periodical or almost periodical (Small world); (ii) uncorrelated and non-periodical; and (iii) strongly correlated non-periodical (Scale Free).

3 Simulation Setup and Results

3.1 PSO Setup

In order to assess the proposed methodology, we used a multimodal benchmark function, named *F6 function*, that was proposed in [14] as a large scale optimization problem. The *F6 function* is a single-group shifted and m-rotated Ackley's function. In all experiments, we used $1,000$ dimensions and m equal to 50. We used 200 particles in all simulations. For each simulation trial, we used $300,000$ fitness function evaluations. We performed simulations for the global, local and dynamic topologies. In the case of the dynamic topology, the threshold of failures for the particles was set to $P_k failures$ =50. The particles were updated according to the Equation 4 with $c_1 = 2.05$ and $c_2 = 2.05$ as indicated in [4].

3.2 The Swarm Influence Graph

Although the communication scheme defines which particles can communicate with one another, this does not mean that a particle actually obtains useful information from all the connected particles. Here, we define useful information as the instantaneous use of the $\mathbf{N}_{best_i}(t)$ by the neighbor particle. Since we aim to assess the flow of useful information, we propose here the concept of influence graph. The influence graph consists of a set of trees and each tree represents an information flow through the swarm. Therefore, the number of trees and their structures have a great significance on the swarm behavior. As an example, lots of trees means that there are many different and independent information flows.

The influence graph is defined at each iteration considering only the active links, *i.e.* the ones in which a useful information was provided. One can observe that the influence graph I_i (where i represents an iteration of the execution) is a directed graph by definition. However, in this paper, the edge direction is removed from the graph in order to simplify the analysis and to make use of some available metrics.

Figure 2 depicts examples of influence graphs (in bold) over the three different swarm topologies studied in this paper. One can observe that there are some sub-graphs for the Ring and Dynamic topologies, whereas the global topology presents a focal point in the influence graph.

(a) Global topology (b) Ring topology (c) Dynamic topology

Fig. 2 Examples of influence graphs over the topology for the three communication topologies.

3.3 Number of Zero Eigenvalues of the Laplacian Matrix

The number of independent components in the influence graph means the number of information flows within the swarm at the current iteration. As mentioned in Section 2.2, the number of zero eigenvalues of the Laplacian matrix corresponds to the number of sub-graphs in the graph. Therefore, the number of eigenvalues of the Laplacian matrix with value equal to zero indicates the connectivity of the Influence matrix. Figure 3 shows the behavior of this quantity as a function of the number of iterations for the three considered topologies.

The Influence graph in the Global topology is an one-component star-like graph and keeps its structure during the whole execution of the algorithm; the number of zero eigenvalues is constant and equal to one. Although the value is equal to one in this case, it does not mean that the influence graph is the same along the entire process. The best particle of the swarm can change along the iterations which causes the center of the star-like topology to change.

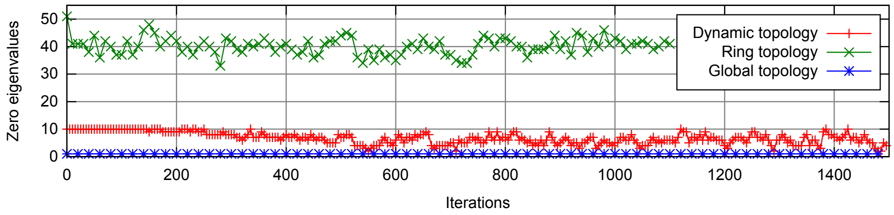


Fig. 3 Number of zero eigenvalues in the Influence matrix for Dynamic, Ring and Global Topologies.

Although the Ring topology is static, the Influence matrix can present different sub-graphs. One can observe that the number of information flows varies through the iterations, but presents a high average value of 40 along the entire process.

In the Dynamic topology, the algorithm starts with 10 information flows and diminishes in average to 5 information flows along the algorithm execution. One must observe that the Dynamic topology presents a balanced behavior between the two static approaches (Global and Ring).

3.4 *R-value*

As described in Section 2.2, the R-value represents a relation between important eigenvalues. Figure 4 presents the behavior of R-value of the Influence graph as a function of the number of iterations for the three considered topologies. A low R-value means that λ_1 is not detached from the rest of the spectrum and it can be seen as a consequence of a periodical structure [7].

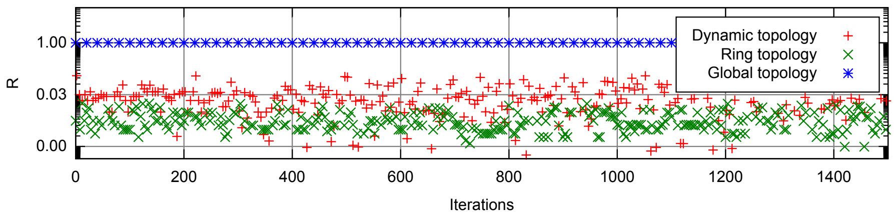


Fig. 4 R-value of the Influence matrix for Dynamic, Ring and Global Topologies.

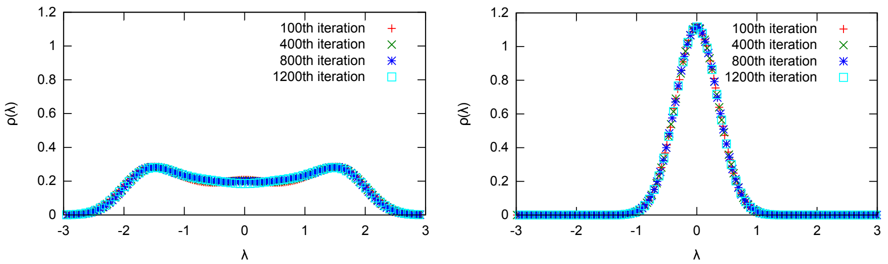
Because of the star-like behavior of influence graph for the Global topology, its R-value is constant and presents the maximum possible value (1), since the extremal eigenvalues λ_1 and λ_N are opposites.

Both the Ring and Dynamic topologies present small R-values, as they produce influence graphs that features quite periodical structures. Again, the Dynamic topology presented a balanced behavior.

3.5 The Density Spectrum

The spectral density has the capacity to represent the frequency of the eigenvalues. Therefore, it is interesting to evaluate the characteristics of the topologies as a function of the number of iterations.

As the Ring and Global topologies present a well known behavior in terms of connectivity and convergence, we first evaluated the evolution of the spectral density of the influence graph as a function of the number of iterations for these two topologies. The results for the iterations 100, 400, 800 and 1200 are depicted in Figure 5. One can observe that the Ring topology presents a bi-modal shape, while the Global topology presents a perfect uni-modal shape. Besides, the shapes are very well defined and do not vary along the iterations.



(a) Ring topology (b) Global topology

Fig. 5 Density spectrum of the swarm influence graph for the statical topologies.

After that, we evaluated the Dynamic topology. Figure 6 shows two different trials of the PSO with the dynamic topology (Run #1 and Run #2). As can be observed, the results for these two independent runs were quite different. In Run #1, the algorithm converged, while in the Run #2 the algorithm got stuck in a local minima. Figure 7 shows the spectral density of the evolution of the influence graph for these two runs for the iterations 100, 400, 800 and 1200. In Run #2, the swarm probably got stuck in a local minima because the topology presented a Global behavior at the beginning of the execution, *i.e.* the shape of the spectral density is perfectly uni-modal around iteration number 100.

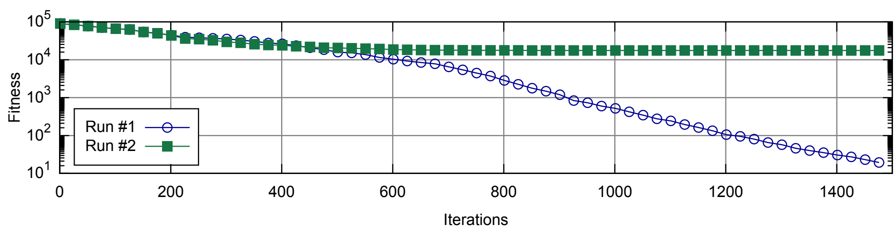
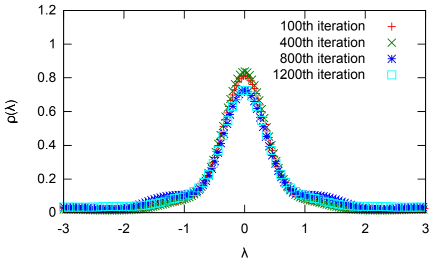
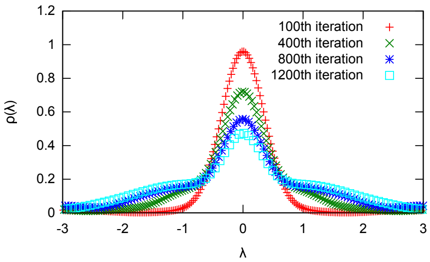


Fig. 6 Two independent runs of the PSO algorithm with the Dynamic topology.



(a) Run #1



(b) Run #2

Fig. 7 Density spectrum of the swarm influence graph in different algorithm executions.

3.6 P_k*failures* Threshold Impact

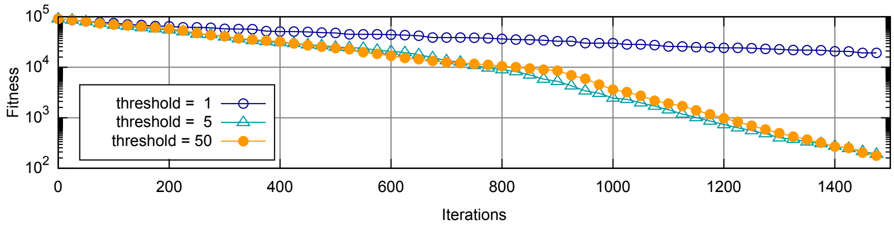
In order to show that we can use our methodology to carry out deeper analyses, we studied the influence of P_k*failures*, which plays a important role in the performance of the Dynamic topology. In general, if P_k*failures* has a low value, the particles will easily try to reconnect. Otherwise, particles will tend to maintain the current topology. One must observe that this value also has an impact on the spectral density of the influence graph.

Figure 8(a) shows the fitness evolution through iterations for P_k*failures* equal to 1, 5 and 50. Figure 8 depicts the density spectra for the three cases. One can observe that in all cases the spectral densities present a combination of the uni-modal and bi-modal curves. The fitness obtained for $P_k failures = 1$ was not satisfactory because, in this case, the topology changes a lot and it diminishes the convergence capability.

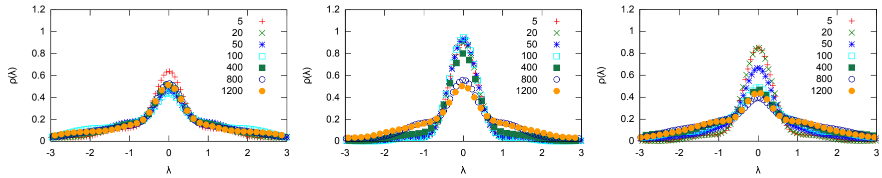
4 Conclusions and Future Works

We proposed in this paper a set of tools based on some Network Science metrics to assess the information flow on the Particle Swarm Optimization algorithms. We observed that it is necessary to assess the influence graph, instead of the topology itself in order to evaluate only the flow of useful information. The tools link the structure features of the influence graph to algebraic quantities, that may be used for further analysis in order to understand the swarm behavior.

We have shown that the swarm with a Dynamic topology has a behavior that is between the two most used static approaches. Moreover, the simulation results indicate that the stagnation can be foreseen by analyzing the density spectrum along the iterations.



(a) Fitness evolution



(b) P_k*failures* threshold = 1 (c) P_k*failures* threshold = 5 (d) P_k*failures* threshold = 50

Fig. 8 Fitness evolution and density spectrum of the swarm influence graph with different P_k*failures* thresholds.

As future works, we intend to use these tools to design high performance dynamic topologies for PSOs by assessing the information flow within the swarm. We also aim to develop variations of the swarm influence graph to recognize different aspects of the swarm communication.

References

1. Albert, R., Barabasi, A.L.: Statistical mechanics of complex networks. Reviews of Modern Physics **74**, 47 (2002). URL doi:10.1103/RevModPhys.74.47
2. Bastos-Filho, C.J.A., Lima-Neto, F.B., Lins, A.J.C.C., Nascimento, A.I.S., Lima, M.P.: A novel search algorithm based on fish school behavior. 2008 IEEE International Conference on Systems, Man and Cybernetics pp. 2646–2651 (2008)
3. Bratton, D., Kennedy, J.: Defining a standard for particle swarm optimization. Swarm Intelligence Symposium, 2007. SIS 2007. IEEE pp. 120 –127 (2007). DOI 10.1109/SIS.2007.368035
4. Clerc, M., Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space. Evolutionary Computation, IEEE Transactions on **6**(1), 58 –73 (2002). DOI 10.1109/4235.985692
5. Dorigo, M., Caro, G.: Ant colony optimization: A new meta-heuristic. In: Proceedings of the Congress on Evolutionary Computation, pp. 1470–1477. IEEE Press (1999)
6. Engelbrecht, A.P.: Computational Intelligence: An Introduction. Wiley Publishing (2007)
7. Farkas, I.J., Derényi, I., Barabási, A.L., Vicsek, T.: Spectra of "real-world" graphs: beyond the semicircle law. Phys Rev E Stat Nonlin Soft Matter Phys **64**(2 Pt 2) (2001). URL http://view.ncbi.nlm.nih.gov/pubmed/11497741
8. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Tech. rep., Erciyes University, Engineering Faculty, Computer Engineering Department (2005)

9. Kennedy, J., Eberhart, R.: Particle swarm optimization. pp. 1942–1948 vol.4 (1995). DOI 10.1109/ICNN.1995.488968. URL http://dx.doi.org/10.1109/ICNN.1995.488968
10. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on, vol. 2, pp. 1671 –1676 (2002). DOI 10.1109/CEC.2002.1004493
11. Lewis, T.G.: Network Science: Theory and Applications. Wiley Publishing (2009)
12. Oliveira-Júnior, M.A.C., Bastos-Filho, C.J.A., Menezes, R.: Using network science to define a dynamic communication topology for particle swarm optimizers. In: R. Menezes, A. Evsukoff, M.C. González (eds.) Complex Networks, *Studies in Computational Intelligence*, vol. 424, pp. 39–47. Springer Berlin Heidelberg (2013). DOI 10.1007/978-3-642-30287-9_5
13. Suganthan, P.: Particle swarm optimiser with neighbourhood operator. In: Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, vol. 3, pp. 3 vol. (xxxvii+2348) (1999). DOI 10.1109/CEC.1999.785514
14. Tang, K., Li, X., Suganthan, P.N., Yang, Z., Weise, T.: Benchmark Functions for the CEC'2010 Special Session and Competition on Large-Scale Global Optimization. Tech. rep., University of Science and Technology of China (USTC), School of Computer Science and Technology, Nature Inspired Computation and Applications Laboratory (NICAL): China
15. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393**(6684), 440–442 (1998). DOI 10.1038/30918. URL http://dx.doi.org/10.1038/30918