Enhanced Performance of Consensus Wireless Sensor Controlled System via Particle Swarm Optimization Algorithm

Safanah Mudheher Raafat[1]
safanamr@gmail.com

Ahmed Mudheher Hasan [1]
60163@uotechnology.edu.iq

Karar Ghalib[1]
karrar.g.ali@gmail.com

[1]Control and System Engineering Dept.
University of Technology
Baghdad Iraq

ABSTRACT

This paper describes the application of consensus optimization for Wireless Sensor Network (WSN) system. Consensus algorithm is usually conducted within a certain number of iterations for a given graph topology. Nevertheless, the best Number of Iterations (NOI) to reach consensus is varied in accordance with any change in number of nodes or other parameters of graph topology. As a result, a time consuming trial and error procedure will necessary be applied to obtain best NOI. The implementation of an intelligent optimization can effectively help to get the optimal NOI. The performance of the consensus algorithm has considerably been improved by the inclusion of Particle Swarm Optimization (PSO). As a case study, variable number of nodes in a network with a random graph topology has been considered. Simulation results show that significant reduction in the NOI and power consumption has been achieved, where it decreased the NOI about 40 iteration; when using PSO for different number of nodes in the specified network.

Key words: consensus algorithm (CA), wireless sensor network (WSN), graph topology, particle swarm optimization (PSO).

تحسين اداء منظومة مسيطرة للكشاف اللاسلكي الجمعي بواسطة خوارزمية حشد الجسيمات الامثلية

د.سفانة مظهر رأفت[1]
أستاذ مساعد

د.أحمد مظهر حسن[1]
مدرس

كرار غالب خطار[1]

قسم هندسة السيطرة والنظم[1]
الجامعة التكنولوجية
بغداد-العراق

الخلاصة

هذا البحث يصف تطبيق تحسين التوافق لشبكة الاستشعار اللاسلكية. وتجرى خوارزمية التوافق عادة لعدد محدد من الخطوات لبنية الرسم البياني. ومع ذلك، فان افضل عدد تكرارات للتوصل لتوافق يختلف طبقا للتغيير بعدد العقد او المعاملات الاخرى لبنية الرسم البياني. كنتيجة لذلك، سيتم تطبيق اجراء الخطا والصواب للوصول الى افضل عدد من الخطوات. تنفيذ خوارزمية الامثلية الذكية يساعد بشكل فعال للحصول على افضل عدد من الخطوات. حيث تم تحسين اداء خوارزمية التوافق بشكل ملحوظ باستخدام طريقة حشد الجسيمات الامثلية. وكدراسة تحليلية، فان عدد مختلف من العقد قد اخذ بنظر الاعتبار في الشبكه لعدد عشوائي من بنية الرسم البياني وتبين النتائج المستحصلة من المحاكاة بتقليل ملحوظ بعدد الخطوات واستهلاك الطاقة وكذلك انخفض عدد الخطوات الى حوالي 40 خطوة باستخدام طريقة حشد الجسيمات الامثلية لعدد مختلف من العقد في شبكة محددة.

1. INTRODUCTION

The sensor network system plays an important role in many military and civilian areas, including area monitoring, health care monitoring, environmental/earth sensing, forest fire detection, and so forth. The main task of sensor network is to measure some environmental parameters such as temperature, moisture degree, pressure, and so on. Network elements (i.e. nodes) measure such parameter independently and the final value is usually obtained in a centralized or decentralized network. In these cases it is often preferred to use the Wireless Sensor (WS) for the forfeit of hardware infrastructures connecting. WS is often containing a transducer, a processing unit, a wireless radio transceiver and a power supply. WS measures certain aspects of the environment. The data is then processed in processing unit and setd to a centralized network; or in a decentralized network, **Akyildiz, et al., 2002**.

The most prominent challenge to design a Wireless Sensor Network (WSN) is how to extend the life of the wireless sensor through the preservation of energy expended, **Bhardwaj, et al., 2001**. Wireless Sensor (WS) has small-sized batteries which are costly. An agreement is required on the sensing parameters in a sensory network agreement algorithm and protocol between nodes. An agreement algorithm must be as uncomplicated as possible and lead the nodes to agree as fast as possible, **Scutari, et al., 2008**.

Distributed consensus algorithm can be used to reach agreement in WSN where it computes the average of an initial set of measurements. The calculation of the average value is done through local information exchange among neighbors. The speed of consensus algorithm to reach the average depends on the states of each sensor. This is important role for energetic reasons to have a smaller number of transmissions among the sensors. Accordingly, the convergence time, that leads to lower energetic cost for each sensor should be reduced, **Giridhar,** and **Kumar, 2005**.

Early research on consensus algorithm focuses on fixed topologies, **Scherber,** and **Papadopoulos, 2004**, where the communication links and the nodes in network are assumed constant throughout time. Study of the consensus algorithm with random network had been presented in, **Xiao,** and **Boyd, 2006**. The authors proposed pair wise gossip defined as that "every two neighbored nodes can be updating their states at each iteration and so on then all nodes reaches to the consensus value". This approach acts slowly since it requires storing all data after each iteration. Additionally, large memory is needed.

Xiao, and **Boyd, 2004, Boyd, et al., 2004**, proposed another approach to change the weight of each edge between sensors that it's required to apply the consensus algorithm. The aim is to find optimal weights to decrease the convergence time. The algorithm had been treated as linear iteration. For large sensor networks, the application of this approach may be unfeasible, since a new topology has to be acquired after each change in the communication between sensors under given computational constraints.

For large scale graph a “random rewiring” method is proposed by **Olfati,** and **Shamma, 2005,** in order to increase the convergence speed in consensus algorithm. However, physically in some applications changing the topology may be difficult.

Another area of recent studies are related to lifetime of sensors; **Sun, et al., 2011**, suggests changing the weights of graph dependent on Euclidian area by using special algorithm called Weighted Dynamic Topology Control (WDTC) algorithm. This algorithm is more complex and does not regard to the performance of network.

Moreover, **Junghun Ryu, et al., 2013** used Borel Cayley graphs to minimize the distance between WS in network that’s will lead increase the number of edge between nodes. While in **Jianping, et al., 2003**, the authors depended on the base station of nodes to increase the lifetime of wireless sensor by using optimal base station location based on computational geometry. Unfortunately, this arrangement had been used in the case of a fixed network in specified area.

In one hand, PSO algorithm is utilized in different optimization problems. On the other hand, it shows superiority at handling optimization problems with network design spaces **Junghun Ryu, et al., 2013**.

The PSO allows for the efficient optimization of complex design spaces and can reach an acceptable optimum solution even when noise and discontinuities exist within the design space. Because of these characteristics, PSO algorithms can be integrated into the optimization framework. The principal idea behind the current effort is to propose a fast, reliable and robust optimization framework that can search for optimal parameters of consensus algorithm for different number of nodes, **Hyunseok, et al., 2013**.

In this paper, strategies of graph theory are applied to model random WSN. Furthermore, the consensus algorithm is implemented to reduce the power consumption. The convergence speed of CA is analyzed and optimized through utilizing powerful optimization method. As a result it increases the capability of the WSN system to speed up the reaching of the goal. Therefore, it reduces the overall energy consumption of a WSN. Therefore, in this paper convergence time should be reduced to eliminate the energy losses. Moreover, fixed topologies in previous work should be considered and perform the algorithm for random networks.

This paper is organized as follows: Section II presents the problem formulation, Section III shows the Consensus in WSN while in section IV the integration between PSO with the consensus algorithm was presented and finally, some conclusions are drawn in Section V.

2. THEORITICAL BACKGROUND

In this section, some basic definitions and notation used in this paper will be presented.

A. The difference between centralized and decentralized

In centralized network sensory, sensors must send their measurement to complex model called fusion center (FC). FC takes the measurement of WSN and makes the final decision. In this type of network it is required an organized set of nodes under medium access control (MAC) and routing protocols require sending the data to the FC, when a sensor fails in network or added to network the re-organization of the MAC and routing protocols is necessary, **Giridhar,** and **Kumar, 2006**.

In decentralized WSN the data process without sending it to the FC and reach decisions locally each sensor can be considering as FC, sensors can organize them and communicated locally. In decentralized networks most requirement of a WSN can be satisfied and can provide reliable results, **Rabbat,** and **Nowak, 2004**.

B. Spectral Graph Theory

In this section, the properties and some definitions that need of spectral graph theory will be reviewed.

Assume a network with N nodes called *vertices* and communication lines between them called edges, the graph topology can be represent as G{V,E} in which the *vertices* V=(1,…,N) and the edges $E \subseteq V \times V$, **Silvana, 2012**.

There are two types that define the flow state between nodes; directed graph if the direction given otherwise undirected graph when there is no direction between nodes.

Definition 1:
The *adjacency* matrix A of G represents the communication between each node with its neighbors

$$a_{ij} = \begin{cases} 1 & if \;\; e_{ij} \in E \\ 0 & if\; else \end{cases} \tag{1}$$

a_{ij} Represent the entries of *adjacency* matrix.
e_{ij} Represent edge between node i and j

Definition 2:
The *in-degree* term of G represent the number of incoming states to the nodes from its neighbors

$$d_i^{in} = \sum_{j=1}^{N} a_{ji} \tag{2}$$

Definition 3:
The *out-degree* term of G represent the number of outgoing states to the nodes

$$d_i^{out} = \sum_{j=1}^{N} a_{ij} \tag{3}$$

Definition 4:
The *degree* matrix is the matrix which its entries are equivalent to the row totals of the *adjacency* matrix

$$D_{ij} = \begin{cases} d_i^{out} \; if \;\; i = j \\ 0 \;\;\; if\; else \end{cases} \tag{4}$$

Definition 5:
The *Laplacian matrix L is* equivalent to the difference between *D* matrix and *A* matrix

$$L = D - A \tag{5}$$

C. Graph theory topologies

There are different models of graphs in graph theory. The most usual topologies in graphs can be seen in **Fig. 1**
***A Ring* network**: in this network each node has only two neighbors called a 2-regular graph in which the spatial distribution looks like a circle.
A *lattice* network: in this network the external node has two neighbors while the internal node has four neighbors in which the spatial distribution looks like a *2D* grid.
A *small-world* network: in this model most of the nodes are not neighbors of other nodes. The nodes connect with each other by established random connection between vertices of nodes, **Alan,** and **Desmond, 2009**.
A *scale-free* network: according to a power law the number of nodes in this model is distributed. This model is very useful for use in internet. A number of nodes in this network can be in the millions, **Guido, 2007**.
A *random geometric* network: contains a set of nodes that's distributed randomly in 2D area. In this model each node is connected with each other if the distance of Euclidean is less than a set of radius, **Penrose, 2003**. This network is used in this paper. Each node is placed in uniformly at random in the unit area, as shown in **Fig. 2**. Each node has (*x, y*) coordinates for some stated radius, R_r nodes (*i, j*) are connected if and only if the following equation is specified, **Alan,** and **Desmond, 2009**:

$$\left(x_i - x_j\right)^2 - \left(y_i - y_j\right)^2 \leq r^2 \tag{6}$$

D. Consensus Algorithm

A Consensus Algorithm is an iterative scheme where its purpose is that each node in wireless sensor network should reach to the same value such as the average from the initial values, **Kenyeres, et al., 2015**.

In consensus algorithm no need to pass the information to the central point, the information of each node are exchanged on a local basis by the nodes with each repetition, so that the value interested should reach in asymptotically way, for example, let's have a network with N-nodes and each node has a scalar value represented by X_i, that defined the state of each node i. The state of measurement sets which are update repeatedly depended on the information that's got from its neighbors. Node i and j achieve consensus if $X_i = X_j$, **Xiao,** and **Boyd, 2004**.

The common function to calculate the average of initial measurement is:

$$\alpha = N^{-1} \sum_{i=1}^{N} X_i(0) \tag{7}$$

All nodes have a local variable $X_i(0)$ at time 0. This equation will be the goal of each node in network however each node should calculate its neighbor's state and update its state variable according to consensus algorithm, where the state of the graph model is given in this paper.

$$x_i(k+1) = x_i + \epsilon \sum_{j \in N_i} \left(x_j(k) - x_i(k)\right) \tag{8}$$

This equation is discrete consensus algorithm, **Kenyeres, et al., 2011**. It updates the state of node dependent on the collect neighbor's state, the nodes must agree on the same parameter of $\epsilon \in (\frac{0.1}{\Delta})$ where Δ is number of degree out in the network dependent on the graph theory connection and ϵ is so-called mixing parameter, **Fagnani,** and **Zampieri, 2008**.

3. DESIGN OF NETWORKING SCENARIO

The network scenario design and simulation prototype is presented while the results obtained are discussed as well in this section.

Consider that there are N wireless sensors placed in a unit area randomly, each node exchange the consensus information with set of neighbors and by using undirected graph can provide scalability networking scenario within area denoted by radius R_r as will be explained later.

A static network of WSN modeled by unidirectional graph theory G (V, E) is considered, while the degree in and the degree out of the node are equal which is known as balanced undirected graph.

Each sensor in network has initially local state $x_i \in \mathbb{R}$ where i the number of nodes. In this paper a state is chosen randomly as shown in **Fig. 3**. The initial states for 20 nodes were chosen randomly by function *rand* in MATLAB.

Through using average consensus algorithm that's run in discrete time Eq.(8), the nodes will collect their neighbor's states and update their state depending on the neighbors states to reach to the consensus value. Each round of the operation states update is called iteration. The more reduction of required iterations to reach the goal the maintaining of energy for WSNs will be greater.

4. SIMULATION RESULTS

The implementations of the consensus network will be discussed in the following subsections:

A. Average Consensus Performance

Discrete consensus program has been developed using MATLAB. As an example, if there are 20 nodes, each node of network runs in discrete consensus algorithm receiving inputs from their neighbors and update its state at each round until reach to the consensus state. Applying consensus algorithm and their effects on the states of nodes and how each state reach to the consensus value is shown in **Fig. 4**. **Fig. 5** shows the states after applying the consensus algorithm and each state equals to the consensus value.

According to discrete consensus algorithm each node reaches to the consensus value after large number of iterations which require more power for wireless sensor to consensus with each other.

The number of iterations or the convergence time is affected by parameter ϵ in Eq (8). It's usually selected as a constant according to $\epsilon \in (\frac{0.1}{\Delta})$, **Olfati-Saber, et al., 2007**.

In this paper ϵ is assumed to be:

$$\epsilon = (\frac{\beta}{\Delta}) \tag{9}$$

Where β is selected as a variable real number between 0 and 1 such that it affects the convergence time. For example, when $\beta = 0.1$ it will require more than 50 iterations to reach to the consensus value, as can be seen in **Fig. 4**. Accordingly more energy will be spent for each WSN.

Through assuming a constant number of nodes (i.e. 20 nodes), **Fig. 6** shows the inverse relationship between the number of iterations and β where increasing β results in decreasing the number of iterations required in order to reach a consensus value. It's clear that when $\beta = 0.9$, it will need only 2 iterations to reach the consensus value, as shown in **Fig. 7**.

Meanwhile increasing the number of nodes will require a different value of β.

If there are 100 nodes with different initial states. For $\beta = 0.1$ it needs more than 75 iterations to reach all nodes to the consensus value as shown in **Fig. 8**.

B. Integration of PSO method with the consensus algorithm

As seen in the results of previous section, when using different number of nodes the value of β should be changed to reach the goal with least number of iterations. That's will need to change the whole topology of the network which is difficult and time consuming since it needs to define new set of number of nodes and select new suitable value for β in order to affect the convergence time, **Silvana, 2012**.

The need to a technique that automatically makes changes in the value of β without changing the whole topology of network is very important. By applying a suitable optimization method the value of β is initialized randomly and then changed automatically and proportionally to the number of WSNs in networks. The optimization method takes into account the number of nodes in network and search for an optimal value of β which affect the convergence time in order to reach the required consensus value for all WSNs.

PSO is a newly invented high-performance optimizer that achieves several highly desirable features, together with the fact that the basic algorithm is very easy to understand and implement.

It has been applied to a diverse number of applications including neural network training and nonlinear systems. PSO is like GAs and other evolutionary algorithms, but it requires less computational memory and fewer lines of code. The PSO conducts its optimization search using a population of particles which correspond to individuals in GAs, where each particle has a specific velocity vector and a position vector to represent a possible solution **Ahmed, et al., 2011**.

Particle swarm optimization (PSO) is characterized by simplicity, efficiency and effectiveness. PSO is heuristic global optimization method and all particles in the swarm are used in a global fitness function, so that the speed of mutation calculation is so fast. PSO is robust to achieve the solution in data aggregation problems, **Olfati, et al., 2007**.

There are three choices for particle in evaluation; the first choice is to insist on itself as optimum solution, the second choice is to move toward the optimum solution itself while each particle remembers its own personal best position that it has found which is called local best, the third choice is to move to the best solution which the population is met. Each particle knows the best position found by other particle in the swarm, which is called global best. The PSO make compromises among these three choices. The PSO update its velocity for each particle through using the local best position (L_{best}) and the global best position (G_{best}) by, **Syed,** and **Sumitha, 2014, Lin,** and **Hong, 2007**.

$$V_i(k+1) = w * V_i(k) + \varphi_1 * r_1 * \left(L_{best} - X_i(k)\right) + \varphi_2 * r_2 * \left(G_{best} - X_i(k)\right) \quad (10)$$

Where:

V_i: *particle velocity*

$X_i(k)$: *current particle postion*

k: *pointer of iteration*

w : *weight of inertia*

φ_1 *and* φ_2 : *inertia constant*

r_1 *and* r_2 : *the random number in the rang* $[0, 1]$

Each particle changes its position dependent on the velocity update according to the following equation, **Lin,** and **Hong, 2007**:

$$X_i(k+1) = X_i(k) + V_i(k+1) \quad (11)$$

In this paper PSO is utilized to choose optimal value of parameter β to be suitable for all numbers of nodes in WSN. For example, given a predefined network of 20 and 100 nodes modeled by random undirected graph, given the PSO parameters, as shown in Table 1, through many experiments on the implementation of PSO algorithm to improve the consensus performance. The value of Table 1 had been selected, three scenarios have been used for the performance evaluation, each scenario has different number of nodes 20, 100 and 150, as shown in Table 2 where the performances of the WSNs before and after inclusion of PSO are presented.

Fig. 9 shows the complete algorithm for the integrated consensus algorithm with the PSO technique. Moreover, a comparison between the proposed work and previous works shows the superiority of the proposed work in saving the energy of the nodes in the network through reducing the required number of iterations as shown in Table 3.

5. CONCLUSIONS

This paper presents the implementation of PSO algorithm to enhance the performance of consensus wireless sensors controlled system. First, graph theory was used to model a random wireless network system. Then, consensus algorithm had been adopted to reach certain value. The required number of iterations to reach consensus had been minimized using PSO algorithm. Different number of nodes had been tested to build different graphs. Simulation experiments had been conducted . Simulation experiments were conducted using MATLAB environment. Saving of energy consumption for WSNs has been realized by 67% improvement compared to stand alone consensus algorithm and finally the results shows a significant reduction in the required number of iterations to reach consensus value.

REFERENCES

- Ahmed, M. H., Khairulmizam, S., Abdul, R., R., 2011, *Optimizing of ANFIS for estimating INS error during GPS outages*, Journal of the Chinese insitiute of Engineers, Vol. 34, No. 7, pp. 967-982.
- Akyildiz, I.F., Weilian Su., Sankarasubramaniam, Y., and Cayirci, E., 2002, *A survey on sensor networks*, IEEE Communications Magazine, Vol.40, No.8, pp. 102-114.
- Alan, T., and Desmond, J., 2009, *A Controllable Test Matrix Toolbox for MATLAB*, ACM Trans. Math. Software, Vol. 35, No. 4, pp. 26:1-26:17
- Bhardwaj, M., Garnett, T., and Chandrakasan, A. P., 2001, *Upper bounds on the lifetime of sensor networks*, Proc. IEEE International Conference on Communications., Helsinki-Finland, Vol. 3, pp. 785-790.
- Boyd, S., Diaconis, P., and Xiao, L., 2004, *Fastest mixing markov chain on a graph*, SIAM Review, Vol. 46, No. 4, pp. 667-689.
- Boyd, S., Ghosh, A., Prabhakar, B., and Shah, D., 2006, *Randomized gossip algorithms*, IEEE Transactions on Information Theory, Vol. 52, No. 6, pp. 2508-2530.
- Chen, Y., Tron, R., Terzis, A., and Vidal, R. 2010, *Corrective consensus: Converging to the exact average*. Proceedings of the IEEE Conference on Decision and Control. https://doi.org/10.1109/CDC.2010.5717925
- Chen, Y., Tron, R., Terzis, A., & Vidal, R. 2011, *Accelerated corrective consensus: Converge to the exact average at a faster rate*. Proceedings of the American Control Conference. https://doi.org/10.1109/ACC.2011.5991097.
- Fagnani, F., Zampieri, S., 2008, *Randomized consensus algorithms over large scale networks*, IEEE Journal on Selected Areas of Communications, Vol. 26, No. 4, pp 634-649.
- Giridhar, A., and Kumar, P.R., 2005, *Computing and communicating functions over sensor networks*, IEEE Journal on Selected Areas in Communications, Vol. 23, No. 4, pp. 755-764.
- Giridhar, A., and Kumar, P.R., 2006, *Toward a theory of in-network computation in wireless sensor networks*, IEEE Communications Magazine, Vol. 44, No. 4, pp. 98-107.
- Guido, C., 2007, *Scale-Free Networks*, Oxford University Press.
- Hyunseok, K., Jinsul, K., and Seongju, 2013, *Particle Swarm Optimization-Based Consensus Achievement of a Decentralized Sensor Network*, Advanced Science and Technology Letters Vol.42, pp.162-166.

- Junghun, R., Jaewook, Y., Eric, N., and Wendy, K., 2013, *Borel Cayley Graph-based Topology Control for Consensus Protocol in Wireless Sensor Networks*, ISRN Sensor Networks, Vol. 2013, pp. 1-15, Article ID 805635.
- Jianping, P., Thomas, H., Lin, C., Yi, S., and Sherman, X., 2003, *Topology control for wireless sensor networks*, MobiCom'03 Proceeding of the Annual International Conference on Mobile Computing and Networking, San Diego, CA-USA, pp. 286-299.
- Kenyeres, M., Kenyeres, J., Skopril, V., 2015, *Split distributed computing in wireless sensor networks*, *Radioengineering*, Vol. 24, No. 3, pp. 749-756.
- Kenyeres, J., Kenyeres, M., Rupp, M., and Farkas, P., 2011, *WSN Implementation of the Average Consensus Algorithm*, 17th European Wireless 2011-Sustainable Wireless Technologies, Vienna, Austria, pp. 1-8
- Lin, C., and Hong, S., 2007, *The design of neuro-fuzzy networks using particle swarm optimization and recursive singular value decomposition*, Neurocomputing, Vol. 71, No. 1-3, pp. 297-310.
- Olfati-Saber, R., Alex, J., and Murray, R.M., 2007, *Consensus and cooperation in networked multi-agent systems*, Proceedings of IEEE, Vol. 95, No. 1, pp. 215-233.
- Olfati-Saber, R., and Shamma, J. S., 2005, *Consensus filters for sensor networks and distributed sensor fusion*, Proceedings of the 44th IEEE Conference on Decision and Control European Control Conference, pp. 6698-6703.
- Penrose, M., 2007, *Random Geometric Graphs*, Oxford University Press.
- Rabbat, M., and Nowak, R., 2004, *Distributed optimization in sensor networks*, Third International Symposium on Information Processing in Sensor Networks, (IPSN'04), Berkeley, California, pp. 20-27.
- Rajagopal, R., and, Wainwright, M. J. 2011, *Network-based consensus averaging with general noisy channels*. IEEE Transactions on Signal Processing, 59(1), 373–385. https://doi.org/10.1109/TSP.2010.2077282.
- Scutari, G, Barbarossa, S, and Pescosolido, L., 2008, *Distributed decision through self-synchronizing sensor networks in the presence of propagation delays and asymmetric channels*, IEEE Transactions on Signal Processing, Vol. 56, No. 4, pp. 1667-1684.
- Scherber, D.S., and Papadopoulos, H.C., 2004, *Locally constructed algorithms for distributed computations in ad-hoc networks*, Third International Symposium on Information Processing in Sensor Networks (IPSN'04), Berkeley, California, pp. 11-19.
- SyedAliFathima, K., Sumitha, T., 2014, *To Enhance the Lifetime of WSN Network using PSO*, International Journal of Innovative Research in Computer and Communication Engineering Vol. 2, No. 1, pp. 1-6.
- Sun, R., Yuan, J., You, I., Shan, X., and Ren, Y., 2011, *Energy-aware weighted graph based dynamic topology control algorithm*, Simulation Modeling Practice and Theory, Vol. 19, No. 8, pp. 1773-1781.
- Silvana, P., 2012, *Distributed Consensus Algorithms for Wireless Sensor Networks: Convergence Analysis and Optimization*, Ph.D Thesis, Universitat Politecnica de Catalunya-Barcelona Tech, Spain.
- Xiao, L. and Boyd, S., 2004, *Fast linear iterations for distributed averaging*, Systems and Control Letters, Vol. 53, pp. 65-78.

Table 1. Parameters for PSO.

Parameters	Value
Fitness function	$\epsilon = \beta /$ (n-1)
Swarm size	Limits of β [0,1]
Correction factor	0.01
Maximum iterations	10
Initial particle position	Input data
Best position so far	1000
Initial velocity	0
Inertia	1.0

Table 2. The performances of consensus algorithm before and after applying PSO algorithm for three different values of nodes: 20,100 and 150.

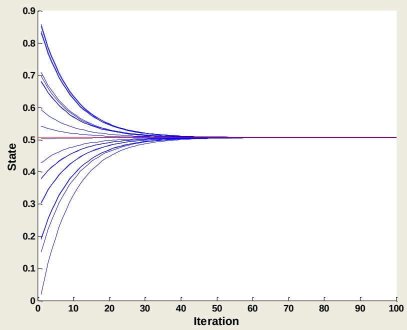
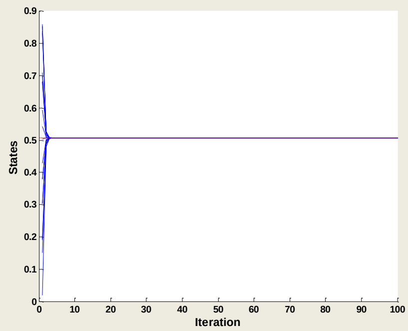
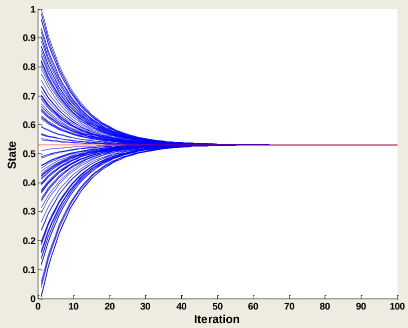
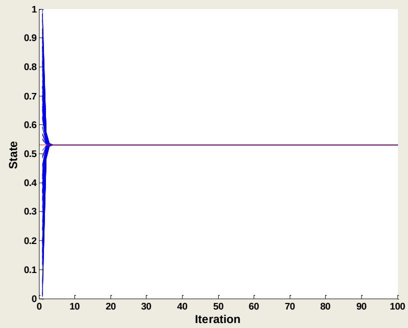
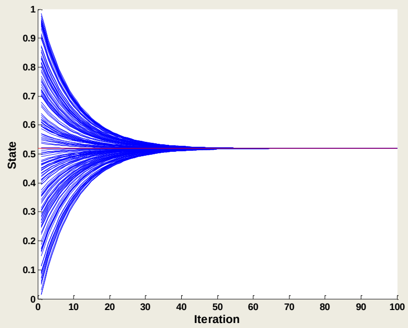
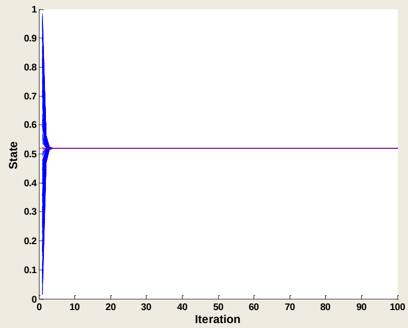
No. of Nodes	Before applying the PSO	After applying the PSO
20		
100		
150		

Table 3. A comparison between the proposed work and previous work.

References	Network topology	Number of Nodes	Number of iterations
Proposed work	random	150	3
(Scherber & Papadopoulos, 2004)	random	200	130
(Chen, Tron, Terzis, & Vidal, 2010)	random	10	24
(Chen, Tron, Terzis, & Vidal, 2011)	fixed	10	62
(Rajagopal & Wainwright, 2011)	fixed	150	10
(J. Kenyeres & Kenyeres, 2011)	random	10	2
(M. Kenyeres, Kenyeres, & Skorpil, 2015)	fixed	24	14

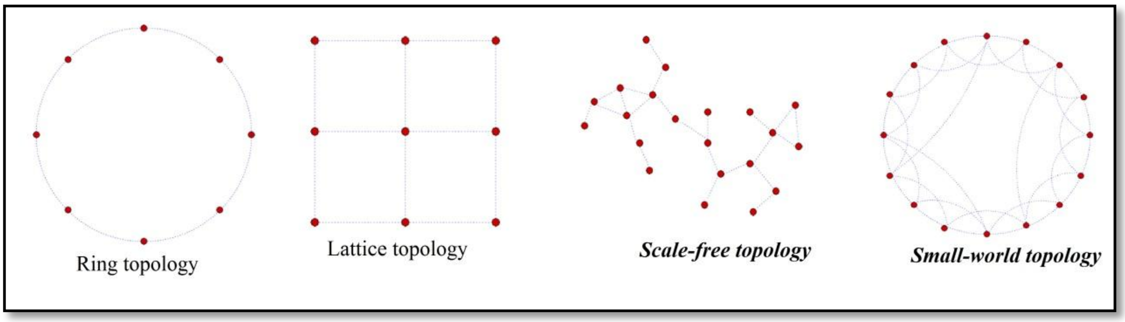


Figure 1. Different graph theory topologies, **Silvana, 2012.**

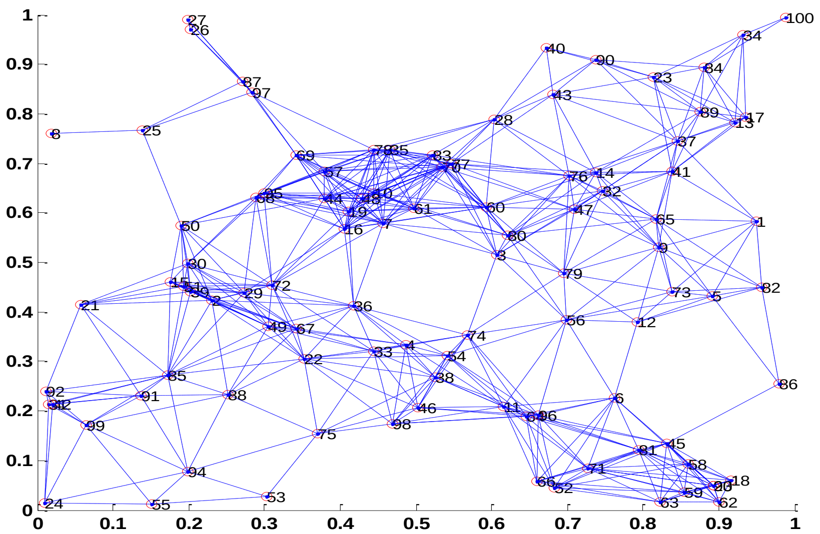


Figure 2. A random geometric graph theory topology with 100 nodes and radius 0.2.

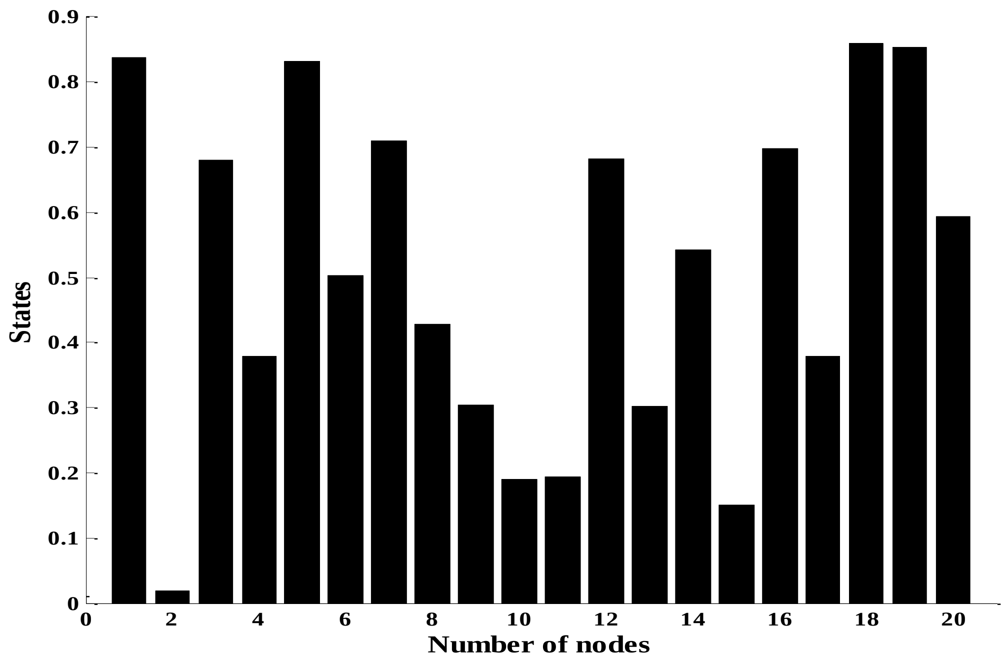


Figure 3. Random initial states for 20 nodes.

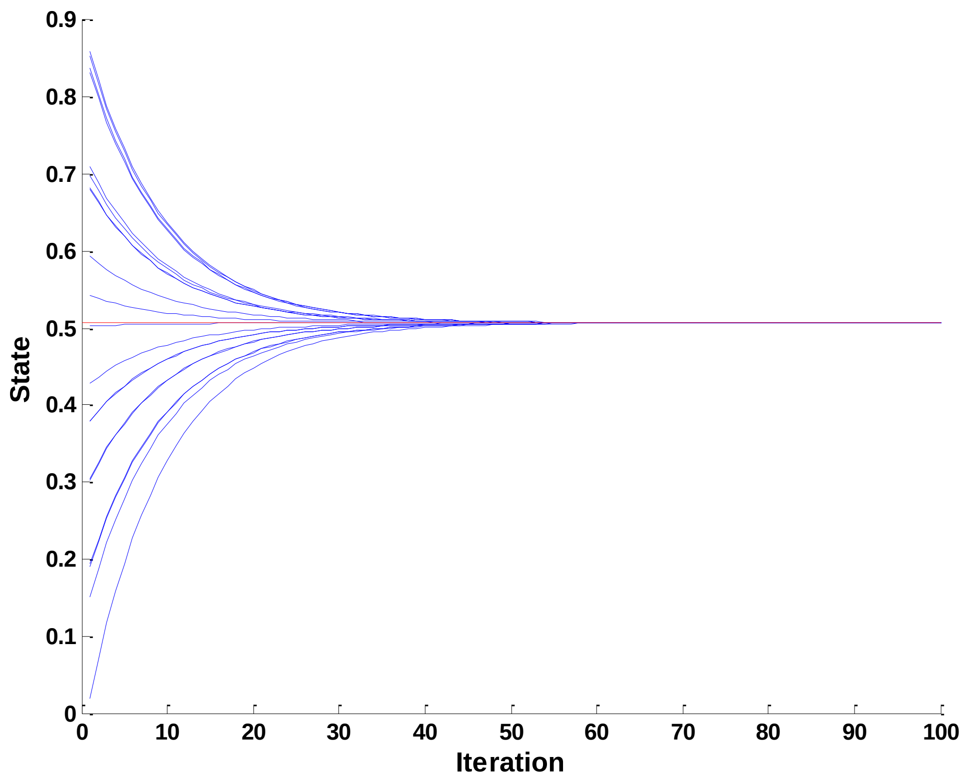


Figure 4. Twenty nodes reach the average consensus value.

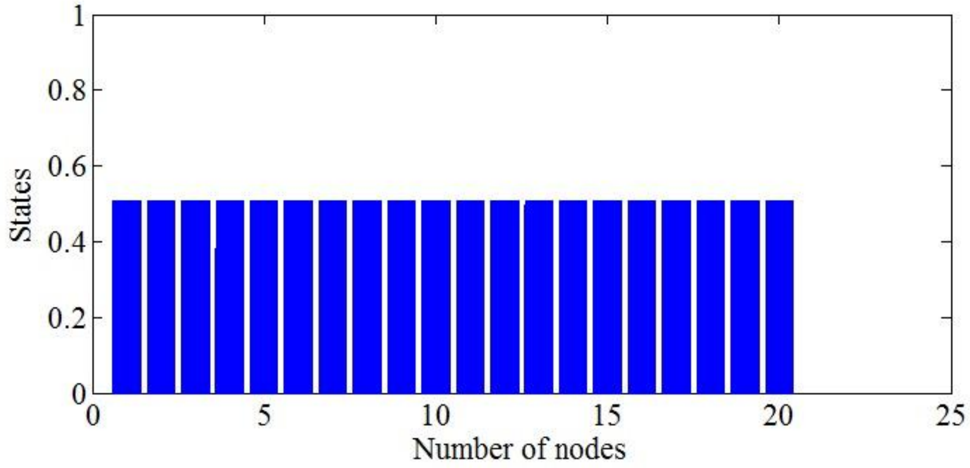


Figure 5. States of nodes after applying discrete consensus algorithm.

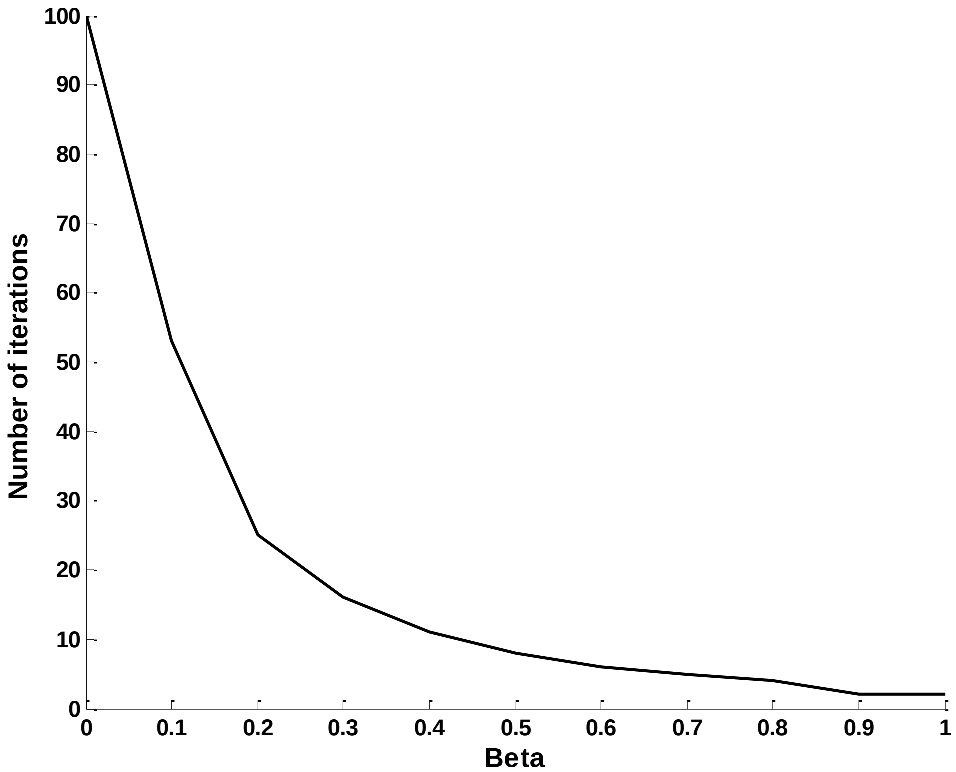


Figure 6. The relation between β and number of iteration.

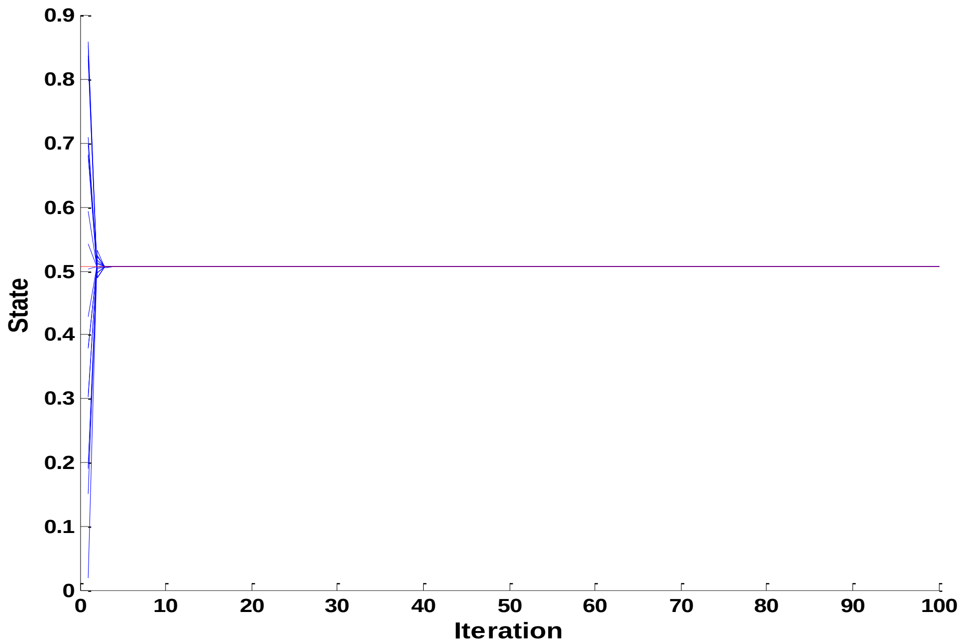


Figure 7. Reach the initial states of 20 nodes to the average consensus value with $\beta = 0.9$.

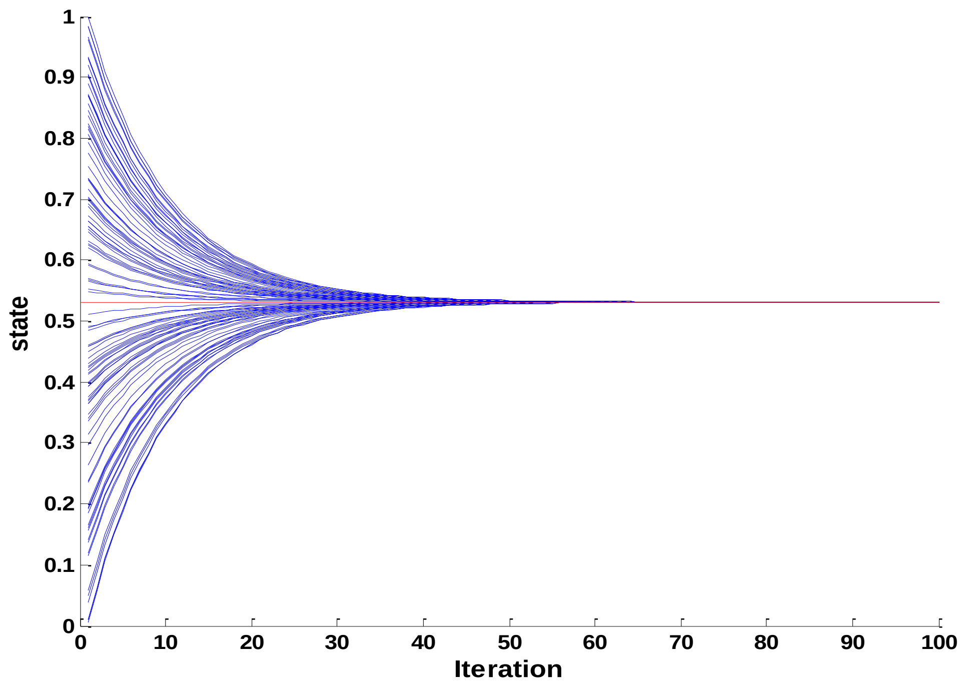


Figure 8. Reach the initial states of 100 nodes to the average consensus value with $\beta = 0.1$.

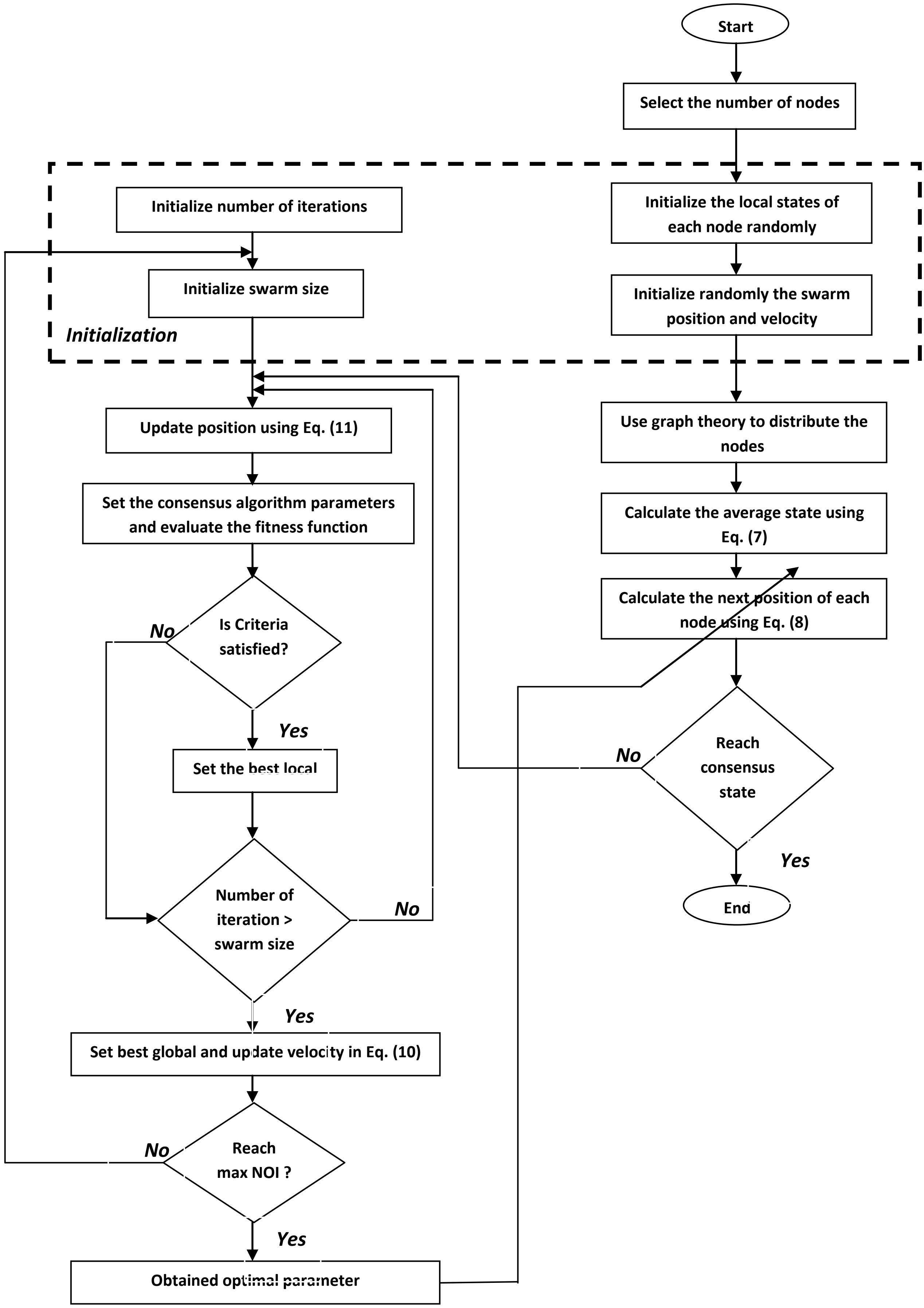


Figure 9. Flowchart for the integrated consensus algorithm with PSO technique.