An Improved Approach for Localization in Wireless Sensor Networks using PSO Algorithm

Seddik Rabhi, Mohammed said Belbali and Mahfoud Haissoune

✉rabhi_seddik@yahoo.fr

Laboratory of Mathematics, Modeling and Applications (LaMMA).Faculty of Sciences and Technology, University of Adrar. Algeria.

Received: 13 March 2021

Accepted: 29 May 2021

Published online: 30 June 2021

Abstract

After setting up a wireless sensor network, the first problem to be faced is the localization of sensors. Researchers have proposed a lot of techniques to solve the localization problem, but the precision resulting of these existing techniques still needs to be improved. In this paper, we propose a modified approach for improving the precision of localization in WSN, based on particle swarm optimization (PSO), that is one of the most successful metaheuristic inspired from nature for optimizing research in a multidimensional space. We adapt the PSO algorithm to search for the coordinates of unknown sensors, randomly distributed in a two-dimensional environment. We simulated our proposed model in the MATLAB simulation environment, and then in order to prove the success of our proposed approach, we conducted an analytical study in one of our experiments where we compared this approach with Simulated Annealing. The obtained results showed the effect of network configuration and the number of particles used on our proposed approach on the one hand, and demonstrated its superiority to simulated annealing on the other hand.

Keywords Wireless Sensor Networks (WSNs), Localization, Particle Swarm Optimization, PSO, Meta-heuristic, Precision.

1. Introduction

Wireless sensor networks (or WSN) were born in the context of technological globalization. They are made up of small units deployed in a certain environment and dedicated to accomplishing a given task. Each unit is a stand-alone microcomputer accompanied by a short-range wireless radio, as well as a series of sensors. The WSN make it possible to measure within an environment the variations of various physical quantities such as temperature, humidity, luminosity, sounds, vibrations, magnetic fields and many others phenomena [1].

Wireless sensor networks have aroused a certain interest in the scientific world, resulting in a considerable technological advance. Especially in the fields: military, medical, agriculture, environmental, precision and many others.

What is interesting and new compared to conventional computers: the sensor is integrated into the real world and has an unknown position and may change over time [2]. It is equipped with sensors which allow it to measure physical quantities linked to the environment and to time and it can communicate with neighboring nodes [1].

In general, localization consists of finding the locality (or position) of one or more objects. Nowadays, there is a real demand for automatic tracking: a company like DHL wants to know where their packages are, the delivery man wants to find his way, the fire chief wants to know where his men are, parents want to know where their children~~.~~

Knowledge of the positions of the sensors in the environment is often desirable, in order to be able to determine the origin of the flow of collected measurements. The purpose of the location methods is to estimate these positions automatically. A localization method in sensor networks is made up of two parts: 1. Estimation of distances, in this first phase the nodes communicate with each other and collect various indicators of radio communication quality between two nodes. The distance estimation can be done on the basis of different indicators: The propagation time of a wave (TDOA: time difference of arrival and TOA: time of arrival) [3]. The signal strength at reception (RSSI: Received Signal Strength Indicator) [4]. The rate of errors corrected during transmissions (LQI: Link Quality Indicator). 2. Derivation of positions, the aim of this second phase is to find the positions of the nodes which best respect the estimated inter-node distances [3]. If we know the position of a few nodes in the network in a certain coordinate system, the positions of other nodes in this coordinate system can be found. In this part the following geometrical principles are used: trilateration, multilateration and triangulation [2].

The existing automatic localization systems are numerous and use several technological paths, they are very different from each other and generally meet different needs (localization in a room or on the scale of the planet, precise or imprecise localization, and localization of one or more objects, static location or target tracking...). But despite all these methods available, the best way to locate is still an open problem. This is due to the lack of resources for storage, energy and capacity of calculation in the WSN. This impairs the accuracy of positioning methods which leads us to look for ways to improve it [2].

In IT (information technology), whenever a problem is needs to be improved, we resort to the optimization. Currently, the most used techniques in optimization are the metaheuristics. Meta-heuristics form a family of algorithms inspired by nature aimed at solving difficult optimization problems.

Several metaheuristic method have been invented to solve the optimization problem such as: simulated annealing (SA) proposed by Kirkpatrick et al. in 1983 [5], artificial immune system 1988 was proposed by Farmer et al. [6] , In 1989, Goldberg published genetic algorithms [7]. In 1992, Dorigo described ant colony optimization [8], In 1995, Russel Eberhart and James Kennedy proposed the particle swarm optimization [9]. In 1997, Storn and Price proposed the differential evolution algorithm [10]. 2008, Simon proposed a biogeography-based optimization algorithm [11], In 2009, Xin-She Yang and Suash Deb proposed cuckoo search [12]. In 2010, Yang [13] proposed the *bat-inspired algorithm*.

Recently some metaheuristics have been adapted to solve the problem of localization in wireless sensor networks, in the main objective is to improve the accuracy of the localization and minimize the use of network resources, such as: simulated annealing of localization named SAL [14], the semi definite programming localization technique [15], genetic algorithm is presented in [16] this metaheuristic minimizes the use of anchors nodes compared to the previous method. Recently, improved method based on Bees optimization algorithms, Firefly and Cuckoo Search algorithms have been proposed in [17] [18] [19]. In [20] and [21] two algorithms based on the fruit fly Optimization algorithm (FOA) are proposed for node localization in WSN.

In this paper, we invited the meta-heuristic of the particle swarm optimization (PSO), for improving the precision of localization in WSN. So, we use the research capacity of particles swarm to find the coordinates of the unknown nodes randomly distributed in a two-dimensional space. The PSO as bio-inspired approach is used, since it has proven its success in several fields of optimization and to benefit from the advantages of meta-heuristics.

The rest of the paper is organized as follows: The next Section explained biological and artificial aspects of particle swarm optimization. In the section 3, the proposed localization approach is presented. Section 4 presents simulation results and analytical study of localization by our approach. The conclusion and future works are stated in Section 5.

2. Particle Swarm Optimization

2.1. Biological Aspect

Particle Swarm Optimization (PSO) is an evolutionary algorithm that uses a population of candidate solutions to develop an optimal solution to the problem. This algorithm was proposed by Russel Eberhart (electrical engineer) and James Kennedy (socio-psychologist) in 1995 [9]. It was originally inspired by the living world, more precisely by the social behavior of animals evolving in swarms, such as schools of fish and grouped flights of birds. In fact, we can observe in these animals relatively complex movement dynamics, while individually each individual has a limited "intelligence", and has only a local knowledge of his situation in the swarm.

Particle Swarm Optimization (PSO) is an evolutionary algorithm that uses a population of candidate solutions to develop an optimal solution to the problem. This algorithm was proposed by Russel Eberhart (electrical engineer) and James Kennedy (socio-psychologist) in 1995 [9]. It was originally inspired by the living world, more precisely by the social behavior of animals evolving in swarms, such as schools of fish and grouped flights of birds. In fact, we can observe in these animals relatively complex movement dynamics, while individually each individual has a limited "intelligence", and has only a local knowledge of his situation in the swarm.

Local information and the memory of each individual are used to decide where to move. Simple rules, such as "stay close to other individuals", "go in the same direction" or "go at the same speed", are sufficient to maintain the cohesion of the swarm, and allow the implementation of complex collective behaviors and adaptive.

The particle swarm is a population of simple agents, called particles. Each particle is considered as a solution of the problem, where it has a position (the solution vector) and a speed. In addition, each particle has a memory allowing it to remember its best performance (in position and in value) and the best performance achieved by the "neighboring" particles (informants): each particle has indeed a group of informants called neighborhood.

This method finds its source in observations made during computer simulations of grouped flights of birds and schools of fish by Reynold, Heppner & Grenander. In other words, it is strongly inspired by the observation of the gregarious relationships of migratory birds, which in order to travel “long distances” (migration, foraging for food, aerial displays, etc.), must optimize their movements in terms of energy expended, time, (etc.), for example the V-shaped formation shown in Fig .1.

Fig .1 Eagles fly in a V shape

The movement of this animals in swarms is complex, this dynamics obey very specific rules and factors:

- Each individual has a certain "limited" intelligence (which allows them to make a decision).
- Each individual must know their local position and have local information on each individual in their vicinity.
- Obey these three simple rules, “stay close to other people”, “go in the same direction” or “fly at the same speed”.

All this factors and rules are essential for maintaining cohesion in the swarm, through the adoption of complex and adaptive collective behavior

2.2. Artificial Aspect

In a two-dimensional search space, the particle i of the swarm is modeled by its position vector Xi(xi1, xi2) and by its speed vector Vi(vi1, vi2) (see Fig .2.). The quality of its position is determined by the value of the objective function at that position (This function is defined according to the problem we are going to study). This particle keeps in memory the best position through which it has ever passed, which we denote by Pbest i = (pbest i1, pbest i2). The best position reached by the particles of the swarm is denoted by Gbest i = (gbest i1, gbest i2). We refer to the general version of PSO, where all particles in the swarm are considered to be neighbors of particle i, hence the Gbest(global best) notation.

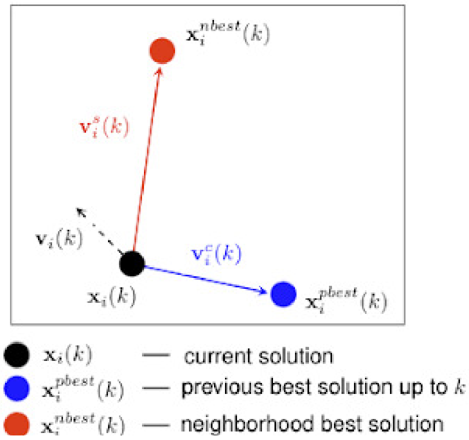

Fig .2 PSO solution updates in 2D

According to the features of the Particle Swarm, the PSO's main steps can be illustrated in the following flowchart of Fig.3. [9]:

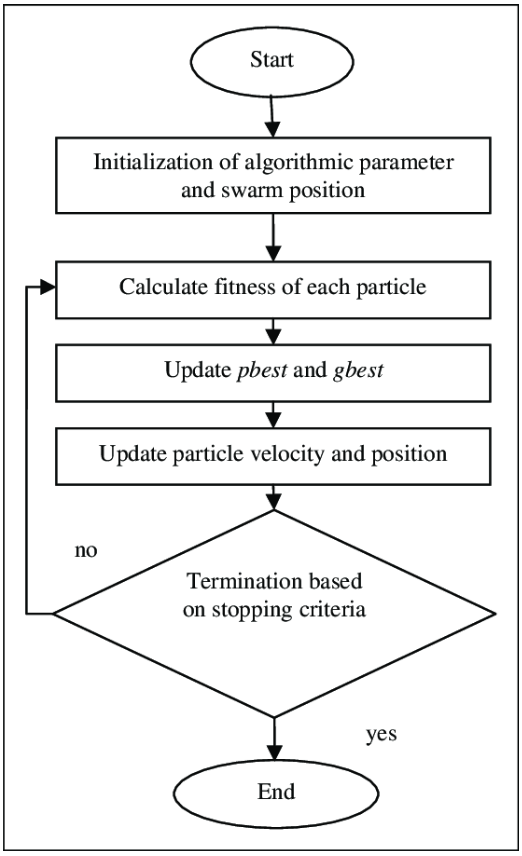

Fig .3 Flowchart of PSO algorithm

3. WSN Localization Using PSO

We have a total number N of the captures represented by nodes deployed in a two-dimensional area of interest represents the environment of WSN, and m> = 0 a fixed number of nodes called anchors whose positions are known a priori either by GPS or by placing it manually during deployment phase. The objective of the localization procedure is to find the positions of n = N-m target nodes whose positions are unknown.

The estimation of positions of the unknown nodes can be formulated as an optimization problem, minimize a fitness function represents the difference between calculated distance and the estimated one by one of measurement technology (RSSI, TDOA, ...). And this is between the unknown node and the anchors in their neighborhood.

If (x, y) the coordinates of unknown node, and (xi, yi) the coordinates of the i^{th} anchor of its direct visibility, the distance between the unknown node and the different anchors will be calculated using the following Euclidean distance equation :

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2} \qquad (1)$$

The estimated distance is represented by $\hat{d}_i = d_i + N_{noise}$knowing that N_{noise}represents the error made during the estimation of the distance by a measurement technology like RSSI, and in the simulation it is calculated by the following Gaussian measurement noise. The fitness to be minimized in the nodes' localization approach can be showed as follows:

$$f(x, y) = \frac{1}{M} * \sum_{i=1}^{M} \left(d_i - \hat{d}_i\right)^2 \qquad (2)$$

Where, M ≥ 3 is the number of neighborhood beacon to node to be evaluated.

So the proposed approach of localization, executes the process illustrated by the following steps:

1. Choose the parameters of the network: area, the number of (N, known nodes and m, unknown nodes) and radios of communication R. then we fix the parameters of PSO algorithm (K: number of individual in the swarm, r1, r2, c1, c2, w and the number of generation).
2. Each anchor in the network diffuse a message allows other nodes to locate ourselves.
3. Unknown nodes with three or more beacon near them are considered **calculable**, while others with less than three wait for the next iterations to determine her location.
4. For all calculable nodes, define the initial search places by centroid method. Then, evaluate this initial position by fitness and consider its values as $Gbest_i$

$$(x_{init}, y_{init}) = \left(\frac{1}{m} \sum_{i=1}^{m} x_i, \frac{1}{m} \sum_{i=1}^{m} y_i\right) \qquad (3)$$

5. The set of K particles is randomly placed around the initial position of the unknown node with radius R. The $Pbest_i$ is initialized with fitness value of this place of each individuals. In addition, the initial velocity of all particles is set to zero.

6. Move the swarm according to:

$$v_{i,j}^{t+1} = wv_{i,j}^{t} + c_1 r1_{i,j}^{t}\left[pbest_{i,j}^{t} - x_{i,j}^{t}\right] + c_2 r2_{i,j}^{t}\left[gbest_{i,j}^{t} - x_{i,j}^{t}\right], j \in \{1,2,\ldots,D\} \quad (4)$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + v_{i,j}^{t+1}, j \in \{1,2,\ldots,D\} \quad (5)$$

7. Evaluating of each new locations of particle using fitness calculated by the equation 2.

8. Update $\vec{P}best_i$ and $\vec{G}best_i$ according to:

$$\vec{P}best_i(t+1) = \begin{cases} \vec{P}best_i(t), si\ f(\vec{x}_i(t+1)) \geq \vec{P}best_i(t) \\ \vec{x}_i(t+1),\ if\ not \end{cases} \quad (6)$$

$$\vec{G}best_i(t+1) = argmin_{\vec{P}best_i}\, f\left(\vec{P}best_i(t+1)\right), 1 \leq i \leq N \quad (7)$$

9. Steps 6 to 8 are repeated until the maximum number of iterations is reached. Then we consider the location of $\vec{G}best_i$as the location of the target node i.

10. Then the unknown node whose location was determined sends a message to help the nodes that were initially considered incalculable to locate them. Thus, our proposed model ensures that all unknown nodes can be located.

4. Simulation Results And Evaluation

To prove the efficacy and ability of our proposed approach, we have implemented and simulated it in a MATLAB environment. For achieving our simulations, we used a computer with the following specifications: laptop Lenovo with CPU I3, 4GB RAM and 500GB HDD and operating system (Windows 10).

In order to adjust the various parameters necessary for the implementation of our proposed algorithm, we have designed a simulator with a user-friendly interface. There are several parameters that we need which lead to a good adaptation of the network configuration and the characteristics of the proposed algorithm. The following Fig .4 shows the interface of our application which offers the possibility of modifying the parameters used in order to obtain good results.

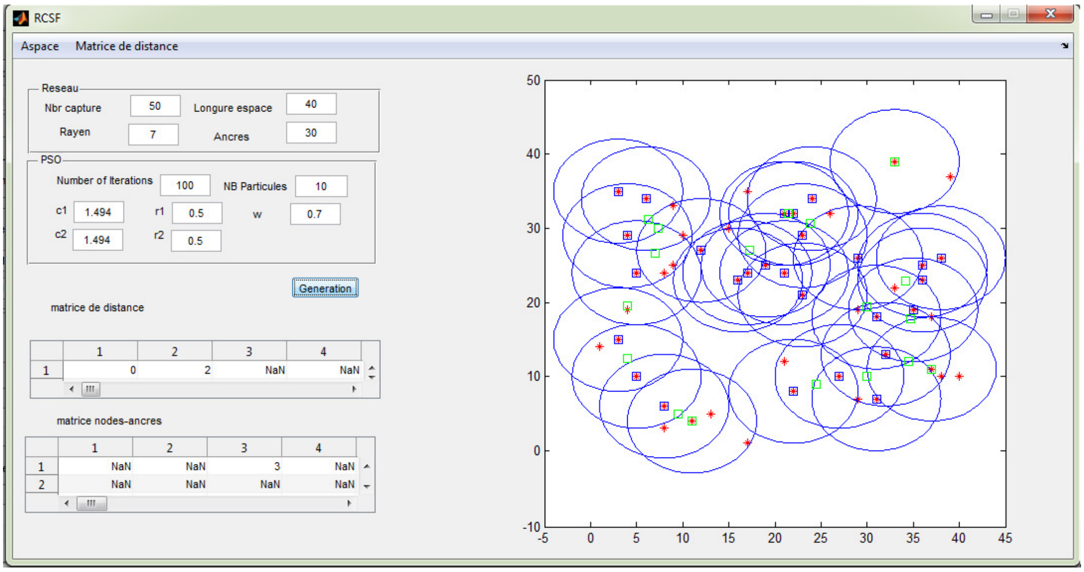

Fig .4. Interface of simulation

4.1. Evaluation Metric

In our simulation, we will focus on the precision of localization, as this is the most critical challenge in determining positions in a wireless sensor network; this metric represents the average of the differences between the actual positions and those estimated for all unknown nodes. The precision (PRE) calculated as follows:

$$PRE_l = \frac{\sum_i^N \sqrt{(X_{fi}-X_{ri})^2+(Y_{fi}-Y_{ri})^2}}{N\times R} \qquad (8)$$

Where,

Xr, Yr: the real positions of unknown node

Xf, Yf: the positions found by the algorithm to be evaluated.

N: number of nodes.

R: Transmission radius.

4.2. The influence of the parameters on precision of our approach

In order to study the effect of network configuration on the efficiency of our proposed approach, we carry out the following studies. Firstly, we vary the density of anchors used between 10% and 30%, the obtained results of the precision of localization are showed in the Fig .5. In this experiment we used the network configuration and algorithm parameters as follows:

- Network configuration: Number of sensor N=100, area size 40x40 m, radius of transmission R= 10m.
- PSO parameters: Number of particle k = 50, c1 = c2 = 1.494, r1 = r2 = 0.5, w = 0.7.

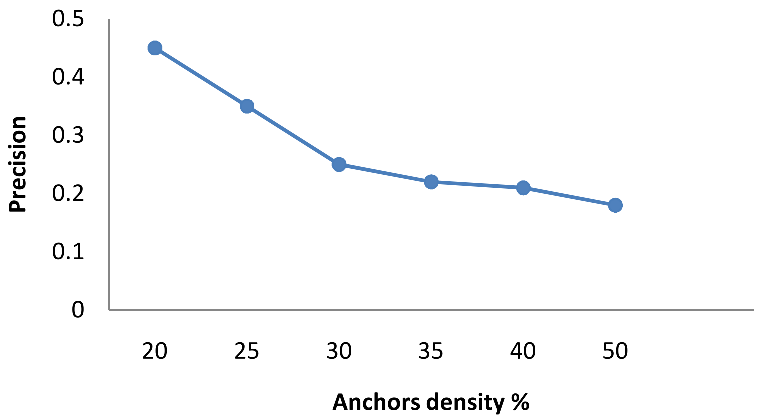

Fig .5. Precision of localization vs. number of anchors

The following figure shows the result of the localization accuracy in our proposed approach, if we vary the transmission radius R used between 10 and 35m. in this second experiment we are using the following parameters:

- Network configuration: Number of sensor N=100, area size 40x40 m, density of anchors = 30%.
- PSO parameters: Number of particle k = 50, c1 = c2 = 1.494, r1 = r2 = 0.5, w = 0.7.

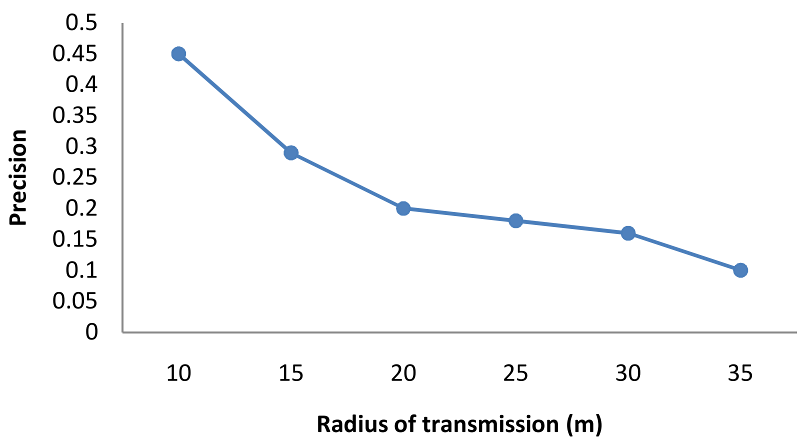

Fig .6 Precision vs. radius of transmission.

Through the two previous diagrams of results, we prove the great effect of network configuration on the accuracy of localization. From the first experiment, we conclude that the accuracy of localization increases with the increase of anchors density. That is confirmed by the second experiment, because increasing the radius of sensors automatically increases the impact size of the anchors not increasing their number, which increases the accuracy of localization.

In the last experiment we evaluated the effect of changing the number of individuals in the swarm. For this, we vary number of particles between 20 to 50 individuals and we fixed the network configuration and the PSO parameters in the following values:

- Network configuration: Number of sensor N=100, area size 40x40 m, density of anchors = 30%, Radius of transmission= 30m.
- PSO parameters: c1 = c2 = 1.494, r1 = r2 = 0.5, w = 0.7.

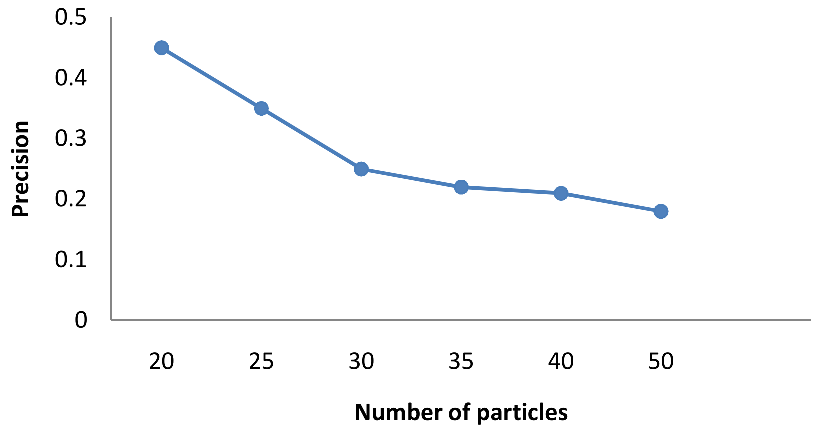

Fig.7.Precision Vs Number of particles.

The simulation result presented in Fig .7 shows that increasing the number of particles will improve the localization performance in the proposed approach; this is because increasing the number of individuals in the swarm expands the search process and increases the potential solutions, which positively affects the accuracy of localization.

4.3. Comparison of proposed approach with Simulated Annealing

In order to ensure comparison justice, we use the same network configuration for two intelligent techniques: Number of sensor N=100, area size 40□40m, density of anchors=30%, Radius of transmission= 30m. The obtained results of the comparison are shown in the Fig .8. These results demonstrate the superiority of our proposed approach over the approach based on the simulated annealing.

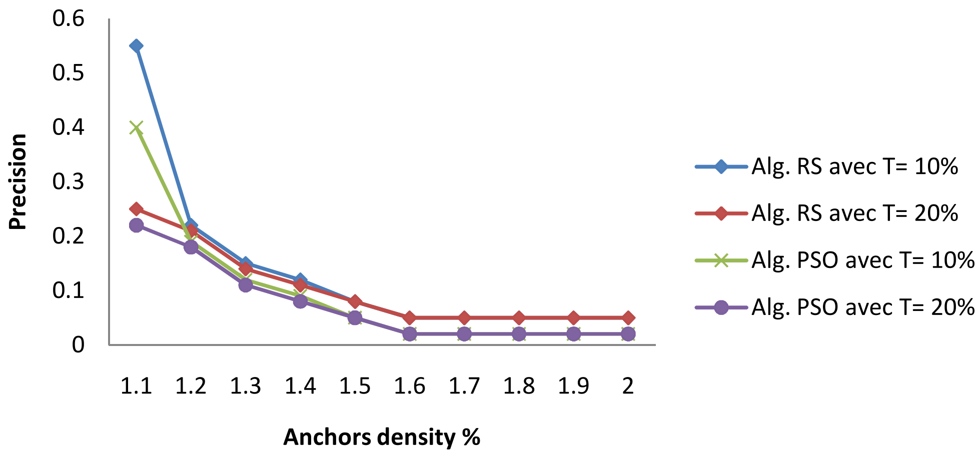

Fig.8 PSO Vs simulated annealing

5. Conclusion

In this paper, we have proposed a modified PSO model to improve precision of localization in wireless sensor networks, based on one of the most efficient meta-heuristic in the combinatorial optimization. We implemented our approach of localization in the MATLAB environment of simulation. To prove the efficacy of our proposed approach, we conducted some experiments. Through it, we demonstrated the effect of network characteristics on the accuracy of localization. Then we finally compared our proposed approach with simulated annealing, the results of comparison proved the superiority of PSO to simulated annealing in term of precision of localization. This approach is intended for static networks and 2D environments, and we hope to provide a version of this method for mobile networks and 3D environments.

References

[1] Boukerche, A., Oliveira, H. A. B. F., Nakamura, E. F., & Loureiro, A. A. F. (2007). Localizationsystems for wireless sensor networks. Wireless Communications, IEEE, 14(6), 6–12.

[2] Paul, A. K., & Sato, T. (2017). Localization in Wireless Sensor Networks: A Survey on Algorithms, Measurement Techniques, Applications and Challenges. Journal of Sensor and Actuator Networks, 6(4), 24.

[3] Cho, H. H., Lee, R. H., & Park, J. G. (2011). Adaptive parameter estimation method for wireless localization using RSSI measurements. Journal of Electrical Engineering and Technology, 6(6), 883-887.

[4] D. Moore, J. Leonard, D. Rus, and S. Teller, “Robust distributed network localization with noisy range measurements,” in Second Int. Conf. on Embedded Networked Sensor Systems Nov. 2004.

[5] Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)

[6] Farmer, J.D., Packard, N.H., Perelson, A.S.: The immune system, adaptation, and machine learning. Phys. D **2**(1–3), 187–204 (1986).

[7] Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1st edn. Studies in Computational Intelligence. Addison-Wesley Longman (1989).

[8] Dorigo, M.: Optimization, learning and natural Algorithms. Ph.D. thesis, Politecnico di Milano, Italy (1992)

[9] Kennedy, J., and Eberhart, R.: ‘Particle swarm optimization’. Proc. IEEE Int. Conf. Neural Networks, Perth, Australia, Jan 1995, pp. 1942-1948

[10] Simon, D.: Biogeography-based optimization. IEEE Transactions on Evolutionary Computation 12(6), 702–713 (2008)

[11] Storn, R.M., Price, K.V.: Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization **11**(4), 341–359 (1997)

[12] Yang, X.S., Deb, S.: Cuckoo search via Lévy flights. In:*World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, Coimbatore, India, IEEE Conference Publications, pp. 210–214. IEEE Press, Piscataway, NJ (2009).

[13] Yang, X.S.: A new metaheuristic bat-inspired algorithm. In: J.R. González, D.A. Pelta, C. Cruz, G. Terrazas, N. Krasnogor (eds.) *Nature Inspired Cooperative Strategies for Optimization(NICSO 2010)*, vol. 284, Chap. 6, pp. 65–74. Springer, Berlin Heidelberg (2010).

[14] Kannan, A. A., Mao, G., & Vucetic, B. (2006). Simulated annealing based wireless sensor network localization. JCP, 1(2), 15-22.

[15] Biswas, P., & Ye, Y. (2004, April). Semidefinite programming for ad hoc wireless sensor network localization. In Proceedings of the 3rd international symposium on Information processing in sensor networks (pp. 46-54). ACM.

[16] Nan, G. F., Li, M. Q., & Li, J. (2007, August). Estimation of node localization with a real-coded genetic algorithm in WSNs. In Machine learning and cybernetics, 2007 International Conference on (Vol. 2, pp. 873-878). IEEE.

[17] Moussa, A., & El-Sheimy, N. (2010, December). Localization of wireless sensor network using bees optimization algorithm. In Signal Processing and Information Technology (ISSPIT), 2010 IEEE International Symposium on (pp. 478-481). IEEE.

[18] Cao, S., Wang, J., & Gu, X. (2012). A wireless sensor network location algorithm based on firefly algorithm. In AsiaSim 2012 (pp. 18-26). Springer, Berlin, Heidelberg.

[19] Goyal, S., & Patterh, M. S. (2014). Wireless sensor network localization based on cuckoo search algorithm. Wireless personal communications, 79(1), 223-234.

[20] Rabhi, S., Semcheddine, F. (2019). Localization in wireless sensor networks using DV-hop algorithm and fruit fly meta-heuristic. Advances in Modelling and Analysis B, Vol. 62, No. 1, pp. 18-23. https://doi.org/10.18280/ama_b.620103.

[21] S. Rabhi., F. Semcheddine and N. Mbarek, An Improved Method for Distributed Localization in WSNs Based on Fruit Fly Optimization Algorithm, Autom. Control Comput. Sci., 2021, vol. 55, no. 3, pp. 286–296.