Original Scientific Article Journal of Decision Analytics and Intelligent Computing
Vol. 3 issue 1, (2023) 113-121 https://doi.org/10.31181/jdaic10005082023t

A method of finding optimal number of clusters in a wireless network based on power efficiency using MOPSO

Harinandan Tunga[1,*] and Debasis Giri[2]

[1] Department of Computer Science & Engineering, RCC Institute of Information Technology, Kolkata, India

[2] Department of Information Technology, Maulana Abul Kalam Azad University of Technology, Haringhata, West Bengal, India

* Correspondence: harinandan.tunga@rcciit.org.in

Received 12 March 2023

Accepted for publication 29 July 2023

Published 5 August 2023

Abstract

Sensors play an important role in monitoring, detecting, recording and recording the physical and environmental conditions of a particular place. These physical conditions mainly include temperature, sound, wind, etc. These nodes are connected to each other via a transmission channel and the nodes are battery-operated. So, energy efficient algorithms are needed to reduce energy consumption in the overall setup and increase the lifetime of sensor nodes. In this paper, we propose a method to solve the problem of routing in "Wireless Sensor Networks", forming clusters such that there are minimal node transfers and the overall energy consumption of the system is reduced. We have implemented Multiple Objective Particle Swarm Optimization (MOPSO) algorithm to devise a technique that considers several parameters like temperature, minimal clustering, routing path distance and energy efficiency to obtain optimal clusters.

Keywords: clustering, energy reduction, Multiple Objective Particle Swarm Optimization (MOPSO), optimality, wireless sensor networks

1. Introduction

Wireless sensor network (WSN) refers to a group of spatially dispersed and dedicated sensors for monitoring and recording the physical conditions of the environment and organizing the collected data at a central location (Deepika and Niranjan, 2015). A wireless sensor network can be thought of as multiple nodes in an area that continuously relays information from one node to the other for processing and communication. A sensor can have both a processing unit as well as relay unit.

Wireless Sensor networks have wide applications. The military initially used sensors, but they got wider applications over time. Sensor networks were developed by different application motivations for different places, such as impassable areas and disaster areas, and they also have monitoring applications on the battlefield in military (Taheriana et al., 2015). A wireless sensor network can have various types of sensor nodes to measure some physical quantities or environmental conditions such as temperature, light, humidity, sound, motion of the pollutant, etc.

The deployment of such networks still suffers from multiple limitations, such as energy consumption, connectivity between the nodes, quality of data transmission and covering the Region of Interests (ROIs). An alternative method for deploying wireless sensor networks called Multi-Objective Particle Swarm Optimization (MOPSO) was proposed by Ibrahem et al. (2018). As wireless sensor nodes are placed in remote, inaccessible areas, they need to operate on a reliable energy source as data transmission needs to be continuously done and any harm to one node can disrupt communication over the entire network setup. Typically, Wireless-Sensor Network nodes operate on battery. This is a constraint on the overall stability of the setup.

Network users face a serious problem with energy loss. Energy-efficient algorithms and load balancing are used during the clustering algorithm. It proposes an algorithm based on Particle Swarm Optimization (PSO) technique to improve network lifetime. It helps in cluster formation as well as selection of Cluster Head (CH) (Yadav et al., 2018).

It is recommended that a Multi-Objective Particle Swarm Optimisation (MOPSO) procedure be utilised while attempting to resolve such an issue (Yarpiz, 2015). Coello Coello and Lechuga (2002) and Coello Coello et al. (2004) came up with the MOPSO proposal. It is a multi-objective variant of PSO that handles multi-objective optimisation problems in a manner comparable to the Pareto Envelope-based Selection Algorithm. It integrates the Pareto Envelope and grid construction system. An analysis of node deployment in a wireless sensor network in a warehouse environment monitoring system is demonstrated by Mao et al. (2019). Sharma et al. (2014) showed an analysis of transmission technology in wireless sensor networks. In the same way as PSO, particles in MOPSO exchange information with one another and work towards improving both the global best particles and their own personal (local) best memories (Kuila and Jana, 2014). In contrast to PSO, however, there is more than one criterion that can be used to regulate and describe the best (Clerc and Kennedy, 2002). Every non-dominated particle in the swarm is assembled into a sub-swarm called the Repository, and from among the members of the Repository, each particle chooses the member that will serve as its best target on a global scale. A domination-based and probabilistic set of rules is applied in order to determine the optimal personal or local particle. The determination of fitness values is accomplished by employing a thresholding strategy in conjunction with several characteristics (Li et al. 2014; Khan et al., 2016). It is therefore able to predict the best possible option. The routing algorithm is developed with a novel particle encoding scheme and fitness function to find the optimal routing tree that connects cluster Heads to the Base Station (Elhabyan and Yagoub, 2015).

Finding the optimal number of nodes to cluster together requires us to take into account the routing in the sensor network, which can provide us with nodes that frequently communicate with one another. We can lower the amount of energy that is sent across the system and bring it closer to a stable state by grouping the nodes together.

The following sections of this document are arranged as follows: Section 2 discusses the assumptions. Model description for the proposed task is presented in Section 3. Experimental results and discussions of the proposed technique are covered in Section 4. The last section is devoted to conclusions and provides some perspectives for future research.

2. Assumptions

Now, in order to discover the optimal routing path, we are taking into account parameters such as velocity and x, y coordinates. We continue to proceed in this fashion for each generation. The Optimality is computed in each state to determine if a proposed solution should be considered or disregarded. Initialize the number of nodes, repository limit, initial energy in the nodes (E_0=starting value of energy, *NodeNum*=number of nodes, *RepLimit*=maximum limit). Initializing quantity of steps and number of iterations in every seed (steps=number of step, iteration=integer value). We initialize transmission rate of nodes or rate of movement of energy from values attained from archive vector set.

The subsequent symbols, along with their meanings are given below in Table 1. These symbols have been utilized in the process of issue conceptualization and solution approach.

Table 1. List of parameters and variables used in the proposed process

Parameters and Variables	Description
xtarget, ytarget	Target solutions
x, y	Coordinates of routing path
HV	Hypermetric volume metric
K	Number of nodes
X	Solution within Ps
Hypercube (x)	Hypercube in the objective space
Distance	Routing distance
PresenceSize	Area of influence of a node
Sink	Its current position
Priority	Order of node in the network
Xi	Current node
Xi-1	Previous node
Rate	Rate of movement of energy between nodes.
E0	initial energy value
NodeNum	number of nodes
Replimit	maximum limit
Energy (ii.PriorityEnergy (jj))	energy of transmission
NewX	Dead nodes
NewV	Gloal best

3. Model description

To find the ideal clustering number, we should consider routing in the sensor network, and it can give us nodes that often communicate with each other. By clustering them, we can reduce energy transmission thus, the system can reach a stable state.

Now, we are considering parameters like velocity and x, y coordinates to find the routing path. The Optimality is computed in each state to consider or disregard the proposed solution, and we continue in this manner for each generation. The process of combining the solutions can be explained as combining the current solution with the target solution. This can be conducted by initially determining the nearest sensor in the target solution, which can be represented in the following equation (1):

$$(xnew, ynew) = (x, y) + Rand * ((xtarget, ytarget) - (x, y)h_k\,(x) = 0 \quad k = 1,2 \ldots K \tag{1}$$

Hypermetric volume metric gives information about the closeness and correlation between a set of non-dominated solutions. It aims to calculate the volume covered by solutions in objective space.

Let x be a solution within $P\ s$, the Hypercube(x) will be initiated by taking into account both W and x as the corners of the Hypercube in the objective space. In this regard, HV can be computed by the volume of the union of the Hypercube as in the following equation (2):

$$HV = volume\ (\cup\ x\ \in Ps\ Hypercube(x)) \tag{2}$$

In fact, the greater value of HV indicates superior performance.

The routing distance is calculated using equation (3):

$$dis\tan ce = \sqrt{(\sin k.x - \Pr iorityX\,(\Pr esenceSize(2)))^2 + ((\sin k.x - \Pr iorityY\,(\Pr esenceSize(2)))^2} \tag{3}$$

Where the PresenceSize represents an area of influence of a node and sink represents its current position. Priority represents the order of nodes in the network. The movement of the node could be found using equation (4):

$$X_i = X_{i-1} + (sign\big((rand * 2) - 1\big) * rand() * Rate) \tag{4}$$

Where X_i is the current node and X_{i-1} is the previous node. Rate signifies the rate of movement of energy between nodes.

Proposed Multi-Objective Algorithm. From the above discussion we are going to design MOPSO based algorithm to determine the optimal number of clusters which is given below (Algorithm 1).

Algorithm 1. MOPSO for determining optimal number of clusters

Input: All parameters (number of nodes and initial energy of each node)

Output: Optimal number of clusters in the network.

Steps:

1. Initialize the number of nodes, initial energy in the nodes, repository limit.
   (E_0=initial energy value, NodeNum=number of nodes, RepLimit=maximum limit)
2. Now we initialize no of steps and number of iterations in each seed.
   (iteration=integer value, steps=number of step)
3. We initialize rate of movement of energy or transmission rate of nodes from values obtained from archive vector set.
4. For each iteration find the energy of transmission after round taking into account routing distance and rate of transmission using equation (5)

$$Energy\big(ii.PriorityEnergy(jj)\big) = E\big(PriorityEnergy(jj)\big) - \big((ETX + EDA) * (4000) + Emp * 4000 * (distance * distance * distance * distance)\big) \tag{5}$$

5. Select clusters based on optimality values, based on condition.
   - $Temp = CurFitness \geq LBest;$
   $Result = (sum(Temp, 2) == ObjNum);$
   OR
   - $Temp = CurFitness \leq LBest;$
   - $Result = (-1) * (sum(Temp, 2) == ObjNum) + Result;$
6. Find dead nodes (which do not have energy left) in them using equation (6). Eliminate them and put in separate clusters.

$$NewX = (Energy \leq 0) * \left(-\frac{XBound}{2}\right) + \sim(Energy \leq 0) * NewX \quad (6)$$

7. For each particle in the swarm:
   (a) Select leader from the archive, and obtain global best using equation (7)
   $$NewV = Weight * V + C1 * \left(rand\left(1, particleSize(2)\right) * (LBestP - X)\right) + C2 * \left(rand\left(1, particleSize(2)\right) * (GBestP - X)\right) \quad (7)$$
   (b) Update clusters as in step 5.
   (c) Update order of nodes, from PSO (non-dominated pairs) using equation (8)
   $$[CurFitness \quad Energy] = MultiObjFitness(GbestAC, GbestVal, X, Y, R, E, next, sender, ETX, EDA, Emp, PlotSize, do, Efs, 2, SenderIndex) \quad (8)$$
   (d) Update energy values of nodes from step 4.
9. Update the archive of non-dominated solutions
10. Compare the solutions obtained from each iteration in PSO non-dominated pairs and check for optimality.
11. Use PSO algorithm for multiple parameters and choose the one that is suitable after comparing.
12. Update routing parameters, x-y co-ordinates and compute total energy of the system.
13. Repeat for each seed (updation), go to step number.

The solution works well if the number of iterations for each step, and number of steps increases allowing the seeding of the archive and generating dominated pairs.

In the above-proposed algorithm, we first initialize parameters and set x-y routing values for each node in the generation. Now for each iteration, we compute distance vectors, x-y coordinates and movement velocity. Next, we compute energy values for each node and then compare non-dominated solutions using a fitness curve and evaluate the optimality using hypermetric volume. This gives us the most optimal solution and the routing path that is to follow. Also, we compute clusters using nearest-neighbor algorithm based on the fitness curve. We calculate the average energy of nodes in the network and compute the total energy in the system. Now we repeat this process for subsequent iterations.

4. Experimental results and Discussions

For our experimental run, we chose 1000 nodes of sensors and ran the algorithm for about 150 iterations. We assigned all the 1000 nodes initial energy of 8 units and placed these nodes in a field of random x-y coordinates for the sake of consideration. The parameter settings that were utilized for the simulation are detailed in Table 2.

Table 2. Parameter Settings

Parameter	Value
Pop Number	30
Velocity Limit	3.83
Repository Limit	100
Initial Energy	8
Number of Nodes	400
Maximum number of rounds	5
xRank	13.33

The minimal number of clusters that are produced from sensor nodes in the network is displayed in Figure 1, and it is shown for each iteration of the algorithm. As the algorithm continues through each iteration, we observe a trend toward a lower average number of clusters. This is consistent with our expectations. This demonstrates that our algorithm is discovering the best routing path, which assists in the grouping of nodes that are related to one another.

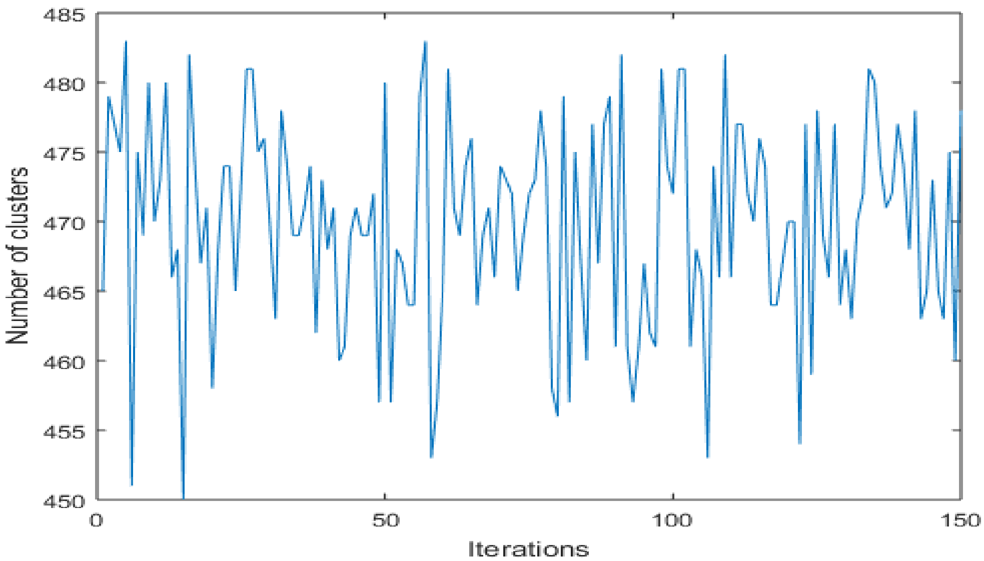


Figure 1. Minimal clusters obtained for 150 iteration of the algorithm

In the beginning of our presentation, we mentioned that one of our goals was to determine the minimum number of clusters necessary so that, following the clustering of related nodes, we could lower the amount of energy that was transmitted throughout our system. Following each cycle, there is a discernible decrease in the amount of energy distributed over the nodes, which, as can be seen in the graphic that was just presented, achieves our goal.

We can observe that the energy values are decreasing as they move closer and closer to the minimum value by looking at Figure 2. The final step, which is depicted by the blue line, has a precipitous decline, which we can recognize as the answer to the problem. Additionally, we discover that the energy values for dead nodes and clusters with minimal values are sharp; nevertheless, the energy values for clusters with bigger values decline as the cluster values increase. In addition, we can observe that as our steps increase, our energy levels have a tendency to decrease, which indicates a stable or viable environment, which is the primary goal that we are working toward. The transmission of energy is drastically decreased if there is a decrease in the distance between frequent nodes (which results in the formation of ideal clusters).

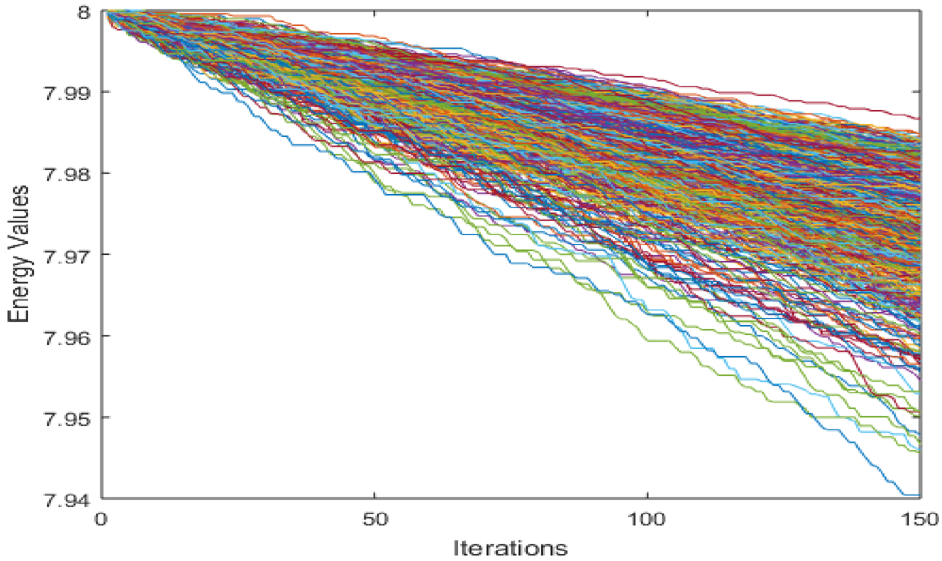
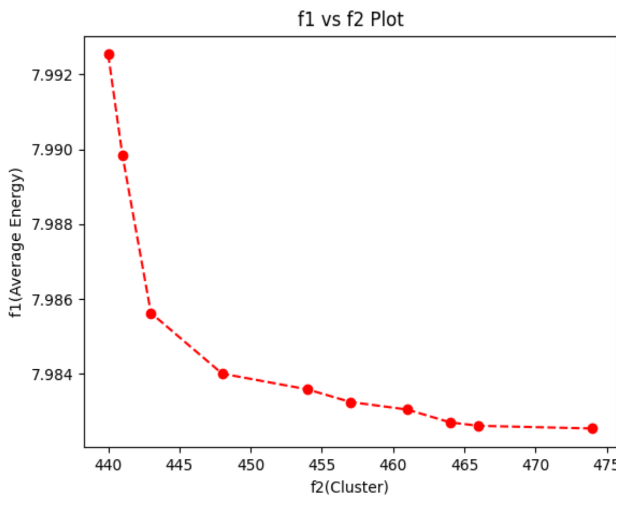


Figure 2. Average energy of 1000 independent nodes for 150 iteration

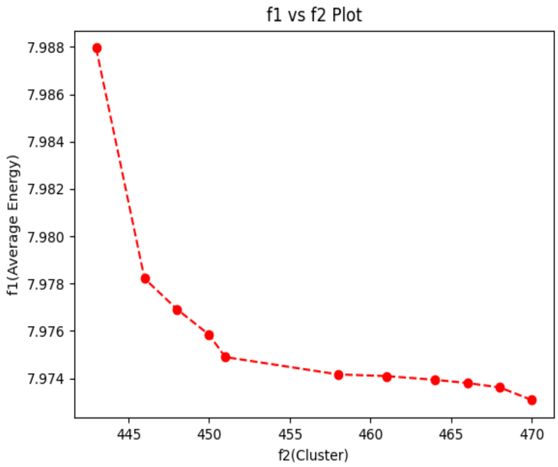
We can see from Figure 3 and Figure 4 for Non-Dominated Pareto Fronts that the plot has a Min-Min nature, which is exactly what we want to see. This brings us back to our original goal. To put it another way, when the energy of the system reduces, the unpredictability of the state in which our solution is found also decreases (or, to put it another way, the solution advances toward stability). This once again demonstrates that the multi-objective approach that we have provided is compatible with our goal of achieving a solution that would enable the creation of a stable network with a minimal amount of energy consumption and a minimal number of clusters.



Cluster (f2)	Average Energy (f1)
440	7.992529
441	7.989842
443	7.98563
448	7.984011
454	7.98359
457	7.983254
461	7.983052
464	7.98271
466	7.982619
474	7.982547

(a) (b)

Figure 3. (a) Interaction of Pareto Non-Dominated solutions for N=100 steps (b) Table of values used for simulation of (a)



Cluster (f2)	Average Energy (f1)
443	7.987957
446	7.978213
448	7.976925
450	7.97584
451	7.974898
458	7.974164
461	7.974094
464	7.973936
466	7.9738
468	7.973616
470	7.973081

(a) (b)

Figure 4. (a) Interaction of Pareto Non-Dominated solutions for N=150 steps (b) Table of values used for simulation of (a)

5. Conclusion

Single and multi-objective (MO) problem formulation in MOPSO for off-line, operationally-constrained, two-dimensional flight route optimisation. When docking a tiny molecule to a bigger receptor molecule, MOPSO is used to find a favourable position and orientation. With the restrictions of temperature, choking, and passivity in mind, MOPSO can be used to optimise the dimensions of the process performance, such as the tool life and the rate at which material is removed. Induced classifiers and the MOPSO algorithm both perform well in terms of the Area Under the Curve (AUC) measure, and MOPSO can handle both numerical and discrete attributes. As conventional economic power dispatch just saves fuel but is unable to manage the environment requirement, a fuzzified MOPSO has been designed to dispatch the electric power taking both into account. In order to absorb a wide spectrum of frequencies and angles, MOPSO was used to create a multilayer coating with a flat surface. The optimal design of an absorber is achieved by optimising the thickness, electric and magnetic properties of each layer, and the reflection coefficient over the specified range.

The authors propose a binary clustering approach, MOPSO, for use in WSNs. It then uses a cluster head selection algorithm to choose the most qualified person to lead that cluster. Very Large Scale Integrated (VLSI) networks use MOPSO for layout design. The approach offers a number of different layout options and generates a well-distributed pareto front. Using inter-node communication, a new method for WSN with an energy-efficient model with good coverage of WSN transmits data to a high-energy communication node. In order to balance the discrete and continuous goals of the power filter components and provide the reactive power compensation services required, a unique shunt power filter design using MOPSO is required. The hybrid power filter compensator employing a C-type filter and fixed capacitor designed with discrete MOPSO is solved using a discrete search optimisation method. Optimisation of the National Air Route Network (ARN) through the resolution of the Crossing Waypoint Location (CWL) problems, improving aviation safety and efficiency. They provide a detailed learning MOPSO to reduce airfare and scheduling conflicts. To find a battery-saving solution for WSN, the PSO-based clustering algorithm takes into account the optimal number of nodes and the power requirements of the sensor nodes. For MANET, MOPSO is used to maximise energy efficiency by minimising network traffic and maximising the number of clusters in an ad hoc network. In this case, we had to balance competing concerns about energy use and temperature. Particle positions were updated using the centroid in velocity-free MOPSO with centroid. The swarm particles should simply have position and no velocity. With technological constraints for optimal scattering parameters and operation bandwidth, the challenge of determining 9 unknown Field-Effect Transistor (FET) model elements is solved.

References

Clerc, M., & Kennedy, J. (2002). The particle swarm–explosion, stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation, 6(1), 58–73.

Coello Coello, C.A, & Lechuga, M.S., (2002). MOPSO: A proposal for multiple objective particle swarm optimization. In Congress on Evolutionary Computation (CEC'2002), vol 2, (pp. 1051– 1056). Piscataway, New Jersey: IEE Service Center.

Coello Coello, C.A., Pulido, G.T. & Lechuga, M.S. (2004). Handling Multiple Objectives With Particle Swarm Optimization. IEEE transactions on evolutionary computation, 8(3), 256-279.

Deepika, & Niranjan, S. (2015). Wireless Sensor Networks. International Journal of Engineering Research & Technology (IJERT), NCETEMS – 2015 Conference Proceedings, 3(10), IJERTCONV3IS10005.

Elhabyan, R.S.Y., & Yagoub, M.C.E. (2015). Two-tier particle swarm optimization protocol for clustering and routing in wireless sensor network. Journal of Network and Computer Applications, 52, 116-128.

Ibrahem M.S., Nazri, M.Z.A., & Othman, Z. (2018). A Multi-Objective Particle Swarm Optimization for Wireless Sensor Network Deployment. International Journal of Engineering & Technology, 7, 140-146.

Khan I., Belqasmi, F., Glitho, R., Crespi, N., Morrow, M., & Polakos, P. (2016). Wireless sensor network virtualization: A survey. IEEE Communications Surveys & Tutorials, IEEE Communications Society, Institute of Electrical and Electronics Engineers 18(1), 553-576.

Kuila, P, & Jana, P.K. (2014). Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach. Engineering Applications of Artificial Intelligence, 33, 127-140.

Li, W., Delicato, F.C., Pires, P.F., Lee, Y.C., Zomaya, A.Y., Miceli, C., & Pirmez, L. (2014), Efficient allocation of resources in multiple heterogeneous Wireless Sensor Network. Journal of Parallel and Distributed Computing, 74(1), 1775-1788.

Mao, J., Xiaoxi, J., & Xiuzhi, Z. (2019), Analysis of node deployment in wireless sensor networks in warehouse environment monitoring systems, EURASIP Journal on Wireless Communications and Networking, 2019, 288.

Sharma, G., Verma, M., & Mishra, N. (2014). Analysis of Transmission Technologies in Wireless Sensor Networks. International Journal of Engineering Research & Technology (IJERT), 3(1), IJERTV3IS10865.

Taherian, M., Karimi, H., Kashkooli, A.M., Esfahanimehr, A., Jafta, T., & Jafarabad, M. (2015). The design of an optimal and secure routing model in wireless sensor networks by using PSO algorithm. Procedia Computer Science, 73, 468-473.

Yadav, A., Kumar, S., & Singh, V. (2018). Network Life Time Analysis of WSNs Using Particle Swarm Optimization. Procedia Computer Science, 132, 805-815.

Yarpiz. Multi-Objective PSO in MATLAB. (2015). https://yarpiz.com/59/ypea121-mopso, Accessed 13 August 2022.