Energy-Efficient Topology to Enhance the Wireless Sensor Network Lifetime Using Connectivity Control

Meysam Yari
Department of Computer Engineering, Islamic Azad University, Shabestar Branch, Shabestar, East Azerbaijan, Iran

Parham Hadikhani
Department of Computer Engineering, Pasargad Higher Education Institute, Shiraz, Fars, Iran

Zohreh Asgharzadeh
Department of Computer Engineering, Payame Noor University, Germi Branch, Germi, Ardabil, Iran

Abstract: Wireless sensor networks have attracted much attention because of many applications in the fields of industry, military, medicine, agriculture, and education. In addition, the vast majority of research has been done to expand their applications and improve their efficiency. However, there are still many challenges for increasing the efficiency in different parts of this network. One of the most important parts is to improve the network lifetime in the wireless sensor network. Since the sensor nodes are generally powered by batteries, the most important issue to consider in these types of networks is to reduce the power consumption of the nodes in such a way as to increase the network lifetime to an acceptable level. The contribution of this paper is using topology control, the threshold for the remaining energy in nodes, and two metaheuristic algorithms, namely SA (Simulated Annealing) and VNS (Variable Neighbourhood Search), to increase the energy remaining in the sensors. Moreover, using a low-cost spanning tree, an appropriate connectivity control among nodes is created in the network in order to increase the network lifetime. The results of simulations show that the proposed method improves the sensor lifetime and reduces the energy consumed.

Keywords: Wireless Sensor Network, Connectivity Control, Lifetime, Meta-heuristic Algorithm, Energy Efficient

Introduction

Wireless sensor networks are one of the kinds of wireless ad hoc network that today have many applications in the fields of industry, military, medicine, education etc. These networks consist of tens to thousands of wireless nodes without monitoring (Vecchio & López-Valcarce, 2015),

http://doi.org/10.18080/jtde.v8n3.255

which are easily deployed in different environments and have defined specific functions for nodes. One of the advantages of a wireless sensor network is its high development speed due to its simplicity, the low cost of implementation, and automatic configuration. However, because of features such as the poor battery life in sensor nodes, it faces challenges (Yetgin, Cheung, El-Hajjar & Hanzo, 2017). Many studies have been carried out by researchers on the characteristics of this network to improve many of its capabilities. Because of the importance of this network in most areas, it has been decided to find a solution to increase the sensors' lifetime by reducing energy loss.

Many (heuristic and metaheuristic) methods have been proposed to increase the network's lifetime, such as clustering of sensor nodes, connectivity control, or a mobile sink. Energy consumption in wireless sensor nodes has a direct relationship with the network lifetime so that, if the process of energy consumption is reduced in the sensor nodes, it is claimed that the network lifetime increases. One of the methods to increase network lifetime is connectivity control in the wireless sensor network (Zhao, Guo, Wang & Wang, 2015).

In this paper, we are looking to create a spanning tree that can increase the network lifetime. A way to achieve this goal is to use a mobile sink and a spanning tree of the sensor nodes in the network. Sensor nodes send their received data to the base station for analysis and statistical reports by way of one or several sinks. Simulation results show that the network lifetime has improved in this study compared to Zhao *et al.* (2015).

In the first part of the article, the introduction of wireless sensor networks is discussed. In the second part, related works are introduced. In the third part, the proposed method is presented and, in the fourth section, simulation of the proposed method is described. In the final section, the conclusion and future works are discussed.

Related Work

A wireless sensor network is categorized as a wireless ad hoc network, a subject which has attracted a lot of attention in terms of academic and industrial studies in recent years (Zhao, Guo, Wang & Wang, 2015). Despite the many advances made in wireless sensor networks, there are still challenges, including energy consumption, routing, scalability, security, and fault tolerance. Routing and finding the most appropriate path among nodes of a wireless sensor network have many benefits, including connectivity control, increasing throughput, and eventually increasing the network lifetime.

In Liang & Liu (2006), the "maximum lifetime algorithm" has been proposed, where nodes are added with a greedy policy to the routing tree one by one. Furthermore, the proposed algorithm seeks to increase the network lifetime without knowing about the queries and their production rates. Only nodes can be added to the routing tree if they are able to increase the network lifetime.

In Heinzelman, Chandrakasan & Balakrishnan (2002), the LEACH algorithm has been described. The algorithm is hierarchical: at one level, it has a number of nodes as head clusters; and, at the next level, the nodes that do not belong to any of the head clusters are members of these clusters. One of the advantages of this method is increased load balancing among all network nodes.

In Tan & Körpeoğlu (2003), two PEDAP and PEDAP-PA protocols are proposed and the goal is to increase the network lifetime and the balanced energy consumption in each node. The results show that the proposed protocols have better performance than LEACH and PEGASIS protocols to increase the network lifetime.

In Gao, Zhang & Das (2010), integer (0-1) linear programming is used to find the optimal mapping among the members. After that, a two-dimensional genetic algorithm is used for optimal routing among members. Finally, a two-phase focused communication protocol is used to support the "maximum value of shortest path" algorithm.

In Zhang & Shen (2008), the problem of balancing energy is formulated as the problem of optimizing the allocation of data transmission by combining the idea of network division based on CORONA and a hybrid routing strategy. The proposed EBDG protocol has better output compared to multicast transmission schemes, direct transfer, and cluster rotation.

In Zhang, Shen, & Tan (2007), they are looking for the load balanced consumption to aggregate the sensor network data. The solution is found by a comparison between data transfer by hop-by-hop routing and direct transfer from the nodes. Thus, two RLN and GCN models are introduced to reduce energy consumption in the network.

In Hua & Yum (2008), the proposed algorithm combines the collected data with the desired routing and presents a smooth approximation function to optimize the problem. As a result, the network lifetime increases by routing and maximizing data aggregation. It also affects network traffic reduction.

In Hao, Wang, Yao, Geng & Chen (2018), a new model is proposed to predict the lifetime of a wireless sensor node on the basis of a Markov model (MPLM). Additionally, TCAMPLM is provided by adjusting the transmit power of sensor nodes to keep energy in the nodes.

In Hou & Zhang (2018), the mobile service computing algorithm is proposed to solve the problem of connectivity control in the wireless sensor network.

In Javadi, Mostafaei, Chowdhurry & Abawajy (2018), a connectivity control algorithm based on a learning automata called LBLATC is proposed. The learning automata chooses the appropriate transmission range of nodes to use the reinforcement signal generated by sensor nodes. The simulation results show that the expressed protocol has proper performance.

In Hadikhani, Eslaminejad, Yari & Mahani (2020), an algorithm is presented to improve the lifetime in wireless sensor networks. This algorithm initially detects a dynamic hole, then bypasses the hole, so that the nodes around the hole consume less energy.

In Zhao, Guo, Wang & Wang (2015), by using a greedy policy and dynamic programming, an innovative connectivity control algorithm (MLS) is proposed that use a mobile sink. The purpose of this method is to increase the minimum node energy in the wireless sensor network, which leads to maximizing the network lifetime. This method has better performance in increasing the network lifetime compared to previous methods.

Wang, Gao, Liu, Sangaiah & Kim (2019) presented a method for energy efficient routing. They used a combination of clustering and mobile sink. First, they divided the network into regions. Then, each region chose a cluster based on residual energy and distance of source node. After clustering, all clusters calculate the routing path and select a path with optimal energy consumption.

In Wang, Cao, Li, Kim & Lee (2017), they proposed an energy efficient routing algorithm with mobile sink. During the routing process they clustered the network by particle swarm optimization based on residual energy and position of node.

Bencan, Panpan, Peng & Dong (2020) proposed an algorithm to prevent energy holes and improve load balance in the network. They used an evolutionary game model for mobile sink based on residual energy and energy consumption of each cluster to make a utility function. A cluster that has greater utility value is selected as the new location of the sink.

In Pandiyaraju, Logambigai, Ganapathy & Kannan (2020), a routing protocol based on Intelligent Fuzzy Rules was provided for the agriculture sector. The purpose of their approach was to improve energy efficiency in the routing process to provide information for an irrigation system.

In Yarinezhad (2019), a new routing algorithm was presented which uses a virtual multi-ring shaped infrastructure to advertise the mobile sink position to the network.

Proposed Method

In this paper, we are looking to enhance the network lifetime by using connectivity control of sensor nodes and considering three limits based on the remaining energy of the sensor nodes. In fact, in this network, a certain threshold for local search is obtained and the changes in communications among nodes happen when their energy level reaches this threshold. In the following, we use the metaheuristic algorithms, which include variable neighbourhood search (VNS) (Hansen, Mladenović, Todosijević & Hanafi, 2017) and simulated annealing (SA) (Hao *et al.*, 2018), to achieve a spanning tree that maximizes the minimum sensor lifetime. We assume that a number of wireless sensor nodes are located in an environment with a mobile sink. a mobile sink is a mechanical node that move through the network to collect data from other nodes and transfer data to the base station; it is used to save sensor energy for multi-hop communication in transferring data to the base station. Sensor nodes in this research are divided into two categories: anchor sensor nodes and normal sensor nodes. Anchor sensor nodes are nodes close to the path of the mobile sink, and the normal nodes are nodes that are farther away from the path of the mobile sink. The normal nodes send received data from the surrounding environment or their next normal node as single or multiple messages to anchor nodes (Zhao, Guo, Wang & Wang, 2015). The anchor nodes send received data from the surrounding environment, along with the data received from their previous normal nodes, to the mobile sink when it is close to the anchor nodes. This connection creates connectivity between sensor nodes and the mobile sink. We want to create a spanning tree from the wireless sensor nodes with the root at the mobile sink that contains all network nodes. Our objective is to maximize the minimum network lifetime so that the death of the first sensor node occurs later. Figure 1 shows the schematic structure of the proposed algorithm.

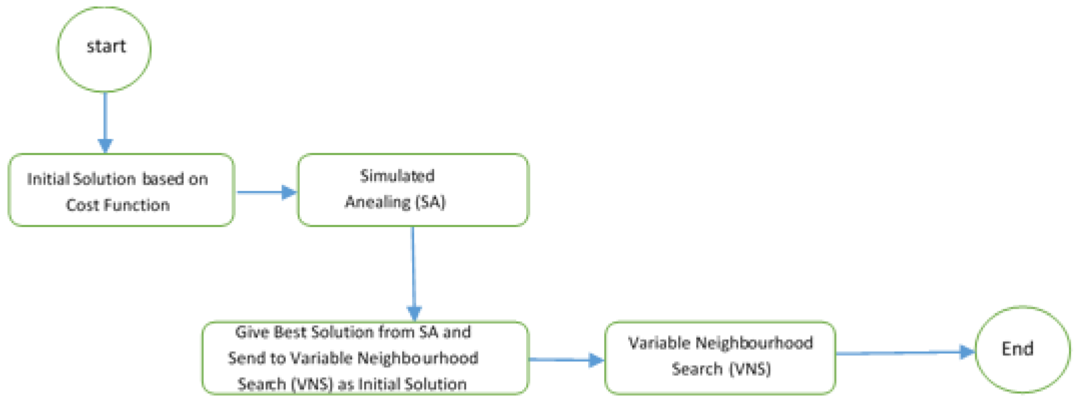

Figure 1: Schematic structure of the proposed algorithm

Energy Consumption Model

Sensor nodes, regardless of what roles they can play in a wireless sensor network, if they have enough energy, they can receive data from their surrounding environment and send the received data to the next node that is closer to the path of the mobile sink, according to the algorithm available in the network. They can also receive data from their neighbours depending on the defined methods in that network, and send the received data along with their data to the next sensor node. Generally, the sensor nodes consume more energy at the time of sending and transmitting the data to the next node. In Heinzelman, Chandrakasan &

Balakrishnan (2002), , according to the first-order radio model, a cover radius is defined based on the energy required in the free multipath space as follows:

$$E_m = \{(E_{elect} + E_{da} \times l) + (E_{amp} \times l \times d^{\alpha})\} \quad (1)$$

In equation (1), E_m is the energy consumed at the sensor node, and E_{elect} is the energy emission for transmitting and receiving data (J/bit). E_{da} is the energy consumed to aggregate sensor node data (nJ/bit/message) and E_{amp} is the emission of energy for the amplifier (pJ/bit/m^2). l is the length of the message and d is the distance of the current node from its neighbour node. The coefficient α is 2 for free-space transmission and 4 for multi-path fading.

Energy consumption in this research is based on equation (1). The most important part of our research is the remaining energy of the nodes. Each sensor node can be a bottleneck to increasing the network lifetime. As a consequence, we choose the connectivity control based on the minimum spanning tree to increase the network lifetime.

Proposed Algorithm

The sensor nodes are randomly distributed in a 1000 metres by 1000 metres environment, for example, and the sink node moves in a predefined path. Each node locally receives its distance from neighbours by using a Hello message. Nodes that are close to the path of the mobile sink are known as anchor nodes and nodes that are farther away from the path of the mobile sink are called non-anchor nodes (normal nodes). Anchor nodes are responsible for sending their data and the data of non-anchor nodes to the mobile sink. The relative load of sensor nodes is based on:

$$l_{(v)} = \frac{(Crx + Ctx(r(v)) \times q(v) - Crx)}{e(v)} \quad (2)$$

This is the cost of energy for the remaining energy in the sensor node. In this equation, v indicates a sensor in the network. C_{rx} indicates energy consumption for receiving data at the node. $C_{tx}(r_v)$ is the energy consumption for transferring data from the node to another by considering the distance between them. $q_{(v)}$ indicates the number of child nodes for which the current node is the parent and $e_{(v)}$, expresses the remaining energy of the current node.

Equation 3 shows that the relative load of a node can be alleviated by reducing the number of its descendent nodes or shortening the transmission radius (Zhao, Guo, Wang & Wang, 2015). Then, for minimizing the maximum relative load of the sensor nodes:

$$minimize: \max\{l(v) |\, v \in V\} \quad (3)$$

Thus, nodes that reduce the load of our spanning tree are added to the tree (Figure 2).

http://doi.org/10.18080/jtde.v8n3.255

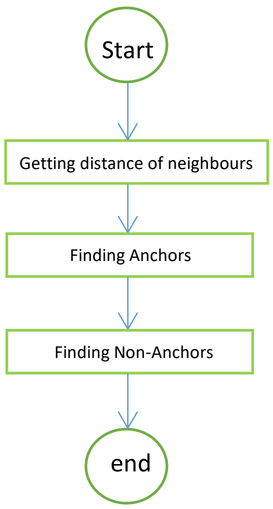

Figure 2. Steps to create spanning tree

Initially, it starts from the root node (the sink).

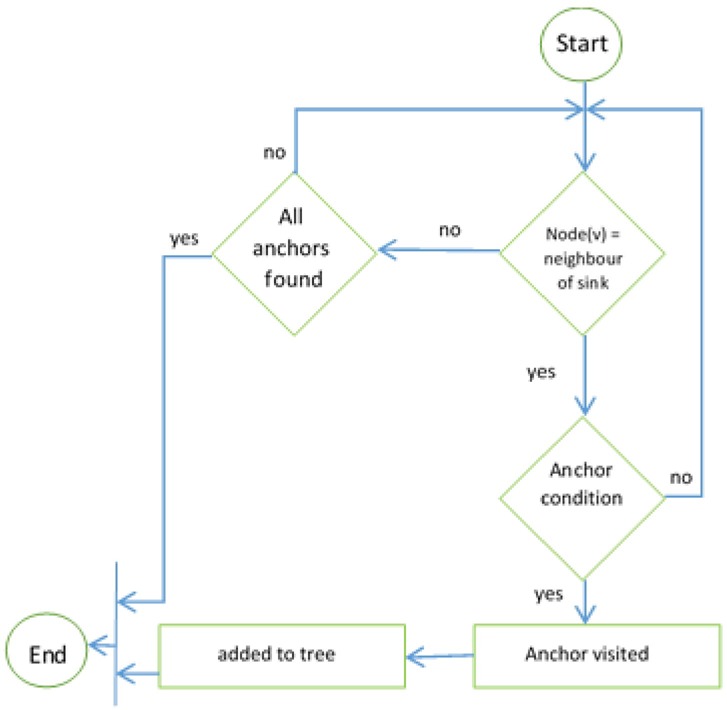

Figure 3. Finding Anchors

As shown in Figure 3, after the anchor nodes are identified and connected to the root node, in the next step, the relative load of neighbour nodes is calculated according to equation (2), if they were not previously members of the spanning tree and their remaining energy has not reached the threshold value (Figure 4). A node that has the lowest relative load (initial selection phase) (Zhao, Guo, Wang & Wang, 2015) is selected as the candidate node. The amount of relative load is obtained on the basis that, if the candidate node is to be attached to the tree, its cost will be its relative load. In the next step (final selection phase), for the nodes that have been selected in the previous step, the loads on candidate nodes in the spanning tree are sorted in ascending order and the largest amount of load is compared with the largest amount of load of the other candidate nodes. Then, the lowest amount is added as the selected node to the spanning tree (Figure 5).

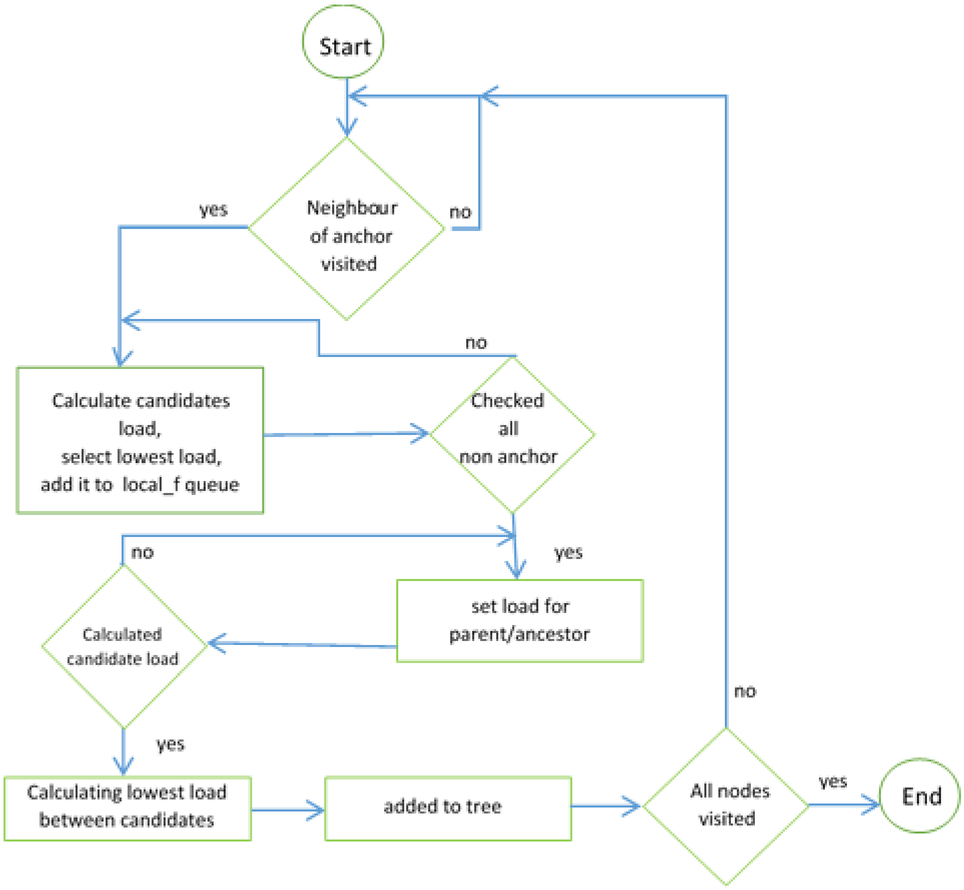

Figure 4. Finding Non-Anchors

Three limits — threshold, warning, and death — are defined for network nodes. In each round of energy simulation, when each node needs to send its data to the next node, the remaining energy is specified in a table. If the remaining energy in a node reaches the threshold (twice the minimum energy stored for each node), we enter the local search step, where that node is labelled and the children of that node should be separated from their parent and the parent selection process should be re-established for them. This reduces the speed of energy loss of the labelled node, and this helps to increase the network lifetime throughout the network. The reason for choosing the threshold as double the minimum value is that, if the remaining energy of the node reaches twice its minimum stored energy, it will have the opportunity to increase its lifetime by entering a local replacement phase. This means that, with two sink moves, the node will not be able to send/receive data in the network and will be removed from the data transfer in the network. But if we can make this node consume less energy in each sink move, it makes its lifetime increase. The warning threshold means that, if the remaining energy in the node is three times the minimum energy stored for the node, we will enter the phase of finding new neighbours. If the remaining energy in the node is less than the minimum amount of energy stored for the node, the condition for the death of the node is fulfilled and that node will be removed from the data transfer in the network.

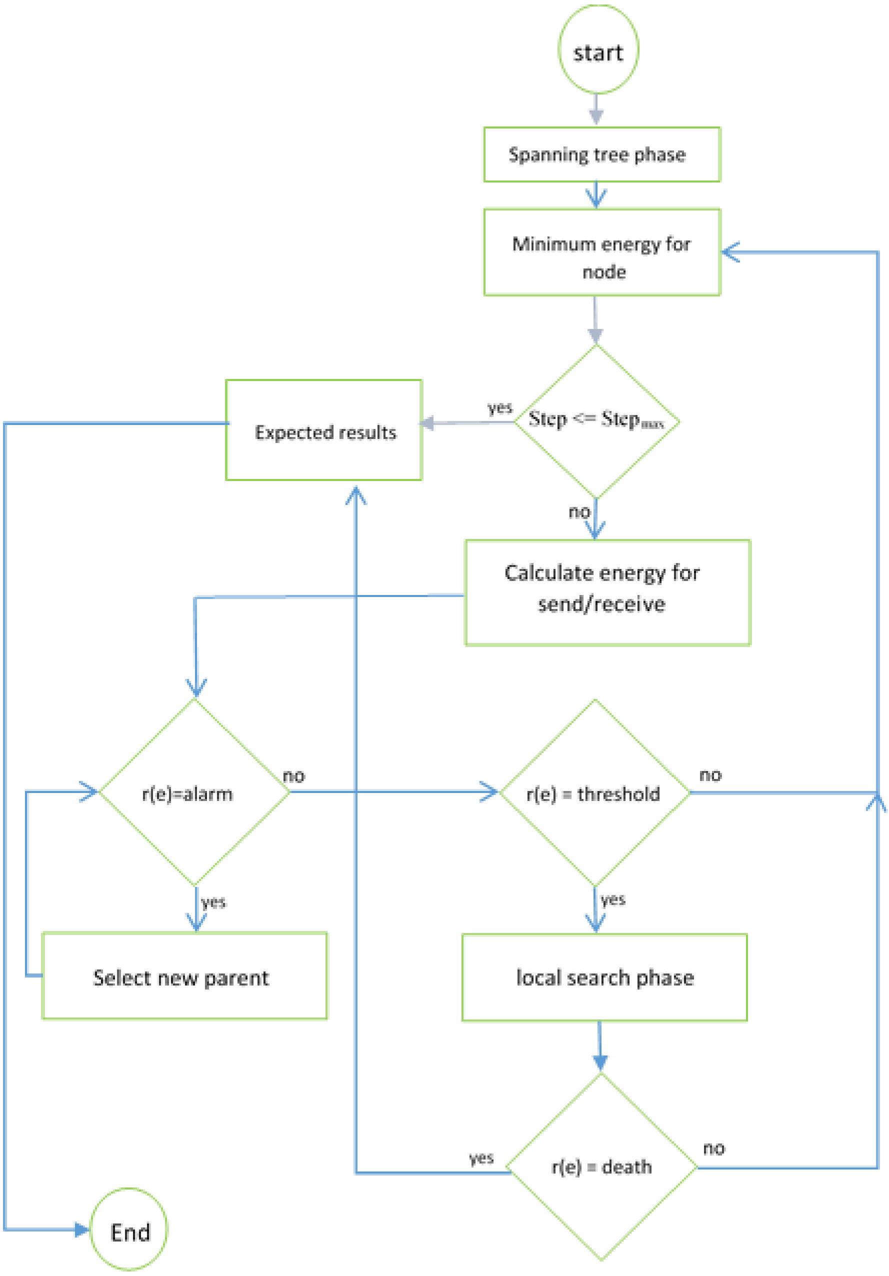

Figure 5. Proposed Method

The reduction of the number of child nodes of a node means shifting the load to subtrees. In the local search phase, the child nodes are separated from that node and a new parent is chosen for them. Thus, the energy consumption of the desired node is reduced. Meanwhile, if a node x that has reached the threshold has only one child and there is no other child in the neighbourhood of that node, the child node is separated from node x and is detached from the network during the local search phase. To prevent being isolated from the network, the warning threshold is used. If a node is placed in such a case, another node is introduced as the parent node of that node.

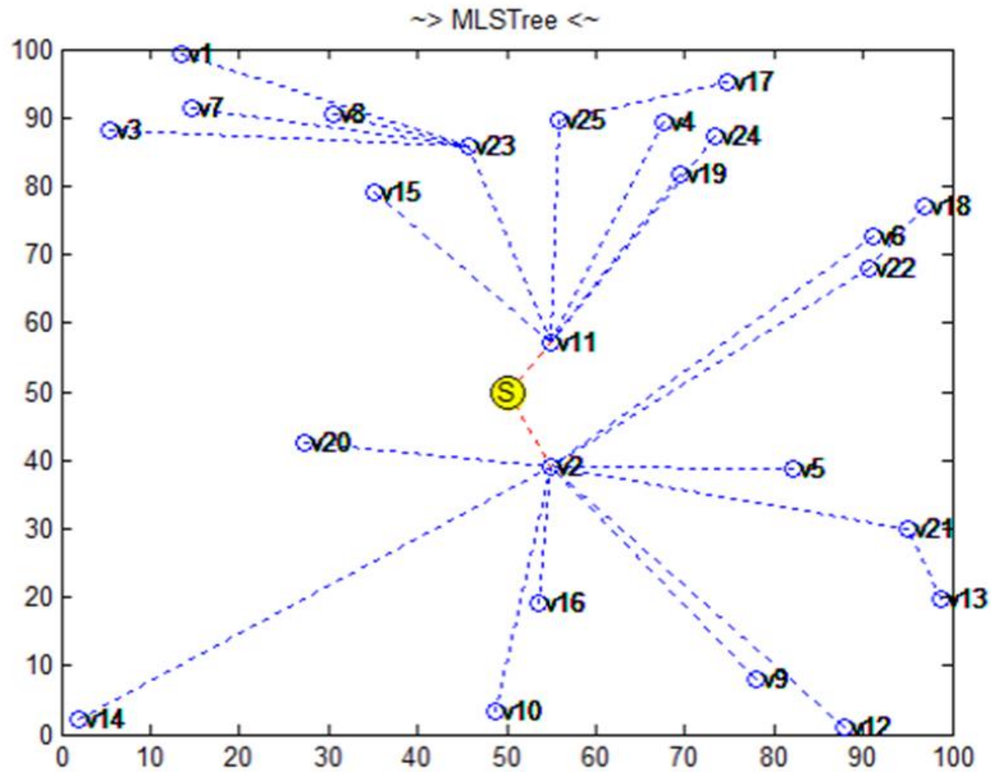

Figure 6. Completed network with the proposed method. S-node is a hypothetical central node in the simulation, which is the starting point for building a spanning tree from this hypothetical node. The nodes connected to the S node by the red dotted lines are the anchor nodes in the network (meaning the nodes closest to the sink node when the sink is moving). Nodes connected to anchor nodes or other nodes with a blue dotted line are normal nodes, whose job is to listen to their surroundings and send the information heard to the next nodes, or as a sink to receive data from previous nodes. Finally, leaf nodes are normal nodes that only listen to their surroundings and send the received data to their next nodes.

As stated earlier, the warning threshold and the threshold are used if, after making a spanning route, the traffic of sending packets on a node increases; then, the load on that node can be divided over the rest of the nodes in the tree. By doing this work, it is possible to claim that the network lifetime has increased over the situation before. To obtain the remaining energy, simulated annealing (Kirkpatrick, Gelatt & Vecchi, 1983) and variable neighbourhood search (Yetgin, Cheung, El-Hajjar & Hanzo, 2017) were used. We are looking for a tree that has least construction cost. The purpose of the cost is to select the node with the minimum load from the tree nodes with the highest load. Figure 6 indicates an example of a completed network.

First, an initial solution is needed to consider a neighbourhood relationship among the nodes. We take the cost of building a tree as the initial solution and enter the simulated annealing phase. For all the sets of responses, changes are made to the generated answers by using the neighbourhood function. Afterward, the cost of the obtained response is compared with the cost of the best answer so far. If the answer is better (minimum) than the best answer so far, the answer is replaced by the better answer. Otherwise, by creating a random number and a

Boltzmann probability function, we are looking to accept the worse answer as well. If this is not true, the best answer will not change (see Algorithm 1).

```
Initialization
Number of Population;
Number of Move toward Neighbours;
Initial temperature and find temperature and T = T0 and α is reduction rate;
Initialize best_sol.cost = Inf;
Pop = Generate Population (with random solution);
If Pop.cost <= best_sol.cost then
best_sol = Pop;
Repeat until stopping condition is met (T <= TF)
create and evaluate new_solution for Number of population and moving;
new_pop = create Neighbour for every member of population;
for I <- 1 to number of pop do
                    if new_pop.cost <= pop.cost
                              Pop = new_pop;
                    elseif rand < TempFunc(new_pop.cost,Pop.cost,T)
                              best_sol = Pop;
          store best_sol;
T = α × T;
```

Figure 7. Algorithm 1. The pseudo code of the proposed method with SA

```
Initialization
   select the set of neighbourhood structures
   NK
   Find an initial solution x
Repeat until stopping condition is met
  Set K = 1;
  Repeat until K = Kmax
        1 – Do_Shaking : Generate a random point X` in NK(X);
        2 – Local_search : X`` is optimum obtained;
        3 – Move / not Move:
          -if X`` is better than X, then x = X`` and k = 1;
          -else K = K + 1;
```

Figure 8. Algorithm 2. Pseudo code of VNS (Yetgin, Cheung, El-Hajjar & Hanzo, 2017)

The next algorithm is Variable Neighbourhood Search (Yetgin, Cheung, El-Hajjar & Hanzo, 2017) (Algorithm 2). The algorithm uses two parameters: "vibration" and "local search". Vibration generates diversity and local search looks for the most appropriate answer. The vibration section creates a fundamental change in the initial response. Since it takes a lot of time to search the entire problem space, using vibration produces diversity in the answer to almost make sure that every state of the answer is checked (see Figure 9).

In the local search phase, if an answer is found better than the current solution, it will replace the current solution. Otherwise, it goes to other areas to compare their answers to the current solution. This work is conducted to the end of the specified time in local search. Finally, the output of this stage is compared with the best answer that already existed. If the solution is better than the current solution, then the new solution replaces the current solution. Otherwise, the vibration is performed on the last answer obtained from the previous step to check again the answers of that range.

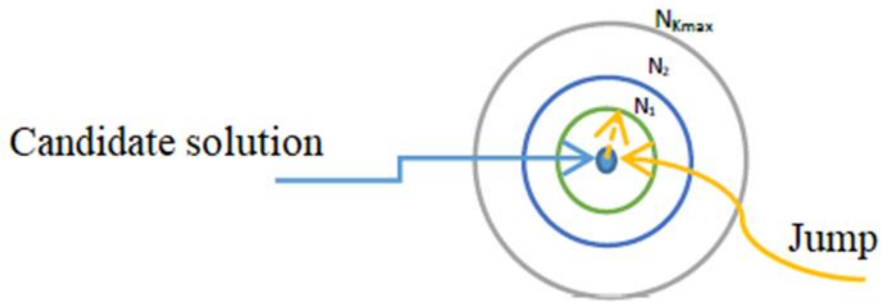

Figure 9. Variable Neighbourhood Search (Yetgin, Cheung, El-Hajjar & Hanzo, 2017)

Simulation and Results

In this section, in order to simulate the proposed method and compare it with Zhao, Guo, Wang & Wang (2015), the ns2 simulation with mannasim is used. Ns2 is an event-based, open-source software package. In this simulation, it is assumed that the simulation environment is 1000 metres by 1000 metres and consists of 100, 130, 160 and 190 wireless sensor nodes along with a mobile sink. For routing, the AODV protocol has been used and the radio antenna is omnidirectional. All parameters are presented in Table 1.

Table 1: Parameters

Parameter	Value
The primary energy of each node	0.2 J
Area of simulation	1000 mx1000 m
Number of sinks	1
MAC type	Mac/802.11
Clustering algorithm	LEACH

To achieve the simulation results, the number of rounds has been executed 25 times, with each execution having 1200 runs. Next, the mean of these 25 executions was obtained and the mean values are recorded in the results.

Figure 10 shows the mean end-to-end delay among nodes, in milliseconds. After 25 executions for 100 nodes in the proposed method, the average delay is 0.5543 ms, while in the method of Zhao, Guo, Wang & Wang (2015) it is 0.67424 ms. When the number of sensor nodes is 130, the average delay in the proposed method is 0.58725 ms; however. in the method of Zhao, Guo, Wang & Wang (2015), it is 0.719874 ms. In addition, when the number of sensor nodes is 160, the average delay is 0.676539 ms, compared to 0.794338 ms in the method of Zhao, Guo, Wang & Wang (2015). Finally, when the number of sensor nodes is 190, the average delay in the proposed method is 0.8725 ms, as opposed to 0.898538 ms in Zhao, Guo, Wang & Wang (2015).

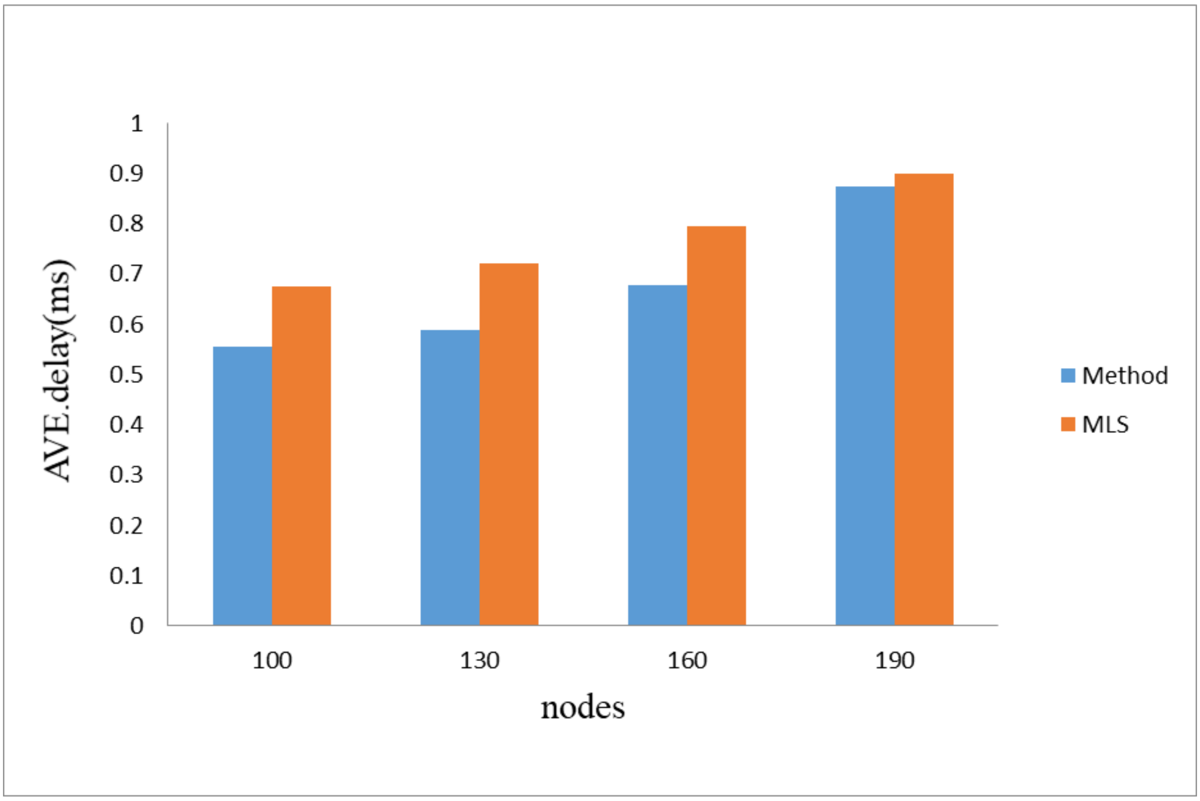

Figure 10. The comparison of the average end-to-end delay in the sensor network between the proposed method and Zhao, Guo, Wang & Wang (2015)

In Figure 11, the energy consumed in the sensor nodes is exhibited. For 100 nodes, the average energy consumption after 25 runs in the proposed method is 17.5 J, whereas in Zhao, Guo, Wang & Wang (2015) it is 18.8 J. For 130 sensor nodes, the average energy consumption in the proposed method is 21.3 J, compared to 24.2 J in Zhao, Guo, Wang & Wang (2015). For 160 and 190 sensor nodes, the proposed method consumes less energy than the method in Zhao, Guo, Wang & Wang (2015): 28.39 J and 32.14 J, respectively, compared to 31 J and 35.83 J.

Figure 12 compares the sensors' lifetimes based on the number of dead nodes between our method and the method proposed in Zhao, Guo, Wang & Wang (2015). A sensor node dies because its battery has limited energy that is consumed over the time of the sensor node's activity (depending on the type of node activity). Thus, its energy decreases as its battery is consumed and not recharged. This causes the sensor node to lose its energy after a while and it can no longer continue to operate, including sensing the surroundings and sending or receiving data. When the number of nodes is 100, the numbers of dead nodes in our method and in Zhao, Guo, Wang & Wang (2015) are 13 and 16, respectively. Further, when the number of nodes is 130, the number of dead nodes in our method is 19, slightly fewer than the 23 dead nodes in Zhao, Guo, Wang & Wang (2015). Similarly, for 160 and 190 sensor nodes, the numbers of dead nodes in our method are 23 and 31 in turn, as opposed to 32 and 40 dead nodes, respectively, in Zhao, Guo, Wang & Wang (2015).

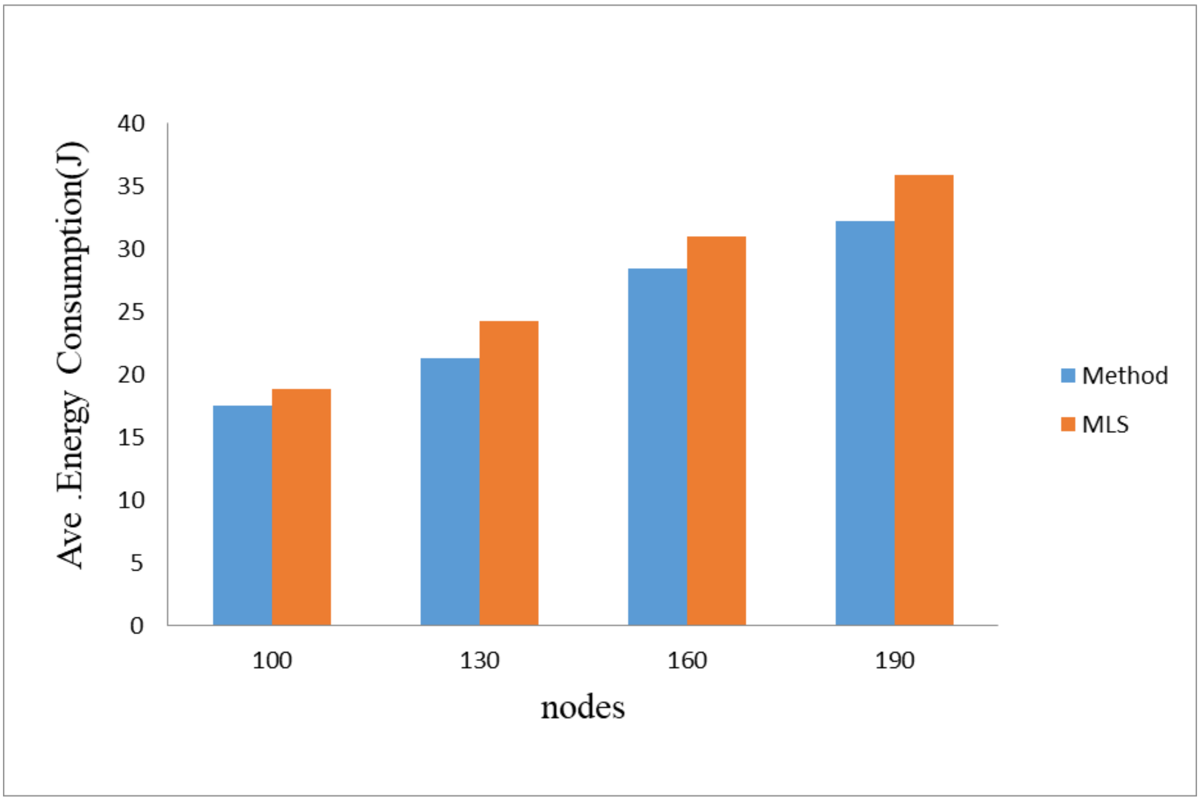

Figure 11. The comparison of the average energy consumed in the sensor network between the proposed method and Zhao, Guo, Wang & Wang (2015)

Figure 13 indicates the comparison of average packet delivery rates among network nodes for the proposed method and Zhao, Guo, Wang & Wang (2015). When the number of nodes is 100, the average packet delivery rate in the proposed method is 85%, compared to 79% in Zhao, Guo, Wang & Wang (2015). When the number of nodes is 130, the average packet delivery rate in the proposed method is 81.84%, as opposed to 78.93% in Zhao, Guo, Wang & Wang (2015). For 160 and 190 sensor nodes, the average packet delivery rates in the proposed method are 84% and 85% in turn, while in Zhao, Guo, Wang & Wang (2015) they are 77% and 79%, respectively.

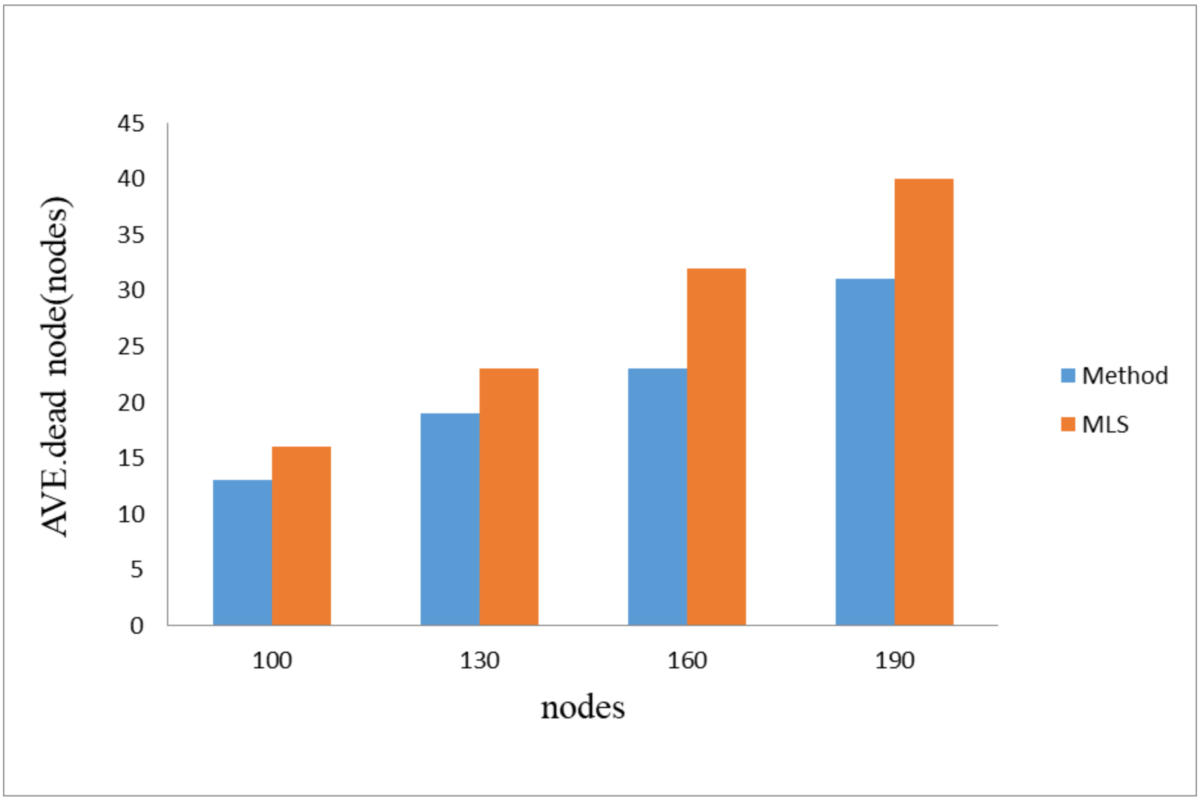

Figure 12. The comparison of the average sensor lifetime in the sensor network between the proposed method and Zhao, Guo, Wang & Wang (2015)

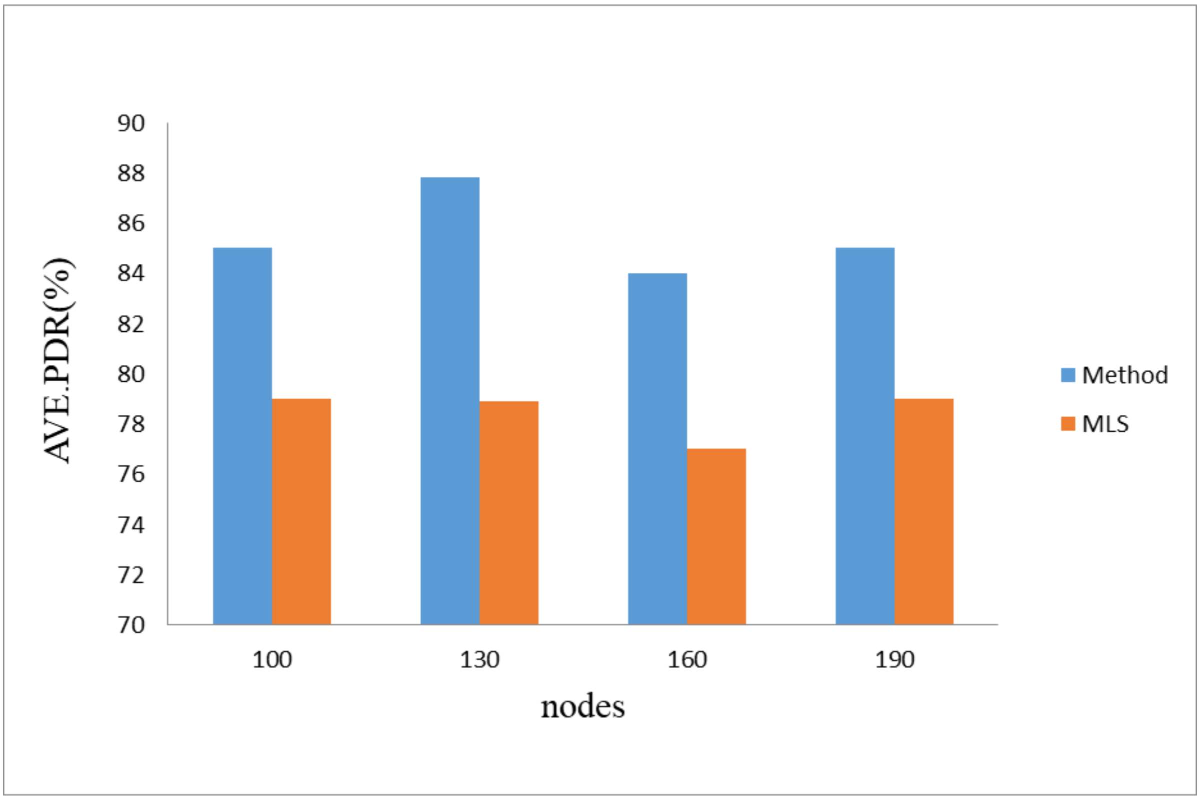

Figure 13. Comparison of average packet delivery rates among network nodes between the proposed method and Zhao, Guo, Wang & Wang (2015)

Conclusion and Future Work

Considering the many applications of wireless sensor networks and the great passion for using them, they still face challenges, including energy consumption. In this study, we developed load balancing by using proper connectivity control between the sensor nodes and the definition of three limits, including thresholds, warnings, and deaths for sensor nodes. Moreover, with the help of two metaheuristic algorithms, SA and VNS, the lowest cost spanning trees for the sensor nodes were found. The results of the simulation show that the network lifetime in the proposed method has been improved and enhanced compared to Zhao, Guo, Wang & Wang (2015).

As future work, in order to evaluate the sensors' lifetimes before forming the connectivity tree, the MPLM method (Hao *et al.*, 2018) can be used. In the proposed method, we can actively use the durability or persistence of sensor nodes in the network. As a result, the connectivity is established among sensor nodes through including the best mode and network status; this improves the network lifetime. The second suggestion is using an information packet that is exchanged among all sensor nodes and includes their current status (active/deactivated). This packet has the status of all sensor nodes and this information can be used for better routing of the mobile sink.

References

Bencan, G., Panpan, D., Peng, C., & Dong, R. (2020). Evolutionary game-based trajectory design algorithm for mobile sink in wireless sensor networks. *International Journal of Distributed Sensor Networks, 16*(3), 1550147720911000.

Gao, S., Zhang, H., & Das, S. K. (2010). Efficient data collection in wireless sensor networks with path-constrained mobile sinks. *IEEE Transactions on Mobile Computing, 10*(4), 592-608.

Hadikhani, P., Eslaminejad, M., Yari, M., & Mahani, E. A. (2020). An energy-aware and load balanced distributed geographic routing algorithm for wireless sensor networks with dynamic hole. *Wireless Networks*, *26*, 507-519. https://doi.org/10.1007/s11276-019-02157-6

Hansen, P., Mladenović, N., Todosijević, R., & Hanafi, S. (2017). Variable neighborhood search: basics and variants. *EURO Journal on Computational Optimization, 5*(3), 423-454.

Hao, X., Wang, L., Yao, N., Geng, D., & Chen, B. (2018). Topology control game algorithm based on Markov lifetime prediction model for wireless sensor network. *Ad Hoc Networks, 78*, 13-23.

Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications, 1*(4), 660-670.

Hou, J., & Zhang, Y. (2018). Mobile-Service Based Approach for Topology Control of Wireless Sensor Networks. *Wireless Personal Communications, 102*(2), 1839-1851.

Hua, C., & Yum, T.-S. P. (2008). Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks. *IEEE/ACM Transactions on Networking (TON), 16*(4), 892-903.

Javadi, M., Mostafaei, H., Chowdhurry, M. U., & Abawajy, J. H. (2018). Learning automaton based topology control protocol for extending wireless sensor networks lifetime. *Journal of Network and Computer Applications, 122*, 128-136.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*(4598), 671-680.

Liang, W., & Liu, Y. (2006). Online data gathering for maximizing network lifetime in sensor networks. *IEEE Transactions on Mobile Computing, 6*(1), 2-11.

Pandiyaraju, V., Logambigai, R., Ganapathy, S., & Kannan, A. (2020). An Energy Efficient Routing Algorithm for WSNs Using Intelligent Fuzzy Rules in Precision Agriculture. *Wireless Personal Communications*, 1-17.

Tan, H. Ö., & Körpeoğlu, I. (2003). Power efficient data gathering and aggregation in wireless sensor networks. *ACM Sigmod Record, 32*(4), 66-71.

Vecchio, M., & López-Valcarce, R. (2015). Improving area coverage of wireless sensor networks via controllable mobile nodes: A greedy approach. *Journal of Network and Computer Applications, 48*, 1-13.

Wang, J., Cao, Y., Li, B., Kim, H.-j., & Lee, S. (2017). Particle swarm optimization based clustering algorithm with mobile sink for WSNs. *Future Generation Computer Systems, 76*, 452-457.

Wang, J., Gao, Y., Liu, W., Sangaiah, A. K., & Kim, H.-J. (2019). Energy efficient routing algorithm with mobile sink support for wireless sensor networks. *Sensors, 19*(7), 1494.

Yarinezhad, R. (2019). Reducing delay and prolonging the lifetime of wireless sensor network using efficient routing protocol based on mobile sink and virtual infrastructure. *Ad Hoc Networks, 84*, 42-55.

Yetgin, H., Cheung, K. T. K., El-Hajjar, M., & Hanzo, L. H. (2017). A survey of network lifetime maximization techniques in wireless sensor networks, *IEEE Communications Surveys & Tutorials*, *19*(2), 828-854.

Zhang, H., & Shen, H. (2008). Balancing energy consumption to maximize network lifetime in data-gathering sensor networks. *IEEE Transactions on Parallel and Distributed Systems, 20*(10), 1526-1539.

Zhang, H., Shen, H., & Tan, Y. (2007). Optimal energy balanced data gathering in wireless sensor networks. Paper presented at the *2007 IEEE International Parallel and Distributed Processing Symposium*.

Zhao, H., Guo, S., Wang, X., & Wang, F. (2015). Energy-efficient topology control algorithm for maximizing network lifetime in wireless sensor networks with mobile sink. *Applied Soft Computing, 34*, 539-550.