Chapter

Intelligent Routing Mechanisms in IoT

Gowrishankar Subhrahmanyam and Ishwari Ginimav

Abstract

Wireless sensor networks (WSN) works on battery in order to communicate with each other. Energy consumption is the challenging issue in WSNs. In the recent few years, green communications have become a major concern in communication research and industries. Its major goal is to minimize the energy consumed by the nodes of the WSN. In order to save energy we need to switch off the extra components that are not in use during low traffic period. The technique where the unused extra components are switched off is called as sleep-scheduling, and the routing algorithm used to implement this is called as sleep-scheduling routing algorithm. In WSN the network is divided into multiple clusters. In each cluster one of the sensor nodes is elected as cluster head (CH) and other sensor nodes act as cluster members (CM). The cluster head collects the data form all the other nodes, removes the redundant data and transmits it to the destination. As the amount of workload is much more on the cluster head, the energy consumed by the cluster head is also more. Therefore to equalize the energy consumption among all the nodes, cluster head rotation is done. This chapter deals with different energy consumption techniques.

Keywords: wireless sensor networks, green IOT, sleep scheduling algorithms, LEACH algorithm, FCM algorithm, GA algorithm, PSO algorithm, GSO algorithm, MINEN algorithm

1. Introduction

Wireless sensor networks (WSNs) [1] is one of the widely used technologies in twenty-first century. The main objective of WSN is that it provides wireless communication by low-cost sensor networks which consumes very limited power and provides multiple functionalities. A typical sensor node [1] is made of four basic components namely: a sensing unit, a processing unit, a communication unit, and a power unit as shown in **Figure 1**. Peer-to-peer communication exists between two nodes. Multi-hop communication exists when the nodes are not in the radio range and the communication between the two nodes is carried out via intermediate nodes. Therefore the WSN provides a very good feature of adding and removing nodes. In WSN the network is divided into number of clusters. In each cluster one of the sensor nodes is elected as cluster head (CH) and other sensor nodes act as cluster members (CM). The cluster head collects the data form all the other nodes, removes the redundant data and transmits it to the destination. As the amount of

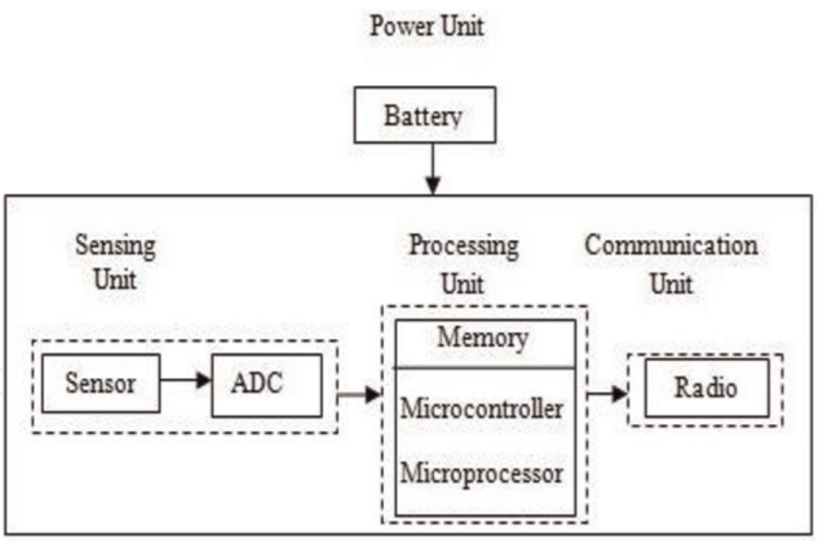

Figure 1.
Sensor node structure.

workload is much more on the cluster head, the energy consumed by the cluster head is also more. Therefore to equalize the energy consumption among all the nodes, cluster head rotation is done. Energy consumption is the challenging issue in WSN. This chapter deals with different energy consumption techniques.

1.1 Characteristics of WSNs

Following are some of the important characteristics of WSN:

1.1.1 Dynamic network topology

Whenever there is an addition or deletion of nodes the network topology keeps updating. This feature adds to the flexibility to the network.

1.1.2 Application specific

The design of the network keeps changing based on requirement of the application.

1.1.3 Energy constrained

As the nodes are movable, there is always a constraint on energy consumption.

1.1.4 Self-configurable

Nodes are set haphazardly in the network. Once the nodes are deployed, they configure themselves based on the requirement.

1.2 WSN routing protocol

In order to reduce the energy consumption in WSN, the different routing protocols are implemented which define how the packets have to be transmitted from source to destination **Figure 2** shows the classification of routing protocols in WSNs [1].

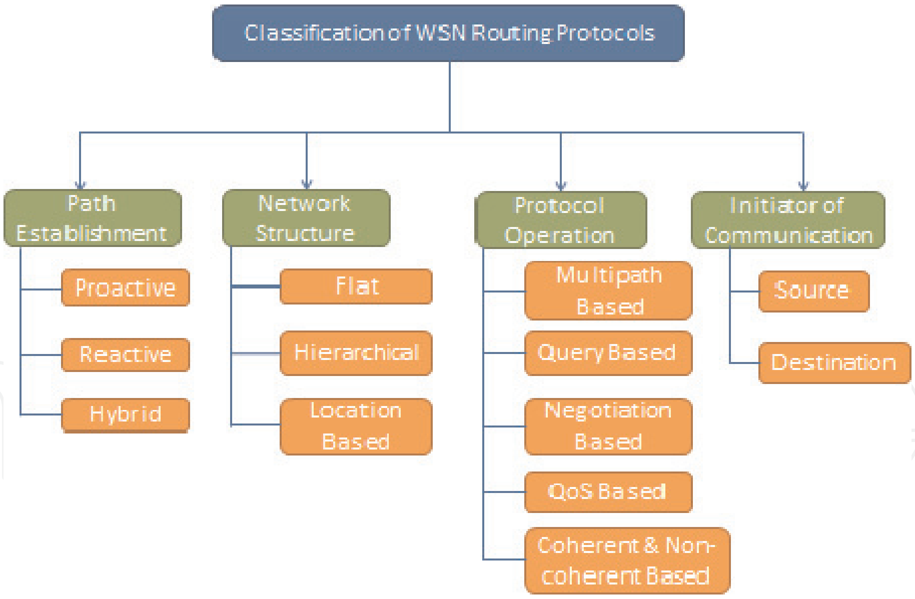

Figure 2.
Classification of WSN routing protocols.

1.3 Green IoT

In the recent few years, green communications have become a major concern in communication research and industries. Its major goal is to reduce the energy consumption of the WSNs. This has come into picture in order to reduce the pollution in the environment and cost. Recent studies have identified that ICT contributes to 10% of the global CO_2 emissions and its contribution is increasing day by day. Research also says that 30–37% of greenhouse gases (GHG) is been produced by ICT sector. Nowadays technology has improved so much that the bandwidths are made larger so as to handle large amount of traffic loads. In such cases in order to save energy we need to switch off the extra components that are not in use during low traffic period. To achieve this, a routing algorithm should be implemented such that the unused network components will be switched off. The technique where the unused extra components are switched off is called as sleep-scheduling, and the routing algorithm used to implement this is called as sleep-scheduling routing algorithm. This chapter mainly deals with sleep-scheduling base green routing algorithm [2].

2. Different routing algorithms

2.1 LEACH routing algorithm

LEACH protocol [3] is an implementation of hierarchical routing protocol. It is a self-organized and self-adaptive routing protocol. The LEACH protocol works on around technique in which each round is considered as one unit. Every round is made out of cluster set-up phase and steady-state for lowering excess energy cost. The steady-state phase is sometimes more time consuming than the set-up phase. One advantage is that the nodes are allowed to elect themselves as the cluster head if required. When a node becomes a cluster head, it can generate a TDMA time schedule for the sensing nodes during which the non-cluster head nodes are turned off when it is not in its time interval. **Figure 3** shows the timeline operation of LEACH.

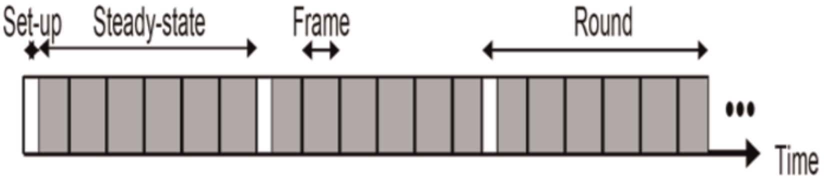

Figure 3.
Timeline showing operation of LEACH.

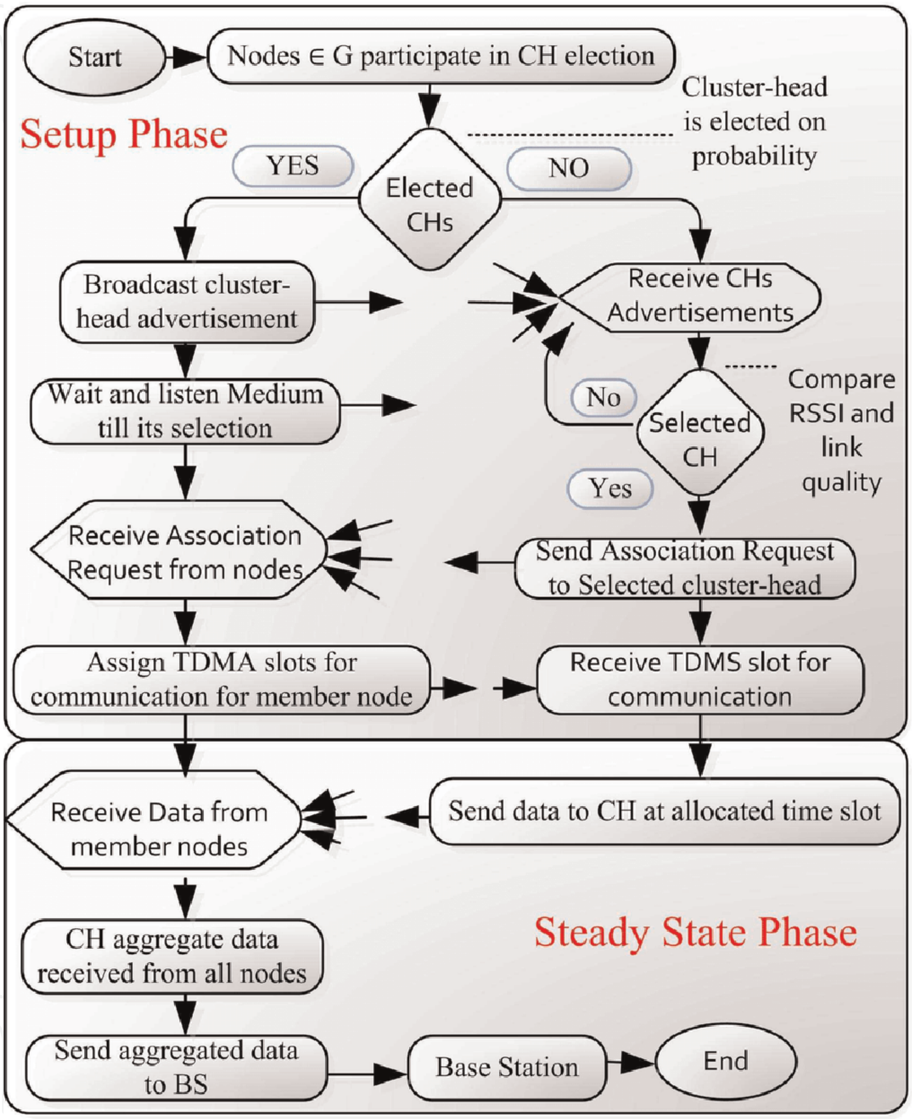
Start
Nodes ∈ G participate in CH election
Cluster-head is elected on probability
Setup Phase
YES
Elected CHs
NO
Broadcast cluster-head advertisement
Receive CHs Advertisements
Wait and listen Medium till its selection
No
Selected CH
Compare RSSI and link quality
Yes
Receive Association Request from nodes
Send Association Request to Selected cluster-head
Assign TDMA slots for communication for member node
Receive TDMS slot for communication
Receive Data from member nodes
Send data to CH at allocated time slot
Steady State Phase
CH aggregate data received from all nodes
Send aggregated data to BS
Base Station
End

Figure 4.
Flow of LEACH routing protocol.

Figure 4 shows the flow of the LEACH routing protocol [4]:

i. **The set-up phase**: At first the LEACH protocol will randomly generate a number between 0 and 1 for each node and selects a cluster head based on that random number generated. A threshold value is identified which is given by the threshold function T(n). If the randomly generated number of a node is less than the threshold value, then that node will be selected as cluster head node.

$$T(n) = \begin{cases} \dfrac{p}{\left(1 - p * \left(rmod\dfrac{1}{p}\right)\right)} & , n \in G \\ 0 & , n \in G \end{cases} \tag{1}$$

where P is the probability of the cluster-head and G is the set of nodes that have never been chosen as cluster-head nodes before 1/p round.

Once the cluster head is selected, it will send a CDMA message to all other normal nodes to inform them to join the cluster head node. Later the cluster head will use TDMA so that every node which is connected to the cluster head node will be allocated time duration for data transmission.

i. **The steady-state**: In this phase the normal node will sense the data and transmit the data to the cluster head node. The data is processed at the cluster head node and then transmitted to the base station.

2.2 Fuzzy C-means

The distribution of nodes in a cluster is one of the major concerns in WSNs. The distribution of nodes in a cluster is said to be good if the distance between a particular node and its cluster head is reasonably small. The communication cost between a node and its cluster head and the unbalanced load distribution can be optimized if the distribution of the nodes in the cluster is maintained and managed properly. Let us consider a network with N nodes. Our main job is to group the nodes into different c clusters in such a way that the distance between every node and its cluster head is reasonably small. It is the responsibility of the base station to compute the cluster head based on the geographical location of the nodes in the WSN. In order to achieve this fuzzy technique [5] is implemented to find the cluster center.

The equation for the distribution of nodes into a cluster is given as:

$$f_{obj} = \sum_{j=1}^{c} \left[\sum_{i=1}^{N} u_{ij}^{m} d_{ij}^{2} \right] \tag{2}$$

where u_{ij} is the degree of belongingness of node i to cluster j, d_{ij} is the Euclidean distance of node I from the centroid of cluster j, and m is the fuzzy control parameter.

There is an update in the value of membership degrees in the course of iteration according to the equation given below:

$$u_{ij} = \frac{1}{\sum_{l=1}^{c} \left[\frac{d}{d_{il}}\right]^{\frac{2}{m-1}}} \tag{3}$$

The equation mentioned below is used to compute the centroid of each cluster:

$$c_j = \frac{\sum_{i=1}^{N} pos(node_i)}{\sum_{i=1}^{N} u_{ij}^{m}} \tag{4}$$

The FCM clustering algorithm is given below:

Algorithm: Fuzzy C-Means (FCM) Algorithm [5]

```
Input: Position of nodes
Output: Center of the cluster
begin
    initialize $\cup f$
    repeat
        for cluster j=1 to c do
            $c_j \leftarrow$ compute cluster centroid
        end for
        update $\cup f$
    until the algorithm converges
    return c
end
```

2.3 Genetic algorithm

Genetic algorithm is based on the genetic principle and concept of natural selection and evolution. Speaking in terms of biology, every individual is obtained by the combination of the parent chromosomes. GA technique follows the same principle of chromosomes. Here in order to evaluate the chromosome we identify a function called as fitness function for the concerned problem. Now the new population is obtained by the operations such as selection, crossover, repair and mutation. The main aim of selection mechanism is to identify the best individuals (parents) for crossover and mutation. The role of crossover mechanism is to exchange the genetic materials of the parents and provide the best genes to the offspring. The role of mutation mechanism is to provide new genetic materials to the offspring.

We explain the routing of data with the help of chromosomes technique, which provides genetic information for genetic algorithm. Consider a given network, on which every chromosome contains a set of genes and each chromosome presents a particular arrangement of nodes in the routing chain [6]. A chromosome is made out of three different things as shown in **Figure 5**, a gene index, gene value and the gene. The gene index tells the location of the node in the network, the gene value tells the node's identification number (ID) and the gene tells what gene is present in the node. Let us consider the size of chromosome to be some integer N, where every integer is unique and lies between 1 and N. These integers represent the individual gene values that make up the chromosome.

i. **Initial population**: A population is nothing but a collection of chromosomes represented as P = $(C_1, C_2, \ldots., C_r)$. In this algorithm the first ever population

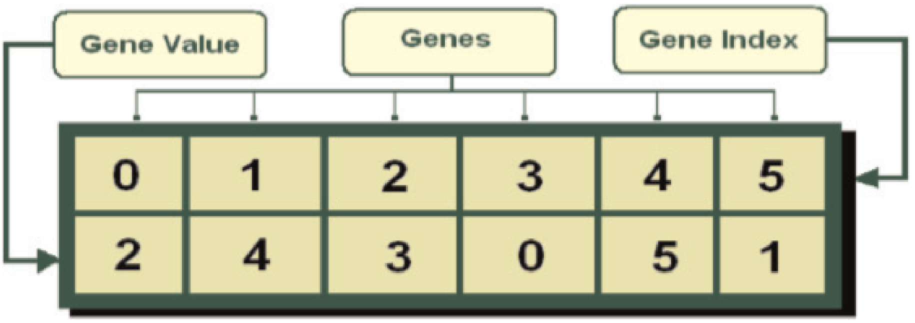

Figure 5.
A chromosome containing 6 genes.

or the initial population is randomly generated by using some random function in order to initiate the algorithm. The initial population consists of r chromosomes. Greater the value of r, greater the probability of getting a better solution. When the chromosomes are generated, care should be taken to see that the distance between two consecutive gene values do not exceed the threshold distance for communication d_{TH}.

ii. **Parent selection**: In this process, each time a selection of two chromosomes is done out of r chromosomes, which will decide which two chromosomes will be mating in order to obtain the offspring.

iii. **Generation**: With the help of crossover and mutation mechanism the new generation will be created.

iv. **Crossover**: Crossover mechanism indicates what combination of the two parent chromosomes can achieve the best offspring.

v. **Evaluation and fitness**: While creating a new generation care should be taken to see that the new generation meets the survival of the fittest condition.

vi. **Repair and mutation**: *Repair:* If the newly generated offspring violates the constraints imposed on it then the algorithm should be repaired in order to obtain the optimized solution. *Mutation:* Mutation adds variation to the next generation. *Cooling schedule:* Anneal temperature is one of the important parameter. In this case, every time the θ of particles approaches a better solution there is a decrement in the value. Let θ_i be the initial temperature and θ_f be the final temperature and t be the cooling time, then the equation of are cooling schedule will be as given below:

$$\theta(t) = \theta_f + \left(\theta_i - \theta_f\right) * \alpha^t \tag{5}$$

```
Algorithm: Genetic Algorithm [6]
Inputs: A set of N sensor nodes along with their position co-ordinates
Output: An ordered sequence of the nodes.
Step 1: Generate the initial population.
Step 2: for N times do:
      Step 2.1: Parents should be selected
      Step 2.2: Crossover has to be performed
      Step 2.3: Evaluate the offspring in order to select or reject it
      Step 2.4: Repair selected offspring and perform mutation
      Step 2.5: Store the produced offspring for the next generation
Step 3: Mark the best offspring in the generation as $C_{best}$
       Increment generation number by 1
Step 4: Find the anneal temperature
       if $\theta(t) \geq \theta_f$ go to step 2
       else required sequence is represented by $C_best$
```

2.4 Particle storm optimization

The particle swarm optimization (PSO) [7] technique was based on the real time activity such as a group of birds flying together, a group of fishes swimming together and the search techniques used by them in order to search their food. It is seen from nature that all winged creatures like birds, fishes etc., all travel together in groups in search of food without colliding with each other. This is possible because they all keep a track of the group (cluster) they belong to and they also have the knowledge of the speed and distance of all the members of that group. So based on the information they have, they change their speed and location.

PSO comprises of a swarm of a predefined size (say N_P) of particles. Every element has a solution to every multi-dimensional issue. For a particular set of particles the dimension D is equal. A particle P_i, $1 \leq i \leq N_P$ has position X_{id}, $1 \leq d \leq D$ and velocity V_{id} in the d^{th} dimension of the hyperspace. The notation for demonstrating the i^{th} particle P_i of the inhabitants as follows:

$$P_i = [X_{i,1}, X_{i,2}, X_{i,3}, \ldots., X_{i,d}] \tag{6a}$$

Each particle is associated with a fitness function which will tell about the superiority of the solution. In order to update the individual position and velocity every particle P_i monitors its individual best called P$best_i$ and global best called G$best$ to achieve the global best position. In every repetition, its position X_{id} and velocity V_{id} and in the d^{th} dimension is modified utilizing the accompanying conditions.

$$\begin{aligned} V_{i.d}(t) &= w * V_{i.d}(t-1) + c_1 * r_1 * \left(Xpbest_{i.d} - X_{i.d}(t-1)\right) \\ &\quad + c_2 * r_2 * \left(Xgbest_d - X_{i.d}(t-1)\right) X_{i.d}(t) \\ &= X_{i.d}(t-1) + V_{i.d}(t) \end{aligned} \tag{6b}$$

where w is the inertial weight, c_1 and c_2 are the non-negative constants called acceleration factor, and r_1 and r_2 are the two different consistently disseminated arbitrary numbers in the range [0, 1].

The modified procedure is iteratively repeated until either an acceptable G$best$ is attained or a fixed number of iterations $t_m ax$ are achieved.

Description of the algorithm

- Every particle is associated with position and velocity.

- Every particle is aware of its position and the value associated with it.

- Every particle is aware of its best position (P$best$) it has ever achieved and the value associated with it.

- Every particle has the knowledge about its neighbors and their best position (G$best$) and the value associated with it.

Every move of a particle is made by considering one of the three possible choices:

- First one is to follow its own way as it has decided.

- Second is to go back to its previous best position.

- Third is to go to the previous best position or current best position of its neighbor.

```
Algorithm: PSO Algorithm [8]
Initialize a population of particles
Do
    for each particle $P_i$ with position $X_{i.d}$ do
        if ($X_{i.d}$ is better than P$best_i$) then
            P$best_i \leftarrow X_{i.d}$
        end_if
    end_for
    Define G$best$ as the best position found so far by any of P's neighbors
    for each particle P do
        $V_{i.d} \leftarrow$ Compute_velocity($X_{i.d}$, P$best_i$, G$best_i$)
        $X_{i.d} \leftarrow$ update_position($X_{i.d}$, $V_{i.d}$)
    end_for
while (a stop criterion is not satisfied):
```

$$\begin{aligned} V_{i.d}(\mathrm{t}) = {} & V_{i.d}(\mathrm{t}-1) + c_1 * r_1 * (\mathrm{P}best_i - X_{i.d}(\mathrm{t}-1)) \\ & + c_2 * r_2 * (\mathrm{G}best_i(\mathrm{t}-1) - X_{i.d}(t-1)) \end{aligned} \tag{6c}$$

$$X_{i.d}(t) = X_{i.d}(t-1) + V_{i.d}(t) \tag{7}$$

2.5 Genetic swarm optimization

The combination of GA and PSO algorithms is the GSO algorithm [7], which formulated to obtain better performance. The main elements of GA are crossover and mutation. The strength of the algorithm is that it is more robust and adaptable. It does not bother about whether the problem is continuous or discrete. What it is bothered is only about whether the problem can be solved. Therefore GA overcomes all the problems and puts light on optimizing the process. One drawback of GA is that its convergence speed very low, whereas, the convergence speed of PSO is relatively fast as compared to GA. Therefore we combine GA and PSO in order to overcome the drawbacks. GSO is different from PSO. The GSO algorithm has combined the core elements of GA i.e., crossover and mutation with PSO so as to improve the performance and efficiency of the PSO algorithm. **Figure 6** shows the process of GSO [7].

Step 1: The numbers of particles are initialized (circuit codes are initialized).

Step 2: The speed and position of each particle are randomly initialized.

Step 3: In this step we calculate the fitness value of each particle. We then store the fitness value and current position of every particle in its P*best*. Later the comparison of all the P*best* is done and we store the fitness value of the best individual in G*best*.

Step 4: The particle swarm is randomly selected and crossover and mutation operations are executed on it.

Step 5: The next step is to update the speed and position with all the particles with expressions (1) and (2).

Step 6: Finally we judge whether the algorithm is achieving convergence criteria. If the convergence touchstone is achieved, then output the best person and the search algorithmic program, otherwise repeat step 3.

When compared with the GA and PSO, GSO mainly has the following advantages:

i. There is much of the improvement in the search capability.

ii. The convergence rate of the partial area is also enhanced.

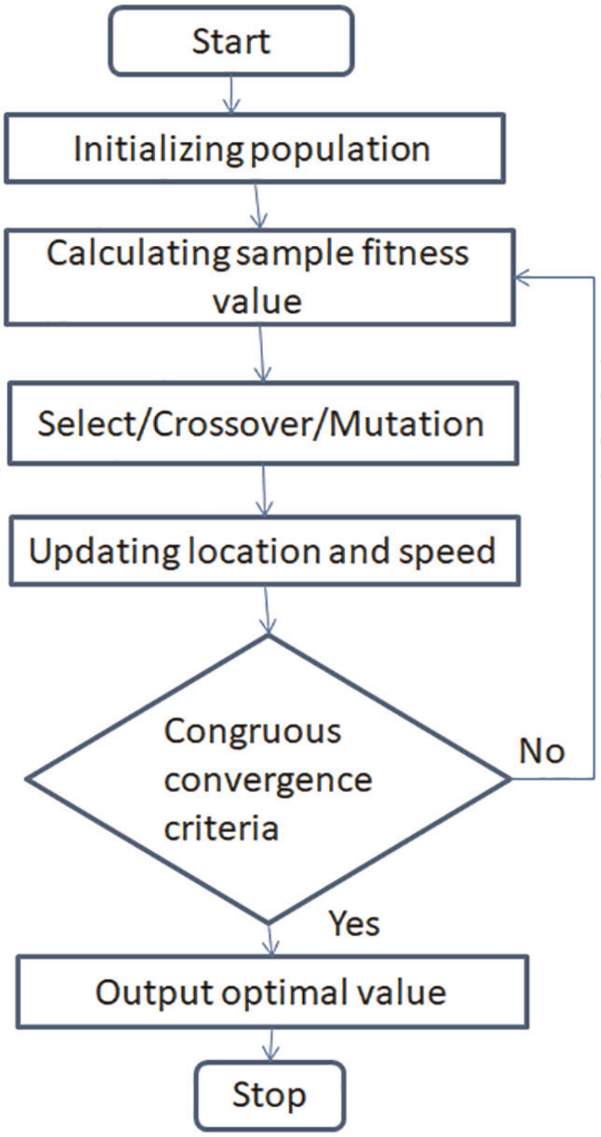
Start
Initializing population
Calculating sample fitness value
Select/Crossover/Mutation
Updating location and speed
Congruous convergence criteria
No
Yes
Output optimal value
Stop

Figure 6.
Process of GSO algorithm.

iii. The partial area convergence stagnation phenomenon which existed in the search appendage of GA is avoided in this algorithm.

iv. The searching truth can be improved and the number of successful evolution looping can be reduced.

2.5.1 Objective function design of GSO

In the optimization process, the fitness functions are given from objective functions. The outcome of the fitness function called the fitness value, shows the ability of individual and adapts to the evolution. By optimizing the process one can achieve greater value, stronger adaptability and preserving the evolution. It should be taken into consideration that the difference among the individual fitness value cannot be neither too big nor too small so as to overcome the fall in local optimum and avoid slow convergence. The fitness value is directly related to the algorithm accuracy and convergence speed. We can get the target function y based on the truth table. The fitvalue is given by:

$$fitvalue = \sum_{1}^{2^n} fitnumber_i \tag{8}$$

where $fitnumber_i$ is the i^{th} input combination of truth table suffice for the evolution circuit and n is the number of inputs.

2.6 Minimum energy

Figure 7 shows the flow of minimum energy (MINEN) [9] algorithm. The steps involved in the execution of the protocol are as follows:

- Firstly we identify the nodes that are not participating in the current scenario. This step is optional.

- Secondly, we form a cluster and elect a cluster head so that all the communications can be carried out through the cluster head.

- Third is, designing a DAG which shows the connections between all the cluster heads and the weights associated with each link.

- Finally, we run the Dijikstra's algorithm to find the shortest path between the cluster head and the base station.

In order that the protocols performs well, we need to make some assumptions such as:

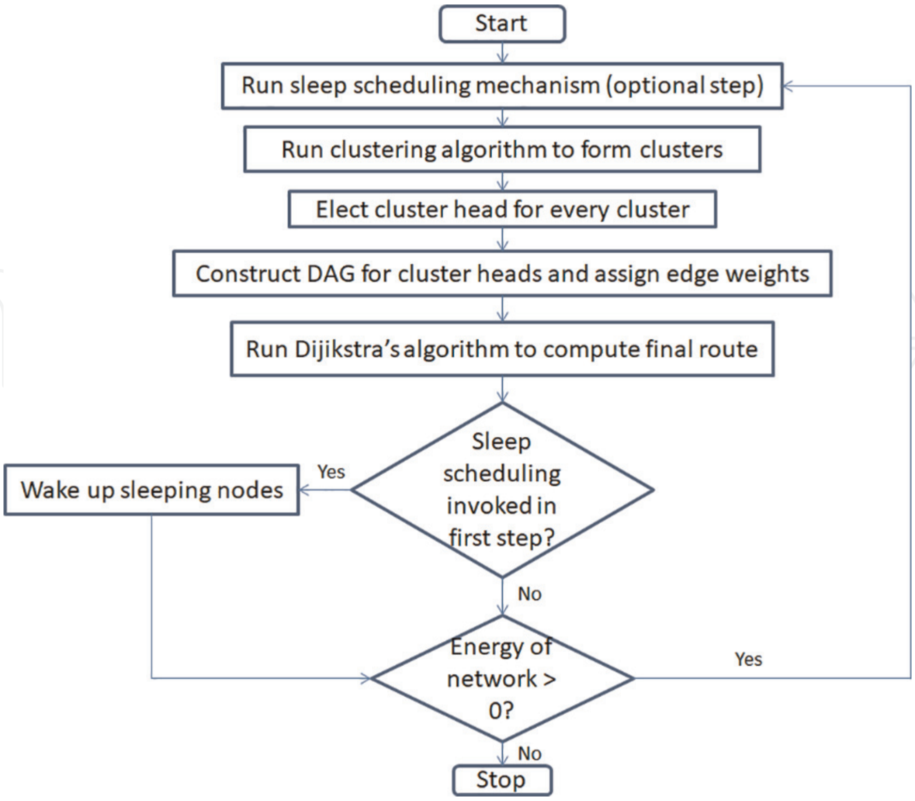
Start
Run sleep scheduling mechanism (optional step)
Run clustering algorithm to form clusters
Elect cluster head for every cluster
Construct DAG for cluster heads and assign edge weights
Run Dijikstra's algorithm to compute final route
Sleep scheduling invoked in first step?
Yes
Wake up sleeping nodes
No
Energy of network > 0?
Yes
No
Stop

Figure 7.
Flow of MINEN algorithm.

- All devices have the same level of energy at the beginning.
- In an IoT network there is only one base station located that is static in nature.
- It is assumed that the base station is provided with infinite energy, i.e., there is no risk of the base station to be shut down due to lack of energy.
- It is assumed that one round of communication is calculated as the time period between the election of new cluster head and successful transmission of messages from all the cluster heads to the base station.

We now explicitly detail each step of the algorithm shown in **Figure 7** in the following subsections [9]:

```
Algorithm: MINEN Algorithm [9]
1. Run the GSO sleep scheduling algorithm (optional)
2. Run clustering algorithm to create cluster set C
3. for c $\in$ C do
4.      for i $\in$ c.devices do
5.           if d.current energy ¡ i.current energy then
6.                d = i
7.      Set c.cluster head = d
8. Create graph G connecting all cluster heads
9. Run Dijkstra on G
10. Return routing path
11. if sleep scheduling is invoked in step 1 then
12.      Wake up all sleeping devices in the network
13. if Number of active devices > 0 then
14.      goto step 1
15. else exit
```

3. Conclusions

In this chapter we have discussed about MINEN routing protocol [9] for IoT-WSN. MINEN protocol is primarily based on cluster rule where the amount of energy utilized is distributed equally among all the devices of the network. This is achieved by using the clustering method, giving every node a chance to become a cluster head and reducing the energy utility of transmitting and receiving messages as well as devices with low residual energies across the link. This is done following the various steps such as, designing a DAG wherein the nodes of the graph act as cluster head. Then we assign appropriate weights to the links of the graph taking into consideration the energy required for transmitting and receiving messages over a link called the energy spent (E_{sf}). The E_{sf} factor enforces energy load balancing across several links. The Dijikstra's Algorithm is used to find the minimum cost (energy) between the cluster head of the sender to the base station. Further, the GSO sleep scheduling algorithm along with the MINEN algorithm would help in enhancing the energy conservation effort. MINEN performs better when compared to GSO. MINEN overcomes the drawback of the two most widely used algorithms namely LEACH and FCM, in terms of network coverage, number of alive nodes and energy dynamics.

For future work we can extend the protocol such that it can be used in IoT where all the nodes are mobile nodes and there no end-to-end path between the source and the destination. Further, work can be progressed to achieve better performance by improving the sleep scheduling algorithm. Work can also be done to improve the clustering techniques, where clusters result in more energy conservation.

Abbreviations

ADV	advertisement message
CDMA	code division multiple access
CH	cluster head
CM	cluster member
DAG	directed acyclic graph
FCM	fuzzy C-mean
GA	genetic algorithm
GHG	greenhouse gases
GMM	Gaussian mixture model
GPS	global positioning system
GSO	genetic swarm algorithm
ICT	information and communication technology
IoT	Internet of Things
LEACH	low energy adaptive clustering hierarchy
MAC	media access control
MEMS	microelectronic mechanical systems
MINEN	minimum energy
PSO	particle swarm algorithm
TDMA	time division multiple access
WSN	wireless sensor networks

Author details

Gowrishankar Subhrahmanyam*† and Ishwari Ginimav*†
BMS College of Engineering, Bengaluru, India

*Address all correspondence to: gowrishankar@bmsce.ac.in and ishwari29@gmail.com

† These authors contributed equally.

References

[1] Dhawan H, Waraich S. A comparative study on LEACH routing protocol and its variants in wireless sensor networks: A survey. International Journal of Computers and Applications. 2014;**95**(8):0975-8887

[2] Dabaghi F, Movahedi Z, Langar R. A survey on green routing protocols using sleep-scheduling in wired networks. Journal of Network and Computer Applications. 2017;**77**:106-122

[3] Yadav L, Sunitha C. Low energy adaptive clustering hierarchy in wireless sensor network. International Journal of Computer Science and Information Technologies. 2014;**5**(3):4661-4664

[4] Available from: https://www.researchgate.net/publication/324943617_Analysis_of_Energy_Efficient_Hierarchical_Routing_Protocols_in_Wireless_Sensor_Networks/figures?lo=1

[5] Tamene M, Rao KN. Fuzzy C-means clustering algorithm for optimization of routing protocol in wireless sensor networks. International Journal of Computer Science and Network. 2016; **5**(2):2277-5420

[6] Chakraborty A, Mitra SK, Naskar MK. A Genetic Algorithm Inspired Routing Protocol for Wireless Sensor Networks

[7] Junbin Z, Jinyan C, Yafeng M, Tianzhen M. Genetic algorithm particle swarm optimization based hardware evolution strategy. WSEAS Transactions on Circuits and Systems. 2014;**13**:274-283

[8] Sudarmani R, Shankar Kumar KR. Particle swarm optimization based routing protocol for clustered heterogeneous sensor networks with mobile sink. American Journal of Applied Sciences. 2013;**10**(3):1546-9239

[9] Vashishth V, Chhabra A, Khanna A, Sharma DK, Singh J. Energy Efficient Routing Protocol for Wireless Internet-of-Things Sensor Networks. Vol. 22019: 1-12