Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/120411

A Hierarchical Approach to Node Localization Based on Fuzzy System to Increase Network Lifetime in Wireless Sensor Networks

M. Karthiga[*] *and R. Venkatesan*

Department of Computer Science and Engineering, PSG College of Technology, Coimbatore-641004, India

Received: 14 Jul. 2017, Revised: 15 Mar. 2018, Accepted: 21 Apr. 2018
Published online: 1 Jul. 2018

Abstract: Energy efficiency is of utmost importance in Wireless Sensor Networks (WSN). A WSN that lives long proves to be more worthy for the effort and cost involved in establishing the network. The standards and protocols in a WSN are expected to be economical in their energy usage. Localization is one such important area, where much contribution has been made by the researchers to improve the accuracy and not much in increasing their energy efficiency. In this work, a hierarchical approach to increase energy efficiency during localization has been proposed. Existing localization approaches, irrespective of the node's remaining energy levels, use a common strategy for all the nodes. To balance the energy usage among nodes, the proposed algorithm uses different strategies based on current energy level. Clustering is performed as part of localization and the cluster heads also serve as localization heads for their cluster members. Range estimation is done using fuzzy system and localization is performed by genetic simulated annealing. Simulation has been carried out in NS2 and the results prove the efficiency of this approach over other approaches in saving energy as well as providing high localization accuracy.

Keywords: Wireless Sensor Networks; Localization; Mobile Anchor; Network Lifetime; Fuzzy System; Genetic Simulated Annealing.

1 Introduction

A Wireless Sensor Network (WSN) is a network of energy constrained, memory constrained, and computation constrained sensor nodes deployed in the sensing field [1]. Energy scarcity has kindled the development of energy efficient protocols to increase network lifetime. Network lifetime is the time until which the first sensor node or group of sensor nodes run out of energy.

This paper introduces an energy efficient approach for localization to increase the network lifetime. Localization methods help sensor nodes identify their physical location after deployment. Broadly the localization algorithms are classified into range-based and range-free.

Range-based algorithms are more accurate and expensive. Range-free algorithms are less accurate and require no expensive hardware. They use messages passed inside the network to localize themselves. Considerable research has been done to increase the localization accuracy of range- based and range-free approaches [2,3,4].

Increasing the accuracy increases communication, which in turn brings down the network lifetime. But quite a number of researches have been dedicated to increase the energy efficiency of the localization strategies as well. They range from simple approaches like maintaining the simplicity of existing algorithms like WCL [5] to sophisticated methods like using additional hardware (directional antennas) [6] or transceiver optimization [7]. Certain other approaches have optimized the communication overhead to achieve energy efficiency as in [8] where the authors have proposed a method where the nodes get to choose optimal anchor pairs. A novel approach was proposed by the authors in [9] where the power of the anchor's transmitter is varied and varying range beacons are received by the sensor nodes. This reduced communication has been used to achieve energy efficiency. The authors of [10] used the concepts of anchor pair selection, hop size modification and reduced communication between the anchor and beacon nodes to achieve energy efficiency. In [11] the authors mitigated

[*] Corresponding author e-mail: karthibtech06@gmail.com

broadcast flooding and duty cycling to achieve energy optimization.

A common factor underlies all the above existing strategies.

The strategies are applicable for all the nodes, in variable of its residual energy. Hence the energy difference among the nodes persists in these algorithms.

Thus the motivation behind the proposed approach, Hierarchical Localization Strategy (HLS), is to improve the lifetime by narrowing down the energy gap among the nodes.

Our previously developed localization strategy namely MAP-GSA [4] achieved an RMSE as low as 1.6 m but at the expense of very high communication. If the number of messages needed by MAP-GSA could be reduced at least for the low energy nodes, the strategy will achieve good accuracy and as well as improve the energy efficiency.

The major differences between the existing energy efficient localization algorithms and the one proposed are, (1) the proposed algorithm does not use a common strategy for localization for all the nodes (2) it uses a cross layer approach. Cluster heads also serve as localization heads for its cluster members (3) the approach does not require special hardware like directional or smart antennas.

In order to improve the network lifetime, energy dissipation of the nodes must be proportional to their battery level. This proportional expenditure makes sure that high and low energy nodes run out of energy more or less around the same time. The proposed HLS works based on this principle. The algorithm classifies nodes as high energy (Master nodes) and medium/low energy nodes (Listener nodes). Master nodes can afford localization with high communication. Localization in listenerNodes has to be achieved with very few messages in order to sustain its remaining battery power. Hence they depend on master nodes.

The proposed approach brings down the average number of messages needed by the listener nodes by a huge number using a fuzzy-based learning process.

Existing fuzzy-based localization systems use fuzzy logic for determining partial solutions to node's locations [12] or for determining the edge weights accurately for weighted centroid algorithms [13]. In some cases fuzzy logic has been used for positioning users in indoor environments [14]. Few researches have used the fuzzy inference system to improve existing localizing technologies like trilateration [15] and ring overlapping approaches [16]. In [17] the authors used fuzzy system for positioning the nodes in a noisy environment.

Unlike these approaches where the fuzzy inference system is used for improving the accuracy, in *HLS* the fuzzy system has been used to achieve energy efficiency in listenerNodes by continuously learning the RF environment and providing nearly accurate range estimates between master and listener Nodes for localization. Three such accurate estimates are sufficient to position the listener Nodes with high accuracy.

The remaining part of the paper is organized as follows. Section 2 gives insight into a sensor node's energy dissipation model. Section 3 details on clustering mechanism used and localization strategy adopted for master nodes. Section 4 details on the localization strategy of listenerNodes. Section 5 illustrates the simulation parameters and various scenarios used for the study. Section 6 investigates the efficiency of the proposed algorithm. Section 7 concludes the work with the inferences made.

2 Energy dissipation model of a sensor node

The fundamental components of a sensor node are sensing unit, transceiver, microcontroller unit (MCU) and a small battery [18]. The energy consumed by the MCU is very less compared to the transceiver unit. Roughly, the energy cost of transmitting data of 1 Kb over 100 meters is approximately as same as executing 3 million instructions [19]. Sensing unit, in continuous event sensing applications consume very high power whereas in sporadic sensing, consumes less power. Thus power optimization in the sensing unit is not efficient as it is highly application specific. Hence optimization in the transceiver unit can substantially reduce the energy consumption.

2.1 Energy States of a Transceiver

The energy needed by a transceiver during transmission and reception [20] is given by,

$$Energy_{\mathrm{tx}} = T_{\mathrm{st}}P_{\mathrm{st}} + \frac{n}{R_b R_c}\left(P_{\mathrm{txElec}} + P_{\mathrm{amp}}\right) \tag{1}$$

T_{st} is the time taken for the transmission circuit to start up. P_{st} is the power needed by the circuit during T_{st}. Thus the startup energy is given by (T_{st} * P_{st}). Let the total bits to be transmitted be 'n', R_b be the bit rate, and R_c be the coding rate.

Hence 'n/(R_b*R_c)' is the time needed for transmission of n bits. P_{amp} is the power consumed by the amplifier and P_{txElec} is the power taken by the rest of the circuit during transmission. Thus the second term gives the energy needed during transmission of 'n' bits.

The energy needed by the receiver during reception [20] is given by,

$$Energy_{\mathrm{rx}} = T_{\mathrm{st}}P_{\mathrm{st}} + \frac{n}{R_b R_c}P_{\mathrm{rxElec}} + nE_{\mathrm{dec}} \tag{2}$$

P_{rxElec} is the power taken by the circuit and E_{dec} is the energy needed to decode a single bit.

Let T be the time at which a decision, whether a node should remain active or be put in a deep sleep state, is taken. Suppose the node is decided to be put in an active state at T. Let 'T_1' be the time of next event after 'T'. Let 'P_{active}' be the power needed if the node is active. So the first term in Eq. (3) gives total energy needed by the node, if the node remains active during T to T_1 or in other words remain in an idle listening state.

Suppose the node is decided to be put in a deep sleep state at T. Let 'T_{down}' be the time taken by the node to achieve deep sleep state and the let power consumed during T_{down} be P_{down}. So the second term in equation (3) gives energy consumed by the node to achieve deep sleep state. The total time the node remains in sleep state is 'T_1-T-T_{down}.' Hence the last term gives total energy needed by the node in deep sleep state.

Thus energy saved when a node is put in a deep sleep state as opposed to being in an idle state is given by,

$$E_{saved} = (T_1 - T)P_{active} - [T_{down}P_{down} + (T_1 - T - T_{down})P_{sleep}] \quad (3)$$

From Equation (1), Equation (2) and Equation (3) it can be inferred that, to reduce the energy consumption during communication, any optimization strategy must reduce the,

1. Total time the transceiver is turned on.
2. Number of times the transmitter and receiver are turned on.
3. Time spent on idle listening.

In simpler terms, reducing the number of messages communicated, it is possible to simultaneously achieve (1) and (2). Localization on demand and putting a node in deep sleep state reduce idle listening and frequent switching between transceiver states.

The proposed work tries to incorporate all the above inferences made from the transmission states.

3 System Model and Localization of High Energy Nodes

The study assumes random placement of 'n' sensor nodes in a two-dimensional sensing field. There are 'm' mobile anchor nodes. Anchor nodes move through the sensing field, broadcasting beacons at regular intervals. Two nodes $Node_i$ and $Node_j$ are neighbors, if they are in communication range of each other. The communication range of all the sensors is the same.

3.1 Clustering Scheme

Each sensor begins a hello timer after deployment and at the beginning of every clustering phase. The timer's value is inversely proportional to the remaining energy of the node [21]. The timer of the Sensor Node (SN) which has the highest energy goes off first. This SN broadcasts its residual energy to all its one hop neighbors. All the receiving one hop neighbors switch off their timers if the received energy is higher than its own residual energy. The SNs categorize themselves into two categories. SNs which have energy levels that are on par with the broadcasting node become the *Master nodes.* Remaining low energy nodes which have energy levels much lesser than the broadcasting node become the *listener nodes.* The broadcasting node is understood as the ClusterHead (CH) by the neighbors and if there are two or more equal energy nodes, one of them is chosen randomly as a CH and the other turns out to be a *master node* of the Cluster. The master nodes have a high duty cycle, and the listener nodes have smaller duty cycles to minimize idle listening. Master nodes are independent in their localization mechanism whereas the listener nodes depend on the master nodes. After a certain time (pre-determined) the clustering begins again with re-election of ClusterHead and master nodes.

Unlike LEACH [22] where a CH cannot be elected again for a few rounds, here the node can be elected as CH again if it still has the highest remaining energy among all its neighbors and the same goes for master nodes.

3.2 Localization of Master Nodes

The master nodes have a comparatively higher duty-cycle. The mobile anchors when they move across the sensing field broadcasts beacons. The master nodes in the anchor's vicinity receive as much beacons as possible and use MAP-GSA for localization. Our prior work to HLS namely MAP-GSA was a highly fine grained localization with a RMSE of 1.6 m. Since the listener Nodes are completely dependent on masters, the accuracy of localization for the master nodes has to be very high to mitigate propagation of localization error. A brief insight to MAP-GSA is given for better understanding of HLS.

3.2.1 Mobile Anchor Positioning (MAP)

Mobile anchors periodically broadcast beacons as they move through the network. All the sensor nodes in the communication range of the anchor node receive the beacons.

The beacons contain the location of the mobile anchor at the time of sending the beacon. A master node receives as many beacons as possible.

At the time of localization, each master node picks up two beacons (S_1 and S_2) which are farthest from each other (Refer Figure 1).

These two beacons mark the boundary of master node's communication range. With those two beacon co-ordinates (S_1 and S_2) as centre and the sensor node's communication range as radius, the sensor node constructs two circles. The two circles must intersect each other as both beacons are within communication range of the sensor node.

Now, the intersection points of the two circles (P_1 and P_2) can be concluded as probable locations of the sensor node.

The reason being,

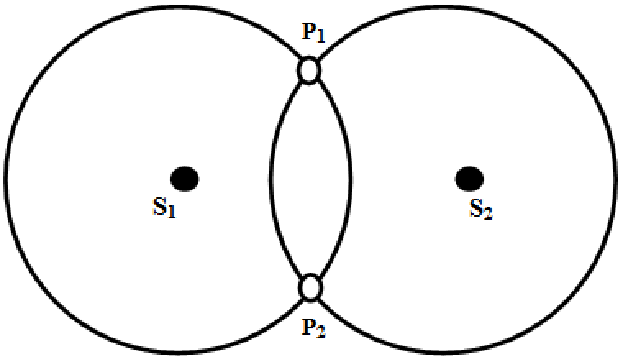

Fig. 1: Probable locations of sensor node determined by MAP.

1). The beacon points lie on the boundary of communication of the sensor node. The communication range of the sensor node and the anchor are the same. Hence, the sensor lies in the boundary of the beacon's communication range.

2). The sensor node should lie in the boundary of communication of both the beacons. Hence the two intersection points of the circles are the probable locations of the sensor node.

The two locations determine the approximate location of the sensor node. Still there remains a non-negligible error of localization. To bring down this error, Genetic Simulated Annealing (GSA) is used.

3.2.2 Genetic Simulated Annealing

The efficiency of evolutionary algorithms in solving critical issues in WSN has been proven.

In this approach, GSA has been used to determine with accuracy the node's location from the approximated location (as a result of MAP). This proceeds in the following phases.

Genetic Encoding

The genes to perform GSA are to be encoded. Since at the end of localization, a two dimensional location of the sensor node is needed, the gene has the following structure. The first chromosome represents the position of sensor node in x axis and the second chromosome represents the position of the sensor node in y axis. Both the chromosomes are real valued.

Population Initialization

Each gene in the population represents a possible location of the sensor node. Each gene in the population is generated around the possible locations (from MAP) of the sensor node in its x axis or y axis or both. This method of population initialization narrows down the solution space. The function to initialize the population determines the accuracy of the calculated locations.

Fitness Function

Let (a,b) represent the anchor's location (from beacon packet received). The gene in the population is represented as (x,y). Let 'current_distance' be the distance between a gene in the population (x,y) and the anchor's location. The actual distance measured between the master node and the anchor is d_i. Master nodes measure many such d_is (via methods like RSSI or TDOA) when the anchor is at locations (a,b). With these definitions, the fitness function for this problem is defined as minimizing the difference between the current_distance and the actual distance (d_i) thereby enhancing localization accuracy. Let the size of initial population be 'n' and the number of measured distances be 'm.' The fitness function for each gene in population is hence given by the following function.

$$\sum_{\substack{1 \le i \le n \\ 1 \le j \le m}} \text{abs}\left(\sqrt{(x_i - a_j)^2 + (y_i - b_j)^2} - d_j\right) \quad (4)$$

Population Selection

The best among the population is selected based on the fitness value of each individual in the population. The selected individuals undergo reproduction which forms the population for the next generation. Roulette wheel selection is the chosen method of selection.

Reproduction

For each individual *i* in the population, a small value Δd is calculated by equation (5). 'Δd' is a function of the difference between the actual_distance (d_i) and current_distance.

$$d = \alpha * \left(\sqrt{(x_i - a_j)^2 + (y_i - b_j)^2} - d_j\right) \quad (5)$$

In Eq. 5, the value of 'α' ranges from 0 to 1. Thus this Δd is used to produce a small change in the corresponding individual, by either adding or subtracting Δd in one of the axes or both.

During reproduction phase, single point crossover is performed on all the individuals to form the next generation. Selection and reproduction can be iteratively performed till the minimal fitness is reached. In order to improve the localization accuracy further, simulated annealing is performed.

Simulated Annealing

Simulated annealing is performed on a small portion of the population. Initially a high temperature is initialized for annealing. The temperature goes down gradually throughout the annealing process. For each individual, a new neighboring solution is created. The decision of whether moving to the new solution depends on the fitness of the new individual and the current temperature. Elaborating, if the fitness of the new solution has improved, the new individual is selected with probability 1. If the fitness has weakened, the new individual is chosen with probability between 0 and 1.

The probability of choosing the new state of the individual is given by the Boltzmann–Gibbs distribution [23]. This is given by Eq. 6

$$e^{d/T} > R(0,1) \tag{6}$$

Here Δd is the difference in fitness of old and the new modified individual. T is the current temperature of the annealing process. R is a random number between 0 and 1.

When the temperature is high, the algorithm allows moving to the new solution even if the new solution does not provide improvement. This step allows the algorithm to skip getting stuck in local minimum and takes risk for arriving at global minimum. As the temperature comes down, the probability of moving to less fit solution decreases.

A random number R is generated. If the value of the decision factor is greater than R, then solution is accepted else it is rejected.

GSA substantially reduces the Root Mean Square Error (RMSE) of localization. MAP-GSA has been proved as an efficient method of localization by various experimentations. But it imposes a tax on the sensor node's battery, as the sensor node has to receive a lot of beacons for localization. The next method of localization works around a way to solve this problem efficiently.

4 Learning the RF Environment by Master Nodes

4.1 Fuzzy System for Localization

The approach proposed in our work puts forward that the fuzzy system which has been widely used to increase localization accuracy can use the learning process to bring down the communication inside the WSN.

Master nodes, during localization, can simultaneously learn about the local RF environment with the help of the beacon messages received. Trivial Position estimating systems depend on the relationship between distance and RSSI for range estimation. Generally, range is calculated from RSSI, based on any of the radio propagation models. But in this work, instead of assuming a single radio propagation model throughout the network's lifetime, learning the path loss pattern from the RF environment every now and then is done, giving more accurate range estimates. fuzzy system is used for this RF environment learning process. This proceeds in following phases.

Learning Phase

Mobile anchors move around the sensing field passing beacons at regular intervals. The beacons convey the anchor's location at the time of sending the beacon message. Whenever mobile anchors come in communication range of a master node, the master nodes receive beacons as part of their localization process.

Since the master node, by now, is already aware of its location, with the beacon received, it is capable of determining the accurate distance between the sending anchor and itself. The receiving master node records the corresponding RSSI of the beacon message as well.

The received RSSI values and calculated distance values are used to define membership functions. Once the learning phase is over, the system has to convert the input RSSI into corresponding distance. Fuzzy inference system does this.

Define Linguistic Variables and Terms

Linguistic variables have values as words or group of words from natural language. RSSI and DISTANCE are input and output linguistic variable for the proposed work. Linguistic values of these variables are {very high, high, medium, low, very low} for RSSI and {very less, less, medium, high, very high} for Distance.

Define Membership Function

A membership function is used to fuzzify crisp inputs to fuzzy inputs during fuzzification. Triangular membership function is well suited for a WSN because of its computational simplicity. The triangular membership function [24] is defined by equation (7).

$$\mu(x) = \begin{cases} 0, & x \le a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ \dfrac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases} \tag{7}$$

μ(x) is the degree of membership varying between 0 and 1. x is the input RSSI value to be fuzzified. (a,b,c) are the three non-collinear points defining each fuzzy bin (each triangle).

Pictorially, it is represented by Figure 2. The master nodes during the learning phase collects RSSI and corresponding distance values. Based on the values, the membership functions are generated by each master node.

Building Fuzzy Inference System

Fuzzy inference system receives a RSSI and generates a corresponding Distance. This is built by human understandable If-then rules. If-then Rules are used to map input to output.

The input is RSSI and output needed is corresponding distance. The If-Then rules are tabulated in Table 1.

Table 1 Fuzzy Rules in Fuzzy Inference System

If RSSI is very low then Distance is very high.
If RSSI is low then Distance is high.
If RSSI is medium then Distance is medium.
If RSSI is high then Distance is less.
If RSSI is very high then Distance is very less.

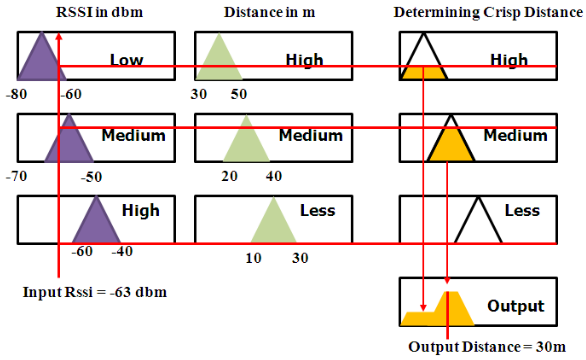

Fig. 2: Calculation of Fuzzy Distance from input RSSI

4.2 *Localization of Listener Nodes*

Localization of the listener nodes happens in two phases. First phase is range estimation phase using the above fuzzy system. This phase approximates the node's distance from few master nodes. Second phase is location estimation phase, which identifies the node's geolocation using the distance estimates from range estimation phase.

Range Estimation Phase

Once each master node builds the inference system, they can help the listener nodes. When a listener node, wants to know its location, it broadcasts a help message. All the master nodes receiving the help message, respond back. The master nodes receiving the help message, identify the RSSI (the work can be extended to work with TOA/TDOA also) of the signal. The inference system uses the rule base to map the RSSI to distance. The input RSSI intersects the input fuzzy bins (refer Figure 2). The intersection points are extended to output distance bins.

This extension truncates the output bins at the height of intersection. The truncated area under each output bin is added to get the total area. The centre of gravity of the total area gives the crisp distance value corresponding to input RSSI value. This crisp distance is sent back as a reply to the listener node.

The listener node receives a message containing location of the master node and the distance between the master node and the listener node. Thus, the listener node identifies its range from all the master nodes in its vicinity.

By now, a listener node has few known locations and its distance from those known locations.

Location Estimation Phase

With few known locations and distances from them, GSA is used to determine the listener node's location with high accuracy.

Genetic Encoding Phase

The genetic encoding phase is similar to the GSA briefed above.

Population Initialization Phase

In the population initialization phase, each gene in the population is a function of the distance and the received location (both obtained from the master nodenode node's message) in its x chromosome or y chromosome or both.

Fitness Function

Let current_distance be the distance between the gene in the population and the master node's location (obtained from the message). Let actual_distance be the measured distance between master node and the listener node (sent by the master node as result of fuzzy inference system). Fitness function for this problem is defined as minimizing the difference between the current_distance (m) and the actual distance (d_{ij}).

Genes are represented by (x,y). The location of the listener node is (a,b). 'd' is the measured distance between the two nodes. Let 'm' be the total number of help messages received. Let 'n' be the population size. Now the fitness function is as follows.

$$\sum_{\substack{1\le i\le n\\ 1\le j\le m}} \text{abs}\left(\sqrt{(x_i-a_j)^2+(y_i-b_j)^2}-d_j\right) \quad (8)$$

Population selection, reproduction and simulated annealing proceeds similar to previously explained GSA. Fuzzy logic accurately estimates the range of listenerNodes from master nodes. With the ranges from known location obtained, genetic algorithm uses the range to search across the solution space for probable location of a sensor node. Simulated annealing searches the neighboring space of the solutions obtained and improves the accuracy by a huge amount.

Table 2: HLS at MasterNode

1: rssi ← Receive help message from listener node
2: fuzz_rssi ← Fuzzify the rssi of input message
3: fuzzy_distance ← Fuzzy inference system gives distance to the corresponding rssi
4: dist ← Defuzzify the fuzzy distance to determine crisp distance
5: Unicast [loc.(location of Master node), dist] to listener node

Table 3 HLS at ListenerNode

1: Broadcast help message to master Nodes in range.
2: [loc,dist] ← Receives location of master node and its range from the masterNode.
3: population ← Initialize population for Genetic Simulated Annealing.
4: Perform selection and reproduction.
5: Select the best among the population and perform simulated annealing.
6: Repeat above two steps iteratively till accurate location of listener node is reached.

Localization happening in the listener node and the master node is briefed in Table 2 and Table 3.

5 Simulation Setup and Performance Evaluation

The simulation setup is as seen as shown in Table 4 for analyzing the results. To study the performance of HLS in terms of its localization efficiency, various positioning parameters (like anchor density) are varied and evaluated. To measure its performance in terms of its energy efficiency, four different scenarios are implemented and tested. The results obtained are the average of 30 simulation runs. HLS is compared with LIP [11]. Section 5.2.4 debriefs LIP.

Table 4 Simulation Settings

Number of Nodes	100
Sensing field	1000m x1000 m
No. of Mobile Anchors	3
Beacon Interval	5s
% of ClusterHeads	10
Transmission Range	100 m
Receiving range	100 m
Initial Energy	(0.1 – 1.0) J
Simulation Time	2500 – 120000 s
Transmit Power	35.28e-3 W
Receiving Power	31.32e-3 W
Idle Power	712e-6 W
Sleep Power	144e-9 W

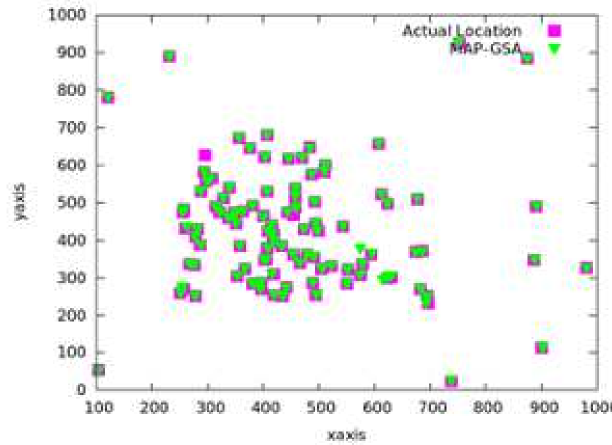

Fig. 3: Actual vs Identified Locations by MAP-GSA

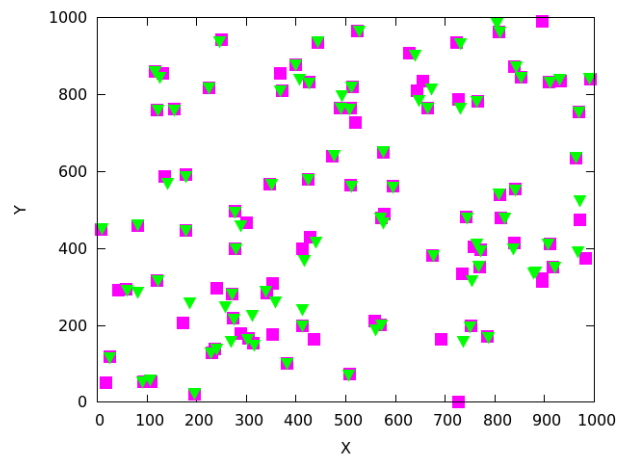

Fig. 4: Actual location vs. location identified by HLS

5.1 Study on Localization Accuracy

5.1.1 Plot of Actual Locations vs Calculated Locations

Fig. 3, Fig 4 and Fig.5 show the plot of actual location vs. identified location in a two-dimensional area.

The pink rectangle represents the actual location and the green triangle represents the calculated locations.

Quantitatively localization accuracy is measured by Root Mean Squared Error (RMSE). RMSE is a quantitative measurement of the difference between the results predicted by an estimator and the actual observed values. RMSE is calculated by Eq. 9, where (x_{ai}, y_{ai}) and (x_{ei}, y_{ei}) are the actual location and estimated location of the Sensor Node 'i' respectively and 'n' is the total number of SNs.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1\ to\ n}(x_{ai}-x_{ei})^2+(y_{ai}-y_{ei})^2} \quad (9)$$

The RMSE of LIP [11] is 14.6 m, HLS is 4.73 m but MAP-GSA has the least RMSE of 1.6 m. But this comes with the price of increased communication.

HLS performs better than LIP in terms of localization accuracy. In LIP since the SNs depend on each other in addition to the anchors for localization, the localization error propagation is high. Lower error in HLS when compared to LIP is attributed to the use of MAP-GSA as the localization strategy for some part of the WSN. Moreover, during the localization of listener nodes, the number of fuzzy bins in the FIS and the use of GSA as the position estimation strategy have highly influenced the RMSE of localization in HLS. Thus HLS could provide a tradeoff between very fine grained accuracy and energy efficiency.

5.1.2 Beacon Interval

Decreasing the beacon interval decreases the RMSE of localization in both MAP-GSA and HLS (Fig. 6). Since the nodes receive beacons from many new positions, the sensor nodes are able to achieve higher localization accuracy.

In HLS the accuracy of the master nodes is increased which positively affects the accuracy of the listenerNodes. In LIP, not much impact is observed as the beacons from the same positions are often broadcasted. Hence decreased

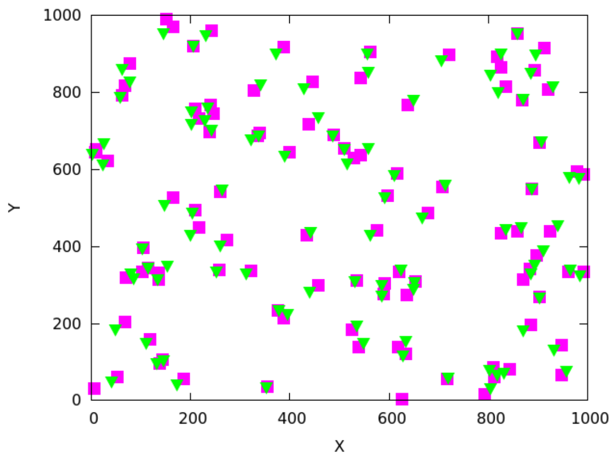

Fig. 5: Actual location vs. location identified by LIP

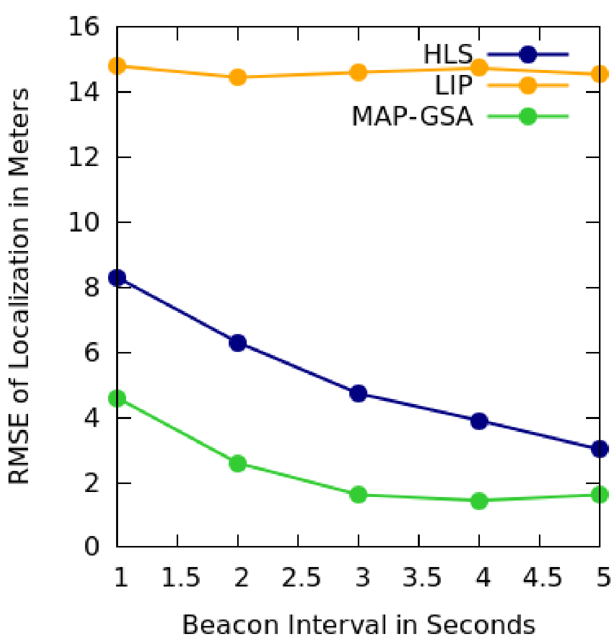

Fig. 6: Impact of Beacon Interval on Accuracy

beacon interval results in unnecessary broadcasts which decrease the energy efficiency of the strategy. Hence the experiments are carried out with beacon interval 5s and the speed of the anchor as 30m/s.

5.1.3 Impact of Number of Fuzzy Bins on Accuracy

The number of fuzzy bins in the fuzzy inference system is an important parameter which directly impacts the localization accuracy of the sensor nodes. When there are

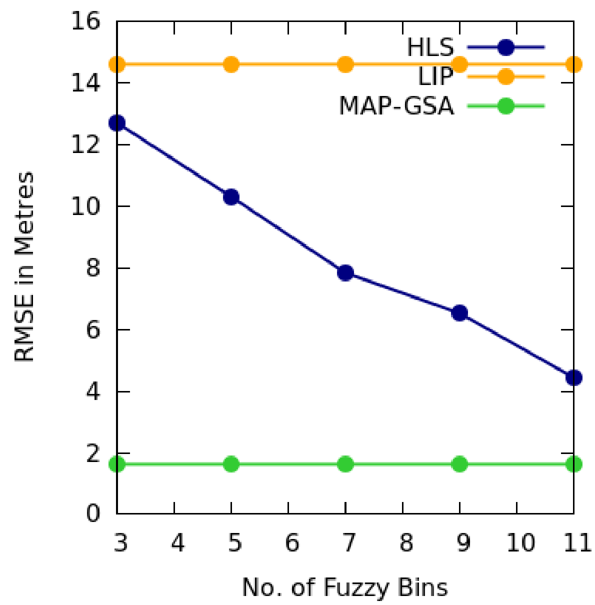

Fig. 7: Impact of Fuzzy Bins on Accuracy

more fuzzy bins, each RSS measurement triggers more fuzzy rules and hence a more accurate range estimate is obtained. Fig. 7 shows the impact of bins on the localization accuracy. The other methods are unaffected by the change in the number of bins.

5.1.4 Impact of Node Density

Node Density affects the performance of LIP but not MAP-GSA and HLS. Unlike LIP, there is no propagation of localization error in MAP-GSA, and HLS. This can be seen from Fig. 8 where node density is plotted against localization error.

5.1.5 Impact of Number of Anchors

The total number of anchors in a WSN has a direct influence on the localization time and accuracy. From Fig. 9 it is observed that HLS and LIP performs well in the presence of many anchors.

But more anchors also imply an increase in communication which affects the energy of a node. So in HLS 3 mobile anchors have been used throughout. Similarly the authors have fixed 5% beacons in LIP. Also it is observed that MAP-GSA is not affected by the number of anchors as it is affected by the beacon interval. This is because though a node benefits by beacons from different positions from many anchors, a node cannot afford to wait indefinitely for the possibility of beacons from new positions. As soon as it starts to receive the beacons, it

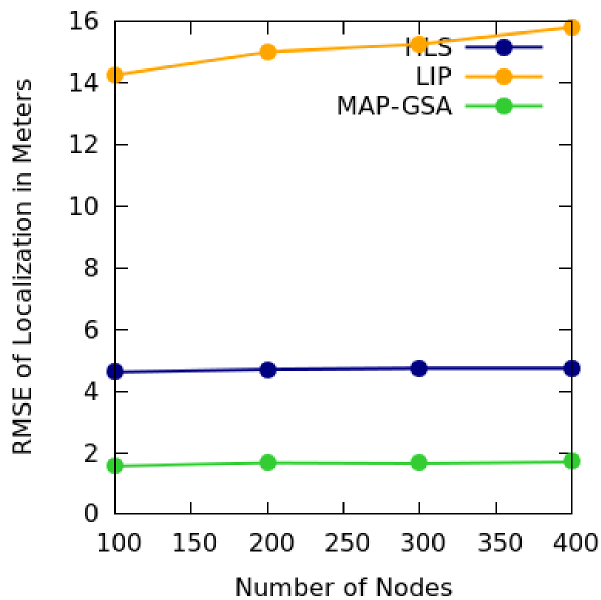

Fig. 8: Impact of Node Density on Accuracy

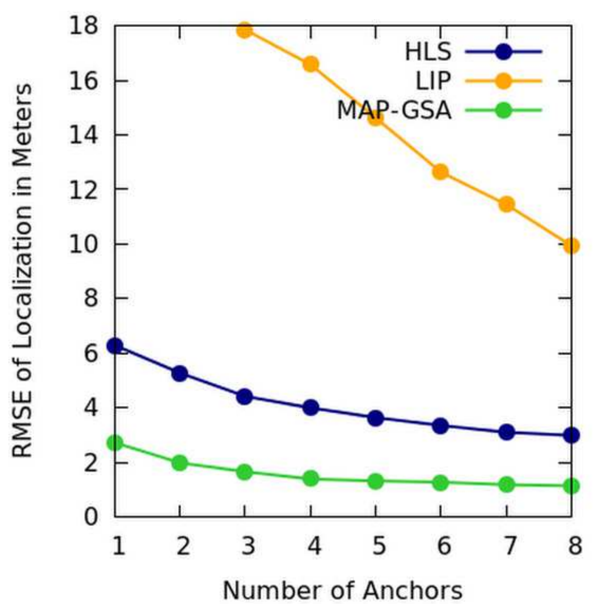

Fig. 9: Impact of Anchors on Accuracy

waits only for a limited amount of time before it begins its localization.

5.2 Study on Energy Efficiency

To evaluate the performance of the HLS algorithm in terms of its energy efficiency, four different scenarios are set up and studied.

5.2.1 Scenario 1 – MAP-GSA

The sensor nodes are randomly deployed in the sensing field. Mobile anchors move around the network passing beacons. In this scenario, there is no hierarchy of nodes based on remaining energy level. All the sensor nodes listen to the medium for messages from anchor. Whenever an anchor comes in its range and broadcasts a beacon, the sensor node stores the message.

As soon as the required number of messages is obtained, localization begins. Here the localization strategy used for all nodes is MAP-GSA. Average energy needed and the lifetime of the WSN for Scenario 1 is shown in Table 5.

Table 5 Avg. Energy and Network Lifetime of Scenario 1

Avg. Energy for one round of localization of high energy Nodes	0.124 Joules
Average Energy for one round of localization of low energy nodes	0.124 Joules
First Node Death	336 s
Half Node Death	3702 s

5.2.2 Scenario 2 – HLS (without Duty Cycling)

The sensor nodes are deployed in the sensing field and mobile anchor moves around the network passing beacons. A formal cluster of nodes do not exist. The sensor nodes are aware of their remaining energy and if the remaining energy is greater than a pre-determined threshold MAP-GSA is used, else FIS and GSA are used for positioning. Average energy needed and the lifetime of the WSN for Scenario 2 is shown in Table 6.

Table 6 Avg. Energy and Network Lifetime of Scenario 2

Average Energy for one round of localization of higher energy nodes	0.259 Joules
Average Energy for one round of localization of low energy nodes	0.0161 Joules
First Node Death	108 s
Half Node Death	2568 s

5.2.3 Scenario 3 – HLS

The sensor nodes are deployed and mobile anchors passes beacons along their path in the network. Hierarchy of nodes exists and clear distinction between master nodes and listener nodes is present. Formal clustering of nodes happens in the network. The elected clusterheads prepare a TDMA schedule for its cluster members to follow and broadcast it [22]. Here, the cluster head and master nodes perform MAP-GSA while listener nodes perform FUZZY estimation and GSA for localization. Average energy needed and the lifetime of the WSN for Scenario 3 is shown in Table 7.

Table 7 Avg. Energy and Network Lifetime of Scenario 3

Average Energy for one round of localization of higher energy nodes	0.023 Joules
Average Energy for one round of localization of low energy nodes	0.0016 Joules
First Node Death	2358 s
Half Node Death	10382 s

5.2.4 Scenario 4- LIP

The Sensor nodes use a localization strategy called the LIP [11]. Similar to HLS, LIP also works in two phases. In the first phase all the sensor nodes invariable of its residual energy receive beacons from static anchors. With the beacons they roughly estimate their position. In the second phase, they refine their estimated location with the help of the neighbor's location and range which is calculated by RSSI. LIP achieves energy efficiency by limiting the flooding of beacon messages. Unlike DV-Hop, only a fraction of messages are flooded. Moreover in the refinement phase, nodes are in low-duty cycle state till neighbors are ready with valid position estimates. Average energy needed and the lifetime of the WSN for Scenario 4 is shown in Table 8.

Table 8 Avg. Energy and Network Lifetime of Scenario 4

Average Energy for one round of localization of higher energy nodes	0.082 Joules
Average Energy for one round of localization of medium energy nodes	0.082 Joules
First Node Death	1610 s
Half Node Death	7740 s

5.2.5 Comparison of Average Energy of Localization of High Energy Nodes

The primary goal of HLS is to make the nodes spend energy proportionately to increase network lifetime. The plot (Fig. 10) showing the average energy consumption of master nodes shows that highest energy consumption happens in Scenario 2 than in Scenario 1.

This is because the master nodes in Scenario 2 in addition to localization also continuously learns about the RF environment and helps the listener nodes. In case of scenario 3 and scenario 4, this energy consumption drastically comes down due to reduced number of help, beacon and hello messages, and duty cycling.

Master nodes in HLS have increased responsibility than in LIP as the listener nodes are completely dependent on them and it also learns the RF environment. This should prove as a setback in terms of energy optimization but

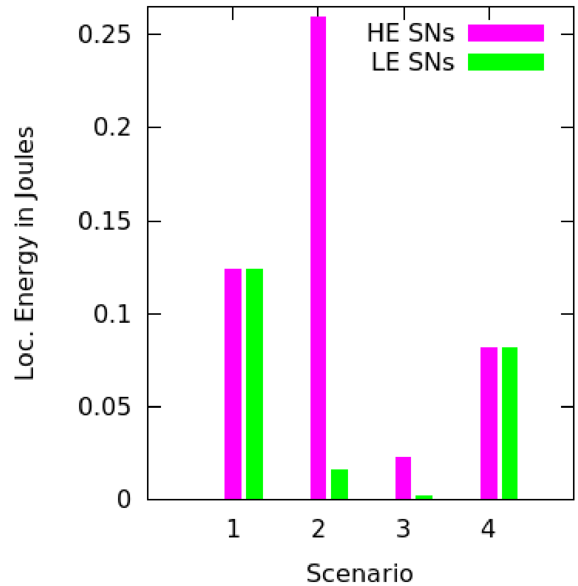

Fig. 10: Mean Loc. Energy for High and Low Energy Nodes

it still can be seen that HLS has an edge over LIP. HLS performs better as there are NIL broadcasts unlike LIP.

Beacon messages are one way from anchor to sensor nodes and their TTL is 1 whereas TTL of a beacon in LIP is fixed randomly and chances of receiving the same beacon again by nodes is high in LIP. Moreover in HLS, no added communication is needed in learning the RF environment as the same beacon messages are used for both learning and localization.

5.2.6 Average Energy of Localization in Low Energy Nodes

Scenario 2 shows an improved performance in case of listener nodes unlike Scenario 1 (Fig. 10). This is because of the drastic reduction in the number of messages needed by the low energy nodes for localization. In case of scenario 1, the lower and higher energy nodes require approximately equal number of messages. But in Scenario 2, the burden of localization falls on the master node and the listener node just needs only a minimum of three messages for localization.

In case of listener nodes, it is intuitive that HLS provides better optimization than LIP because of the drastic reduction in the number of beacons needed.

5.2.7 Mean Residual Energy and Mean Standard Deviation of Residual Energy across Rounds

Though residual energy is important for a sensor Node's lifetime, standard deviation of residual energy of the

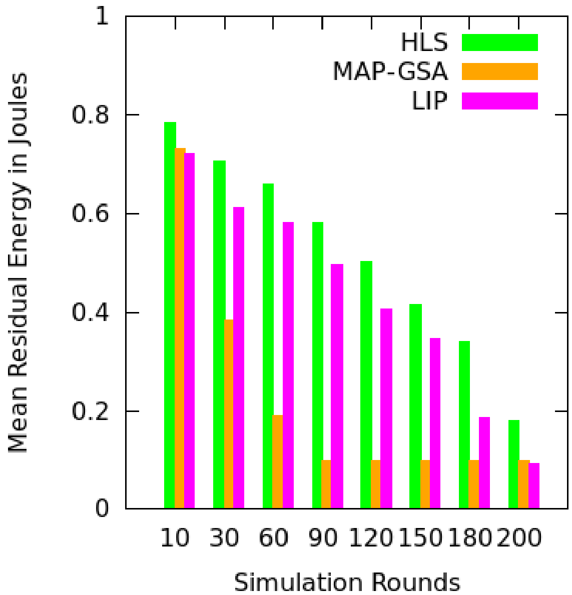

Fig. 11: Average residual energy across rounds

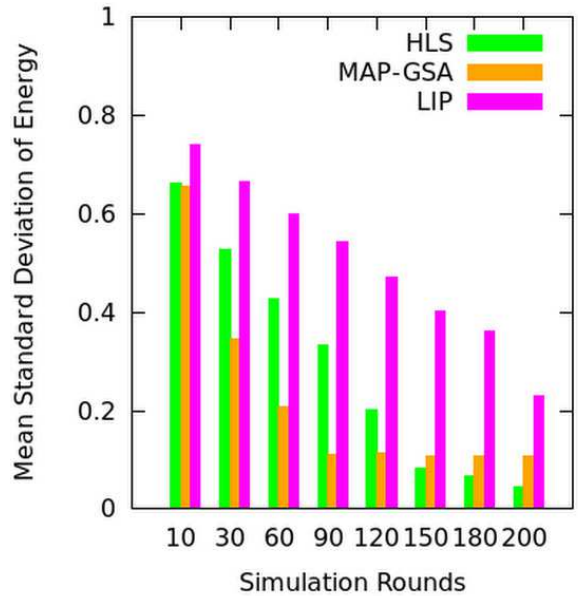

Fig. 12: Mean Std. Dev. Of Residual Energy

nodes plays a still more important role in determining the network lifetime (Fig.11 and Fig.12).

If the energy gap among the high and low energy nodes increases or remains the same, the low energy nodes runs out of energy quicker than the high energy nodes breaking the network connectivity. This WSN becomes futile even when some nodes still have battery power left. Hence decreasing the MSD of the nodes' residual energy has been the motivation of this work.

The energy difference among the nodes remains the same throughout in LIP as LIP provides the same optimization for all the nodes in the WSN. However, HLS caters the needs of the listener nodes whose participation is very less when compared to the master nodes.

5.2.8 Network Lifetime

Network lifetime is often measured by First Node Death (FND) and Half Node Death (HND). FND is the time at which the first node runs out of energy in a WSN. HND is the time at which half of the nodes in a WSN run out of energy. Fig. 13 shows a comparative study of FND and HND respectively for all the four scenarios. LIP fares well when compared to MAP-GSA but FND as well as HND are high in case of HLS when compared to all the other scenarios.

Proportional consumption of energy, zero broadcasts, nil collisions, and re-transmissions during localization have provided HLS an edge over LIP in terms of energy efficiency. HLS has improved the network lifetime by 46% in case of FND and by 34 % when HND is considered.

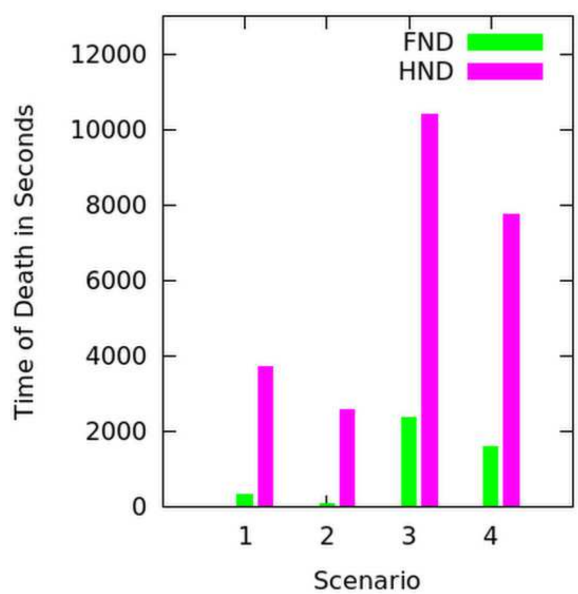

Fig. 13: Plot of FND and HND

6 Conclusion

Simulation results prove that fuzzy-based HLS can be concluded as a highly energy efficient localization technique. This technique does not involve complex algorithms or special hardware devices like directional antennas for reducing the energy needed. Unlike other localization algorithms, the proposed work also does not assume that nodes are active throughout the process of localization. The work uses the existing network structure like clusters and sleep/wakeup duty schedule and performs the localization.

Different strategies for nodes with different energy levels also go easy on the battery life. The proposed work completely shifts the burden of communication on the master nodes. The listener nodes are thin clients which need a very few number of messages (three messages) for localization. All these advantages have been effective in decreasing the number of messages needed by listener nodes to 3. HLS also improves the network lifetime by 46% when compared to LIP.

The RMSE of MAP_GSA is 1.62. The RMSE of LIP is 14.6 m and HLS is 4.73. Though HLS performs better than LIP, it does not achieve the RMSE of MAP-GSA. But this small difference in error can be neglected if higher priority parameters (like network lifetime) are to be maintained.

References

[1] B.M. Sadler, Fundamentals of energy-constrained sensor network systems, IEEE A&E Systems Magazine, Vol. 20, No. 8, pp. 17-35 (2015).

[2] G. Han, H. Xu, T. Q. Duong, J. Jiang and T. Hara, Localization algorithms of wireless sensor networks: a survey, Telecommunication Systems, Vol. 52, No. 4, pp. 2419-2436 (2013).

[3] U. Nazir, N. Shahid, M. A. Arshad and S. H. Raza, Classification of localization algorithms for wireless sensor network: A survey, Proceedings of the IEEE International Conference on Open source systems and technologies (ICOSST), pp. 1-5 (2012).

[4] S. Sivakumar, R. Venkatesan and M. Karthiga, Evolutionary approaches for minimizing error in localization of wireless sensor networks, International Journal of Sensor networks, Vol. 17, No. 1, pp. 17-26 (2015).

[5] A. Kaur, P. Kumar, and G. P. Gupta, A weighted centroid localization algorithm for randomly deployed wireless sensor networks, Journal of King Saud University-Computer and Information Sciences (2017).

[6] B. Zhang and F. Yu, Lox-complex energy-efficient localization algorithm for wireless sensor networks using directional antenna, IET Communications, Vol. 4, No. 13, pp. 1617-1623 (2010).

[7] M. Taheri and S. A. Motamedi, Transceiver optimization for ToA-based localization of mobile WSN, Journal of Circuits, Systems and Computers, Vol. 25, No. 09 (2016).

[8] S. Lee, B. Koo and S. Kim, RAPS: Reliable anchor pair selection for range-free localization in anisotropic networks, IEEE Communications Letters, Vol. 18, No. 8, pp. 1403-1406 (2014).

[9] M. Farooq-I-Azam, Q. Ni and E. A. Ansari, Intelligent energy efficient localization using variable range beacons in industrial wireless sensor networks, IEEE Transactions on Industrial Informatics, Vol. 12, No. 6, pp. 2206-2216 (2016).

[10] G. Sharma and A. Kumar. Modified Energy-Efficient Range-Free Localization Using Teaching–Learning-Based Optimization for Wireless Sensor Networks, IETE Journal of Research, pp. 1-15 (2017).

[11] A. Martirosyan and A. Boukerche, LIP: an efficient lightweight iterative positioning algorithm for wireless sensor networks, Wireless Networks, Vol. 22, No. 3, pp. 825-838 (2016).

[12] D. F. Larios, J. Barbancho, F. J. Molina and C. León, LIS: Localization based on an intelligent distributed fuzzy system applied to a WSN, Ad Hoc Networks, Vol. 10, No. 3, pp. 604-622 (2012).

[13] A. Kumar and V. Kumar, Fuzzy logic based improved range. free localization for wireless sensor networks, Int. Sci. Index, Electr. Comput. Eng, Vol. 7, No. 5, pp. 497-505 (2013).

[14] T. Garcia-Valverde, A. Garcia-Sola, H. Hagras, J. A. Dooley, V. Callaghan and J. A. Botia, A fuzzy logic-based system for indoor localization using WiFi in ambient intelligent environments, IEEE Transactions on Fuzzy Systems, Vol. 21, No. 4, pp. 702-718 (2013).

[15] B. R. El Samadony, R. A. Ramadan and N. M. Darwish, Enhanced trilateration localization using fuzzy logic for WSN, Int. J. Emerg. Technol. Adv. Eng, Vol. 5, No. 6, pp. 378-385 (2015).

[16] A. S. Velimirovic, G. L. Djordjevic, M. M. Velimirovic and M.D. Jovanovic, Fuzzy ring-overlapping range-free (FRORF) localization method for wireless sensor networks, Computer Communications, Vol. 35, No. 13, pp. 1590-1600 (2012).

[17] H. Chenji and R. Stolero, Toward accurate mobile sensor network localization in noisy environments, IEEE Transactions on Mobile Computing, Vol. 12, No. 6, pp. 1094-1106 (2013).

[18] Ian F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, A survey on sensor networks, IEEE Communications Magazine, Vol. 40, No. 8, pp. 102-114 (2002).

[19] N. Kaur, V. Walia and R. Malhotra, Activation matrix oriented base station implementation for energy optimization in wireless sensor networks, International Journal of Computing and Corporate Research, Vol. 4, No. 4 (2014).

[20] H. Karl and A. Willig, Protocols and Architectures for Wireless Sensor Networks, Chichester, U.K: Wiley (2005).

[21] A. Thakkar and K. Kotecha, Cluster head election for energy and delay constraint applications of wireless sensor network, IEEE Sensors Journal, Vol. 14, No. 8, pp. 2658 -2664 (2014).

[22] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, Proceedings of the IEEE Hawaii International Conference on System Sciences, Maui, Hawaii, USA, pp. 1-10 (2000).

[23] M. N. S. Swamy and K. L. Du, Search and Optimization by Metaheuristics, Cham, Switzerland: Birkhauser (2016).

[24] J. S. R. Jang, C. T. Sun and E. Mizutani, Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence, Englewood Cliffs, NJ: Prentice-Hall (1997).

M. Karthiga completed her B.Tech in Information Technology from Amrita Viswa Vidya Peetham, Coimbatore in 2010. She completed her Master's degree (ME) in Computer Science and Engineering at PSG College of Technology, Coimbatore, India affiliated to Anna University in the year 2012. Currently she is pursuing her Ph.D at Anna University, Chennai, India. Her research interests include wireless sensor networks, cloud computing, networks and evolutionary computing.

R. Venkatesan received his BE (Hons) from Madras University in 1980. He completed his Masters degree in Industrial Engineering from Madras University in 1982. He obtained his second Masters degree MS in Computer and Information Science from University of Michigan, USA in 1999. He was awarded with PhD from Anna University, Chennai in 2007. He is currently Professor and Head in the Department of Computer Science and Engineering at PSG College of Technology, Coimbatore, India. His research interests are in simulation and modelling, software engineering, algorithm design, software process management