International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I4P221
Volume 70 Issue 4, 245-257, April 2022

Original Article

Aquila Optimized Localization of Mobile nodes in Heterogeneous WSN with Reduced Complexity using MCL square

Swapna M P[1], G. Satyavathy[2]

[1,2]*Department of Computer Science, Sri Ramakrishna College of Arts & Science for Women, Coimbatore, India.*

[1]mpswapna77@gmail.com

Received: 23 February 2022 Revised: 05 April 2022 Accepted: 19 April 2022 Published: 26 April 2022

Abstract *- Localization of mobile nodes in Heterogeneous Wireless Sensor Network (HSWN) requires more research and experiments. The majority of the localization protocols discuss locating the static nodes in the wireless network. This paper proposes the localization of mobile nodes in an HWSN, considering energy efficiency. The protocol Aquila Optimized Monte Carlo Localization(AOMCL) is a novel attempt to combine the mobile node localizing algorithm MCL and the new swarm intelligence algorithm, Aquila Optimizer. The protocol AOMCL reduces the sampling and filtering process of traditional MCL. AOMCL localizes the unknown nodes by generating an MCL square around the location-aware anchor nodes. The method efficiently reduces the time and complexity of localizing the unknown nodes. The experimental analysis of AOMCL in the Matlab simulator illustrates that the proposed protocol, AOMCL, has high localization accuracy, better localization coverage, and reduced complexity compared with the existing protocols, DEMCL, RMCL, and QMCL.*

Keywords – *Filtering, HWSN, Localization, Location Prediction, MCL Square, Sampling.*

1. Introduction

Wireless Sensor Network (WSN) is a collection of sensor nodes deployed in ad-hoc environments to monitor and gather data from the environment. These sensor nodes are either static or mobile, and they transmit the collected data to sink nodes for further processing at the destination [1, 2]. With the advancement in the Internet of Things (IoTs) and smart applications like healthcare, homes, cities, etc., connecting physical things with the digital world [3,4] has become mandatory. WSN plays an inevitable role in such applications.

The deployment of sensor nodes should be efficient to optimize the quality of the network in all aspects like network coverage, lifetime, and connectivity[5,6]. Deployment of sensor nodes follows either a deterministic scheme or a random scheme. In the previous method, the location of each sensor node is predetermined, but for large-scale networks having thousands of nodes [7,8], this method is infeasible. The drawback of such deployment is that it does not ensure complete coverage and connectivity.

Once deployed, the sensor nodes start monitoring and collecting information from the environment. The collected data becomes useful when the sensor's location information and the timestamp are wrapped. Location information is important to retrieve the location identity of the observed events, identify and locate target objects, assess the quality of coverage, facilitate geographic routing algorithms for position-aware data processing, etc. Determining the exact location or position of sensor nodes in WSN is called the localization problem.

1.1 Motivation

Global Positioning System(GPS) is a known technique to locate a gadget, but it is not practical in WSN, as providing GPS on each sensor node (that is energy constrained) is not feasible. Studies reveal that GPS consumes high power and reduces accuracy in indoor and urban areas [9-15]. Owing to this, many localization algorithms were proposed by researchers, and the localization problem is a trending research area.

1.2 Objective

Localization methods are broadly classified into two range-based and range free. Out of the two, range-free algorithms are less expensive and highly preferred than range-based methods, as additional hardware requirement is nullified and complexity is reduced in range-free methods. The popular range-free localization protocols are the centroid positioning algorithm[16], DV-HOP positioning algorithm[17], amorphous algorithm[18], etc. Though these algorithms have proved higher localization accuracy, network robustness, and efficient energy consumption, they are all suitable for static sensor nodes. Mobility of the sensor nodes in WSN is not regarded in most of the proposed localization algorithms, which is a mandate for current WSNs such as IoT, underground mining, etc.

1.3 Problem Statement

Mobile node localization of WSN was earlier performed by the periodic localization and calculation of static nodes, resulting in high communication costs and tedious calculations[19-21]. Monte Carlo node Localization(MCL) algorithm with its particle filtering scheme is a perfect solution for localizing mobile nodes. Still, it has certain limitations like inaccurate positioning and location coverage, more number of anchor nodes, less efficiency, etc. This paper proposes a novel model for determining the location of mobile nodes in WSN with reduced effort and time. The proposed algorithm AOMCL is an innovative scheme for localizing the mobile nodes in a Heterogeneous WSN, coalescing MCL with Aquila optimizer. AOMCL has the following characteristics

1. Identifying suitable Anchor nodes based on residual energy of nodes
2. Location estimation of Mobile nodes by generating an MCL square reduces the localization error as the search area is minimized.
3. Optimized the location information using a bio-inspired Aquila optimizer.

The remaining sections of the paper are Section 2 details the literature review and related works. The preliminaries and basic concepts of the proposed work are specified in Section 3. Section 4 eloborates the phases and framework of the proposed AOMCL algorithm. Section 5 analyzes the experimented results and has detailed discussions.

2. Related Works

Range-free localization algorithms are popular and widely used as they don't require specific hardware for the location estimation of nodes[22] and use network connectivity for localization. Many researchers proposed the effect of heterogeneity of nodes on range-free localization methods. In “Low-Cost optimization for HWSN”[23], the localization protocol for heterogeneous WSN with varied transmission capabilities is proposed. The initial estimation of the locations is through the analysis of hop progress, a correction mechanism that minimizes localization error. “Effective range-free localization with elliptical distance correction in HWSN” [24] is a range-free elliptical distance correction method for accuracy in distance estimation. “Node localization based on multiple transmission power levels”[25] is an algorithm for localizing nodes in HWSN with multi-scale virtual factors. High localization accuracy is achieved using multiple transmission powers at different levels. Mobility of nodes was not considered in the above-proposed methods.

Two positioning methods are used to localize mobile nodes in WSN; one approach is to use a static positioning scheme in a dynamic network which is done periodically to assess the location of nodes. The method includes high communication costs and tedious calculations and negatively impacts localization accuracy if the maximum speed at which a node’s movement rises. The second method is the Monte Carlo Localization for mobile nodes proposed by Hu and Evans [26]. The Monte Carlo Box (MCB) algorithm was proposed by Baggio and Langendoen [27], which improved sampling efficiency and energy consumption using anchor box and sample box concepts. “Mobile node localization based on fuzzy theory"[28] has accurate filtering conditions that overcome the drawback of the traditional MCL with reduced localization time. “Adaptive MC method for dynamic sensor nodes” [29] is an improved protocol with the dead reckoning method and ensured high positioning accuracy. But the method requires additional hardware. “Time sequence-based MCL algorithm"[30] represents the sampling area formed using feedback signals from nearby one-hop anchor nodes. Here the positioning task becomes inaccurate when the anchor nodes are scanty. In [31], MCL is combined with the differential evolution algorithm for positioning accuracy. But the scheme consumes more energy due to the hardware equipment used for measuring the distance.

Meta-heuristic optimization algorithms inspired by nature that mimic biological or physical phenomena are the popular ones used in real-world applications [32]. These are straightforward and flexible algorithms and are used to solve many tricky and complex optimization problems. Swarm intelligence algorithms are extensively used in optimization protocols based on the behavior of swarms of creatures, the most popular of them is the “Particle Swarm Optimization”(PSO)[33]. Other algorithms include “Ant Colony Optimization Algorithm (ACO)” [34], “Bat Algorithm (BA)” [35], “Grey Wolf Optimizer (GWO)” [36], “Cuckoo Search (CS) Algorithm” [37], “Whale Optimization Algorithm (WOA)” [38], “Harris Hawks Optimizer (HHO)” [39], etc. Aquila Optimizer (AO) [40] is a new algorithm in the series that simulates Aquila's hunting tricks and methods for its prey. Aquila uses different methods for different types of prey. For rapid-moving prey, it uses global exploration, while for slow-moving prey, it exhibits local exploitation features. Optimization has stepped into networking to find the efficient route between source and destination. Bio-Optimization based routing protocols give the best solution to routing issues [41-45]

3. Preliminaries of the proposed work

3.1 Mobile node localization by Monte Carlo Scheme

The Monte Carlo approach was frequently used in robotics [46,47], with the concept that a robot does localization depending on its movement and the perception of its environment. This theory was extended by Hu and Evans[48] to localize sensors in hostile environments and irregular terrain. Localization algorithms based on the mobility of sensor nodes are a real-time model and a prime requisite for WSN. Monte Carlo Localization(MCL) guarantees localization on mobile sensor nodes and ensures accuracy in estimating positions.

MCL algorithm for mobile nodes is as follows.

- It is assumed that time is segregated into discrete intervals. At every time interval, a node relocalizes (being a mobile node)
- A set of N random samples, $L_0 = \{l_0^0, l_0^1, \ldots\ldots, l_0^{N-1}\}$, is selected by the sensor node from the deployment area(L_t).
- The prediction and filtering process is carried out on the samples.

3.1.1 Prediction

- At time t, a new collection of samples are produced by sensor nodes based on the preceding set, L_{t_1}.
- From a known location l_{it_1} of L_{t_1}, arbitrary location l_{it} is estimated in a circular region with the radius v_{max}. Considering the Euclidean distance $d(l_1,l_2)$ between any two points l_1 & l_2 and the even distribution of velocity between(0, v_{max}), the current probable position estimation would be distributed as

$$p(l_t, l_{t-1}) = \begin{cases} \frac{1}{\pi v_{max}^2}, & d(l_t,l_{t-1}) < \text{vmax} \\ 0, & d(l_t,l_{t-1}) \geq \text{vmax} \end{cases} \tag{1}$$

3.1.2 Filtering

- All the irrelevant and unfeasible locations are eliminated from the new collection of samples.
- Removal is based on the position information obtained from a group(X) of anchors that are believed to be within the sensor node's radio range (r). And from the group(M) of anchors who are neighbors of the first group (lying outside the radio range), the filtering formula is as follows

$$f = \forall x \in X, d(f,x) \leq r \Delta \forall s \in M, d(f,x) \leq 2r \tag{2}$$

- The prediction and filtering process is repeated until the desired number of samples has been obtained, until the effective sample list (N_e) is higher than the threshold list((N_t).
- The mean of all potential locations from the sample set *Lt* predicts a sensor's location at time t.

3.2 Aquila Optimizer

The novel meta-heuristic swarm intelligence, the Aquila algorithm, is based on the hunting ability of Aquila. The bird exhibits the hunting nature of humans and follows different strategies for hunting different prey. It flexibly switches among the strategies and hunts with speedy action, powerful feet, and claws. The Aquila optimization procedure is a population-based method. The optimization initiates with the population of identified solutions as candidates(X) ranging between the boundaries UB and LB. The optimal solution is determined from the best solutions in each iteration, using the formula in equation 3.

$$X_{ij} = rand \times (UB_j - LB_j) + LB_j \quad i = 1,2,\ldots,N \; j = 1,2,\ldots Dim \tag{3}$$

where N indicates the total population(candidate solutions) and Dim is the volume of the problem. The mathematical model of Aquila Optimizer has the following steps

3.2.1 Expanded Exploration

The Aquila identifies the presence of prey in a specific area with its ability to fly high and vertical stoop. It explores the search space extensively. The wide analysis and exploration give a clear picture of the prey resulting in a vertical dive. This behavior of expanded exploration can be represented as

$$X_1(t+1) = X_{best}(t) \times \left(1 - \frac{t}{T}\right) + (X_M(\text{t}) - X_{best}(t) \times r) \tag{4}$$

Where $X_1(t+1)$ *is the solution generated by the first search,* $X_{best}(t)$, represents the best position identified, the approximate position of prey. X_M(t) is the average position of all the locations in the present iteration,

$$X_M(t) = \frac{1}{N}\sum_{i=1}^{N} X_i\,(t) \tag{5}$$

the current iteration is t, while the maximum number of iterations is T, N represents the population size, and r is a random value between 0 and 1

3.2.2 Narrowed Exploration

Once Aquila identifies the prey, it circles the prey to focus it and then attacks. This is represented as the contour flight attack, where the explored space of the previous step is narrowed down for an attack. This is mathematically represented as

$$X_2(t+1) = X_{best}(t) \times LV(D) + X_R(t) + (y-x) * r \tag{6}$$

where D specifies the dimension space,

$$LV(D) = s \times \frac{i \times \sigma}{|j|^{\frac{1}{\varphi}}} \tag{7}$$

is the flight distribution function with s being a constant value(0.01), i & j are random numbers ranging from 0 to 1, φ is a constant value(0.5) and

$$\sigma = \left(\frac{\Gamma(1+\varphi) \times sin(\frac{\pi\varphi}{2})}{\Gamma(\frac{1+\varphi}{2}) \times \varphi \times 2^{(\frac{\varphi-1}{2})}} \right) \tag{8}$$

y and x are spiral searches given as

$$y = r \times \cos(\theta) \tag{9}$$

$$x = r \times \sin(\theta) \tag{10}$$

$$r = rc \times 0.00565 \times D_1 \tag{11}$$

$$\theta = -0.005 \times D_1 \times \frac{3 \times \pi}{2} \tag{12}$$

rc is the number of search cycles ranging from 1 to 20, $D_1$is the number from 1 to D(dimension size)

3.2.3 Expanded Exploitation

Upon identifying the prey accurately, Aquila makes the initial attack on the prey by descending vertically and observing the prey's reaction. It’s a slow descent attack. This is represented as

$$X_3(t+1) = \left(X_{best}(t) - X_M(t)\right) \times \propto -r + (U-L) \times r + L) \times \delta) \tag{13}$$

Where $\propto$ and δ are the parameters to adjust exploitation with a value (0.1), U is the higher limit, L is the lower limit of the problem, respectively, and r is the random value as utilized in the previous steps. $X_{best}(t)$ & $X_M(t)$ indicates the best position and the value of positions, respectively.

3.2.4 Narrowed Exploitation

Aquila gets closer to the prey, attacks from land, and grabs the prey. This is represented as the optimized location of the identified prey. This behavior is shown as

$$X_4(t+1) = \left(Q_f \times X_{best}(t) - (M_1 \times X(t) \times r\right) - M_2 \times LV(D) + r \times M_1) \tag{14}$$

where Q_f is the quality function to stabilize search strategies calculated as

$$Q_f(t) = t^{\frac{M_1}{M_2}} \tag{15}$$

M1 represents various motions of Aquila during the tracking sequence of its prey. M2 is the decreasing value from 2 to 0, representing the flight slope from step1 (first location) to step 4(last location-t)

$$M_1 = 2 \times rand(\,) - 1 \tag{16}$$

$$M_2 = (1-T)^2 \tag{17}$$

t & T is the current iteration and the maximum iteration respectively, r is the random value (0 to 1).

Algorithm1 – Aquila Optimizer

Initialize the population(P) and parameters of AO
While(stop condition is not met) do
Compute the fitness values of P_i
Determine the best solution $P_{best}(t)$
For (i=1,2,3,4,5…..,N) do
Update $P_M(t)$,x,y,G1,G2,LV(D) [Exploration-I]
If $t \leq \frac{2}{3} * T$ then
Update P_i using equation 4
If Fitness($P_1(t+1)$< Fitness (P(t)) then
P(t)= ($P_1(t+1)$
If Fitness ($P_1(t+1)$)< Fitness($P_{best}(t)$) then
$P_{best}(t)= P_1(t+1)$
End if
End if
Update P_i using equation 6 [Exploration-II]
If Fitness ($P_2(t+1)$< Fitness (P(t)) then
P(t)= ($P_2(t+1)$
If Fitness ($P_2(t+1)$)< Fitness($P_{best}(t)$) then
$P_{best}(t)= P_2(t+1)$
End if
End if
Else
Update P_i using equation 13 [Exploitation-I]
If Fitness ($P_3(t+1)$< Fitness (P(t)) then
P(t)= ($P_3(t+1)$
If Fitness ($P_3(t+1)$)< Fitness($P_{best}(t)$) then
$P_{best}(t)= P_3(t+1)$
End if
End if
Update P_i using equation 14[Exploitation-II]
If Fitness ($P_4(t+1)$< Fitness(P(t)) then
P(t)= ($P_4(t+1)$
If Fitness ($P_4(t+1)$)< Fitness($P_{best}(t)$) then
$P_{best}(t)= P_4(t+1)$
Endif
Endif
Endif
Endfor
Endwhile
Return P_{best}

4. Aquila Optimized Monte Carlo Localization(AOMCL) of Mobile nodes in HWSN

4.1 Heterogeneity of Network Nodes

Heterogeneous Wireless Sensor Network has nodes of different manufacturers with varied battery levels, sensing, and transmission capabilities. In HWSN, the coverage area and transmission range of nodes varies.

4.2 Detailed Phases of AOMCL

Step 1 – Selection of Anchor Node from the Heterogeneous set of Nodes

Nodes in HWSN vary in terms of initial energy and residual energy. A candidate for being an Anchor Node must have a high energy level. Otherwise, it may lead to the death of the node. Selection of the apt node to be the anchor is vital in HWSN to sustain the energy and lifetime of the network. The parameters, Node Degree and Residual Energy of a node are considered for appropriate selection of candidates to become Anchors.

Node Degree (N_d) is the count of one-hop neighbors of a node (N).

$$N_d = \sum N_j \text{, } where\ N_j \in group\ of\ one - hop\ neighbors\ of\ N \tag{18}$$

The residual energy($E_R(N)$) of a node is given as

$$E_R(N) = E_I - (E_T + E_C) \tag{19}$$

It is the difference between the initial energy from the combined energy utilized for transmission and computation.

Step 2 – Location estimation of unknown nodes using MCL Square

Though MCL is the most favored and accurate algorithm for mobile node localization, it has certain setbacks like having a bigger sampling area, low sampling efficiency, and a dead sampling cycle. The sampling and filtering process consumes more energy and time for the energy-constrained WSN nodes. To mitigate the discussed problems of traditional MCL, the proposed algorithm has the following improvements in setting the sampling area.

4.2.1 Sampling area by MCL Square generation

MCL square is an innovative method of reducing the sampling area. Minimizing the sampling area implies a reduced search area, which could reduce localization errors. This method is intended to generate good samples and decrease the filtration process's effort. It creates the deployment area for localizing the node. A square is built for every unknown node (N) based on the nearby anchor nodes (either one-hop or two-hop). The generated MCL square is an overlapped region of all the nearest anchors. N's identification of nearby anchor nodes is by using Received Signal Strength Indication(RSSI). The closest anchor node is determined when the RSSI value is high. The unknown node broadcasts a message to get the RSSI signals from the nearest anchor nodes. As represented in figure 1, the grey portion is the generated sampling area.

The box coordinates are (X_{min}, X_{max}) and (Y_{min}, Y_{max}).

$$X_{min} = max_{i=1}^{n}(x_i - r) \tag{20}$$
$$X_{max} = min_{i=1}^{n}(x_i + r) \tag{21}$$
$$Y_{min} = max_{i=1}^{n}(y_i - r) \tag{22}$$
$$Y_{max} = min_{i=1}^{n}(y_i + r) \tag{23}$$

where (x_i, y_i) are the coordinates of the node(i), n is the totality of nearby anchors, r represents the radio range of one-hop anchors, and 2r replaces this in the case of two-hop anchors. A node draws samples from the generated square region, restricting the generation of more accurate and appropriate samples.

The above-specified square of sampling area is applicable if the sample set is empty (initially), but when there is a sample set available previously, then the coordinates of the square must include the movement of the node with a speed of v_{max}, and it is given as

$$X_{min} = max\ (x_{min}, x_{t-1}^{i} - v_{max}) \tag{24}$$
$$X_{max} = min(x_{max}, x_{t-1}^{i} + v_{max}) \tag{25}$$
$$Y_{min} = max\ (y_{min}, y_{t-1}^{i} - v_{max}) \tag{26}$$
$$Y_{min} = max\ (y_{min}, y_{t-1}^{i} + v_{max}) \tag{27}$$

Fig. 1 Sampling area by MCL square generation

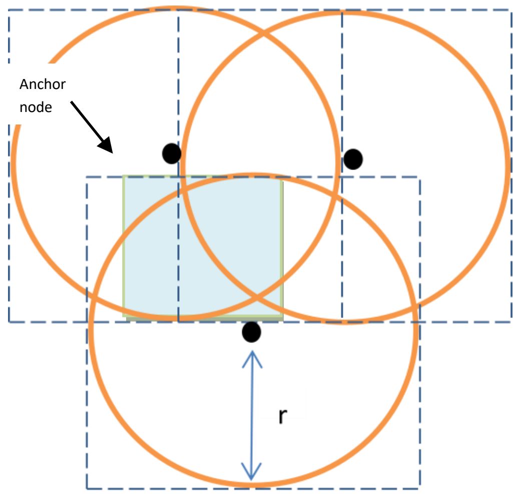

Fig. 1 Sampling area by MCL square generation

4.2.2 Predicting the location

Based on the location information of anchor nodes(i) within the MCL square at time t, and with the assumption that the node is likely to move randomly in any path within the square at speed between 0 and v_{max}, location is predicted as

$$p(l_t \,|l_{t-1}) = 1 \tag{27}$$

$$provided\ X_{min}^i \leq X_t^i \leq X_{max}^i \ \&\ Y_{min}^i \leq Y_t^i \leq Y_{max}^i\ , otherwise\ 0$$

4.2.3 Filtering the samples

Considering r as the radio range, O as the set of anchors that are one-hop, T as two-hop anchors, the filtering process is specified as

$$p(Ob_t|l_t^i) = 1, \tag{28}$$

If $\forall s \in O, d(l_t, s) \leq r \wedge \forall s \in T, d(l_t, s) \leq 2r$, otherwise 0

Where $d(l_t, s)$ is the Euclidean distance between the anchor s and the generated sample l_t, ob_t is the set of observations at the time t. The above three steps are repeated until the maximum count of samples is generated to fill the set.

4.2.4 Location Estimation of the samples is given as

$$\frac{\sum_{i=1}^{N} l_t^i}{N} \tag{29}$$

Step 3 – Optimization of predicted locations by Aquila Optimizer(AO)

The goal of the optimization is to normalize the distance between the estimated position and the anchor node. AOMCL doesn't use the weighted average of the generated samples to localize the unknown nodes; here, the optimal node position is determined using AO within the MCL square. The generated samples of the above phase are AO's initial population(X). The nodes with minimum fitness are retained during the iterative process of AO as specified in Algorithm 1. The fitness function is given by equation 30.

$$\sum_{i=1}^{N} \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} - d_{ik} \tag{30}$$

where d_{ik} denotes the estimated distance between unknown node i and anchor node k, (x_i,y_i) and (x_k,y_k) are the coordinates of the unknown node and anchor node. N represents the total count of anchor nodes within a one-hop distance.

4.3 Phases AOMCL

Figure 2 illustrates the phases of the proposed AOMCL.

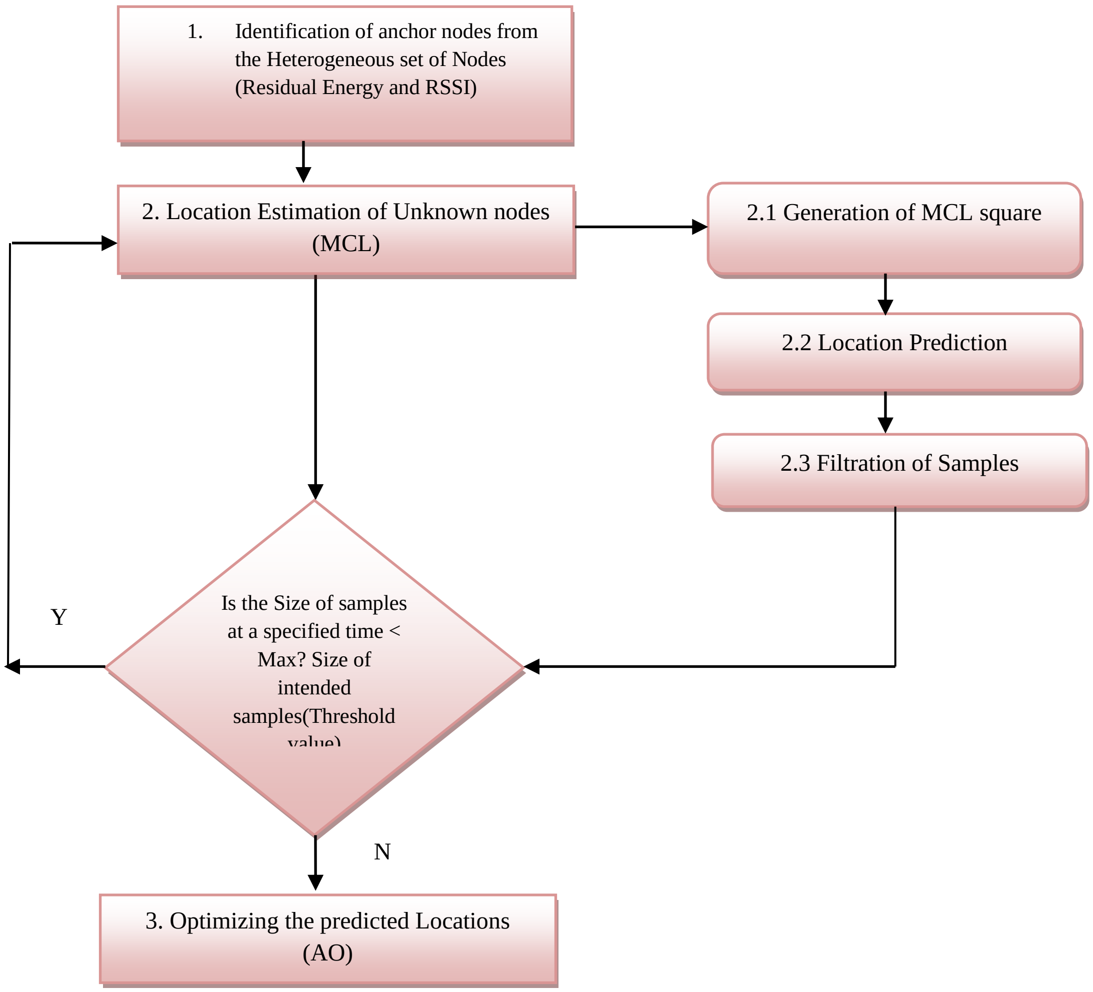

Fig. 2 Flowchart representing the phases of AOMCL

5. Simulation Results & Discussions

The set-up of the simulation environment in Matlab 2015b is as specified in the table. Random WayPoint is the most used model for mobile networks. Each node in the network moves at speeds ranging from 0 and Vmax. A node can reach another node within the radio range 'r .'A heterogeneous network is a group of nodes with a difference in the transmission power of each node. The variation in transmission powers results in varied communication ranges for nodes. The communication range of nodes is within 10 m to 30 m. The simulation experimented with the proposed algorithm with the existing algorithms RMCL, DEMCL, and QMCL, for positioning accuracy/localization error, positioning coverage, anchor node density, and energy consumption.

- ✓ The Localization Error (LE) variation in the node's actual and estimated position.
- ✓ Positioning coverage/Localization coverage is the percentage of nodes whose location is determined.
- ✓ Energy Consumption depends on the algorithm's communication cost and computational complexity.

Table 1. Simulation settings for experimenting with AOMCL

Parameters	Value
Number of Nodes	250-300
Area	500*500m
Number of Anchor nodes	30
Initial Energy of Node	1000J
Starting DTV	0.5
Transmission Range	100m
Model of Mobility	Randomway Point
Speed of Mobility	4 m/s to 45 m/s

5.1 Localization Error based on Anchor node density

The count of anchor nodes attributes to the cost of the mobile WSN. High positioning accuracy must be maintained with low anchor node density. Figure 3 is the effect of Localization Error(LE) with the change in density of anchor nodes on the AOMCL localization algorithm. The localization error of AOMCL is relatively low for all node densities than that of the existing protocols. As anchor nodes in the network increase, the filtered samples are closer to posterior distribution probability, and hence the error in localization reduces. As the number of anchor nodes increases, the distance to the unknown node decreases, resulting in the quick generation of MCL square that produces reduced search area, reduced filtration process, best sampling set, and better localization accuracy. Table 2 gives the numerical representation.

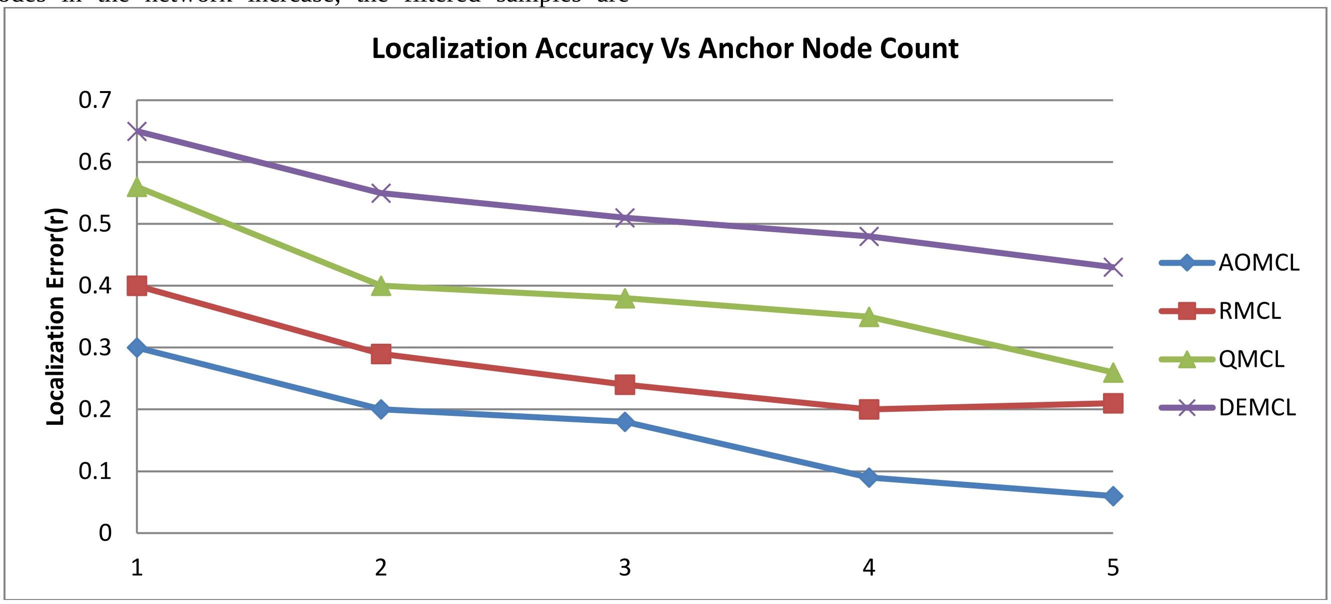

Fig. 3 Localization Error percentage of AOMCL

Table 2. Localization Accuracy of AOMCL

Anchor nodes / Algorithm	1	2	3	4	5
AOMCL	0.3	0.2	0.18	0.09	0.06
RMCL	0.4	0.29	0.24	0.2	0.21
QMCL	0.56	0.4	0.38	0.35	0.26
DEMCL	0.65	0.55	0.51	0.48	0.43

5.2 Localization Error based on moving speed

With the increase in V_{max}, the activity of the node increases per unit time, leading to enhanced communication with the anchor nodes. Thus, the invalid samples can be filtered more effectively, allowing better localization accuracy. One of the major problems of MCL was the unrestricted size of the sampling area that got enlarged as V_{max} is increased, resulting in poor localization accuracy. When V_{max} increases within a specified range, the localization accuracy will increase, but the effect may reverse beyond the range. From figure 4, it is evident that the existing protocols have better localization accuracy than MCL due to reduced sampling area. In AOMCL, the sampling area is further reduced due to MCL square and proves to have low localization error compared to the counterparts. Table 3 represents the numerical representation of the analysis.

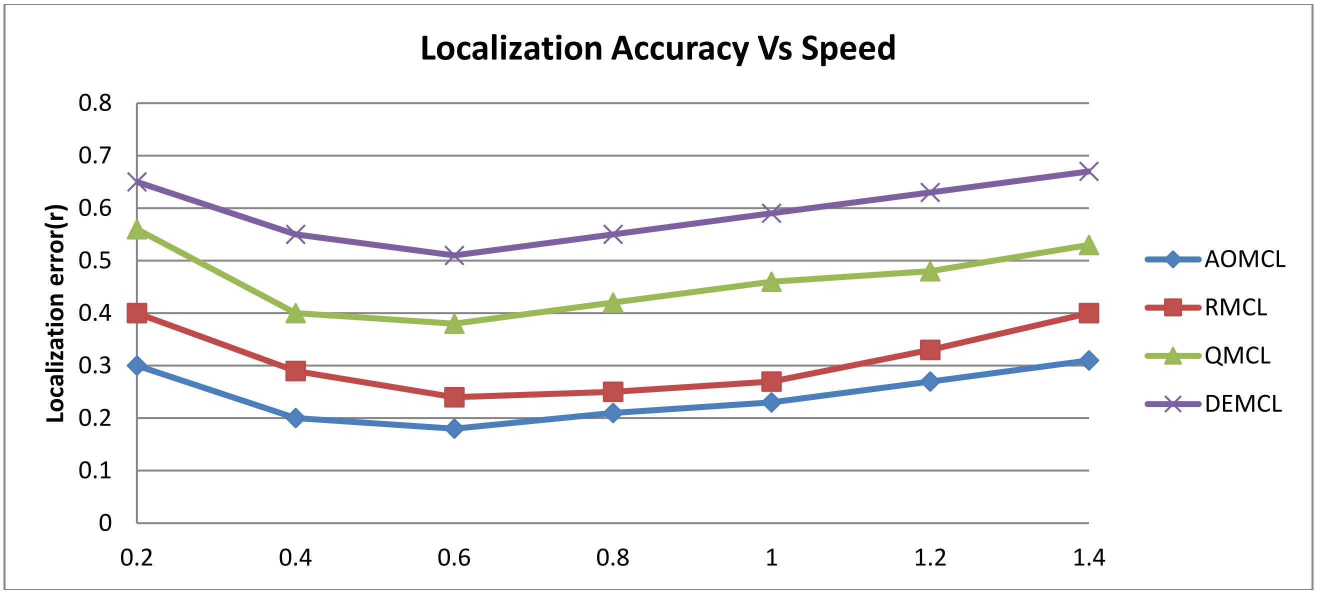

Fig. 4 Location Accuracy against speed(V_{max})

Table 3. Localization Accuracy of AOMCL with respect to Speed

Vmax (r distance per time) Protocol	0.2	0.4	0.6	0.8	1	1.2	1.4
AOMCL	0.3	0.2	0.18	0.21	0.23	0.27	0.31
RMCL	0.4	0.29	0.24	0.25	0.27	0.33	0.4
QMCL	0.56	0.4	0.38	0.42	0.46	0.48	0.53
DEMCL	0.65	0.55	0.51	0.55	0.59	0.63	0.67

5.3 Location Coverage Rate

The ratio of nodes successfully located to all unknown nodes in a given time period defines the location coverage. As shown in Figure 5, the AOMCL algorithm exhibits the highest location coverage rate, which is approximately 92~97% compared to the existing DEMCL, RMCL, and QMCL algorithms. The generation of MCL square based on anchor nodes around the unknown nodes reduces the search and sampling area, resulting in high localization coverage.

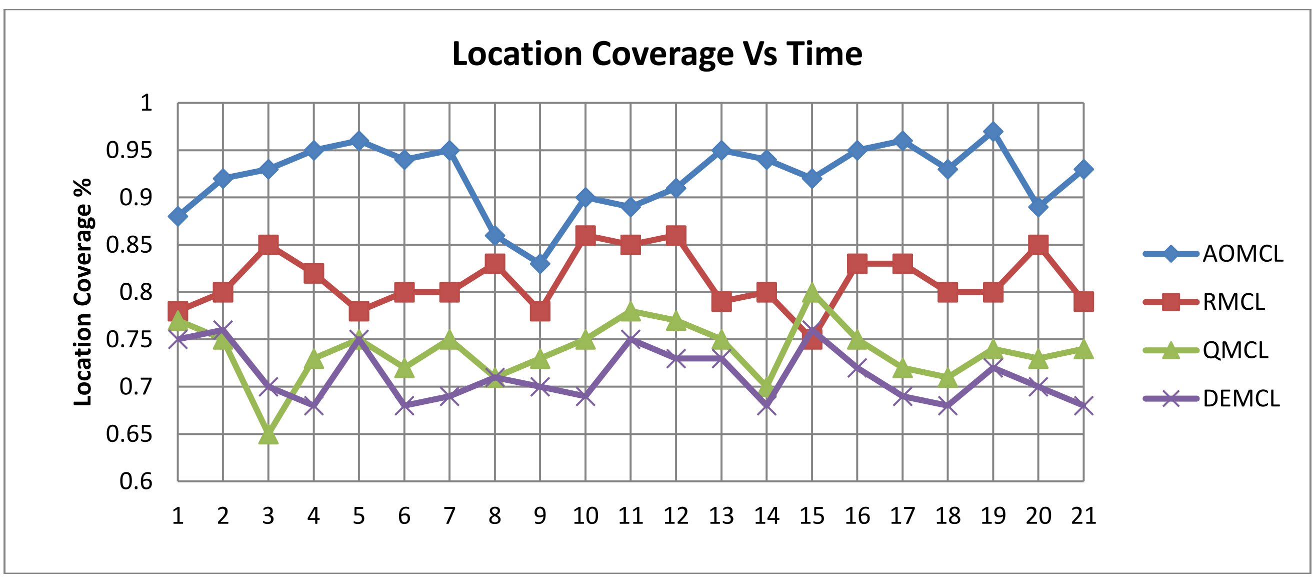

Fig. 5 Location Coverage percentage of AOMCL

5.3 Computational Complexity

The energy utilization and time consumption of the protocols are computational complexity measures. Figure 6 demonstrates the computational time of different algorithms based on varied node densities. The proposed AOMCL considers the best nodes with high residual energy as anchor nodes. Being HWSN, the energy level of nodes varies. Hence this mechanism guarantees energy retention and efficiency. The complexity of the actual MCL is reduced in AOMCL with the generation of MCL square and effective sample set within the region. With the increase in anchor nodes, sampling rounds get reduced. It is analyzed that, at a lower node count, the time taken is somewhat similar for all the algorithms. But, as node density goes up, the computational time is higher for DEMCL, RMCL & QMCL.

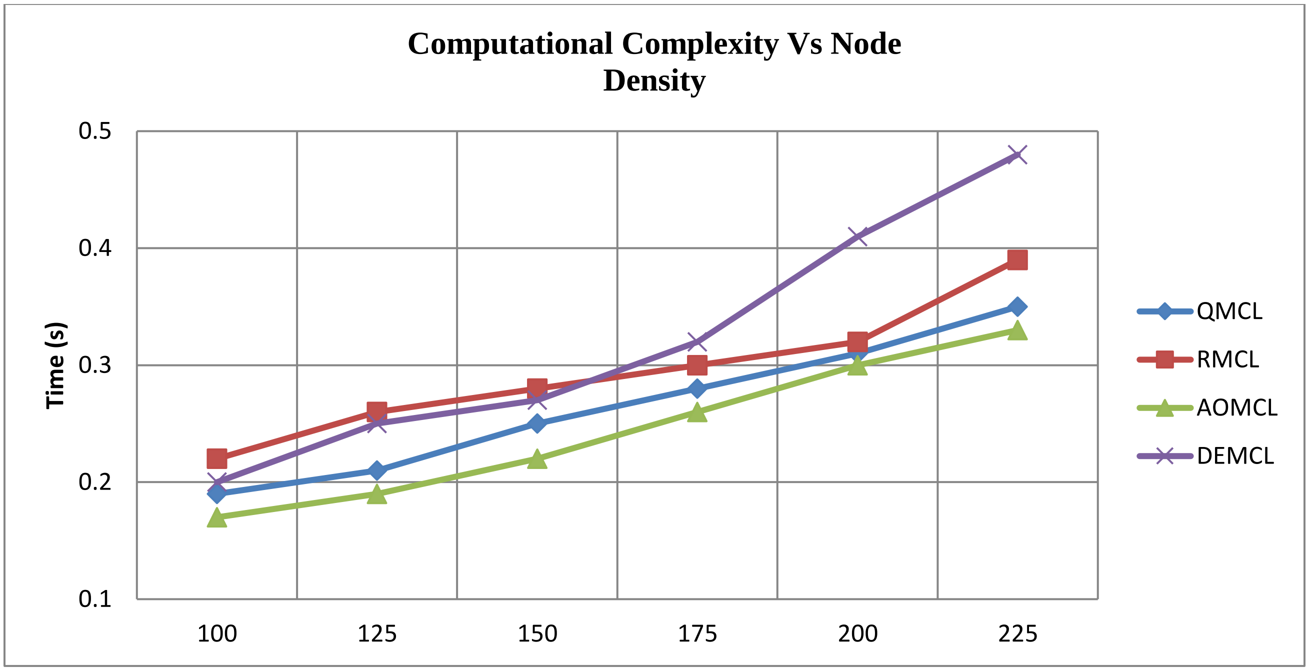

Fig. 6 Computational Complexity of AOMCL

Table 4. Computational complexity values of AOMCL

Nodes / Algorithm	100	125	150	175	200	225
QMCL	0.19	0.21	0.25	0.28	0.31	0.35
RMCL	0.22	0.26	0.28	0.3	0.32	0.39
DEMCL	0.2	0.25	0.27	0.32	0.41	0.48
AOMCL	0.17	0.19	0.22	0.26	0.3	0.33

6. Conclusion

Localization of mobile nodes in WSN was earlier done using periodic localization and calculation of static nodes. It was a hectic task with high communication costs and tedious calculations. Monte Carlo node Localization(MCL) algorithm with particle filtering scheme is an efficient method for localizing mobile nodes. It has limitations like inaccurate positioning and low location coverage, more number of anchor nodes, less efficiency, etc. Many variations of MCL were proposed for effective results, but the heterogeneity of sensor nodes was not keenly considered. The proposed work, Aquila Optimized Monte Carlo Localization(AOMCL), localizes the unknown heterogeneous nodes of HWSN with reduced complexity using MCL square. The research work is a coalition of mobile node localization protocol, MCL, and novel metaheuristic optimization protocol AO. The proposed protocol AOMCL was experimented with using Matlab. The results disclose that the proposed protocol is superior in localization accuracy and coverage percentage to the existing algorithms RMCL, DEMCL, and QMCL. The energy utilization is efficient as the complexity and searching space are reduced with the introduction of the MCL square. The work can be extended in the future to decrease the anchor node density in the network to minimize the hardware component and overall cost.

References

[1] Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless Sensor Networks: A Survey. Comput Netw, 38(4) (2002)393-422.

[2] Singhm, Khilar PM.Mobile Beacon-Based Range-Free Localization Method for Wireless Sensor Networks. Wirel Netw,23(4) (2017) 1285-1300.

[3] Yan, X., Yang, Z., Song, A., Yang, W., Liu, Y., & Zhu, R., A Novel Multihop Range-Free Localization Based on Kernel Learning Approach for the Internet of Things. Wireless Personal Communications, 87(1) (2016) 269–292.

[4] Yao, X., Zhou, J., Lin, Y., Li, Y., Yu, H., & Liu, Y., Smart Manufacturing Is Based on Cyber-Physical Systems and Beyond. Journal of Intelligent Manufacturing, 30(8) (2019) 2805–2817.

[5] Moh'd Alia, O., & Al-Ajouri, A., Maximizing Wireless Sensor Network Coverage With Minimum Cost using A Harmony Search Algorithm. IEEE Sensors Journal, 17(3) (2017) 882–896.

[6] Adulyasas, A., Sun, Z., & Wang, N.,Connected Coverage Optimization for Sensor Scheduling in Wireless Sensor Networks. IEEE Sensors Journal, 15(7) (2015) 3877–3892

[7] Abu-Mahfouz, A. M., & Hancke, G. P., Alwadha Localization Algorithm: Yet More Energy Efficient. IEEE Access, 5(5) (2017) 6661–6667.

[8] Kirci, P., & Chaouchi, H., Recursive and Ad Hoc Routing-Based Localization in Wireless Sensor Networks. Computer Standards & Interfaces, 44 (2016) 258–263.

[9] Pandey S, Varma S., A Range-Based Localization System in Multihop Wireless Sensor Networks: A Distributed Cooperative Approach. Wirel Pers Commun 86(2) (2016) 615–634.

[10] Alippi C, Vanini G., A RSSI-Based and Calibrated Centralized Localization Technique for Wireless Sensor Networks. Fourthannual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06). IEEE, (2006) 5.

[11] Wang P, Xue F, Li H, Cui Z, Chen J., A Multi-Objective Dvhop Localization Algorithm Based on NSGA-II in the Internet of Things.Mathematics, 7(2) (2019) 184.

[12] Yan J, Qiao R, Tang L, Zheng C, Fan B., A Fuzzy Decision-Based WSN Localization Algorithm for Wise Healthcare. China Communications, 16(4) (2019) 208–218.

[13] Zou D, Chen S, Han S, Meng W, an D, Li J, Zhao W., Design A Practical WSN-Based Fingerprint Localization System. Mobile Networks and Applications, (2019) 1–13.

[14] Ruz ML, Garrido J, Jim'enez J, Virrankoski R, V'azquez F., Simulation Tool for Analyzing Cooperative Localization Algorithms for Wireless Sensor Networks. Sensors 19(13) (2019) 2866.

[15] Xiong H, Sichitiu ML., A Lightweight Localization Solution for Small, Low Resources WSNS. J Sens Actuator Netw 8(2) (2019) 26.

[16] Bulusu N, Heidemann J, Estrin D., GPS-Less Low-Cost Outdoor Localization for Very Small Devices. IEEE Pers Commun, 7 (2000) 28–34.

[17] Niculescu D, Nath B (2003) DV Based Positioning in Ad Hoc Networks. Telecommun Syst, 22 (2003) 267–280.

[18] Nagpal R, Shrobe H, Bachrach J (2003) Organizing A Global Coordinate System From Local Information on an Ad Hoc Sensor Network. Lect Not Comp Sci, 2634 (2003) 333–348.

[19] J. Xing, Y. Ren, L. Yang, J. Sha, and J. Sun, Distributed Grid-Based Localization Algorithm for Mobile Wireless Sensor Networks, Springer, Berlin Heidelberg, (2012).

[20] H. Akcan, V. Kriakov, H. Brönnimann, and A. Delis, Managing Cohort Movement of Mobile Sensors Via GPS-Free and Compass-Free Node Localization, Journal of Parallel & Distributed Computing, 70(7) (2010) 743–757.

[21] S. Fujii, A. Uchiyama, T. Umeda, H. Yamaguchi, and T. Higashino, Trajectory Estimation Algorithm for Mobile Nodes using Encounter Information and Geographical Information, Pervasive and Mobile Computing, 8(2) (2012) 249–270.

[22] Singh SP, Sharma S., Range Free Localization Techniques in Wireless Sensor Networks: A Review. Procedia Computer Science, 57 (2015) 7–16

[23] A. El Assaf, S. Zaidi, S. Affes, N. Kandil, Low-Cost Localization for Multihop Heterogeneous Wireless Sensor Networks, IEEE Trans. Wirel. Commun. 15 (1) (2015) 472–484, Doi: 10.1109/TWC.2015.2475255 .

[24] W. Wu, X. Wen, H. Xu, L. Yuan, Q. Meng, Efficient Range-Free Localization using Elliptical Distance Correction in Heterogeneous Wireless Sensor Networks, Int. J. Distrib. Sens. Netw. 14 (1) (2018). Doi: 10.1177/1550147718756274

[25] Y. Xing, W. Huangfu, X. Dai, X. Hu, Node Localization Based on Multiple Radio Transmission Power Levels for Wireless Sensor Networks, in in International Conference on 5G for Future Wireless Networks, Springer, Cham, (2017) 19–27.Doi: 10.1007/978-3-319-72823-0_3. April

[26] L. Hu, D. Evans, Localization for Mobile Sensor Networks, in Tenth International Conference on Mobile Computing and Networking (Mobicom’04), Philadelphia, Pennsylvania, USA, (2004) 45–57.

[27] A. Baggio and K. Langendoen, Monte Carlo Localization for Mobile Wireless Sensor Networks, Ad Hoc Networks, 6(5) (2008) 718–733.

[28] L. Jianpo, S. Ming, Y. Xie, and S. Jisheng, A MCL Mobile Node Localization Algorithm Based on Fuzzy Theory, Computer Applications and Software, 30(12) (2013) 147–150.

[29] Z. Juqin, C. Yangyan, S. Xiaoping, and X. Guo, Research on Node Localization Algorithm Based on Adaptive Monte Carlo Algorithm for Dynamic Sensor Networks, Applied Laser, 36(4) (2016) 446–450.View At: Google Scholar.

[30] T. I. A. N. Haoshan, L. I. Cuiran, X. I. E. Jianli, and L. I. A. N. G. Yingxin, Node Localization Algorithm for WSN Based on Time Sequence Monte Carlo, Chinese Journal of Sensors and Actuators, 29(11) (2016) 1724–1730.

[31] M. Qin and R. Zhu, A Monte Carlo Localization Method Based on Differential Evolution Optimization Applied Into Economic Forecasting in Mobile Wireless Sensor Networks, EURASIP Journal on Wireless Communications and Networking, Article ID 32, 2018(1) 2018.

[32] Abualigah, L.; Diabat, A. Advances in Sine Cosine Algorithm: A Comprehensive Survey. Artif. Intell. Rev. 54 (2021) 2567–2608.
[33] Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the IEEE International Conference on Neural Networks (ICNN '93), Perth, WA, Australia, IEEE: Piscataway, NJ, USA, 27(1) (1995).
[34] Dorigo, M.; Birattari, M.; Stutzle, T. Ant Colony Optimization. IEEE Comput. Intell. 1 (2006) 28–39.
[35] Yang, X.S. A New Metaheuristic Bat-Inspired Algorithm. in Nature Inspired Cooperative Strategies for Optimization (NICSO); Springer: Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. Adv. Eng. Softw. 69 (2014) 46–61.
[36] Gandomi, A.H.; Yang, X.S.; Alavi, A.H. Cuckoo Search Algorithm: A Metaheuristic Approach to Solve Structural Optimization Problems. Eng. Comput. 29 (2013) 17–35.
[37] Mirjalili, S.; Lewis, A. the Whale Optimization Algorithm. Adv. Eng. Softw. 95 (2016) 51–67.
[38] Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H.L. Harris Hawks Optimization: Algorithm and Applications. Future Gener. Comput. Syst. 97 (2019) 849–872.
[39] Abualigah, L.; Yousri, D.; Elaziz, M.A.; Ewees, A.A.; Al-Qaness, M.A.A.; Gandomi, A.H. Aquila Optimizer: A Novel Meta-Heuristic Optimization Algorithm. Comput. Ind. Eng. 157 (2021) 107250.
[40] J. Ramkumar and R. Vadivel, Multi-Adaptive Routing Protocol for Internet of Things Based Ad-Hoc Networks, Wirel. Pers. Commun., (2021) 1–23. Doi: 10.1007/S11277-021-08495-Z.
[41] J. Ramkumar and R. Vadivel, Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks, Int. J. Comput. Digit. Syst, 10(1) (2020) 1063–1074. Doi: Http://Dx.Doi.Org/10.12785/Ijcds/100196.
[42] J. Ramkumar and R. Vadivel, Improved Wolf Prey Inspired Protocol for Routing in Cognitive Radio Ad Hoc Networks, Int. J. Comput. Networks Appl., 7(5) (2020) 126–136. Doi: 10.22247/Ijcna/2020/202977.
[43] J. Ramkumar, R. Vadivel, and B. Narasimhan, Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network, Int. J. Comput. Networks Appl., Doi: 10.22247/Ijcna/2021/210727.
[44] J. Ramkumar and R. Vadivel, Whale Optimization Routing Protocol for Minimizing Energy Consumption in Cognitive Radio Wireless Sensor Network, Int. J. Comput. Networks Appl., 8(4). Doi: 10.22247/Ijcna/2021/209711.
[45] F. Dellaert, D. Fox, W. Burgard, S. Thrun, Monte Carlo Localization for Mobile Robots, in IEEE International Conference on Robotics and Automation (ICRA99), Detroit, Michigan, USA, (1999).
[46] S. Thrun, D. Fox, W. Burgard, F. Dellaert, Robust Monte Carlo Localization for Mobile Robots, Artificial Intelligence 128 (1–2) (2001) 99–141.
[47] L. Hu, D. Evans, Localization for Mobile Sensor Networks, in Tenth International Conference on Mobile Computing and Networking (Mobicom'04), Philadelphia, Pennsylvania, USA, (2004) 45–57.
[48] J. Y. Lu and C. Wang, A New Monte Carlo Mobile Node Localization Algorithm Based on Newton Interpolation, EURASIP Journal on Wireless Communications and Networking, 2018(1) (2018) Article ID 161.