A Study on Wireless Sensor Node Localization and Target Tracking Based on Improved Locust Algorithm

Tan SONGHE, Qin Qi
Hechi University, School of Big Data and Computer, Hechi 546300, China

***Abstract*—To improve the positioning accuracy of wireless sensor nodes and ensure the target tracking effect, a wireless sensor node positioning and target tracking method based on an improved locust algorithm is proposed. The DV Hop algorithm is used to calculate the minimum hops and average hops distance between the unknown node and each anchor node to obtain the location of the unknown node, realize the rough positioning of wireless sensor nodes, and analyze the positioning error to determine the positioning accuracy target function; The improved locust algorithm is used to solve the positioning accuracy objective function to obtain the sensor node positioning results with the minimum error; The target tracking model and the target is calculated. According to the target observation information obtained by all sensor nodes, the target state in the wireless sensor network model is tracked using the probability hypothesis density filtering algorithm. The test results show that the algorithm has better performance, the spatial evaluation index results are all lower than 0.020, and the individual distribution in the solution set is better; The location of each unknown node in different node distribution states can be obtained; The positioning error under the surface and plane is less than 0.012; The maximum error of target tracking is 0.142m; It can track single target and multiple targets.**

Keywords*—Improved locust algorithm; wireless sensors; node localization; target tracking; target state; unknown node position

I. Introduction

This Nodes in wireless sensor networks are a crucial part of the network, they have the ability of autonomy, can be organized autonomously, and do not require additional wiring, which provides them with excellent flexibility [1]. The primary function of wireless sensor networks is to monitor the surrounding conditions, such as water quality monitoring and forest fire protection monitoring. The location information of these sensing nodes is crucial for the network to realize sensing because, without location information, the measurement data is meaningless in real situations [2], [3]. To acquire the location information of these environmental data, it is essential to solve the self-localization problem of sensor nodes first. Node localization refers to the technical means by which the sensor nodes collaborate to determine their position coordinates [4]. With the position information of nodes, target tracking, movement trajectory prediction, etc. can be realized, and at the same time, it is also important for the construction of a network topology map and calculation of network coverage [5]. In target tracking, algorithms are used to estimate the velocity, position, angle, and other characteristic quantities of the target and predict its movement trajectory. However, in practical applications, localization accuracy is crucial for unknown nodes, which directly affects the validity of the data collected by the nodes [6], [7], [8]. Beacon nodes are categorized into fixed and mobile beacon nodes. In some scenarios, the distribution of nodes may be uneven due to cost or terrain constraints, failing localization of some unknown nodes due to insufficient reference beacon nodes.

In study [9], an adaptive distance-based localization (ARBL) algorithm based on the trilateral measurement and reference node selection is studied. The algorithm locates nodes according to the evaluation results of node geometry, to improve the positioning accuracy of sensor nodes and provide a basis for target tracking; If the evaluation of node geometry is not accurate enough or has errors, it will lead to the inaccuracy of node positioning results obtained by this method, thus affecting the effect of target tracking. The study in [10] carried out relevant research on the large-scale positioning accuracy of wireless sensor nodes underwater, combined with the special underwater environment and the underwater performance of the network, proposed a distance-based multilateral accumulation method, which will move the gateway node on a fixed track, and collect the underwater node information to locate the node position independently; The particularity of underwater environment will cause signal transmission attenuation, multipath effect, and other problems. At the same time, the performance of the underwater sensor network is also affected by water quality, water flow, and other factors, which may lead to the decline of node positioning accuracy. In study [11], the distance between two nodes is calculated, and the distance between unknown nodes and beacon nodes is obtained through the results of hop value and size. Finally, the probability distance estimation value is calculated, and the corresponding distance of this result is the node position; however, this method may have measurement errors and uncertainties. Measurement error and uncertainty will affect the exact estimation of the distance between nodes, and then affect the positioning accuracy and tracking effect of nodes. In the study [12], to achieve the accurate positioning of wireless sensor nodes, the signal strength, distance, and other parameters are optimized according to the quaternion backtracking search optimization algorithm by calculating the distance between anchor nodes and beacon nodes, to improve the positioning accuracy, reduce the positioning error and improve the target tracking effect; Due to the use of backtracking search optimization algorithm, more computing resources and time may be required to complete the precise positioning of nodes. In addition, the convergence of the algorithm also needs reasonable adjustment and parameter setting to guarantee the steadiness and accuracy of the results.

Considering that in wireless sensor node localization and target tracking, it is usually carried out under conditions of uneven node density, strong signal interference, or rapid target movement. By introducing a pairing self-learning mechanism to improve the locust algorithm, individuals in the population can learn from each other, share search experience and knowledge, thereby improving overall performance. Therefore, in the application of wireless sensor node localization and target tracking, the improved locust algorithm is used for wireless sensor node localization and target tracking, in order to more accurately locate nodes, track targets faster, and demonstrate stronger robustness in complex environments.

II. Wireless Sensor Node Localization

A. Rough Localization Algorithm for Wireless Sensor Nodes Based on DV Hop

To achieve wireless sensor network node location, the DV Hop algorithm is used in this paper. The working process of this algorithm is simple, easy to implement, and does not need distance-measuring equipment. It is a classic wireless sensor network node location algorithm without distance measuring. DV Hop location can be divided into three stages, namely connectivity detection, distance estimation, and unknown node location estimation [13]. The detailed stages are displayed below:

1) Determine the minimum number: The anchor node broadcasts the packet in a flooded manner and other nodes store and forward the packet and record the minimum number of hops.

2) Calculate the average hopping distance, using the formula as follows:

$$\bar{S}_k = \frac{\sum_{k\neq u}\|X_k - X_u\|_2}{\sum_{k\neq u} B_{k,u}} \tag{1}$$

where, $\bar{S}_k$ and $X_k = [x_k, y_k]^T$ represent the average distance per jump and the horizontal and vertical coordinates of the anchor node k, respectively; the $B_{k,t}$ is the minimum number of hops for the shortest path between anchor node k and u.

Computing the distance between the unknown nodes m and the anchor node k:

$$d_{m,k} = B_{k,u} \times \bar{S}_k \tag{2}$$

3) Estimation of unknown node locations: The unknown node location is calculated using the trilateral or multilateral localization method, assuming that the deployment environment has n anchor nodes, whose matrix is given by:

$$d_{m,k} \times \begin{bmatrix} (x_1 - x_m)^2 + (y_1 - y_m)^2 \\ (x_2 - x_m)^2 + (y_2 - y_m)^2 \\ \vdots \\ (x_n - x_m)^2 + (y_n - y_m)^2 \end{bmatrix} = \begin{bmatrix} d_{m,1}^2 \\ d_{m,2}^2 \\ \vdots \\ d_{n,m}^2 \end{bmatrix} \tag{3}$$

where: (x_m, y_m) are the coordinates to be found. $(x_1, y_1) \cdots (x_n, y_n)$ are the coordinates; the $d_{m,1} \cdots d_{n,m}$ is the distance between the unknown node u

By the method of least squares, the coordinates to be found are:

$$X = (A^T A)^{-1} \begin{bmatrix} d_{m,1}^2 \\ d_{m,2}^2 \\ \vdots \\ d_{n,m}^2 \end{bmatrix} + A^T B \tag{4}$$

where, T denotes the number of orders. X denotes the position matrix, and $X = [x, y]$, A and B denote the known node and unknown node coordinate matrices, respectively; the matrix formulas are:

$$A = \begin{bmatrix} 2(x_n - x_1) & 2(y_n - y_1) \\ 2(x_n - x_2) & 2(y_n - y_2) \\ \cdots & \cdots \\ 2(x_n - x_m) & 2(y_n - y_m) \end{bmatrix} \tag{5}$$

$$B = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ x_2^2 - x_n^2 + y_2^2 - y_n^2 + d_n^2 - d_2^2 \\ \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix} \tag{6}$$

Solving Formula (3), the position of the unknown sensor node can be obtained. The whole localization process is shown in Fig. 1.

B. Objective Function Determination for Coarse Localization Accuracy of Wireless Sensor Nodes

Disturbed by environmental factors as well as by the measurement equipment itself [14], the measurement distance d has a certain deviation from the actual distance between the unknown node and the anchor point, but this interference cannot be avoided in practice, so the $n-1$ dimensional random error vectors σ is introduced in Formula (4), then Formula (4) is converted to:

$$AX + \sigma = B \tag{7}$$

If E_i is the distance error between the node to be located and the ith anchor node, then the solution error function calculation formula of the position (x, y) of the sensor node to be located is:

$$f(x, y) = B \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2} - E_i \tag{8}$$

To make the unknown sensor node localization accuracy higher, then it is to minimize the value of Formula (8), i.e.

$$min\, f(x, y) = B \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2} - E_i \tag{9}$$

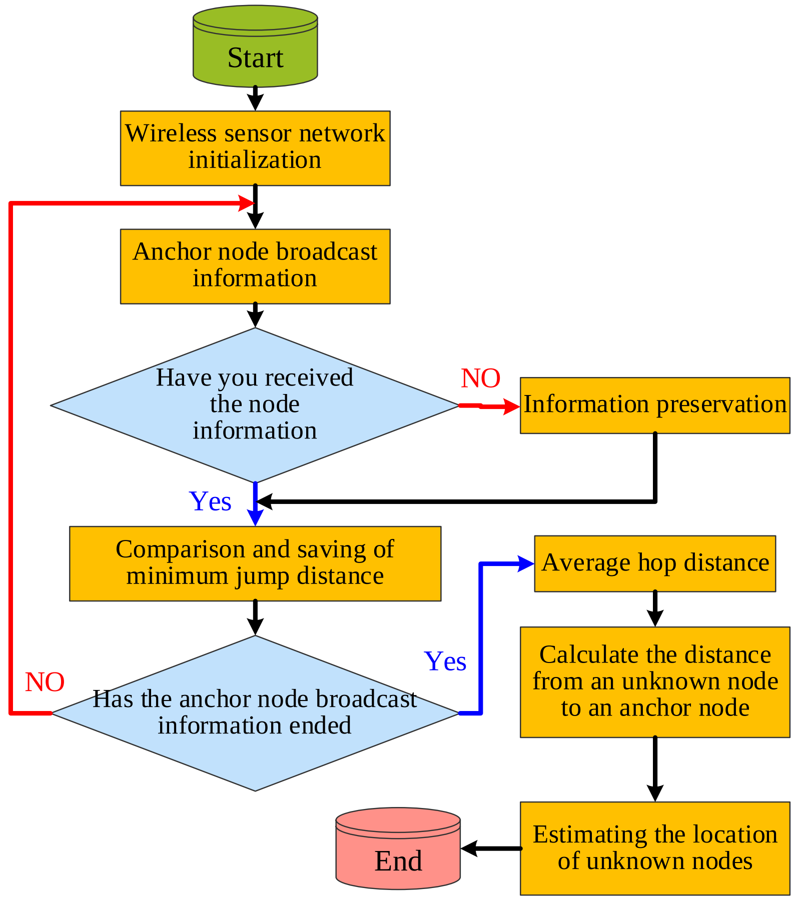


Fig. 1. Rough localization process of wireless sensor nodes based on DV Hop.

According to Formula (9), the sensor localization problem is converted into a localization error minimization objective problem, and the result of minimizing the localization error can be obtained by solving this Formula [15], the coordinate point $\tilde{X} = [\tilde{x}, \tilde{y}]$ corresponding to this result is the precise sensor node position.

C. *Coarse Localization Objective Function Solving for Wireless Sensor Nodes Based on Improved Locust Algorithm*

1) *Principles of the improved locust algorithm*

a) The locust algorithm: The Grasshopper Optimization Algorithm (GOA) is a metaheuristic biomimetic optimization algorithm proposed by Saremi et al. in 2017. It has high search efficiency and fast convergence speed, and the algorithm's unique adaptive mechanism can balance the global and local search processes well, with good optimization accuracy. The algorithm utilizes the local development of larvae and the global search of adults in a two-stage approach to achieve the target search. After completing the coarse positioning of wireless sensor nodes through the above subsections, to ensure the accuracy of the positioning results, the minimization of the error of the sensor node position is designed as an objective function, using this objective function $\min f(x, y)$ as the initial population for the improved locust algorithm, since the objective function is the result of the minimum error of the positions of all nodes, that is, the minimum positioning error of each coordinate point $X = [x, y]$, so the paper uses the set of individual coordinates to replace the objective function, using X_i to represent that each error solution is each individual in the population [16], the objective function reaches the optimal value by adjusting the parameters such as the position and velocity of the individuals in the population.

The algorithmic bionic principle is to map the small-scale movement behavior of locusts in the larval stage to the local exploitation of short steps, and the large-scale movement behavior of locusts in the adult stage to the global exploration of long steps, and the process of searching for the food source is the algorithmic optimization process. The formula related to the locust's swarming behavior is:

$$X_i = S_i + G_i + H_i \tag{10}$$

where, X_i denotes the position of the individual i; and G_i is the force of gravity on the individual. H_i is the wind force on the individual. S_i is the individual's community force [17], which is calculated by the formula:

$$S_i = \sum_{j=1, j\neq i}^{N} s(d_{ij})\, d'_{ij} \tag{11}$$

where d_{ij} denotes the interval between two individuals i and j . d'_{ij} denotes the unit vector. s denotes the strength function of the community force, which is calculated as follows:

$$s(r) = fe^{\left(\frac{r}{l}\right)} - e^{-r} \quad (12)$$

where f and l denote the strength of attraction and the attraction step, respectively.

As the spacing between individuals gradually increased,$s$is gradually not possible to generate community forces. Therefore, the algorithm restricts the position of individuals in the population to between [1, 4]. If all individuals are in their comfort zone, the position remains constant.

b) Improved locust algorithm: The Locust algorithm in the objective function solution, the algorithm has advantages, but its global search capability is general, the convergence rate is slower, and the initial parameters of the dependence of the initial parameter are large [18], affecting its ability to search for the optimal. Therefore, the article introduces a pairing self-learning mechanism to enable individuals in the population to learn from each other and improve overall performance. At the same time, a decreasing factor based on hyperbolic functions is used to optimize the individual movement range, so that the algorithm can maintain a larger search range in the early stages of iteration to explore the global optimal solution, and in the later stages, the search range is reduced to accelerate convergence. Optimize the content of the following:

In a population, actively learning from excellent neighboring individuals can improve the IGOA algorithm to introduce pairing self-learning to realize individual information exchange and improve the quality of the population. The specific process is to sort individuals according to their fitness, and then divide locusts into two groups according to their fitness values, namely self-learning groups X_o and sample clusters X_υ. The X_o group consists of the top half of individuals with better fitness values, the$X_\upsilon$group consists of the other half of the remaining poorly adapted individuals.

MakeX_i^o and X_i^υrespectively represent the position of locust i in groups$X_o$andX_υ, then the individuals in both groups satisfy the following conditions:

$$\begin{cases} f(X_1^o) \leq f(X_2^o) \leq, \dots, \leq f(X_n^o) \\ f(X_i^\upsilon) \leq f(X_2^\upsilon) \leq, \dots, \leq f(X_n^\upsilon) \end{cases} \quad (13)$$

where, the fitness values of X_i^o and X_i^υ are denoted by$f(X_i^o)$and$f(X_i^\upsilon)$respectively.

The main purpose of this optimization is that the individuals with the worst fitness inX_o learn [19] from the individual with the best fitness inX_υ, while the best individual in$X_o$learns from the worst individual inX_υ, and so on, as a way to expand the variability between the self-learning individuals and the sample individuals. According to the differences between the two, the individual position in$X_o$is updated, and the update formula is:

$$\tilde{X}_i^o = X_i^o + DX_i^o \quad (14)$$

where: D denotes the exchange operator variant; the$\tilde{X}_i^o$represents the new position of individual$i$pairing after self-learning.

Based on Formula (14), utilizing a merit-based retention strategy, to determine the final individual position in the groupX_o, which was calculated by the formula:

$$X_i^o = \begin{cases} \tilde{X}_i^o, f\left(X'^o_i\right) < f(X_i^o) \\ X_i^\upsilon, f\left(X'^\upsilon_2\right) \geq f(X_2^\upsilon) \end{cases} \quad (15)$$

c) Optimization for declining factor η of individual movement range.

Individuals in the search process between locusts attraction, repulsion, and comfort zones are all determined byη, the coefficient has a large impact on the algorithm's ability to look for the optimal best answer due to the fact that$\eta$is linearly varying and drops too quickly that the beginning of the algorithm, resulting in the locusts not being able to traverse more of the solution space, and that the linear variation characteristic$\eta$of the algorithm leads to insufficient global exploration ability at the early stage of the algorithm, and at the late stage of the algorithm, the parameter decreasing trend is too smooth, which is easy to stagnate near the local optimum prematurely [20], and the speed of convergence decreases. To solve this problem, a curve function$\eta(t)$is proposed to replace the parameterη, enabling a better balance between the global exploration and local exploitation capabilities of algorithms. The formula for $\eta(t)$is:

$$\eta(t) = -\eta_{\max} \operatorname{csc} h\,(\beta t)^2 \quad (16)$$

$$\beta = \frac{1}{T_{\max} \ln\left(\frac{\eta_{\min}}{\eta_{\max}}\right)} \quad (17)$$

where, t is the current iteration number, the β is the intermediate quantities. T_{max} is the maximum number of iterations; the η_{max} and η_{min} indicate the maximum and minimum values ofη.

From this formula, it can be seen that$\eta(t)$is still a decreasing function and by the hyperbolic function property, $\eta(t)$has a larger value and slower descent in the early stage allowing the algorithm to explore the whole world with a larger step size in the early iteration; at the same time, the smaller value and faster descent in the late iteration make the algorithm accelerate the convergence speed and realize the optimization of the algorithm. According to the above steps to complete the optimization of the algorithm, the optimization process is shown in Fig. 2.

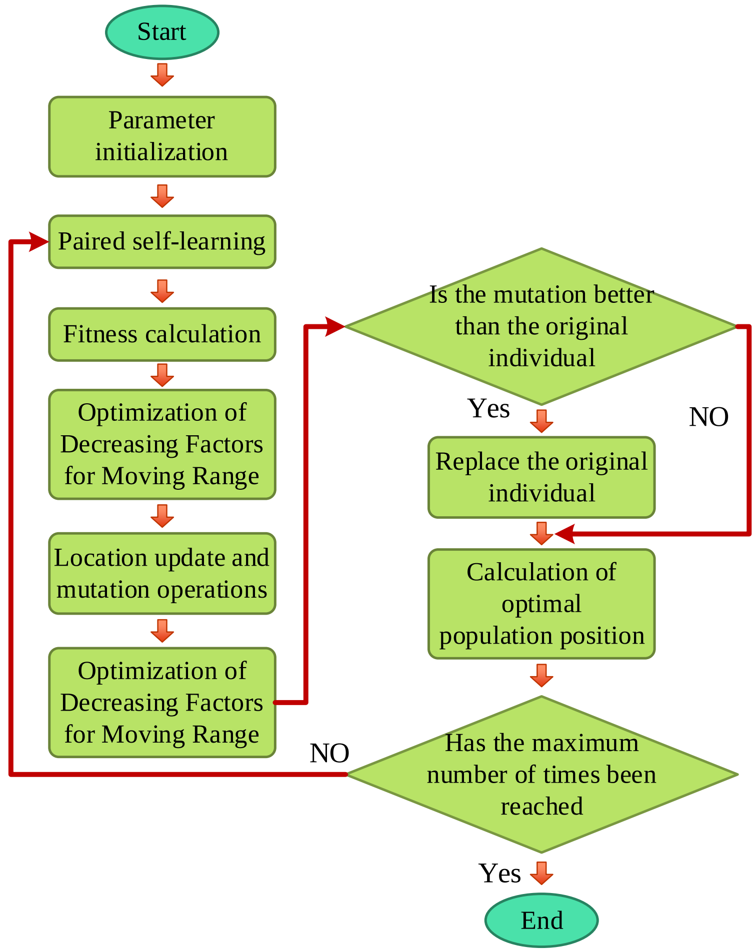


Fig. 2. Optimization process for rough localization of wireless sensor nodes based on improved locust algorithm.

The optimization solving steps for the localization objective function of the improved locust algorithm are described below:

Step 1: Setting the parameters of the locust population, mainly including the population size n, spatial dimension $D = 2$, maximum number of iterations L, a decreasing factor that reduces the range of movement of individual locusts η, its maximum value η_{max} and the minimum value η_{min}; initializing locust populations X_i.

Step 2: Calculating the fitness value $f(i)$ for each average hop distance by the following formula:

$$f(i) = \|y' - C\hat{\mu}_i\|_2 \tag{18}$$

where, $\hat{\mu}_i$ denotes the estimation vector and y' denotes the measurement vector of the objective function; the C denotes the sensing matrix.

Step 3: After completing the calculation of the fitness value according to the Formula (5), the given values should be adjusted in descending order to obtain the maximum value of the amount, and the result corresponding to this value is the individual optimal value X_{best}.

Step 4: Update the parameters η, position X_i^o and the probability P_s of mutation, the formula is:

$$\eta(t) = \eta \, sin\left(\frac{\pi}{2} \times \frac{l}{L} + \pi\right)(\eta min_{max})_{min} \tag{19}$$

$$X_i^o = \eta\left(\sum_{\substack{j=1 \\ j\neq 1}}^{N} \eta \frac{d_{\max} - d_{\min}}{2} sd_{ij}\hat{d}_{ij}\right) + \hat{T}_d \tag{20}$$

$$P_s = 0.3 - 0.3 \times \left[\frac{1}{e-1} \times \left(e^{\frac{l}{L}} - 1\right)\right] \tag{21}$$

where, l stands for the current number of iterations. d_{max} and d_{min} are the upper and lower bounds for the d dimension. d_{ij} stands for the Euclidean interval between individual i to individual j. $\hat{d}_{ij}$ indicates the unit vector of an individual i pointing to the individual j; $\hat{T}_d$ is the optimal solution for the d dimension; the e denotes a natural constant.

For each result X_{best}, the variation operation is performed, and the calculation formula is:

$$X_i(t+1) = w_1 p_1 [X_{best} - X_i^o] + w_2 p_2 [\tilde{X}(k) - X_i(t)] \tag{22}$$

where, w_1 and w_2 is the weight parameter, the X_{best} is the optimal solution, , and $\tilde{X}(k)$ is a randomized result, the p_1 and p_2 are coefficients that obey the Cauchy distribution.

Step 5: Judge whether the fitness of the mutated individual is greater than that of the original individual, if it is greater than that of the original individual, then the original individual is retained; otherwise, the mutated individual is retained.

Step 6: Compute population's overall fitness, rank, and update the position of the optimal individualX_{best}.

Step 7: Determine whether to reach the maximum number of iterations, if the maximum number of iterations, then go to the next step; otherwise, repeat the operation from step 4.

Step 8: Output the global optimum value of the$\tilde{X} = [\tilde{x}, \tilde{y}]$.

After performing the above operations on all individuals, the above operations are performed again with a new population, and the optimal solution, i.e., the sensor node with the smallest localization error, is finally obtained by repeated cycles.

III. Wireless Sensor Networks for Target Tracking

A. *Target Tracking Model Construction for Wireless Sensor Networks*

According to the above subsection to complete the wireless sensing node localization after the target tracking, in the tracking before the need to construct the wireless sensor network target tracking model [21], the model is constructed according to the calculation results of $\tilde{X} = [\tilde{x}, \tilde{y}]$ and the structure of the constructed target tracking model for wireless sensor network is displayed in Fig. 3.

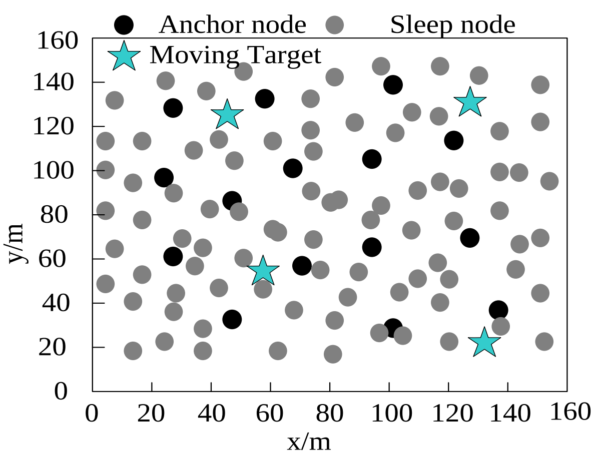


Fig. 3. Structure of wireless sensor network target tracking model.

In this model for target tracking, the sensor nodes do not need to be connected to the aggregation node for each tracking information, the nodes in the network can compute the position information of the target locally, and the sensor nodes are connected to the aggregation node only when they need to transmit the tracking information to the aggregation node [22]. Each sensor node in the tracking model can know its exact location coordinates through the result of$\tilde{X}$. In the process of tracking, the position coordinates of the sensor nodes are considered as a priori information. To achieve accurate tracking of the target [23], time synchronization has been implemented in the network. Sensor nodes have three operating states: active, listening, and sleeping. Sensor nodes in the active state can sense targets and communicate with neighboring sensor nodes. Sensor nodes in the sleep state can neither sense the target nor communicate with the neighboring sensor nodes, and the most energy-efficient state is the sleep state. The sensor node periodically changes and activates if it receives a message that the target is approaching.

N sensors are deployed in the target area.$S_1, S_2, \cdots, S_N$represent the respective positions, when the tracked target is active in the network of sensor nodes [24], the state is$\tilde{X}_t = [x_t, y_t, v_t^x, v_t^y]$at the moment$t$, of which$(x_t, y_t)$is the position of the target at the moment t.$(v_t^x, v_t^y)$is the velocity in the x and y direction, the dynamic model of the target is displayed below:

$$\begin{cases} \tilde{X}_t = Rx_{t-1} + Gu_{t-1} \\ Z_t^i = f_t^i(x_t) + \varepsilon_t^i \end{cases} \tag{23}$$

where, $t+1$ represents b time of $t+1$; R is the transfer matrix, the G is a constant matrix; the ε_k^i and u_{k-1} denote measurement noise and process noise, respectively. f_t^i denotes the measurement function. X_t and Z_t^i denote, respectively, the state vector of the tracked target and the measurement vectors of the$i$th individual nodes. Then the observation matrix of each node is$Z_t = [(z_t^1)^T, (z_t^2)^T, \cdots, (z_t^N)^T]^T$.

The estimator of the target state is given by:

$$\begin{cases} \hat{x} = E(x_t|Z_t) \\ \bar{x} = E(x_t|Z_{t-1}) \end{cases} \tag{24}$$

$$\begin{cases} P_t = \sum t|t-1 \\ M_t = \sum t|t-1 \end{cases} \tag{25}$$

where:$\hat{x}_t$and$M_t$represent the target estimation state and error matrix, the$\bar{x}_t$and$P_t$represent the target prediction state and error matrix.

Using an extended Kalman filtering algorithm in the form of information to calculate $\bar{x}_t$, $\hat{x}_t$, P_t and M_t, the calculation formula is as follows:

$$\bar{x}_t = R\hat{x}_{t-1} \tag{26}$$

$$\hat{x}_t = \bar{x}_t + K_t\left(Z_t - H_{t,c}\bar{x}_t\right) \tag{27}$$

$$P_t = RM_{t-1}R^T + GQG^T \tag{28}$$

$$M_t^{-1} = P_t^{-1} + \sum_{i=1}^{N} \left(H_t^i\right)(R^i)^{-1}H_t^i \tag{29}$$

where:$K_t$denotes the set of non-target-generated interference measures.$H_{t,c}$denotes a mapping projection.Qdenotes the actual distance between the target and the sensor node; the$H_t^i = \frac{\partial f_t^i}{\partial \bar{x}_t^T}$.

B. *Target State Tracking Based on Probabilistic Hypothesis Density Filtering Algorithm*

After completing the target tracking model construction of the wireless sensor network according to the above subsection, the probabilistic hypothesis density filtering algorithm is used in the paper to track the target state in the model. The target tracking problem is essentially a nonlinear filtering problem [25], and the optimal Bayesian recursive filtering algorithm is used in the paper to solve the problem, and the expression of the target state tracking is as follows:

$$f_{t+1|t}(\tilde{X}|Z^{(t)}) = \int^{\int} f_{t+1|t}(\tilde{X}|q) f(q|Z^{(t)})_{t|t} \tag{30}$$

$$f_{t+1|t+1}(\tilde{X}|Z^{(t+1)}) = \lambda^{-1} f_{t+1}(Z_{t+1}|\tilde{X}) f_{t+1|t}(\tilde{X}|Z^{(t)}) \tag{31}$$

Of which:$\tilde{X}$is the set of localized target states in the sensor network model, containing variables such as the number of tracked targets and state parameters, and in case of multi-target tracking [26], then, the$\tilde{X} = \{\tilde{X}_1, \tilde{X}_m\}$, denotes the existence of multiple goals, and the state variables of the goals satisfy$\tilde{X}_1 \neq \tilde{X}_m$; $Z_t$indicates the target observations acquired by all sensor nodes at the time$t$, the$Z^{(t)}$indicates all observations prior to the time t, i.e., the $Z^{(t)} = Z_1, \cdots, Z_t$; $L_{Z,t}(\tilde{X}) = f_y(Z|\tilde{X})$ is the observed likelihood function; the $f(\tilde{X}|q)_{t+1|t}$ is the multi-objective state set Markov state transfer density; the$f(\tilde{X}|Z^{(t)})_{t|t}$is the posterior probability density of the set of multi-objective states at the moment$k$; the$\lambda$is the normalization parameter, which is calculated as follows:

$$\lambda = \int f_{t+1}(Z_{t+1}|\tilde{X}) f_{t+1|t}(\tilde{X}|Z^{(t)}) \delta\tilde{X} \tag{32}$$

where:δ represents the observation factor.

The algorithm is applied in such a way as to reduce the computational complexity of the algorithm by using first-order statistical moments $\psi_{t|t}$ replacing the probability distribution density function, then:

$$\psi_{t|t}(\tilde{X}|Z^{(t)}) = \int f_{t|t}(\tilde{X} \cup q|Z^{(t)}) \delta q \tag{33}$$

Where:$q$denotes the weighting factor.

The detailed steps of target state tracking based on the probabilistic hypothesis density filtering algorithm are described below:

Step 1: Measurement Screening.

When the target enters the monitoring range of the sensor network at the moment t, the relevant node is triggered to measure and obtain the distance value$l_j(t)$between the target. Compare$l_j(t)$to a pre-stored threshold. When$l_j(k)$, ie., the node is close enough to the target, the node sends the measurement data, otherwise it does not send the data and the node enters the dormant state when it is not greater than the threshold value;

Step 2: Measurements are routed to the base station.

The cluster head node of a cluster of nodes formed by the set of observed state nodes receives the measurements sent by the member nodes and routes the data to the base station through the reverse multi-propagation tree.

Step 3: The base station performs information fusion on the measurement data.

Firstly, the observation data from the base station are predicted as a means of estimating the target's motion state, which is given by the following formula:

$$\psi_{t+1|t}(x) = b_{t+1|t}(x) + \int \begin{pmatrix} p_s(x) f_{t+1|t}(x|q) + \\ b_{t+1|t}(x|q) \end{pmatrix} \psi(q)_{t|t} \tag{34}$$

where: the multivariate Gaussian distribution density is used as the probability density $f_{t+1|t}(x|q)$, $p_s(x)$ denotes the probability that the target set exists at the next moment, the $b_{t+1|t}(x|q)$ is the distribution density of the derived set, the$b_{t+1|t}(x)$is the distribution density of the freshman set.

Upon receiving a target observation$Z_{t+1}^{[j]}$sent by a sensor network node, the base station corrects the predicted target motion state, and at the same time, outputs the filter value of the target state to complete the tracking of the target at that moment. Finally, the observation is transferred to calculate the motion state of the target at the next moment, and the correction formula is as follows:

$$\psi_{t+1|t+1}(x) \cong F_{t+1}(Z_{t+1}|x) \psi_{t+1|t}(x) \tag{35}$$

$$F_{t+1}(Z_{t+1}|x) = \sum_{z^{[j]} \in Z_{t+1}^{[j]}}^{\Sigma_D^{[j]}} \frac{p_D(x) L_z(x)}{\hat{c}(z^{[j]}) + \psi[p_D L_z{}^{[j]}]_{t+1|t}} \tag{36}$$

where:$Z_{t+1}^{[j]}$is a subset of the observation setZ_{t+1}, i.e., the observations of the$j$th individual sensors.$L_z(x) = f_{t+1}(z|x)$is likelihood function, the$\lambda = \lambda_{t+1}$is poisson distributed scanning error, probability density $c(z) = c_{t+1}(z)$; $p_D(x)$ is the probability of detecting the target.

After the correction is completed, the state filter value of the tracked target is output, and the tracking of the target at that moment is completed.

IV. ANALYSIS OF RESULTS

In this paper, Matlab 2017 toolbox programming is used for simulation experiments to test the application impact of the method in this study on wireless sensor node positioning and target tracking. Network simulation parameters and algorithm parameters are shown in Table I.

TABLE I. PARAMETER SETTING RESULTS

Parameter	Numerical value
Network range /m	**150×150**
Number of sensor stages/piece	**115**
Communication radius /m	**25**
Number of anchor nodes/piece	**15**
Unknown number of nodes/pieces	**100**
Power /dBm	**31**
Path Loss Exponent	**2**
Spatial dimension/dimension	**2**
Distribution of sensor nodes	**Uniform distribution of anchor nodes, unknown Random distribution of nodes**
Population size	**100**
iterations	**300**
The maximum value of the decreasing factor for individual movement range	**1**
The minimum value of the decreasing factor for individual movement range	**4**
weight	**0.5**

In the process of sensor node localization, the method solves the localization objective function by improving the locust algorithm. To verify the application performance of the method, the space evaluation index (SM) is used to measure the standard deviation of the minimum distance from each solution to other solutions in the approximate solution set, reflecting the uniformity. The index value is between 0 and 1, and the smaller the value is means that the more uniform the distribution of individuals in the solution set, the better the solution performance of the algorithm. The calculation formula of this index is:

$$\Gamma_{sm} = \sqrt{\sum_{i=1}^{N} \frac{\left(d_i - \bar{d}\right)}{N-1}} \tag{37}$$

where:$d_i$denotes the distance between solutions; its average value is denoted by$\bar{d}$;$N$is the number of individuals in the non-dominated solution set.

Under different numbers of individuals in the non-dominated solution set, the method in the paper is used to localize the sensor nodes and obtain the localization objective function during the solution process, with the gradual increase in the number of sensor nodes, the result of theΓ_{sm}during the solution procedure. Table II indicate the outcomes.

After analyzing the test results in Table II, it is concluded that: when using the method in the paper for solving the objective function of wireless sensor localization, with the gradual increase in the number of sensor nodes under different numbers of non-dominated solution sets of individuals, all theΓ_{sm}results during the solving process of the method are lower than 0.020, and the uniformity of the distribution of individuals in the solution set is good.

TABLE II. MEASUREMENT RESULTS OF SPATIAL EVALUATION INDICATORS

Number of nodes	**Number of individuals in the non-dominated solution set**		
	5	**10**	**15**
10	**0.011**	**0.015**	**0.016**
20	**0.012**	**0.014**	**0.014**
30	**0.015**	**0.011**	**0.013**
40	**0.009**	**0.016**	**0.011**
50	**0.014**	**0.011**	**0.009**
60	**0.013**	**0.013**	**0.007**
70	**0.016**	**0.012**	**0.012**
80	**0.011**	**0.015**	**0.011**
90	**0.008**	**0.011**	**0.010**
100	**0.010**	**0.013**	**0.015**

TABLE III. LOCALIZATION RESULTS OF 10 UNKNOWN NODES

Communication radius /m	Random distribution	Regular distribution
2	(10.2,33.7)	(20.5,45.1)
4	(23.5,55.8)	(17.6,22.5)
6	(27.7,9.88)	(37.6,46.2)
8	(28.1,12.6)	(12.3,62.3)
10	(31.4,40.6)	(20,15.5)
12	(35.2,46.8)	(31.1,55.3)
14	(33.5,68.9)	(41.3,17.4)
16	(40.4,22.2)	(44.4.51.2)
18	(41.3,21.5)	(30,35.5)
20	(48.1,2.6)	(50.5,15.6)
22	(46.2,33.8)	(62.3,30.2)
24	(60,15)	(48.1,50.5)

To check the localization performance of the method, the localization results of the method in the paper for the location of unknown sensor nodes in both cases of random and regular distribution of sensor nodes, with the gradual increase of the communication radius, due to the limited space, the results are only presented randomly for the localization results of 10 unknown nodes, as displayed in Table III.

After analyzing the test results in Table III, it is concluded that: the sensor nodes in the random distribution and regular distribution of two cases, respectively, through the text of the method of sensor node localization, can obtain the location of each unknown node, so the method of the text of the wireless sensor node localization capabilities.

To further verify the sensor node localization effect of the method in the paper, the unknown sensors are localized by the method in both planar and curved surfaces, and the position results of each sensor are obtained, and the average localization error $\overline{\varepsilon}$ is used in the study as an evaluation index, the result of this index is between 0 and 1, and the larger value indicates the higher positioning accuracy, and the formula of this index is:

$$\bar{\varepsilon} = \frac{\sum_{i=1}^{M} \sqrt{(x_e - x_i)^2 (y_e - y_i)^2}}{M} \tag{38}$$

where:(x_e, y_e)denotes the sensor localization result of the method in the text; the(x_i, y_i)denotes the result of the actual position of the wireless sensor; the$M$iindicates the number of unknown nodes.

Based on the above formula, the method in the paper is calculated under different numbers of unknown nodes, and due to the limited space, the results only present the positional localization results of any 10 unknown sensing nodes, as shown in Table IV.

TABLE IV. LOCATION RESULTS OF 10 UNKNOWN SENSOR NODES

Unknown number of nodes/pieces	Planar distribution	Surface distribution
10	0.006	0.008
20	0.006	0.007
30	0.006	0.009
40	0.007	0.011
50	0.009	0.008
60	0.005	0.01
70	0.004	0.007
80	0.008	0.009
90	0.009	0.012
100	0.006	0.011

After analyzing the test results in Table IV, it is concluded that in the two cases of planar distribution and curved surface distribution, using the method in the paper to locate the unknown sensors, with the gradual increase in the number of unknown nodes, the results of the unknown node localization of the wireless sensors $\overline{\varepsilon}$ values are below 0.012, where the localization of unknown nodes of wireless sensors in the plane, the maximum value $\overline{\varepsilon}$ is 0.009; the localization of unknown nodes of wireless sensors on the surface, the maximum value $\overline{\varepsilon}$ is 0.012. The localization accuracy of the method is high and meets the standard of curve sensor localization accuracy.

To verify the target tracking effect of the method, the target moving trajectory tracking error ℓ_{ERR} is used in the paper during the survival period of the sensing network as an evaluation index, the value of this index ranges from 0 to 1, with larger values indicating poorer tracking effects and smaller values indicating better tracking effects. The formula for ℓ_{ERR} is:

$$\ell_{ERR} = \frac{\sum_{J=1}^{t} \bar{\varepsilon}}{T} \tag{39}$$

Where: T Indicates the number of target motion trajectories.

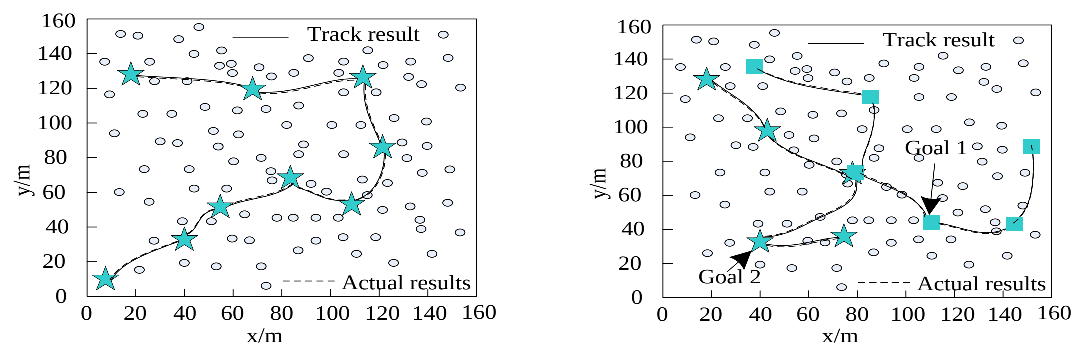
Reference [9], reference [10], reference [11], and reference [12] as the comparison method of the methods in the text, based on the Formula (31) to calculate the five methods in different target distances, for the target tracking error results, the test results are shown in Table V.

TABLE V. RESULTS OF TARGET TRACKING ERROR AT DIFFERENT TARGET DISTANCES (M)

Target distance /m	Reference [9] Method	Reference [10] Method	Reference [11] Method	Reference [12] Method	The method in the text
3	**0.205**	**0.199**	**0.188**	**0.223**	**0.112**
6	**0.211**	**0.205**	**0.194**	**0.255**	**0.105**
9	**0.223**	**0.21**	**0.202**	**0.241**	**0.116**
12	**0.208**	**0.224**	**0.213**	**0.236**	**0.109**
15	**0.213**	**0.213**	**0.206**	**0.228**	**0.135**
18	**0.224**	**0.206**	**0.215**	**0.216**	**0.142**
21	**0.277**	**0.214**	**0.207**	**0.219**	**0.128**
24	**0.219**	**0.222**	**0.196**	**0.254**	**0.131**
27	**0.198**	**0.209**	**0.219**	**0.246**	**0.141**
30	**0.207**	**0.201**	**0.1216**	**0.234**	**0.125**

After analyzing the test results in Table V, it is concluded that under different target distances, five methods are used for target tracking, and the maximum tracking errors of reference [9], reference [10], reference [11], and reference [12] are 0.277m, 0.224m, 0.219m, and 0.255m respectively; The maximum tracking error is 0.142m after target tracking by the method. Hence, this method has a better target tracking effect, because the method in this paper builds a target tracking model based on the sensor node position after accurate positioning, so it can improve the target tracking accuracy.

To further verify the tracking effect of the method on the target, in the constructed wireless sensor network, the method in the paper is used in single target and multi-target (only 2 nodes as an example) tracking, for the tracking results of the method on the two kinds of targets, the test results are shown in Fig. 4.



(a) Single target tracking results.

(b) Multi-target tracking results.

Fig. 4. Target tracking results.

After analyzing the test results in Fig. 4, it is concluded that: in the constructed wireless sensor network, the method in the paper can accurately realize the tracking of the target trajectory when single-target and multi-target tracking are carried out respectively; in single-target tracking, the target trajectory can be accurately determined when the nodes are unevenly distributed; in multi-target tracking, the method can accurately track the two targets when the two nodes overlap and are separated. Under both single-target and multi-target tracking, the tracking results and the actual outcomes exhibit a strong concordance, and the tracking effect is better.

In order to further validate the effectiveness of the proposed method, the positioning errors of our method, basic locust algorithm, references [9], [10], [11], and [12] were tested based on three environments: smart home environment (50m * 50m), campus environment (100m * 100m), and large industrial park environment (200m * 200m). The results are shown in Table VI.

TABLE VI. POSITIONING ERROR RESULTS IN DIFFERENT REAL-WORLD SCENARIOS

Method	Smart home environment		Campus environment		large industrial park environment	
	Average positioning error/m	Positioning error standard deviation/m	Average positioning error/m	Positioning error standard deviation/m	Average positioning error/m	Positioning error standard deviation/m
The method in the text	0.60	0.04	1.31	0.08	2.65	0.15
Basic locust algorithm	0.85	0.07	1.75	0.12	3.25	0.22
Reference [9] Method	0.84	0.06	1.64	0.11	3.15	0.19
Reference [10] Method	0.88	0.08	1.75	0.13	3.31	0.21
Reference [11] Method	0.92	0.09	1.82	0.14	3.44	0.22
Reference [12] Method	0.98	0.15	1.95	0.15	3.67	0.24

As shown in Table VI, the average and standard deviation of the positioning error of our method are lower than those of other methods, indicating that our method can provide more accurate node position information. This indicates that in the same network environment, our method effectively improves the positioning accuracy of wireless sensor nodes by improving the locust algorithm, thereby improving the effectiveness of target tracking. In summary, it can be seen that in different network ranges, the method proposed in this paper can improve the effectiveness of target tracking while ensuring positioning accuracy. Compared with the basic locust algorithm and other methods in the literature, the method proposed in this paper has better performance and higher reliability. This proves the effectiveness of the wireless sensor node localization and target tracking method based on the improved locust algorithm.

Considering that node density can have an impact on the final wireless sensor node localization and target tracking. Therefore, under three different node densities: low density (one node per 25m × 25m area), medium density (one node per 10m × 10m area), and high density (one node per 5m × 5m area), the target tracking accuracy standard deviation of our method, basic locust algorithm, references [9], [10], [11], and [12] were compared. The results are shown in Table VII.

TABLE VII. TARGET TRACKING RESULTS UNDER DIFFERENT NODE DENSITIES

Method	Average standard deviation at low density/m	Average standard deviation at medium density/m	Average standard deviation at high density/m
The method in the text	0.090	0.065	0.048
Basic locust algorithm	0.129	0.098	0.072
Reference [9] Method	0.155	0.124	0.095
Reference [10] Method	0.141	0.106	0.084
Reference [11] Method	0.147	0.113	0.088
Reference [12] Method	0.136	0.101	0.078

According to Table VII, at all node densities, the average standard deviation of our method is significantly smaller than other methods, indicating that our method has better stability and accuracy in localization accuracy. As the node density increases, the average standard deviation of all methods decreases, indicating that an increase in node density helps to improve localization accuracy. In high-density nodes, the average standard deviation of our method is only 0.048m, which is much smaller than other methods, further proving the excellent performance of this method in high-density node environments.

V. CONCLUSION

The node localization problem in wireless sensor networks is an essential and crucial issue need to be solved, and the target tracking is also the wireless sensor network in military and civil and other fields are heavily used, the research centers on the wireless sensor network node localization technology and the single active target tracking technology, for how to enhance the node's self-localization accuracy in the wireless sensor network. The application effect of this method is tested, and the results show that the algorithm has high applicability, can improve the positioning accuracy of sensor nodes, and the tracking effect of target trajectory is good. Based on the above, it can be concluded that the method proposed in this article can accurately obtain the position of unknown nodes in different node distribution states by combining the coarse localization of DV Hop algorithm and the fine optimization of improved locust algorithm. The localization error in both surface and plane is less than 0.012 meters, demonstrating excellent localization performance. In addition, using the probability hypothesis density filtering algorithm, this method can effectively track single and multiple targets in the wireless sensor network model, with a maximum error of only 0.142 meters, verifying its effectiveness in the field of target tracking.

Future research will explore more advanced filtering algorithms and positioning technologies to improve the accuracy and efficiency of target tracking. In addition, considering the challenges in practical applications such as communication interference and node failure, future research will also further focus on the stability and reliability of algorithms in practical environments.

Competing of Interests

The authors declare no competing of interests.

Authorship Contribution Statement

Qin Qi: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.

Tan Songhe: Methodology, Software, Validation.

References

[1] R. Álvarez, J. Díez-González, P. Verde, R. Ferrero-Guillén, and H. Perez, "Combined sensor selection and node location optimization for reducing the localization uncertainties in wireless sensor networks," Ad Hoc Networks, vol. 139, p. 103036, 2023.

[2] S. Rani, H. Babbar, P. Kaur, M. D. Alshehri, and S. H. Shah, "An optimized approach of dynamic target nodes in wireless sensor network using bio inspired algorithms for maritime rescue," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 2, pp. 2548–2555, 2022.

[3] M. Agoramoorthy and I. R. Praveen Joe, "Hybrid cuckoo–red deer algorithm for multiobjective localization strategy in wireless sensor network," International Journal of Communication Systems, vol. 35, no. 4, p. e5042, 2022.

[4] L. Baidar, A. Rahmoun, M. Mihoubi, P. Lorenz, and S. Birogul, "A hybrid Harrison Hawk optimization based on differential evolution for the node localization problem in IoT networks," International Journal of Communication Systems, vol. 35, no. 9, p. e5129, 2022.

[5] A. Achroufene, "RSSI-based geometric localization in wireless sensor networks," J Supercomput, vol. 79, no. 5, pp. 5615–5642, 2023.

[6] M. Nain, N. Goyal, L. K. Awasthi, and A. Malik, "A range based node localization scheme with hybrid optimization for underwater wireless sensor network," International journal of communication systems, vol. 35, no. 10, p. e5147, 2022.

[7] A. M. Khedr, S. S. Rani, and M. Saad, "Optimized Deep Learning for Congestion-Aware Continuous Target Tracking and Boundary Detection in IoT-Assisted WSN," IEEE Sens J, vol. 23, no. 7, pp. 7938–7948, 2023.

[8] H. Mohd Zali, M. K. A. Mahmood, I. Pasya, M. Hirose, and N. Ramli, "Narrowband and wideband EMW path loss in underwater wireless sensor network," Sensor Review, vol. 42, no. 1, pp. 125–132, 2022.

[9] J. Luomala and I. Hakala, "Adaptive range-based localization algorithm based on trilateration and reference node selection for outdoor wireless sensor networks," Computer Networks, vol. 210, p. 108865, 2022.

[10] M. Nain, N. Goyal, L. K. Awasthi, and A. Malik, "A range based node localization scheme with hybrid optimization for underwater wireless sensor network," International journal of communication systems, vol. 35, no. 10, p. e5147, 2022.

[11] P. Tripathy and P. M. Khilar, "An ensemble approach for improving localization accuracy in wireless sensor network," Computer Networks, vol. 219, p. 109427, 2022.

[12] M. Nain, N. Goyal, S. Rani, R. Popli, I. Kansal, and P. Kaur, "Hybrid optimization for fault - tolerant and accurate localization in mobility assisted underwater wireless sensor networks," International Journal of Communication Systems, vol. 35, no. 17, p. e5320, 2022.

[13] R. Arya, "Exploiting perturbed and coalescent anchor node geometry with semidefinite relaxation for sensor network localization," Physical Communication, vol. 52, p. 101606, 2022.

[14] T. K. Mohanta and D. K. Das, "Multiple objective optimization-based DV-Hop localization for spiral deployed wireless sensor networks using Non-inertial Opposition-based Class Topper Optimization (NOCTO)," Comput Commun, vol. 195, pp. 173–186, 2022.

[15] M. Kumar, N. Goyal, R. M. A. Qaisi, M. Najim, and S. K. Gupta, "Game theory based hybrid localization technique for underwater wireless sensor networks," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 11, p. e4572, 2022.

[16] S. Rani, H. Babbar, P. Kaur, M. D. Alshehri, and S. H. Shah, "An optimized approach of dynamic target nodes in wireless sensor network using bio inspired algorithms for maritime rescue," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 2, pp. 2548–2555, 2022.

[17] K. Park et al., "An energy-efficient multimode multichannel gas-sensor system with learning-based optimization and self-calibration schemes," IEEE Transactions on Industrial Electronics, vol. 67, no. 3, pp. 2402–2410, 2019.

[18] E. Suganya and C. Rajan, "An adaboost-modified classifier using particle swarm optimization and stochastic diffusion search in wireless IoT networks," Wireless Networks, vol. 27, no. 4, pp. 2287–2299, 2021.

[19] A. A. Rizvi, D. Yang, and T. A. Khan, "Optimization of biomimetic heliostat field using heuristic optimization algorithms," Knowl Based Syst, vol. 258, p. 110048, 2022.

[20] K. P. Rani, P. Sreedevi, E. Poornima, and T. S. Sri, "FTOR-Mod PSO: A fault tolerance and an optimal relay node selection algorithm for wireless sensor networks using modified PSO," Knowl Based Syst, vol. 272, p. 110583, 2023.

[21] X. Cheng, Y. Sun, W. Zhang, Y. Wang, X. Cao, and Y. Wang, "Application of deep learning in multitemporal remote sensing image classification," Remote Sens (Basel), vol. 15, no. 15, p. 3859, 2023.

[22] G. A. Castillo, B. Weng, W. Zhang, and A. Hereid, "Reinforcement learning-based cascade motion policy design for robust 3d bipedal locomotion," IEEE Access, vol. 10, pp. 20135–20148, 2022.

[23] J. Guerrero, A. Chemori, J. Torres, and V. Creuze, "Time-delay high-order sliding mode control for trajectory tracking of autonomous underwater vehicles under disturbances," Ocean Engineering, vol. 268, p. 113375, 2023.

[24] T. Ishikawa, K. Hamamoto, and K. Kogiso, "Trajectory tracking switching control system for autonomous crawler dump under varying ground condition," Autom Constr, vol. 148, p. 104740, 2023.

[25] M. Yamagami, L. N. Peterson, D. Howell, E. Roth, and S. A. Burden, "Effect of handedness on learned controllers and sensorimotor noise during trajectory-tracking," IEEE Trans Cybern, vol. 53, no. 4, pp. 2039–2050, 2021.

[26] M. Najimi and V. S. Sadeghi, "Energy - efficient compressive sensing for multi - target tracking in wireless visual sensor networks," International Journal of Communication Systems, vol. 35, no. 16, p. e5307, 2022.