Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.047996

ARTICLE

A Layered Energy-Efficient Multi-Node Scheduling Mechanism for Large-Scale WSN

Xue Zhao, Shaojun Tao, Hongying Tang, Jiang Wang* **and Baoqing Li***

Science and Technology on Microsystem Laboratory, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, 200050, China

*Corresponding Authors: Jiang Wang. Email: jiang.wang@mail.sim.ac.cn; Baoqing Li. Email: sinoiot@mail.sim.ac.cn

Received: 24 November 2023   Accepted: 13 March 2024   Published: 25 April 2024

ABSTRACT

In recent years, target tracking has been considered one of the most important applications of wireless sensor network (WSN). Optimizing target tracking performance and prolonging network lifetime are two equally critical objectives in this scenario. The existing mechanisms still have weaknesses in balancing the two demands. The proposed heuristic multi-node collaborative scheduling mechanism (HMNCS) comprises cluster head (CH) election, pre-selection, and task set selection mechanisms, where the latter two kinds of selections form a two-layer selection mechanism. The CH election innovatively introduces the movement trend of the target and establishes a scoring mechanism to determine the optimal CH, which can delay the CH rotation and thus reduce energy consumption. The pre-selection mechanism adaptively filters out suitable nodes as the candidate task set to apply for tracking tasks, which can reduce the application consumption and the overhead of the following task set selection. Finally, the task node selection is mathematically transformed into an optimization problem and the genetic algorithm is adopted to form a final task set in the task set selection mechanism. Simulation results show that HMNCS outperforms other compared mechanisms in the tracking accuracy and the network lifetime.

KEYWORDS

Node scheduling; pre-selection; target tracking; WSN

1 Introduction

1.1 Background

Wireless sensor network (WSN) is comprised of many miniature sensor nodes that are deployed in a monitored area. These nodes form a self-organizing network through multi-hop wireless communication. In military scenarios, WSN is often applied for target tracking. However, sensor nodes are mostly deployed in harsh field environments and difficult to replace batteries. Therefore, many researchers aim to extend the lifetime of WSN from different perspectives, such as regulating the data reporting process at the zone level through a general event-thresholding model in heterogeneous WSN [1], obtaining suitable routing protocols for WSN [2], finding efficient relay selection [3] and so on. In this paper, we mainly aim to extend the network lifetime from the perspective of scheduling nodes to remove redundancy in WSN for target tracking. The dilemma is that improving tracking accuracy requires more nodes to track the target and consumes more energy [4,5]. Therefore, it is important to investigate how to balance tracking accuracy and energy consumption to prolong the network lifetime while obtaining favorable tracking performance [6].

To save energy, nodes are put in a sleep state where only the sensing module is turned on, while the computation and communication modules are turned off until a target is sensed. After being awakened by the target, the nodes enter the active state and are organized into a cluster to track the target. How to select the appropriate task nodes from the awakened nodes to perform the tracking task, obtain better tracking accuracy, and extend the network lifetime is the research motivation of this paper.

1.2 Contributions

The main contributions of this paper are summarized below:

Firstly, we propose a cluster head (CH) election mechanism, which innovatively introduces the changing trend of the distance between the target and the node to obtain the optimal CH. This reduces the frequency of CH rotation and thus reduces energy consumption.

Secondly, we develop a pre-selection mechanism that pre-selects a portion of cluster members to apply for tracking the target. Some unsuitable nodes are prevented from transmitting data to CH, and some transmission consumption is reduced.

Finally, a mathematical problem is formulated and solved to obtain the final task set from the candidate task set. The tracking accuracy, the total energy consumption, and the balance degree of the residual energy of cluster members are jointly considered. Simulation experiments are conducted and the results show that the proposed heuristic multi-node collaborative scheduling mechanism (HMNCS) has the most superior performance.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 introduces the system model. The proposed protocol is explained in Section 4. Section 5 includes simulation experiments and performance analysis. Finally, Section 6 concludes the paper.

2 Related Work

2.1 Existing Researches

In WSN for target tracking, there are many researchers dedicated to research on node scheduling. In [3], Energy proficient clustering (EEPC) was proposed to reduce energy consumption. The placement and energy level are used to select the CH and the relay nodes are obtained by calculating particle fitness value with their velocity and location. In [7], researchers proposed an improved Unscented Kalman Filter (UKF) that considers the uncertainty of the network and unpredictability of target motion. They utilized a genetic adaptive multi-objective algorithm to obtain a better set of task nodes without knowing the number of sensors to be selected. In [8], a scheduling mechanism based on a Type-2 Fuzzy Logic System (T2FLS) and a software agent approach was developed. The mechanism integrates energy, the number of neighboring nodes, and available bandwidth. In [9], researchers applied target state dynamics to predict the target trajectory and used the particle swarm algorithm for network optimization. A coordinated tracking mechanism was designed to save overall network energy consumption. In [10], Zhu et al. proposed an environment-adaptive event-driven robust set affiliation fusion estimation mechanism to schedule the required number of nodes to track the target. Experiments demonstrated that it reduces energy consumption and computational complexity while ensuring localization accuracy. In [11], the sensor selection problem was transformed into a multi-objective optimization framework, and a new mutual information upper bound (MIUB)–based sensor selection mechanism was proposed. The authors studied several sensor selection strategies, such as how to obtain a task set when minimizing the number of selected sensors or minimizing the difference between the performance obtained when all sensors transmit data and the performance obtained when only task nodes transmit data. In [12], researchers proposed a GPB1-UIF (the Generalized Pseudo-Bayesian estimator of first order and the unscented information filter) algorithm with estimation feedback and an information quality metric framework to obtain the task set using a multi-objective optimization strategy. In [13], Fu et al. analyzed the target detection probability and residual energy of sensors, proposing a new energy-balanced sensor node scheduling algorithm to balance tracking quality and network lifetime. In [14], Alagha et al. considered factors such as residual energy, power cost, and data confidence, proposing a new data-driven active node selection mechanism that uses genetic and greedy algorithms to select the optimal task set. In [15], the scheduling problem was transformed into a partially observable Markov decision process (POMDP) optimal policy problem by analyzing the intrinsic relationship between tracking performance and energy consumption. A dynamic cluster membership scheduling (DCMS) algorithm was proposed to solve the trade-off problem. In [16], spatial and temporal correlations among nodes were utilized instead of just spatial correlation. With the support of reconstructed data from the previous moment, a constrained convex relaxation plus rounding method was applied to obtain the task set, and the proposed mechanism was experimentally shown to extend the network lifetime. In [17], an accuracy enhancement algorithm and an active node selection algorithm based on the robust interval fusion preference aggregation (IF&PA) method were developed to significantly extend the network lifetime while providing high-accuracy measurements. Measurement uncertainty can be reduced even with a small number of sensors. In [18], game theory was used to reduce redundancy, and only one active node was periodically selected in each region to transmit data. Extensive simulation results showed that the proposed mechanism effectively reduced energy waste and extended the network lifetime. Finally, in [19], a cooperative node selection mechanism considering the spatial correlation among nodes was proposed to dynamically select the optimal set of nodes for the tracking task, significantly reducing energy consumption without degrading tracking accuracy. In [20], the proposed mechanism selects the set of sensors based on the location of the target and the mobile sink, thereby extending the network lifespan and reducing complexity. In [21], the researchers proposed an innovative glowworm swarm optimization technique (SS-GSO) to appropriately divide the clusters and obtain the optimal CHs. The residual energy, total energy consumption, and other factors are considered to efficiently schedule the nodes. The simulation results show that SS-GSO performs better in energy consumption and lifetime compared with other techniques. In [22], the researchers developed a long-term dynamic resource allocation algorithm to obtain the optimal resource scheduling solution for accurate and consecutive tracking. The algorithm offloads time-sensitive tasks to sink to improve computing efficiency.

2.2 Remaining Weaknesses

It can be seen that researchers have explored different ways to eliminate redundancy from multiple perspectives in Section 2.1. However, certain limitations still exist within the current approaches. Firstly, there is a lack of comprehensive consideration for the rotation process. Since target tracking is an ongoing task, frequent rotations can lead to substantial energy wastage. Secondly, in the majority of existing mechanisms, all cluster members are required to transmit pertinent information to the CH for selection, without taking into account the energy consumption involved in this process. In reality, if certain criteria can be established for cluster members to halt the transmission of information from nodes that do not contribute to prolonging the network lifespan, it could result in significant energy savings.

3 System Model

3.1 Network Model

Without losing generality, we consider an L ∗ L monitored area where N nodes are randomly deployed. The target enters the area from the boundary and moves within the area according to certain rules until leaving the network. To save energy and continuously monitor the target, before the node perceives the target, only the sensing module is turned on, and the computing and transmission modules are turned off, that is, the sensor node is in a dormant state. After sensing the target, the node is awakened and enters the wake-up state, which means all modules are turned on. After awakened nodes are self-organized into a cluster, CH collects and fuses the data about the target from task nodes, and then transmits it to the sink. Here the assumptions are as follows:

1. Sink is in the center of the region and has unlimited energy.

2. The transmission radius of the sensor is more than twice the sensing radius to ensure that all nodes in the cluster can communicate with each other.

3. Nodes can obtain their positions through the global positioning system (GPS) or any other positioning algorithm.

Fig. 1 shows the schematic diagram of the network scene.

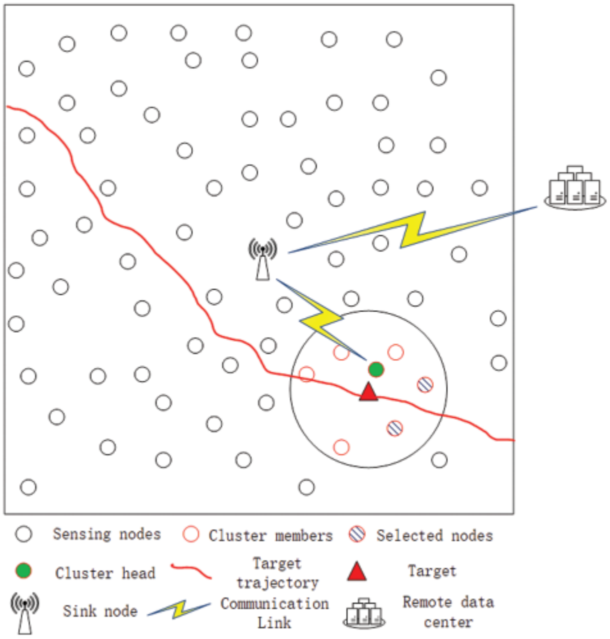


Figure 1: The schematic diagram of the network scene

3.2 Target Movement Model

We denote the state of the target at k^{th} moment is $X_K = \left[x_k, v_k^x, y_k, v_k^y\right]^T, k = 1, 2, 3, \cdots$, where x_k and y_k are the position coordinates of the target in the horizontal and vertical directions, respectively, and v_k^x and v_k^y are the velocities of the target in the horizontal and vertical directions, respectively.

The state transition equation of the target is as follows:

$$X_k = F \cdot X_{k-1} + W_{k-1} \tag{1}$$

$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$, W_{k-1} is the process noise at $k-1^{th}$ moment and W_k is a zero mean Gaussian white noise. The covariance matrix of W_k is C. The noise intensity matrix is $Q = diag\left([q_1, q_2]\right)$, and the process noise driving matrix is $G = \begin{bmatrix} \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix}$, so $C = G \cdot Q \cdot G^T$. The observation equation of the node is as follows:

$$y_k^i = h\left(X_k\right) + v_k^i \tag{2}$$

$$h\left(X_k\right) = \sqrt{\left(x_k - x_i\right)^2 + \left(y_k - y_i\right)^2} \tag{3}$$

$v_k^i \sim N\left(0, R\right)$, and v_k^i is the observation noise of node i at k^{th} moment. The observation noise and the process noise are independent of each other. In this paper, UKF is used to estimate the state of the target [10].

3.3 Energy Consumption Model

The most common energy consumption model is adopted, as shown below [13]:

$$E_{Tx}\left(l, d\right) = \begin{cases} l \cdot E_{elec} + l \cdot \varepsilon_{fs} \cdot d^2, & d < d_0 \\ l \cdot E_{elec} + l \cdot \varepsilon_{amp} \cdot d^4, & d \geq d_0 \end{cases} \tag{4}$$

$$E_{Rx}\left(l\right) = l \cdot E_{elec} \tag{5}$$

$$d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{amp}} \tag{6}$$

where ε_{fs} and $\varepsilon_{\mathrm{amp}}$ are the energy consumed by each bit of data through the power amplifier for the free space and multi-path wireless channel model, respectively, and d_0 is the threshold distance between the two models. E_{elec} is the energy consumed by each bit of data transmitted or received by the sensor. d is the distance between the sender and the receiver.

4 The Proposed Protocol

The proposed protocol HMNCS will be described in detail in this section. Section 4.1 describes the process of target tracking. Section 4.2 presents the CH selection mechanism. Section 4.3 includes the pre-selection mechanism. Finally, the task set selection mechanism is introduced in Section 4.4.

4.1 The Overall Framework of HMNCS

Throughout the journey of the target within the network, a set of strategies has been devised to enhance tracking accuracy and prolong the network lifespan. These strategies encompass the CH selection mechanism, the pre-selection mechanism, and the task set selection mechanism, collectively referred to as the heuristic multi-node collaborative scheduling mechanism (HMNCS), illustrated in Fig. 2.

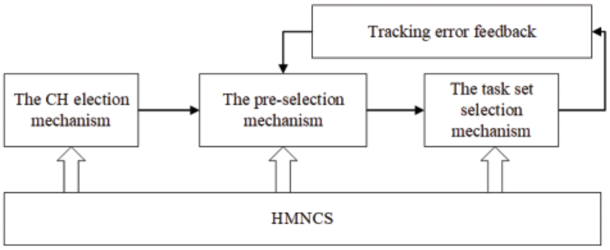

Tracking error feedback
The CH election mechanism
The pre-selection mechanism
The task set selection mechanism
HMNCS

Figure 2: The block diagram of HMNCS

The target enters WSN from the area boundary, triggering the awakening of the node responsible for sensing the target. In the initial round, since no node possesses complete information about the cluster at the outset, the first awakened node assumes the role of CH, while the remaining nodes become cluster members. If an awakened node fails to receive the announcement message transmitted by other nodes, it broadcasts its announcement message, declaring itself as the temporary CH, and awaits join messages from other nodes. However, if the awakened node successfully receives the announcement message from other nodes, it sends a join message to the temporary CH, seeking admission into the cluster. The temporary CH responds by sending a confirmation message to the awakened nodes, verifying their inclusion in the cluster. Subsequently, the temporary CH employs the CH selection mechanism to determine the actual CH and broadcasts the information in the cluster. Once the cluster members are aware of the true CH, the temporary CH relinquishes its role and becomes a cluster member.

After a cluster is formed, each node obtains the number of awakened nodes in its neighborhood by exchanging information between itself and neighbor nodes. Then it determines whether it applies to CH to be a task node according to the pre-selection mechanism, which will be introduced in Section 4.3. If it meets the pre-selection conditions, it will send an applying message including the residual energy and the observation of the target state to CH. CH derives the task set by applying the task set selection mechanism, which will be described in Section 4.4, and broadcasts the scheduling result in the cluster. Those selected task nodes keep awake to track the target, while those non-task nodes enter the dormant state to save energy. Task nodes keep tracking and send observations to CH. CH fuses the collected information, uses UKF to obtain the estimated target state, and then sends it to the sink. In the second and subsequent rounds, those nodes not aware of the target get out of the cluster, and those newly awakened nodes join the cluster. And as the target moves, CH and the task set may need to rotate. CH determines whether it needs rotating or not. If it does, CH obtains the new CH according to the CH selection mechanism and broadcasts it in the cluster. The newly designated CH assumes the responsibilities of the CH, while the former CH transitions into a cluster member or departs from the cluster entirely. CH also needs to determine whether the task set needs rotating or not. When the task set needs rotating, CH gets the new task set by using the task set selection mechanism and broadcasts it. Task nodes diligently track the target, while non-task nodes enter a dormant state to conserve energy. This process iterates until the target leaves the monitored area.

In this paper, we assume that the target moves slowly, it is usually a long time from a task set selected to it is rotated. During the process, those nodes that meet the pre-selection conditions only send an apply message once. Hence the energy consumption of applying accounts for a small part of total consumption and is acceptable.

4.2 The CH Election Mechanism

The proposed CH election mechanism is shown in Algorithm 1.

Algorithm 1: The CH election algorithm

```
Step 1: Compute the average residual energy of cluster members according to Eq. (7);
Step 2: Score the nodes whose residual energy is more than the average residual energy $e_{th}$ in Eq. (7) according to Eq. (8);
Step 3: The node with the highest score becomes CH.
```

$$e_{th} = \frac{1}{n}\sum_{i=1}^{n} e_i^{res} \tag{7}$$

$$score\left(i\right) = \sum_{i=1}^{n} \frac{1}{1+\log\left(dis\left(i,v\right)+1\right)} + \frac{1}{1+\log\left(dis\left(i,\sin k\right)+1\right)} + \gamma_i \tag{8}$$

$$\gamma_i = \begin{cases} \gamma_0 & \text{if } dis\left(i,\text{target}\right)_k \le dis\left(i,\text{target}\right)_{k-1} \ \ and\ k > 1 \\ 0 & \text{if } dis\left(i,\text{target}\right)_k > dis\left(i,\text{target}\right)_{k-1} \ \ or\ \text{k} = 1 \end{cases} \tag{9}$$

e_i^{res} is the residual energy of i^{th} node, and e_{th} is the average residual energy of cluster members. $dis\left(i,v\right)$ is the Euclidean distance between i^{th} node and v^{th} node, and $dis\left(i,\sin k\right)$ is that between i^{th} node and sink. γ_i is used to indicate whether the target is getting closer or farther from the node. If they are getting closer or keep the same distance, γ_i is equal to a positive constant γ_0. If it is getting farther, γ_i is equal to 0. In the beginning, it is impossible to obtain the trend of target movement, so it is specified that γ_i of nodes in the first round after the target appears are equal to 0.

When CH meets the constraints in Eq. (10) or Eq. (11), it needs rotating.

$$e_{CH} < \lambda \cdot e_{th}, \lambda \in (0,1] \tag{10}$$

$$dis\left(CH, target\right) > r_{sense} \tag{11}$$

In Algorithm 1, the first and second parts of Eq. (8) are included to make the transmitting distance between cluster members and CH short to reduce energy consumption. The average residual energy of CLUSTER MEMBERs is used as a threshold to avoid nodes with low energy becoming CH. In Eq. (10), when the residual energy of CH is less than a certain value, CH needs to rotate to make sure the CH node has enough energy to complete its task in the next round. It should be noted that the energy consumption of a CH exceeds that of a cluster member. If nodes with low residual energy were to become CH, they would have a short lifespan, leading to an accelerated emergence of coverage holes, an increased frequency of CH rotation, and energy wastage [10]. Moreover, the relative position between the target and the node is also taken into account. Nodes that are approaching the target are more likely to become CH, thus delaying CH rotation caused by failure to sense the target.

4.3 The Pre-Selection Mechanism

Considering redundancy, all nodes do not have to participate in tracking tasks. Instead, nodes transmit pertinent information to CH to express their interest in becoming task nodes, and CH selects a set of task nodes from these applicants. To minimize energy consumption during the application process and alleviate the computational load on the CH, a pre-selection mechanism is developed. This mechanism enables each node to decide whether or not to apply based on factors such as residual energy, estimation error, node density, and other relevant criteria.

The requirements which nodes need to satisfy to apply are as follows:

$$k > 1 \tag{12}$$

$$\left|y_k - y_k^r\right| > \delta \tag{13}$$

$$y_k^r = \begin{cases} \hat{y}_k & \textit{if the node ever sent message} \\ 0 & \textit{otherwise} \end{cases} \tag{14}$$

$$\frac{e^c}{e^{res}} < b \cdot \frac{N_k^d}{N_k^s} \tag{15}$$

$$N_k^d = c \cdot \varepsilon_{k-1} \tag{16}$$

where y_k is the observed Euclidean distance of a wake-up node in k^{th} round, and y_k^r is its reference value in the k^{th} round. If the node has ever been selected as a task node of the current target, y_k^r is equal to $\hat{y}_k$, and $\hat{y}_k$ is the observed Euclidean distance that the node transmitted to CH last time. If the node has not been selected as a task node of the current target, y_k^r is equal to 0. e^c is the expected energy consumption in the current round if the node performs the task. e^{res} is the residual energy of the node. N_k^d is the number of reference neighbor nodes in k^{th} round, which can be obtained by information exchange in the neighborhood. N_k^s is the number of awakened neighbor nodes.

Eq. (13) requires that the absolute difference between the observation y_k and the reference y_k^r is greater than δ. This is to avoid some unnecessary transmission when the target moves slowly and the observation changes slightly. The left of Eq. (15) is the ratio of expected energy consumption e^c to residual energy e^{res} of the node. The right of Eq. (15) is b times the ratio of the number of awakened neighbor nodes N_k^s to the number of reference neighbor nodes. If N_k^s is relatively small, each awakened node in this neighborhood is supposed to be more likely to apply to give CH more choices to get enough information about the target. N_k^d is positively correlated with the estimation error ε_{k-1} of the previous round. If ε_{k-1} is large, more nodes are supposed to get involved in tracking. Therefore, the nodes that have lower expected energy consumption, higher residual energy, and less awakened neighbor nodes are more likely to apply.

The nodes that meet the pre-selection requirements apply to CH to become a task node, and those nodes that do not meet the conditions turn into a dormant state directly until the next task set rotation or is awakened by the next target.

The flow chart for determining whether to apply is shown in Fig. 3.

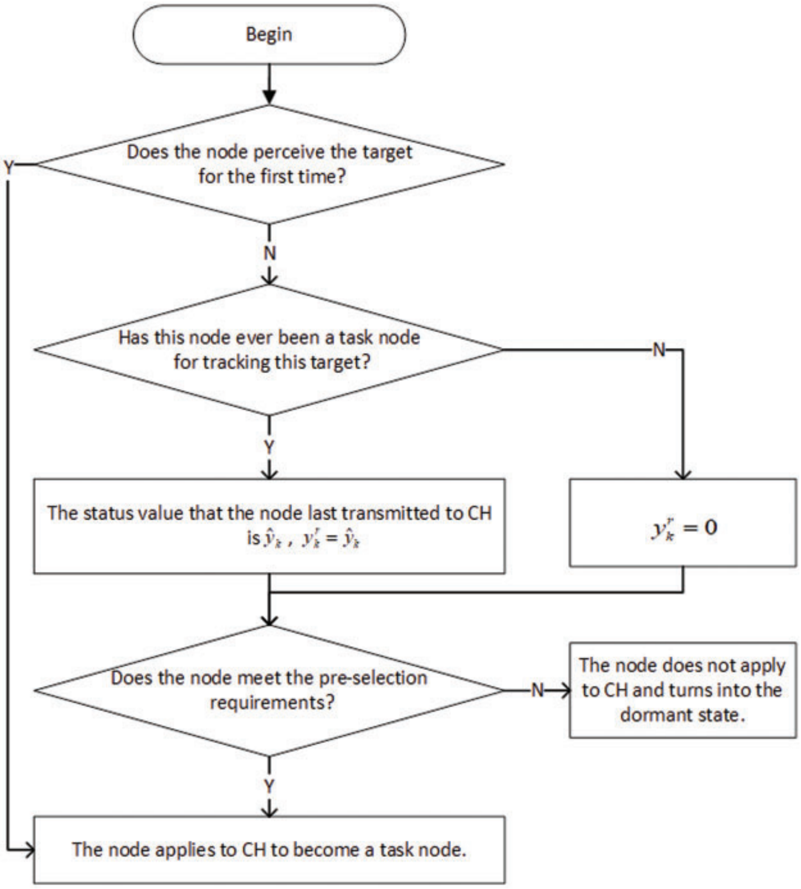

Begin
Does the node perceive the target for the first time?
Y
N
Has this node ever been a task node for tracking this target?
N
Y
The status value that the node last transmitted to CH is $\hat{y}_k$, $y_k^r = \hat{y}_k$
$y_k^r = 0$
Does the node meet the pre-selection requirements?
N
The node does not apply to CH and turns into the dormant state.
Y
The node applies to CH to become a task node.

Figure 3: The flowchart of the pre-selection mechanism

4.4 The Task Set Selection Mechanism

The selection of the task node set takes into account both the estimation accuracy and the network lifespan, where the latter is influenced by the total energy consumption and the energy balance level. In this paper, UKF is employed to estimate the true state of the target. The trace of the error covariance matrix in UKF serves as an indicator of the average square error in state estimation [10]. The total energy consumption is the sum of the energy consumption of each node in WSN, and the energy balance level is measured by the variance of the residual energy sequence of nodes in the cluster. Their expressions are as follows:

$$\varepsilon_{TS} = tr\left(P\right) \tag{17}$$

$$E_{TS}^{C} = \sum_{i} e_i^c \tag{18}$$

$$\mathrm{var}_{TS} = \frac{1}{n}\sum_{i}\left(e_i - \frac{1}{n}\sum_{i} e_i\right)^2 \tag{19}$$

where ε_{TS} is the average square error obtained by using observations of task nodes to estimate the real target state through UKF, and P is the error covariance matrix in UKF. e_i^c is the expected energy consumption of i^{th} node if it becomes a task node. E_{TS}^{C} and var_{TS} are the expected total energy consumption of the network and the expected variance of the residual energy sequence of nodes in the cluster, respectively, after the task set TS completes the tracking task in the current round.

The optimization objective function for selecting the task set is as follows:

$$\begin{aligned}
&\min\left[\alpha\cdot\tilde{\varepsilon}_{TS}+\beta\cdot\tilde{E}_{TS}^{c}+\chi\cdot\widetilde{\mathrm{var}}_{TS}\right]\\
&\alpha+\beta+\chi=1\\
&0<\alpha,\beta,\chi<1
\end{aligned} \tag{20}$$

α,β,χ are the weights of average square estimation error, energy consumption, and energy balance level, respectively. $\tilde{\varepsilon}_{TS}$, $\tilde{E}_{TS}^{C}$, $\widetilde{\mathrm{var}}_{TS}$ are their normalized values, respectively.

Eq. (20) is an optimization problem. If the exhaustive algorithm is used to get the results, there will be $C_m^1+C_m^2+\cdots+C_m^m=2^m-1$ candidate task sets (assuming there are m nodes in the cluster), and the number increases with m at an exponential rate. Therefore, it is necessary to apply an algorithm with low complexity to solve this problem.

In this paper, the genetic algorithm is adopted. CH is always in the task set, and task nodes are selected from those applying nodes. Hence the length of the chromosome is equal to the number of applying nodes, and each gene on the chromosome represents a node. Each node has two states: Selected and unselected, so binary encoding can be perfectly applied to this case. Each gene has two values, 1 for selected and 0 for unselected. Each chromosome corresponds to a task set. Suppose that there are eight applying nodes whose numbers are 11, 32, 45, 8, 55, 135, 146, and 16, respectively. If the chromosome code is "01110100", the second, third, and sixth nodes are selected, that is, the corresponding task set is 32, 45, 8, 135, as shown in Fig. 4.

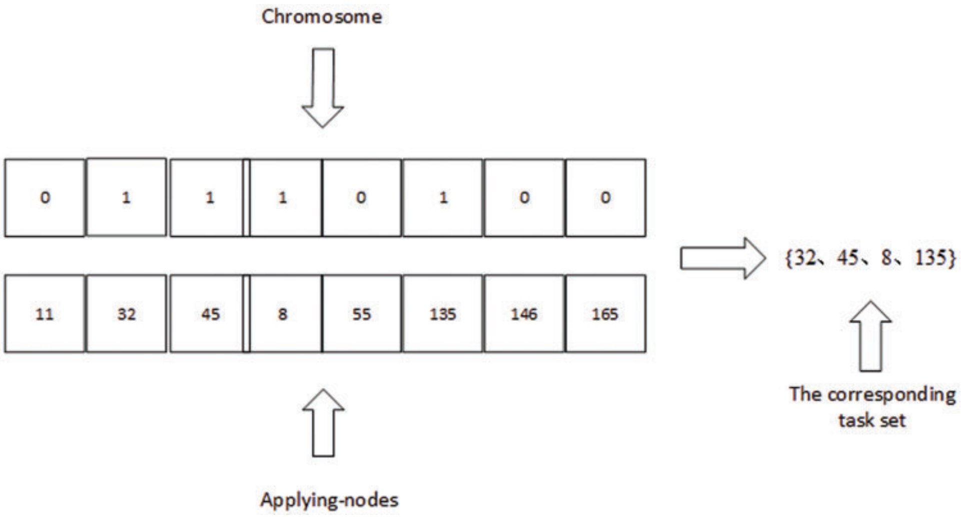


Figure 4: The comparison diagram of chromosomes and task sets

According to the objective function, the fitness function of the candidate task set is set, as shown in Eq. (21).

$$\begin{aligned} & f\left(TS\right) = \alpha \cdot \tilde{\varepsilon}_{TS} + \beta \cdot \tilde{E}^{c}_{TS} + \chi \cdot \widetilde{\mathrm{var}}_{TS} \\ & \alpha + \beta + \chi = 1 \\ & 0 < \alpha, \beta, \chi \ < 1 \end{aligned} \tag{21}$$

Due to significant differences in absolute values, it is necessary to normalize the average square estimation error, energy consumption, and variance. Normalization is carried out for all chromosomes in the population. The task set selection algorithm is shown in the Algorithm 2.

Algorithm 2: The task set selection algorithm

```
Input: The set  of applying nodes;
Output: The task set;
Step 1: Initialize the population based on the number of applying nodes;
Step 2: Calculate the fitness value of each individual according to Eq. (21), and the number of iterations is 1;
Step 3: Has the number of iterations exceeded the maximum value? If it has, go to step 10. If it has not, execute step 4;
Step 4: According to the roulette principle, n descendants are selected and the population is updated. The individuals with small fitness values are more likely to be selected;
Step 5: For the adjacent odd and even individuals in the new population (such as the first and the second, the third and the fourth, excluding the second and the third, and so on), determine whether to cross by the cross probability. If so, randomly select the crossing position to obtain two new descendants; If they do not cross, the two individuals become descendants. Update population;
Step 6: For each individual in the population, determine whether it is mutated by the mutation probability. If it is mutated, randomly select a gene to change to obtain a descendant; If there is no mutation, the individual directly becomes a descendant. Update population;
Step 7: Calculate the fitness value of each individual in the population;
Step 8: Select the individual with the lowest fitness value in the population and record it as the best individual in the current round;
Step 9: Increase the number of iterations by 1, and go to step 3;
Step 10: Select the individual with the lowest fitness value from the best individuals in all iteration rounds to become the best individual obtained by genetic algorithm;
Step 11: Convert the optimal individual into a task set, which is the  final task set.
```

5 Simulation Experiments and Performance Analysis

5.1 Simulation Scenarios and Parameter Settings

In this section, the performance of HMNCS, the dynamic chain-based collaboration mechanism (DCBC) [23], the greedy balance replace heuristic algorithm (GBRHA) [13] and the node scheduling algorithm for all-directional intrusion detection (NSAID) [24] are evaluated on MATLAB. In DCBC, the target tracking task is completed by forming a dynamic tracking chain around the target. GBRHA is an energy-balanced sensor node scheduling algorithm. By analyzing the target detection probability and the residual energy of the sensor nodes, multiple sensor nodes are scheduled for cooperative target tracking. In NSAID, the monitored region is divided into the outermost and inner regions, which can effectively save energy while ensuring the quality of all-around detection.

In the simulation, the monitored area is set to 300 m ∗ 300 m, and the sink is in the center of the area, with infinite energy. To evaluate the performance of HMNCS under different deployment densities, experiments with 400, 600, and 800 nodes are carried out. The sensing radius of nodes is set to 40 m, and the transmission radius is set to twice the sensing radius to ensure that all nodes in the cluster can communicate with each other. The sampling interval is 1 s, and every 5 s is a round. The target enters the network from the random position of a boundary and moves according to the state transition Eq. (1), until it leaves the network. Fusion energy consumption means the energy consumed by fusing received data packets and its own data packets. The settings of simulation parameters are shown in Table 1.

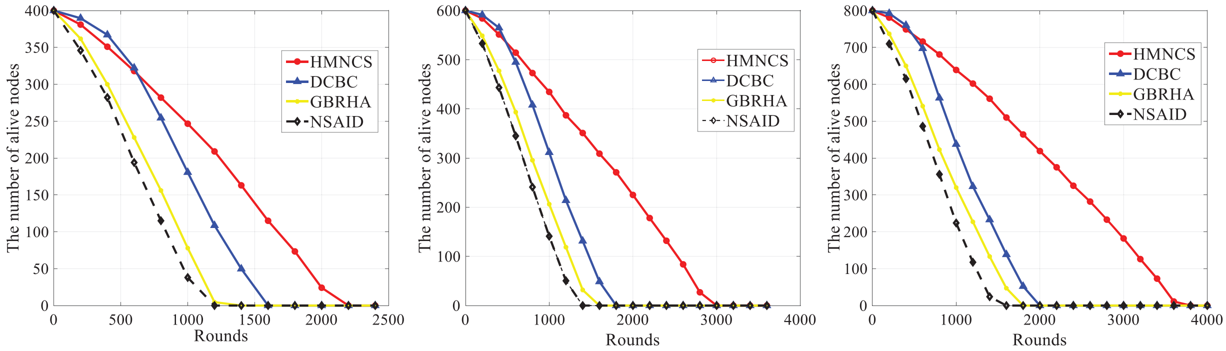
Table 1: The settings of simulation parameters

Parameter	Value
Network scale	300 m ∗ 300 m
The sink coordinates	(150 m, 150 m)
Sensing radius	40 m
Transmission radius	80 m
The energy consumption of fusion	0.0002J/packet
Initial energy	[0.1, 0.2, 0.3] J
Delta_w	10^(−3)
Variance of observation noise R	1
Weight of normalized energy consumption β	0.5
Weight of normalized variance x	0.2
e_{elec}	1
ε_{fs}	0.5
ε_{amp}	0.2
The length of the control packet	48 ∗ 8 bits
The length of the data packet	600 ∗ 8 bits
The number of nodes	[400, 600, 800]
Q	delta w ∗ diag{1,1}
The sampling interval T	1 s
Weight of normalized estimation error	0.3
Fixed parameter about the movement trend	1

The performance criteria used in this paper include the number of alive nodes, the total energy consumption, the tracking error, and the network lifespan. The lifespan can be usually measured in terms of FND (first node death), HND (half node death), and LND (last node death). However, in large-scale WSN, the death of the first node has little effect on the connectivity and coverage of the network, so HND and LND are adopted. The average value and variance of the tracking error sequence from deployment to the round when the last node runs out of energy are calculated to indicate the tracking performance.

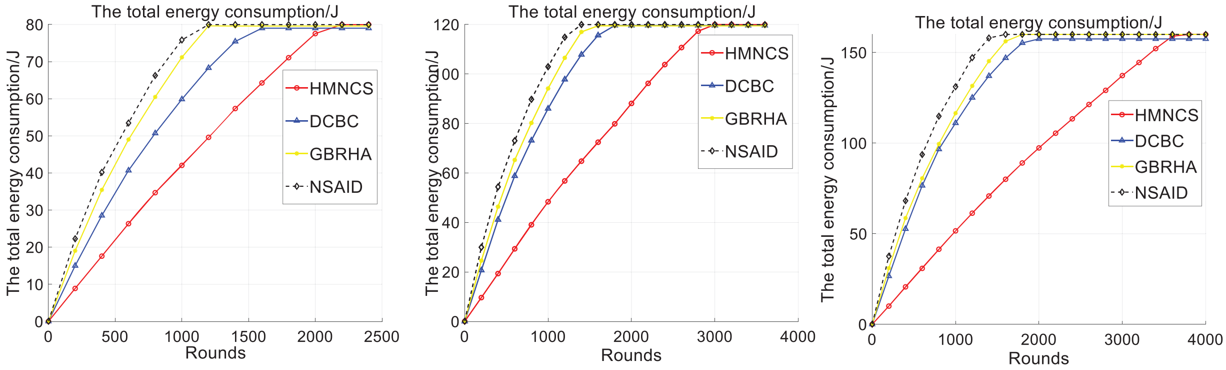
5.2 Performance Analysis

The curves of the number of alive nodes and the total energy consumption of HMNCS, DCBC, GBRHA, and NSAID, when 400, 600, and 800 nodes are deployed in the network, are plotted in Figs. 5 and 6, respectively. In the long term, the number of alive nodes of HMNCS is higher than that of other compared mechanisms after the same rounds with different node deployment densities. In other words, the nodes of HMNCS die slower than those of other mechanisms, and the total energy consumption of HMNCS is the lowest after the same rounds. This is because HMNCS uses various methods to reduce energy consumption. Firstly, in the CH selection mechanism, not only the residual energy of nodes, the distance to other nodes, the distance to the sink, etc., are considered, but also the relative movement trend between the target and each node is taken into account. The nodes that are closer and closer to the target are more likely to be CH, which reduces the rotation energy consumption. Further, using the event trigger mechanism avoids a part of applications and saves energy. Taking energy consumption and energy balance factors into account when the task set is finally obtained, all these operations result in lower energy consumption and longer network lifetime. As the deployment density of nodes increases, it is found that the advantage of HMNCS over the compared mechanisms increases, indicating that HMNCS is more suitable for scenarios with high node density.



(a) 400 nodes (b) 600 nodes (c) 800 nodes

Figure 5: The number of alive nodes under different densities



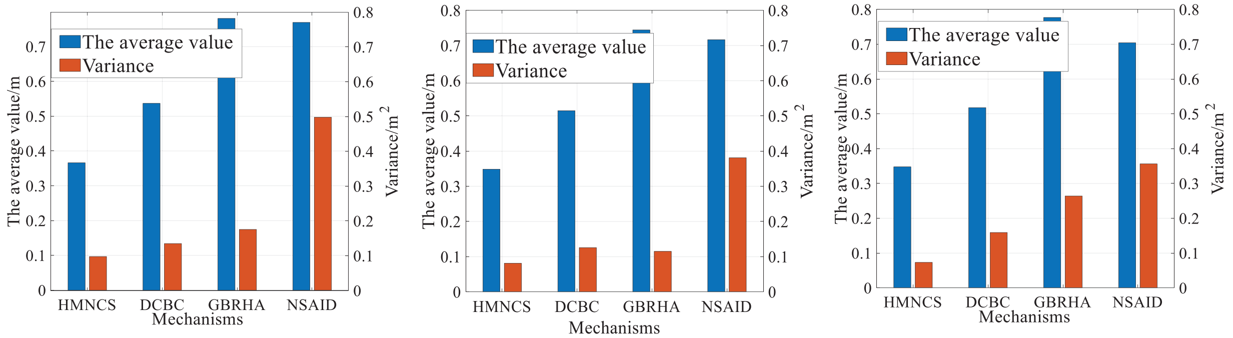
(a) 400 nodes (b) 600 nodes (c) 800 nodes

Figure 6: The total energy consumption under different densities

However, in the early stage after deployment, the number of alive nodes of HMNCS is slightly lower than that of DCBC, and the gap decreases with the increase of node deployment density. In

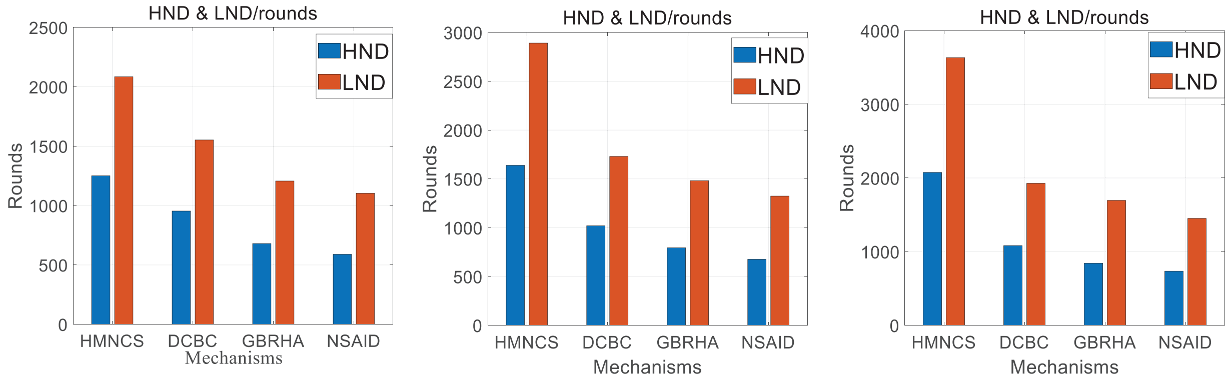
HMNCS, after the target appears, a cluster is formed in WSN. Then CH and a task set are selected, and only some nodes track the target, while the rest of the CLUSTER MEMBERs enter the dormant state. In DCBC, the nodes that sense the target form a tracking chain, and the chain head is selected. Data of all nodes is transmitted to the chain head from both ends of the tracking chain node-by-node and fused. Each node in the cluster transmits and fuses data, but the transmission distance is short in DCBC. In HMNCS, only the task set nodes in the cluster transmit data directly to CH, but the transmission distance is longer than that of DCBC. In DCBC, the energy consumption of each node is low, so nodes die slowly in the beginning. However, there are so many nodes participating in data transmission that the total energy consumption is large. In HMNCS, task nodes consume more energy, while non-task nodes hardly consume energy. As a result, the number of alive nodes in the network is slightly lower than that of DCBC in the early stage after deployment, but with the increase of rounds, the advantages of HMNCS are gradually emerging, and the number of alive nodes is the highest.

Fig. 7 shows the average tracking error and variance of tracking error from deployment to the round of last node death for 400, 600, and 800 nodes. Fig. 8 shows HND and LND under three deployment densities. The average value and variance of tracking error of HMNCS are lower than those of compared mechanisms, and HND and LND are the greatest, i.e., HMNCS is the best in terms of tracking performance and network lifetime.



(a) 400 nodes (b) 600 nodes (c) 800 nodes

Figure 7: The tracking error under different densities



(a) 400 nodes (b) 600 nodes (c) 800 nodes

Figure 8: HND & LND under different densities

The factors considered in HMNCS are relatively comprehensive. In the pre-selection mechanism, the tracking error of the previous moment is used to adjust the number of applying nodes at the current moment. Moreover, the average square estimation error is considered in the optimization objective function. The expected energy consumption and the residual energy of nodes are used to filter whether to apply or not. Energy consumption, and energy balance factors, which directly affect the network lifespan, are also included in the objective function. By setting appropriate weights for different factors, good performance can be achieved in both network lifespan and tracking performance.

It is worth noting that DCBC cannot achieve high tracking accuracy in the later stage due to its fast node death rate. The early period of high tracking accuracy does not last for a long time, so tracking performance is not good in the long run.

For all the mechanisms, the lifetime of WSN increases with the increase in node density. Among them, the WSN that uses HMNCS has the most extended lifespan, so it is more suitable for high-density WSN.

Based on the above simulation results, it can be concluded that the performance of HMNCS is superior to that of the compared mechanisms. In every step of HMNCS, such as the CH election mechanism, the pre-selection mechanism, and the optimization objective function of getting the final task set, the accuracy, energy consumption, and energy balance factors are considered in different ways. Therefore, HMNCS achieves a clever balance between the tracking accuracy and the network lifetime and obtains good results in both aspects of performance over the long term.

6 Conclusion

In this paper, an innovative two-layer node scheduling protocol called the Heuristic Multi-Node Collaborative Scheduling mechanism, is proposed to obtain optimal scheduling results during target tracking in WSN. HMNCS consists of three parts, which are cluster head election, pre-selection, and task set selection mechanism, respectively. By reducing the redundancy and making those necessary nodes track the target, this work balances the tracking performance and the network lifetime as much as possible. Simulation results confirm the superiority of the proposed mechanism. For HND, HMNCS is improved by 31.1%, 83.9%, and 112.3% compared with DCBC, GBRHA and NSAID, respectively. For LND, HMNCS is improved by 34.3%, 72.8%, and 89.0% compared with DCBC, GBRHA and NSAID, respectively.

With the development of unmanned aerial vehicles (UAVs) in recent years, it is very common that wireless sensor network deployed on the ground cooperate with UAVs to fulfill tasks. UAVs can be regarded as movable sensor nodes, and all nodes form a three-dimensional heterogeneous wireless sensor network. In such circumstances, how to schedule nodes and plan the flight paths of UAVs efficiently to achieve reliable data collection and other requirements in the actual scenario is our future research work.

Acknowledgement: The authors would like to express their gratitude to the members of the research group for their support.

Funding Statement: This work is supported by the Project Program of Science and Technology on Micro-System Laboratory, No. 6142804220101. The author who received the grant is identified by Xue Zhao.

Author Contributions: The authors confirm contributions to the paper as follows: Study conception and design: Xue Zhao; simulation: Shaojun Tao and Hongying Tang; analysis, interpretation of results: Jiang Wang; draft manuscript preparation: Baoqing Li. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The authors confirm that the data and materials supporting the findings of this study are not applicable as there were no specific datasets or materials used.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. S. H. Abdul-Qawy *et al.*, "An enhanced energy efficient protocol for large-scale IoT-based heterogeneous WSN," *Sci. African*, vol. 21, pp. e01807, 2023. doi: 10.1016/j.sciaf.2023.e01807.

[2] T. Khurana, S. Singh, and N. Goyal, "An evaluation of Ad-hoc routing protocols for wireless sensor networks," *Int. J. Adv. Res. Comput. Sci. Electron. Eng.*, vol. 1, no. 5, pp. 27–30, 2012.

[3] K. Guleria, A. K. Verma, N. Goyal, A. K. Sharma, A. Benslimane and A. Singh, "An enhanced energy proficient clustering (EEPC) algorithm for relay selection in heterogeneous WSNs," *Ad Hoc Netw.*, vol. 116, no. 3, pp. 102473, 2021. doi: 10.1016/j.adhoc.2021.102473.

[4] J. Wang, H. Han, H. Li, S. He, P. Kumar Sharma and L. Chen, "Multiple strategies differential privacy on sparse tensor factorization for network traffic analysis in 5G," *IEEE Trans. Ind. Inform.*, vol. 18, no. 3, pp. 1939–1948, 2022. doi: 10.1109/TII.2021.3082576.

[5] M. Maheswari and R. A. Karthika, "A novel QoS based secure unequal clustering protocol with intrusion detection system in wireless sensor networks," *Wirel. Pers. Commun.*, vol. 118, no. 2, pp. 1535–1557, 2021. doi: 10.1007/s11277-021-08101-2.

[6] Y. Zhang and L. J. Gao, "Sensor-networked underwater target tracking based on grubbs criterion and improved particle filter algorithm," *IEEE Access*, vol. 7, pp. 142894–142906, 2019. doi: 10.1109/ACCESS.2019.2943916.

[7] M. Anvaripour, M. Saif, and M. Ahmadi, "A novel approach to reliable sensor selection and target tracking in sensor networks," *IEEE Trans. Ind. Inform.*, vol. 16, no. 1, pp. 171–182, 2020. doi: 10.1109/TII.2019.2916091.

[8] G. S. Kori and M. S. Kakkasageri, "Agent driven resource scheduling in wireless sensor networks: Fuzzy approach," *Int. J. Inf. Technol.*, vol. 14, no. 1, pp. 345–358, 2022. doi: 10.1007/s41870-021-00829-1.

[9] D. P. Bhagat, "Tracking of moving target in wireless sensor network with improved network life time using PSO," *Wirel. Pers. Commun.*, vol. 127, no. 2, pp. 1225–1239, 2021.

[10] H. Zhu, H. Chen, and M. Luo, "Adaptive event-driven robust set membership estimation for received-signal-strength-based moving targets localization," *IEEE Internet Things J.*, vol. 9, no. 14, pp. 12825–12835, 2022. doi: 10.1109/JIOT.2021.3138784.

[11] N. Cao, S. Choi, E. Masazade, and P. K. Varshney, "Sensor selection for target tracking in wireless sensor networks with uncertainty," *IEEE Trans. Signal Proces.*, vol. 64, no. 20, pp. 5191–5204, 2016. doi: 10.1109/TSP.2016.2595500.

[12] Z. Li, L. Zhang, Y. Cai, and H. Ochiai, "Sensor selection for maneuvering target tracking in wireless sensor networks with uncertainty," *IEEE Sens. J.*, vol. 22, no. 15, pp. 15071–15081, 2022. doi: 10.1109/JSEN.2021.3136546.

[13] P. Fu, H. Tang, Y. Cheng, B. Li, H. Qian and X. Yuan, "An energy-balanced multi-sensor scheduling scheme for collaborative target tracking in wireless sensor networks," *Int. J. Distrib. Sens. Netw.*, vol. 13, no. 3, pp. 155014771769896, 2017. doi: 10.1177/1550147717698968.

[14] A. Alagha, S. Singh, R. Mizouni, A. Ouali, and H. Otrok, “Data-driven dynamic active node selection for event localization in IoT applications—A case study of radiation localization,” *IEEE Access*, vol. 7, pp. 16168–16183, 2019. doi: 10.1109/ACCESS.2019.2894956.

[15] B. Wu, Y. Feng, H. Zheng, and X. Chen, “Dynamic cluster members scheduling for target tracking in sensor networks,” *IEEE Sens. J.*, vol. 16, no. 19, pp. 7242–7249, 2016. doi: 10.1109/JSEN.2016.2597544.

[16] W. Chen and I. J. Wassell, “Optimized node selection for compressive sleeping wireless sensor networks,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 827–836, 2016. doi: 10.1109/TVT.2015.2400635.

[17] L. I. Khudonogova and S. V. Muravyov, “Interval data fusion with preference aggregation for balancing measurement accuracy and energy consumption in WSN,” *Wirel. Pers. Commun.*, vol. 118, no. 4, pp. 2399–2421, 2021. doi: 10.1007/s11277-021-08132-9.

[18] S. Hu, G. Li, and G. Huang, “Dynamic spatial-correlation-aware topology control of wireless sensor networks using game theory,” *IEEE Sens. J.*, vol. 21, no. 5, pp. 7093–7102, 2021. doi: 10.1109/JSEN.2020.3043748.

[19] J. Feng, B. Lian, and H. Zhao, “Coordinated and adaptive information collecting in target tracking wireless sensor networks,” *IEEE Sens. J.*, vol. 15, no. 6, pp. 3436–3445, 2015. doi: 10.1109/JSEN.2014.2388234.

[20] L. Saadatdoost, H. Motameni, S. M. H. Andargoli, and R. Ghazalian, “Sensing node selection and mobile sink displacement in the environments with multiple targets,” *IET Wirel. Sens. Syst.*, vol. 11, no. 4, pp. 147–160, 2021. doi: 10.1049/wss2.12013.

[21] S. Bharany, S. Sharma, N. Alsharabi, E. T. Eldin, and N. A. Ghamry, “Energy-efficient clustering protocol for underwater wireless sensor networks using optimized glowworm swarm optimization,” *Front. Mar. Sci.*, vol. 10, pp. 6945, 2023. doi: 10.3389/fmars.2023.1117787.

[22] L. Zhou, S. Leng, Q. Liu, H. Chai, and J. Zhou, “Intelligent sensing scheduling for mobile target tracking wireless sensor networks,” *IEEE Internet Things J.*, vol. 9, no. 16, pp. 15066–15076, 2022. doi: 10.1109/JIOT.2021.3125530.

[23] J. Feng and H. Zhao, “Dynamic nodes collaboration for target tracking in wireless sensor networks,” *IEEE Sens. J.*, vol. 21, no. 18, pp. 21069–21079, 2021. doi: 10.1109/JSEN.2021.3093473.

[24] K. Lin, T. Xu, J. Song, Y. Qian, and Y. Sun, “Node scheduling for all-directional intrusion detection in SDR-based 3D WSNs,” *IEEE Sens. J.*, vol. 16, no. 20, pp. 7332–7341, 2016. doi: 10.1109/JSEN.2016.2558043.