

Research Article

Adaptive Sensor Activation Algorithm for Target Tracking in Wireless Sensor Networks

Wei Zhou, Weiren Shi, Xiaogang Wang, and Kai Wang

College of Automation, Chongqing University, Chongqing 400030, China

Correspondence should be addressed to Weiren Shi, wrs@cqu.edu.cn

Received 9 December 2011; Revised 28 February 2012; Accepted 22 March 2012

Academic Editor: Aditya Karnik

Copyright © 2012 Wei Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Target tracking is an important application of wireless sensor networks where energy conservation plays an important role. In this paper, we propose an energy-efficient sensor activation protocol based on predicted region technique, called predicted region sensor activation algorithm (PRSA). The proposed algorithm predicts the moving region of target in the next time interval instead of predicting the accurate position, by analyzing current location and velocity of the target. We take these nodes within the predicted region as waiting-activation nodes and establish activation strategy. The fewest essential number of sensor nodes within the predicted region will be activated to monitor the target. Thus, the number of nodes that was involved in tracking the target will be decreased to save energy and prolong the network's operational lifetime. The simulation results demonstrate the effectiveness of the proposed algorithm.

1. Introduction

Wireless sensor networks (WSNs) employ a large number of intelligent sensor nodes with sensing, processing, and wireless communicating capabilities to implement complicated tasks in the specified sensing field. The rapid development of microelectronic technologies, embedded microprocessors, wireless communications, and networking technologies has made wireless sensor networks of large scale applicable to a wide range of applications, such as environmental monitoring, industrial sensing and diagnosis, situational awareness on the battlefield, space exploration, and healthcare [1]. One of the most important research areas of wireless sensor networks is target tracking, in which sensors monitor and report the locations of mobile targets [2–4].

The target tracking system in wireless sensor networks usually includes three phases: (1) target detection, (2) target localization, and (3) prediction and notice [5, 6]. In the first phase the selected nodes are activated to monitor and collect the location information of the mobile target using acoustic signal or images of targets [7, 8]. Then, the cluster head locates the target using the collected information. After locating the target, the cluster head uses a prediction model to predict the next location of the target and activates

the appropriate nodes to continue monitoring the target. If the prediction fails to track the target, the cluster head activates the additional sensor nodes to recapture the lost target. That implies a better prediction model can significantly reduce energy consumption, for fewer redundant sensor nodes will be activated.

Obviously, more activated sensor nodes around the target can achieve better tracking accuracy, particularly when there are some uncertainties in sensor detection. However, the activated sensor nodes need to consume energy. The more sensor nodes are activated, the more energy consumption is produced. Thus, we would like to select the fewest essential number of nodes dedicated for the task and at the same time other nodes stay in the sleeping status to save energy for prolonging the network life. High-efficiency energy strategy is a very important aspect of the target tracking protocols. How to balance the quality of the target tracking and network energy consumption has become a research focus [9–12]. In order to reduce energy consumption, we can reduce the data in communication, the frequency of communication, and the number of nodes in the tracking, by using prediction.

The cluster-based protocols are commonly used to reduce the energy usage of energy-sensitive wireless sensor

networks where sensor nodes are battery powered. In cluster-based protocols, a noncluster sensor node detects the target and then it forwards the information to its cluster head. Next, the cluster head collects and propagates the information to a sink. Chen et al. [7] have proposed a dynamic clustering algorithm for distributed target tracking using acoustic information. The hierarchical sensor network system was composed of sparsely placed high-capability nodes and densely spaced low-end nodes. The high-capability nodes acted as cluster heads and the low-end nodes acted as cluster members. Cluster heads closest to the target became active with higher probability than cluster heads that were farther from the target. Similarly, the probability that a cluster member sent data to the cluster head was proportional to its distance to the target. The authors of [13] presented a continuous object detection and tracking algorithm, based on a hybrid static/dynamic clustering scheme for the monitoring of continuous objects in wireless sensor networks. However, it lacks a missing track recovery mechanism. In [14], the authors also introduced a dynamic clustering mechanism for target tracking in wireless sensor networks.

For reducing the number of nodes involved in tracking a target and saving energy, the prediction-based techniques are used to predict the upcoming location of mobile target. When a sensor node detects target, it forwards the target information to its cluster head. The information contains location, velocity, and moving direction of the target. After calculating and predicting the location of target, the cluster head wakes up all/some nodes nearby the destination or all/some nodes on and around the route of the moving target from current position to the destination. Many filtering algorithms have been used for target prediction. Lin et al. [15] proposed a Compressed Kalman Filtering algorithm to reduce the totality of transmitted data to reduce the energy consumption. Compared with the traditional Kalman algorithm, they compressed the symmetric covariance matrix into a diagonal matrix to reduce the totality of transmitted data. In [16], two distributed particle filter tracking algorithms were proposed for tracking mobile targets in cluster-based underwater sensor networks. Li [17] proposed a collaborative multisensor tracking strategy using a novel adaptive particle filtering algorithm to predict and track the target for sensor networks. However, these filters require storage of many parameters, such as location, velocity, and acceleration, so the energy consumption is very large. For the prediction strategy, due to the uncertainty and unpredictability of real-world targets' motion, it is difficult to guarantee high accuracy of the prediction. Meanwhile we cannot guarantee the target will not change its moving direction beyond the prediction. The prediction result will be poor when the target can move with large accelerations, such as animal or people.

To adapt the real-time changes in velocity and direction of a moving target, Chen et al. [18] proposed a distributed sensor activation algorithm that enabled reliable tracking with the simplest binary-detection sensors. When the sensor that detected the target would broadcast a "wake" message to neighboring sensors, the one hop neighboring sensors were activated with a probability to detect targets or sleep to save

energy. However, in the dense sensor network, the energy dissipation will increase obviously. Guo et al. [9] proposed an energy-efficient algorithm by limiting the number of sensors used to track a target through monitoring their data quality and by limiting the amount of data being sent to the cluster head. The system allocated a weight to the sensor data and only sensors whose weight value was above the set threshold were allowed to participate in tracking. The weight that the sensor sent data to the cluster is proportional to its distance to the target. Meanwhile, they reduced the amount of redundant data in the network by considering the spatial relationship between neighboring sensors. Arienzo and longo [19] proposed a collaborative tracking algorithm in the cluster-based sensor networks. The algorithm adopts a greedy-type strategy to select informative sensors to track the target. In the tracking phase, the bootstrap particle filter and the unscented particle filter are used. However, the energy consumption will increase along with the accuracy of filtering prediction decreasing. In [20], the author proposed a novel target tracking algorithm to detect and monitor the path of a moving target. They activated a minimum subset of nodes while maintaining coverage and network connectivity, in order to keep more nodes in sleeping and minimize the energy consumption. However, sensor node in wireless sensor networks is not very reliable. If the sensor nodes among the minimum subset cannot work well, the tracking accuracy will decrease obviously. Also an adaptive sensor activity scheduling was proposed to detection and dynamic footprint tracking of spatial-temporal events, where concepts of Statistical Mechanics were employed to stochastically activate the sensor nodes [21].

In this paper, we propose an energy-efficient sensor activation protocol based on predicted region technique. The predicted region sensor activation algorithm (PRSA) improves the energy-quality tradeoff by predicting the moving region of target in the next time interval instead of predicting the accurate position. The PRSA algorithm can reduce the number of nodes that are involved in tracking the target and decrease the missing rate that is caused by the traditional prediction schemes. The proposed algorithm predicts the moving region in the next time interval by analyzing current location and velocity of the target. The moving region is defined as a circle around the current coordinates of the target with a radius of moving range of target in one second (assuming that the sampling interval is one second). Assume that the target's moving region in one second is less than a setting value, and then we can prove that the moving region of target in the next time interval must be in the intersection region which combines the communication range of two nodes closest to the target, as Figure 2 shows. Namely, target location in the next time interval must be in the communication intersection region. We take these nodes within the communication intersection region as waiting-activation nodes and establish activation strategy by judging the location relation between the target and the waiting-activation nodes. We activate the fewest essential number of sensor nodes within the intersection region to monitor the target, in order to ensure the target

can be detected at any time and reduce the number of nodes involved in tracking the target for prolonging the network's operational lifetime.

The rest of this paper is organized as follows: in Section 2, we present the tracking system model, including target motion model and sensing model. Section 3 presents the main contribution of this paper, the adaptive sensor activation algorithm. Simulation results and detailed comparisons are presented in Section 4. Finally, concluding remarks are given in Section 5.

2. Tracking System Descriptions

2.1. Target Motion Model. We consider that there is only one target in the monitoring region and assume the target moves in a two-dimensional plane, then the target motion of the tracking system can be described as follows:

$$X_{k+1} = FX_k + Gw_k, \quad (1)$$

where the target state $X_k = [x(k), y(k), \dot{x}(k), \dot{y}(k), \varphi]^T$, therein $x(k)$, $y(k)$, $\dot{x}(k)$, $\dot{y}(k)$, and φ represent the position, velocity, and the turn rate, respectively, $w_k \sim N(0, Q_k)$ is the process noise, F and G can be described as

$$F = \begin{bmatrix} 1 & 0 & \frac{\sin jT}{j} & -\frac{1-\cos jT}{j} & 0 \\ 0 & 1 & -\frac{1-\cos jT}{j} & \frac{\sin jT}{j} & 0 \\ 0 & 0 & \cos jT & -\sin jT & 0 \\ 0 & 0 & \sin jT & \cos jT & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

$$G = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

2.2. Sensing Model. When a target comes into monitoring region, the acoustic signal from the moving target is detected by the sensor nodes. At time k , the sensed acoustic energy z_k^i of the i th node is given by [22]

$$z_k^i = \lambda_i \frac{S_k}{\|L_k^T - L^i\|^\alpha} + n_k^i, \quad (3)$$

where L_k^T and L^i denote the target location and the node location, respectively, S_k denotes the acoustic energy of the target, $n_k^i \sim N(0, R_k^i)$ denotes the measurement noise, λ_i denotes the gain factor of the i th node, and $\alpha \approx 2$ denotes the signal attenuation parameter.

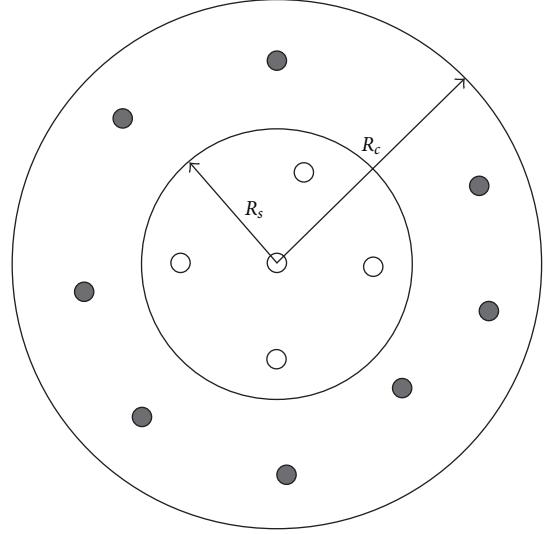


FIGURE 1: Sensing model and communication model.

In our application, each measurement taking part in localization is calculated by taking the average of all samples during the localization interval to minimize the measurement error.

3. Proposed Algorithm

3.1. Problem Description. In our model (see Figure 1), we assume that the communication range of sensor nodes is R_c , and R_s is the sensing range. In order to guarantee the direct communication between the nodes which can simultaneously monitor the target, we define $R_c = 2R_s$. So the sensing region is defined as $d < R_s$ and communication region is defined as $d < R_c$.

For single prediction strategy, due to the uncertainty and unpredictability of real-world targets' motion, it is difficult to guarantee high accuracy of the prediction. Therefore, we predict the moving region of target in the next time interval instead of predicting the accurate position. To reduce the number of nodes that are involved in tracking the target and guarantee the target can be detected in the next time interval, we limit the target within the communication intersection region of two nodes closest to the target in the next time interval, as Figure 2 shows. In Theorem 1, as long as maximum velocity of the target is limited (satisfies $v_k T \leq \min(R_c - d_k^i, R_c - d_k^j)$), we will prove that no matter how the acceleration and moving direction of the targets change, the moving region of target in the next time interval must belongs to the communication intersection region. Thus, we can take these nodes within the communication intersection region as waiting-activation nodes and establish activated strategy to detect the target.

Theorem 1. As Figure 2 shows, A_s^i and A_s^j are the sensing regions of nodes S_i ($i = 1, 2, 3, \dots, n$) and S_j ($j = 1, 2, 3, \dots, n$), A_c^i and A_c^j are the communication regions of S_i and S_j . Assume

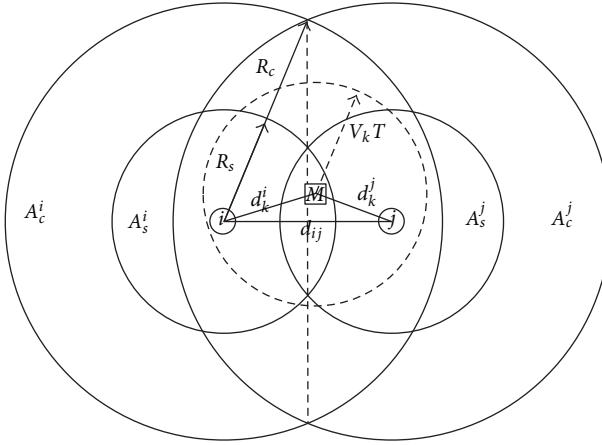


FIGURE 2: The relation between target and nodes.

S_i and S_j are the monitoring nodes closest to the target. Target $M(x_k, y_k) \in A_s^i \cap A_s^j$, and the moving velocity of the target is $v_k m/s$. If $v_k T \leq \min(R_c - d_k^{im}, R_c - d_k^{jm})$, then $M(x_{k+1}, y_{k+1}) \in A_c^i \cap A_c^j$.

Proof. Assume that (x_i, y_i) is the coordinate of S_i and (x_j, y_j) is the coordinate of S_j . d_k^i and d_k^j are the distances between the target and the nodes S_i and S_j . v_{kx} and v_{ky} denote the target's velocity of x -direction and y -direction, respectively.

The distance between the target and the i th node at time k is given by

$$d_k^i = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}. \quad (4)$$

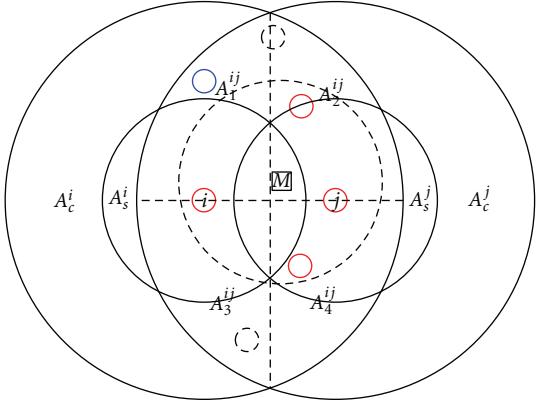
So, the distance between the target and the i th node at time $k+1$ is given by

$$\begin{aligned} d_{k+1}^i &= \sqrt{(x_{k+1} - x_i)^2 + (y_{k+1} - y_i)^2} \\ &= \sqrt{[(x_k + v_{kx}T) - x_i]^2 + [(y_k + v_{ky}T) - y_i]^2} \\ &= \sqrt{(d_k^{im})^2 + (v_k T)^2 + 2v_{kx}T(x_k - x_i) + 2v_{ky}T(y_k - y_i)}, \end{aligned} \quad (5)$$

where $0 \leq x_k - x_i \leq R_s$, $0 \leq y_k - y_i \leq R_s$ and $0 \leq d_k^i \leq R_s$ are known. Assume that $d_k^i > d_k^j$ at time k and $v_k T \leq R_c - d_k^i$ is satisfied. Then we have

$$\begin{aligned} d_{k+1}^i &\leq \sqrt{(d_k^i)^2 + (v_k T)^2 + 2v_{kx}T R_s + 2v_{ky}T R_s} \\ &= \sqrt{(d_k^i)^2 + (v_k T)^2 + 2v_k T R_s} \\ &\leq \sqrt{(d_k^i)^2 + (R_c - d_k^i)^2 + 2R_s(R_c - d_k^i)} \\ &= \sqrt{2(d_k^i)^2 + 2R_c^2 - 3R_c d_k^i} \\ &\leq R_c. \end{aligned} \quad (6)$$

As a result, it is concluded that the target $M(x_{k+1}, y_{k+1}) \in A_c^i \cap A_c^j$ at time $k+1$. \square



○ Tracking node
○ Monitoring node
○ Sleeping node

FIGURE 3: The communication intersection region.

According to the theorem, we can draw the conclusion that the target should be within the communication intersection region, so long as the moving region of the target satisfies $v_k T \leq \min(R_c - d_k^i, R_c - d_k^j)$, no matter how the acceleration and moving direction of the targets change. Therefor, we can estimate the region that the target will appear in the next time interval. Namely, the moving region of target in the next time interval belongs to the communication intersection region of the two nodes closest to the target. Thus, we activate fewest essential number of sensor nodes within the intersection region to monitor the target's location in the next time.

3.2. Algorithm Description. In Figure 3, the communication intersection region (short for intersection region in the following) of nodes S_i and S_j is separated into four sub-regions ($A_k^{ij}, k = 1, 2, 3, 4$) in the paper. The nodes which can monitor the target within the intersection region called tracking nodes, and the nodes are activated but do not monitor the target called monitoring nodes, and the others nodes are sleeping nodes.

When the target appears in the monitoring region, the nodes which monitor the target are activated. They estimate the location of the target and send the measured values to the cluster head. The cluster head selects two nodes S_i and S_j which are closest to the target and activates some nodes within their intersection region to monitor the target in the next time interval. That is because the two nodes closest to the target can form a biggest intersection region to cover the target. The two cluster members S_i and S_j would broadcast a “wake” message to their neighboring nodes (in their one hop). The neighboring nodes of S_i and S_j will be activated partially by implementing the activation algorithm (Algorithm 1).

In order to facilitate implementing the activation algorithm, we treat the tracking nodes in the intersection region

```

Begin
Calculate the number of neighbor of  $S_i, S_j$ ;
If  $num\_neighbour = 0$  then
    Active  $S_i, S_j$  neighbor node with probability  $P$ ;
If  $num\_neighbour = 1$  then
    If target and node in the same region then
        Active this node and its neighbor node with probability  $P$ ;
    Else
        Active  $S_i, S_j$  neighbor node with probability  $P$ ;
If  $num\_neighbour \geq 2$  then
    If there are nodes in  $A_1^{ij} \cup A_2^{ij}$  and  $A_3^{ij} \cup A_4^{ij}$  respectively then
        If target in  $A_1^{ij} \cup A_2^{ij}$  then
            Active one node in  $A_1^{ij}$  and  $A_2^{ij}$  respectively;
            Active one or two node in  $A_3^{ij} \cup A_4^{ij}$  randomly;
        Else
            Similar to the target in  $A_1^{ij} \cup A_2^{ij}$ ;
        Else If there are nodes in  $A_1^{ij} \cup A_2^{ij}$  but not in  $A_3^{ij} \cup A_4^{ij}$  then
            If target in  $A_3^{ij} \cup A_4^{ij}$  then
                If the motion of target trends to  $A_1^{ij} \cup A_2^{ij}$  then
                    Active node in  $A_1^{ij}$  and  $A_2^{ij}$  closest the target;
                Else
                    Active  $S_i, S_j$  neighbor with probability  $P$ ;
            Else
                Active one node in  $A_1^{ij}$  and  $A_2^{ij}$  respectively;
        Else
            Similar to the target in  $A_1^{ij} \cup A_2^{ij}$  but not in  $A_3^{ij} \cup A_4^{ij}$ ;

```

ALGORITHM 1: Active strategy for nodes.

as “sleeping nodes” except the two closest nodes. However, we activate the “sleeping nodes” priority. The real sleeping nodes will be activated only when there are no or not enough “sleeping nodes” within the intersection region. Therefore, Algorithm 1 can be described in details as follows.

n is the number of nodes within the intersection region of S_i and S_j , where n does not include S_i and S_j . When $n \geq 2$, there are three situations.

(i) If there are nodes within the regions $A_1^{ij} \cup A_2^{ij}$ and $A_3^{ij} \cup A_4^{ij}$, respectively:

when the target is in the region $A_1^{ij} \cup A_2^{ij}$, two nodes closest to the target are activated from the regions of A_1^{ij} and A_2^{ij} , respectively, and one or two nodes in the region $A_3^{ij} \cup A_4^{ij}$ are activated randomly to monitor the target. The situation that the target is in the region $A_3^{ij} \cup A_4^{ij}$ is similar to the region $A_1^{ij} \cup A_2^{ij}$.

(ii) If there are nodes only within the region $A_1^{ij} \cup A_2^{ij}$:

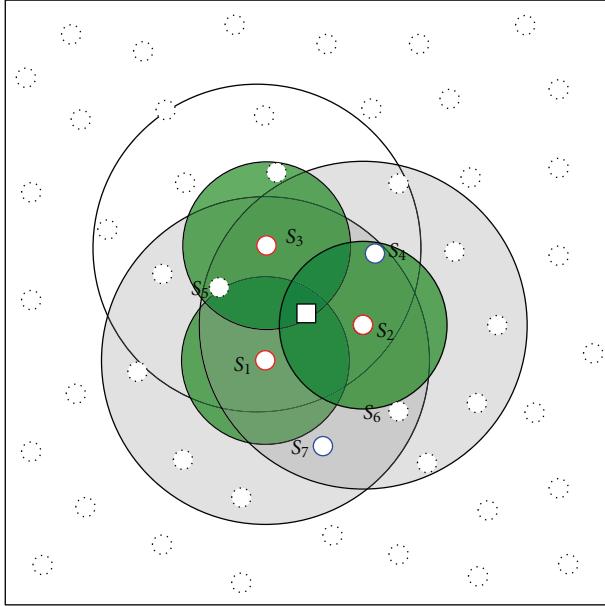
when the target is in the region $A_1^{ij} \cup A_2^{ij}$, two nodes closest to the target are activated from the regions of A_1^{ij} and A_2^{ij} , respectively. When the target is in the region $A_3^{ij} \cup A_4^{ij}$, this means there are no nodes within this region. And we can predict the motion tendency of the target by its current position and velocity. If the motion of the target trends to the region $A_1^{ij} \cup A_2^{ij}$, then the nodes within the regions A_1^{ij} and A_2^{ij} which are closest to the target are activated; otherwise,

the monitoring region should be enlarged by activating the neighboring nodes of the nodes S_i and S_j with the probability P .

(iii) If there are nodes only within the region $A_3^{ij} \cup A_4^{ij}$, the nodes activation algorithm is similar to the situation that there are only nodes within the region $A_1^{ij} \cup A_2^{ij}$.

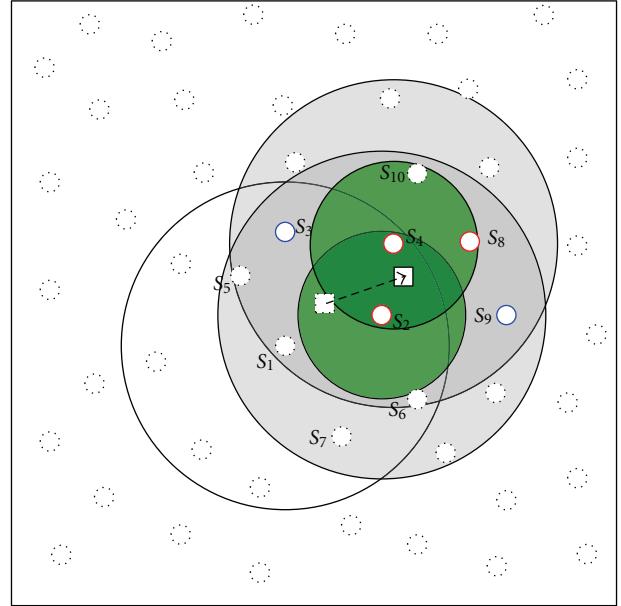
When $n < 2$, this means the target is in the sparse region of the network and we cannot ensure the target will be tracked accurately in the next time interval. Thus, a sparse network activated strategy is presented in the paper. If there is no node within the intersection region, then the neighboring nodes of S_i and S_j are activated with the probability P to monitor the target. If there is one node in the intersection region and this node is in the same subregion with the target, then the only node should be activated and its neighboring nodes are activated with the probability P to monitor the target. Otherwise, the neighboring nodes S_i and S_j should be activated with the probability P to enlarge the monitoring region.

According to the analysis in Section 3.1, the target will appear in the intersection region of S_i and S_j in the next interval only when the target’s moving region in one second satisfies $v_k T \leq \min(R_c - d_k^i, R_c - d_k^j)$. Thus, we can monitor the target with a high probability in the next time interval using the proposed algorithm. As Figure 4 shows, the target is detected by three nodes marked as red dot. S_1 and S_2 are the nodes closest to the target, then “wake” messages



□ Target
 Monitoring node
 Tracking node
 Sleeping node

FIGURE 4: The activation algorithm.



□ Target
 Monitoring node
 Tracking node
 Sleeping node

FIGURE 5: The working mode of nodes.

are broadcasted to the one hop scope (the gray scope). The nodes S_k ($k = 3, 4, 5, 6, 7$) within the intersection region will receive the “wake” messages, and the other nodes out of the intersection region will drop it. We can see that the target is within the region $A_1^{ij} \cup A_2^{ij}$, and the node S_3 is a “sleeping node” (tracking node), so S_3 within the region A_1^{ij} will be activated preferentially instead of the real sleeping node S_5 . Meanwhile, the node S_4 within the region A_2^{ij} is activated and the node S_7 within the region $A_3^{ij} \cup A_4^{ij}$ is activated randomly. Thus, there will be five nodes S_k ($k = 1, 2, 3, 4, 7$) that are activated in the next time interval to monitor the target, and other nodes will work at sleeping mode.

3.3. Node Working Mode. In order to reduce energy consumption of the network and ensure the network can monitor the target within the tracking region at any time, the nodes cannot work at a fully closed status. Therefore, we divide the work modes for nodes into the following situations.

(a) *Tracking Mode.* Working in the tracking mode, the nodes can sense, send, and receive messages. The microprocessor is activated and the nodes can estimate the position of the target.

(b) *Monitoring Mode.* The nodes can sense and receive messages in the monitoring mode. The microprocessor is on standby, and it can be activated only when the sensor data exceed the threshold or command information reaches.

(c) *Sleeping Mode.* Working at the sleeping mode, the nodes would close the sense module and communication module. And the microprocessor goes to sleeping status. However, the nodes will turn to monitoring status until the timer or the radio-trigger awakes them.

The proposed PRSA algorithm is a distributed tracking algorithm instead of a centralized one. The working mode of each sensor node relies on its neighboring sensors, but not the central server. The normal sensor nodes are switched between the tracking mode, the monitoring mode, and the sleeping mode. The specific implementation is described as in Algorithm 2.

When a target comes into the sensing regions, the nodes that monitor the target will turn to the tracking mode and keep the tracking status until the target leaves the sensing regions. When the target leaves the sensing regions, the tracking nodes will turn back to the sleeping mode. If the tracking node is one of the nodes that closest to the target of last time interval, the tracking node will broadcast a “sleep”

message to neighboring sensors in its one hop. If the neighboring nodes don’t detect the target and they are not the activated nodes at present moment, then they will turn to the sleeping mode. In this way, the time of the nodes at the monitoring mode will be shortened and the energy consumption of the network will be reduced.

At time t , the nodes which participate in tracking the target are shown in Figure 4. At time $t + 1$, the target leaves the monitoring region of the node S_1 (as Figure 5 shows). Then the node S_1 goes into sleeping mode and sends “sleep” message to its neighboring nodes. And the target is

```

Begin
  Switch the working mode of sensor do
    Case sleeping mode
      If timer is up then
        Change to monitoring mode;
      Else If get activation message then
        Change to monitoring mode;
      Else wait;
    Case monitoring mode
      Detecting target;
      If target detected then
        Broadcast “wake” message to one hop neighbour;
        Change to target mode;
      Else
        Change to sleeping mode until receive “sleep” message;
    Case target mode
      If the closest node then
        Broadcast “wake” message to one hop neighbour;
        Tracking target;
      Else
        Tracking target;
      If target disappear then
        Broadcast “sleep” message to one hop neighbour;
        Change sleeping mode;
  End

```

ALGORITHM 2: Working mode of nodes.

monitored by the nodes S_2 and S_4 then the nodes $S_k (k = 3, 8, 9)$ of the intersection region are activated by the activated strategy. Therefore, the nodes $S_k (k = 2, 3, 4)$ drop the “sleep” message, and the node S_7 is switched from the monitoring mode to the sleeping mode. It means we can make the nodes which are activated but not detected target turn to sleep. This can keep the amount of activated nodes at the fewest essential number in every moment, to reduce the energy consumption of the network.

When the sleeping nodes are awaked by the timer, they begin to monitor the target. The monitoring nodes will turn back to sleep if they do not detect the target. If the sleeping nodes are activated by the activation strategy described in Section 3.2, they will work on the monitoring mode. And the monitoring nodes will turn to the tracking mode unless they monitor the target. Or else they will remain in the monitoring mode until they receive the “sleep” message. Once the node receives the “sleep” message, it will turn to the sleeping mode immediately, which can avoid the long-term usage of one single node.

4. Simulation Results

The energy consumption is closely related to the number of sleeping sensor nodes, monitoring sensor nodes, and tracking sensor nodes. E is the energy cost in one step, so the energy model can be described as [23]

$$E = N_{\text{sleep}} \cdot P_{\text{sleep}} + N_{\text{detect}} \cdot P_{\text{detect}} + N_{\text{tracking}} \cdot P_{\text{tracking}}, \quad (7)$$

where $P_* = W_{\text{se}} t_{\text{se}} + W_{\text{re}} t_{\text{re}} + W_{\text{tr}} t_{\text{tr}} + W_{\text{sl}} t_{\text{sl}}$, N_* is the number of nodes in corresponding states. Where W_{se} , W_{re} , W_{tr} , and W_{sl} represent the power usage per time unit of sensing, transmitting, receiving, and radio-trigger, respectively, t_{se} , t_{re} , t_{tr} , and t_{sl} represent the active time of sensing, transmitting, receiving, and radio-trigger during the period t .

In this paper, we analyze the performance of our algorithm via simulations. The network model for simulation consists of uniformly and randomly deployed 250 nodes in a square area over $500 \text{ m} \times 500 \text{ m}$. We assume the communication coverage range is 60 m, the sensing coverage range is 30 m, and the time interval is set as one second. In Figure 6, The blue line is the actual trajectories of the moving target. We detect the target 50 times when the target cross the scope, where 62 nodes (the red circles) are activated, only 38 nodes (the red circles with “+”) participate the tracking. To reduce the energy consumption, most nodes (the blue circles) in the network work are at sleeping mode.

We focus on the energy consumption as performance by comparing our energy consumption results with distributed sensor activation algorithm (DSA2) [18] and Naive method, where each sensor node monitors its sensing range all the time and reports the location of target to the sink node periodically [24].

Figure 7 shows the simulation result of increasing sensing range. We run the test with 250 sensor nodes, and we increase the sensing range from 25 m to 50 m in increment of 5 m. From the result, we can find that the total energy consumption in our approach is much less than the consumption energy of Naive and DSA2 approach. Although the number

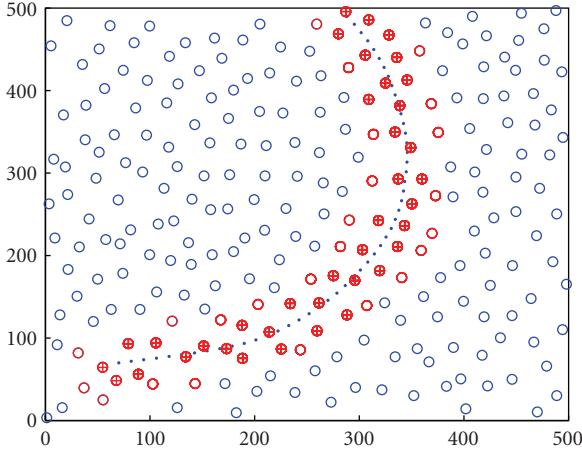


FIGURE 6: The actual trajectory of the moving target and the activated nodes.

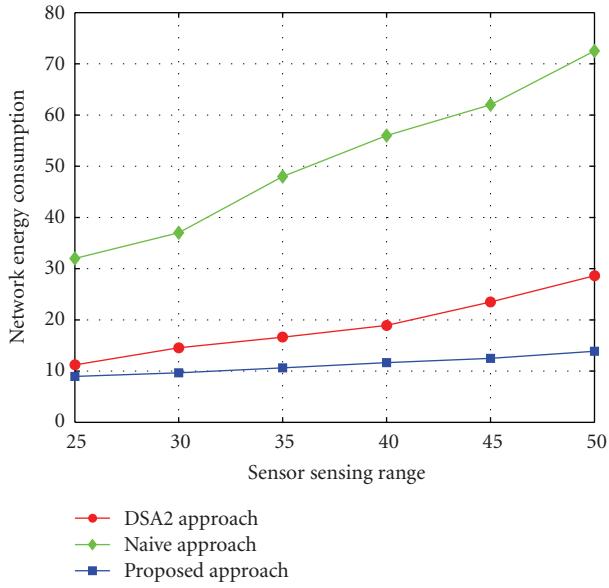


FIGURE 7: The relation between energy consumption of the proposed approaches, DSA2 approaches, and Naive for different sensing ranges.

of nodes in the intersection region will increase as the sensing range gets large, our proposed algorithm does not activate more nodes with the increasing number in the intersection region. Only fewest essential nodes are activated to track the target, so we can reduces energy consumption by reducing the number of nodes involved in tracking the target.

Figure 8 shows the effect of deploying different sensor nodes on the total energy consumption of our approach and other existing methods. We set the sensing range of the nodes at 30 m, communication range at 60 m. The sensor nodes are varied from 200 to 500 in increment of 100 nodes. With the increase of the sensor nodes, the nodes within the intersection region will increase, just like the situation of large sensing range. The effect of large number nodes is almost the same with large sensing range, and the energy

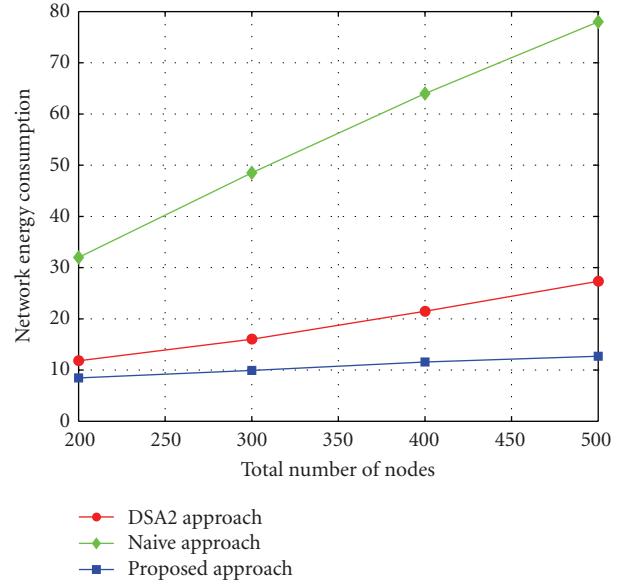


FIGURE 8: The relation between the energy consumption and network scale in the proposed approach, DSA2 approach and Naive.

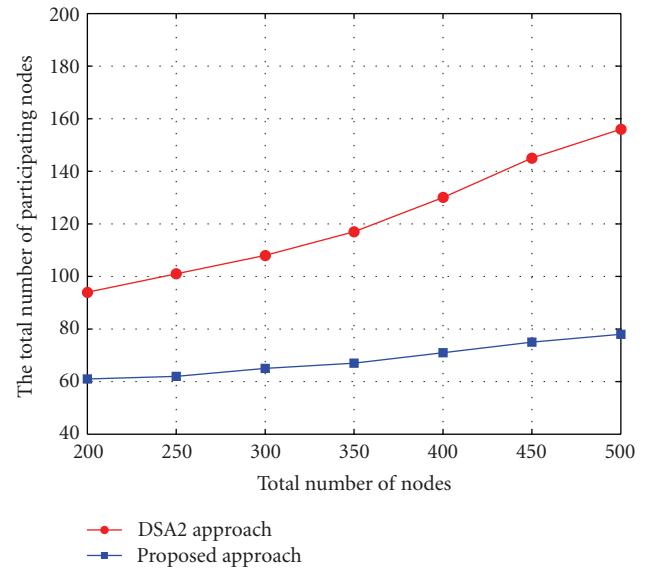


FIGURE 9: The relation between the total number of deployed nodes and the total number of participating nodes in the proposed approach and DSA2 approach.

consumption slightly increases with the increase of the sensor nodes.

The relation between the total number of deployed nodes and the total number of participating nodes in our proposed approach and DSA2 approach is shown in Figure 9. From the figure, we can find that the total number of participating sensor nodes in our approach is less than the DSA2 approach. Thus, that implies our approach can reduce the energy consumption and prolong the lifetime of the network.

In Figure 10, different velocities of the moving target are taken into consideration in our approach and DSA2

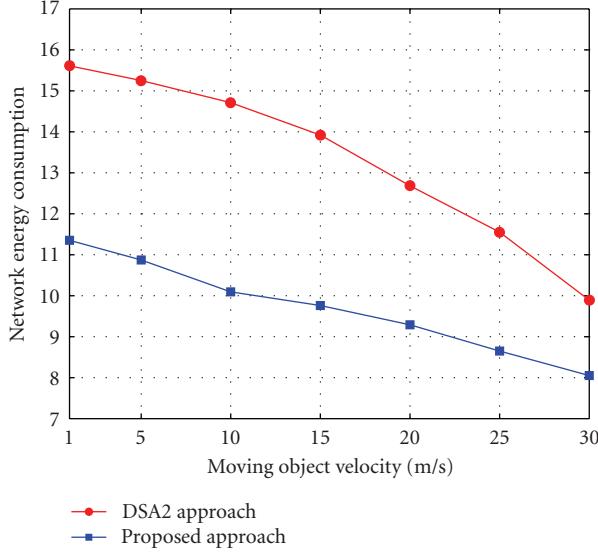


FIGURE 10: The effect of target velocity on the energy consumption in the proposed approach and DSA2 approach.

approach. When the velocity is large, just like the situation of small sensing, the effect of large velocity of the target is almost the same with small sensing range of sensors. The result shows that with the increase of the velocity of the moving target, the network energy consumption will decrease. That is because the number of the active nodes decreases, while the target velocity increases and this will reduce the network energy dissipation.

Figure 11 shows the impact of the moving target velocity on the network energy consumption and missing rate in our approach and DSA2 approach. The missing rate is defined as the percentage that nodes fail to detect the target. From the result, we can see that the missing rate is very low when $V_{\max} \leq 30 \text{ m/s}$, due to the target to be moving in the communication intersection region. The missing rate will increase when $V_{\max} > 30 \text{ m/s}$, because the sensing range is 30 m, which is too small for this velocity. The target may move out of the intersection region, or even out of the sensing range of the activated nodes, so the tracking rate will decrease.

Once there is no node within the intersection region of S_i and S_j , the neighboring nodes of S_i and S_j (in their one hop) will be activated with the activation probability P to monitor the target. Figure 12 shows the impact of the activation probability P on missing rate of the network. We run the test in a sparse network with 150 sensor nodes, and we increase the activation probability P from 0 to 1 in increment of 0.1. The result shows that the missing rate will decrease with the increase of the activation probability P . That is because the number of the active nodes increase, while the activation probability P increases in the sparse network.

From the simulation results, we can conclude that if the target's moving region in one second satisfies $v_k T \leq \min(R_c - d_k^i, R_c - d_k^j)$, then more energy savings can be

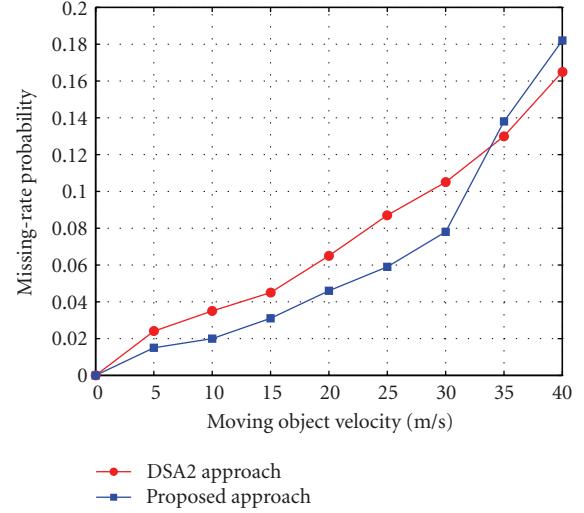


FIGURE 11: The effect of target velocity on the missing-rate probability in the proposed approach and DSA2 approach.

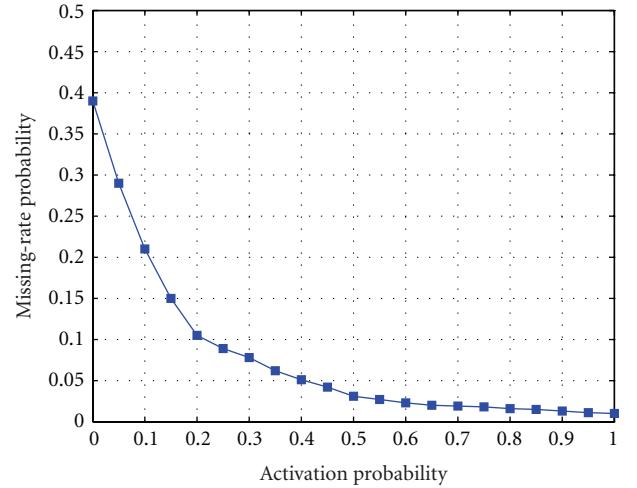


FIGURE 12: The effect of the activation probability P on the missing-rate probability.

obtained, as well as low number of participating nodes, and low missing rate.

5. Conclusion

Focusing on the energy problems in the target tracking of wireless sensor networks, we propose an energy-efficient sensor-activated strategy. Due to the uncertainty and unpredictability of real-world targets' motion, it is difficult to guarantee high accuracy of the prediction by using traditional prediction strategy. In this paper, we predict the moving region of target in the next time interval that is proved to be in the communication intersection region instead of predicting the accurate position. Then we establish an activated strategy to activate the fewest essential number of sensor nodes within the communication intersection region to monitor the target's location in the

next time by analyzing the position relation between the target and the nodes. The proposed algorithm can reduce the number of nodes being involved in tracking the target to prolong the network's operational lifetime. Simulation studies show that the proposed algorithm provide significant energy savings, while side effect on target tracking rate is not too negatively improved in terms of energy-quality tradeoff.

Acknowledgments

This work was supported by the National Science and Technology Major Project (2009ZX07528-003-09), and Key Scientific and Technological Projects of Chongqing (CSCT, 2010AA2036).

References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [2] S. P. M. Tran and T. A. Yang, "OCO: Optimized Communication Organization for target tracking in wireless sensor networks," in *Proceedings of the 13th International Symposium on Temporal Representation and Reasoning*, pp. 428–435, Taichung, Taiwan, 2006.
- [3] Z. Wang, Z. Wang, H. Li, X. Shen, and X. Sun, "Tracking and predicting moving targets in hierarchical sensor networks," in *Proceedings of the IEEE International Conference on Networking, Sensing and Control (ICNSC '08)*, pp. 1169–1173, Sanya, China, April 2008.
- [4] H. W. Tsai, C. P. Chu, and T. S. Chen, "Mobile object tracking in wireless sensor networks," *Computer Communications*, vol. 30, no. 8, pp. 1811–1825, 2007.
- [5] Y. C. Tseng, S. P. Kuo, H. W. Lee, and C. F. Huang, "Location tracking in a wireless sensor network by mobile agents and its data fusion strategies," *Computer Journal*, vol. 47, no. 4, pp. 448–460, 2004.
- [6] Z. Guo, M. Zhou, and L. Zakrevski, "Optimal tracking interval for predictive tracking in wireless sensor network," *IEEE Communications Letters*, vol. 9, no. 9, pp. 805–807, 2005.
- [7] W. P. Chen, J. C. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 258–271, 2004.
- [8] X. Sheng, Y.-H. Hu, and P. Ramanathan, "Distributed particle filter with GMM approximation for multiple targets localization and tracking in wireless sensor network," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, pp. 181–188, Los Angeles, Calif, USA, 2005.
- [9] M. Guo, E. Olule, G. Wang, and S. Guo, "Designing energy efficient target tracking protocol with quality monitoring in wireless sensor networks," *Journal of Supercomputing*, vol. 51, no. 2, pp. 131–148, 2010.
- [10] M. Zoghi and M. H. Kahaei, "Adaptive sensor selection in wireless sensor networks for target tracking," *IET Signal Processing*, vol. 4, no. 5, pp. 530–536, 2010.
- [11] W. Guojun, W. Huan, C. Jiannong, and G. Minyi, "Energy-efficient dual prediction-based data gathering for environmental monitoring applications," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 3516–3521, Kowloon, China, March 2007.
- [12] G. K. Atia, V. V. Veeravalli, and J. A. Fuemmeler, "Sensor scheduling for energy-efficient target tracking in sensor networks," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4923–4927, 2011.
- [13] W.-R. Chang, H.-T. Lin, and Z.-Z. Cheng, "CODA: a continuous object detection and tracking algorithm for wireless ad hoc sensor networks," in *Proceedings of the 5th IEEE Consumer Communications and Networking Conference*, pp. 168–174, Las Vegas, Nev, USA, 2008.
- [14] G.-Y. Jin, X.-Y. Lu, and M.-S. Park, "Dynamic clustering for object tracking in wireless sensor networks," in *Proceedings of the 3rd International Symposium on Ubiquitous Computing Systems (UCS '06)*, pp. 200–209, Seoul, Republic of Korea, 2006.
- [15] J. Y. Lin, L. H. Xie, and W. D. Xiao, "Target tracking in wireless sensor networks using compressed Kalman filter," *International Journal of Sensor Networks*, vol. 6, pp. 251–262, 2009.
- [16] Y. Huang, W. Liang, H. B. Yu, and Y. Xiao, "Target tracking based on a distributed particle filter in underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 8, pp. 1023–1033, 2008.
- [17] X. Li, "Collaborative multi-sensor tracking in mobile wireless sensor networks," *International Journal of Sensor Networks*, vol. 8, pp. 233–246, 2010.
- [18] J. Chen, K. Cao, K. Li, and Y. Sun, "Distributed sensor activation algorithm for target tracking with binary sensor networks," *Cluster Computing*, vol. 14, no. 1, pp. 55–64, 2011.
- [19] L. Arienzo and M. Longo, "Energy-efficient collaborative tracking in wireless sensor networks," *International Journal of Sensor Networks*, vol. 9, no. 3-4, pp. 124–138, 2011.
- [20] A. M. Khedr and W. Osamy, "Effective target tracking mechanism in a self-organizing wireless sensor network," *Journal of Parallel and Distributed Computing*, vol. 71, pp. 1318–1326, 2011.
- [21] A. Srivastav, A. Ray, and S. Phoha, "Adaptive sensor activity scheduling in distributed sensor networks: a statistical mechanics approach," *International Journal of Distributed Sensor Networks*, vol. 5, no. 3, pp. 242–261, 2009.
- [22] X. Sheng and Y. H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 53, no. 1, pp. 44–53, 2005.
- [23] L. Yang, C. Feng, J. W. Rozenblit, and H. Qiao, "Adaptive tracking in distributed wireless sensor networks," in *Proceedings of the 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems*, pp. 103–111, Potsdam, Germany, 2006.
- [24] S. P. M. Tran and T. A. Yang, "Evaluations of target tracking in wireless sensor networks," in *Proceedings of the 37th Technical Symposium on Computer Science Education (SIGCSE '07)*, pp. 97–101, Houston, Tex, USA, 2007.

