DOI: 10.21917/ijct.2020.0327

AN ENERGY EFFICIENT HYBRID CLUSTERING ALGORITHM COMBINED WITH PREDICTION METHOD FOR TARGET TRACKING IN WIRELESS SENSOR NETWORKS

Ri Man Gun[1], Pak Jin[2], O Ju Hyok[3] and Ri Chol Ho[4]
[1,2,3]*Communication Faculty, Kim Chaek University of Technology, D.P.R of Korea*
[4]*Computer Engineering Faculty, Pyongyang University of Science and Technology, D.P.R of Korea*

Abstract

Target tracking in WSN has attracted a great attention duo to its growing application potential in different fields. One of the main problems for target tracking in WSN is to maximize network lifetime by reducing energy consumption as well as guaranteeing the target tracking quality at a certain level. Among different target tracking schemes, hybrid clustering resolves the boundary problem and guarantee the target tracking quality because the static cluster and the on-demand dynamic cluster take turns each other to track the target in hybrid clustering scheme. However, huge amount of energy can be consumed due to the frequent formation and dismiss of redundant dynamic clusters when the target zigzags between a static cluster and a dynamic cluster or when the movement of target makes overlapped dynamic clusters to be formed continuously. In order to resolve this kind of problems, in this paper, a hybrid clustering algorithm combined with prediction method is proposed so that energy consumption due to the overforming of dynamic clusters could be reduced and the target tracking quality could be guaranteed simultaneously. Furthermore, a scheme to adjust the size of predicted clusters and the length of target interval time, according to prediction error and target speed, is applied to guarantee the target tracking quality of the prediction-based clustering algorithm. The results of extensive simulation experiment show that the proposed scheme can guarantee the target tracking quality and extend network lifetime significantly although a huge amount of energy is consumed due to overforming and overdismissing dynamic clusters.

Keywords:

Wireless Sensor Networks, Energy Consumption, Quality of Tracking, Hybrid Clustering, Prediction-based Clustering

1. INTRODUCTION

A target tracking is one of the important applications of wireless sensor networks. Wireless sensor networks are used as a basis of many practical target tracking applications such as tracking and monitoring of military targets, natural disaster reliefs, tracking and observation of wild animals, biomedical healthcare and observation, tracking human beings in congested and limited areas, tracking transportation vehicles like cars in highways etc. [1], [2]. Generally, the main aim of the moving target tracking is to detect the existence of target and to estimate its locations during the moving into the monitoring area and to relay the information of target to the base station immediately [3].

In target tracking in WSNs, essential characteristics of WSNs must be considered; such as limited processing ability and memory capacity, short communication range of sensors, limited resource of energy, low data transmission rate and etc. Such characteristics of WSNs make it difficult to design and realize moving target tracking algorithm using WSNs. Especially, in multi-target tracking, large overhead for communications can be occurred due to much more packets switching between the neighboring sensors and multi-hop communication, thus, moving target tracking task might not be performed without interruption.

Target tracking algorithms using WSNs proposed so far can be classified into 5 categories such as tree-based algorithm, cluster-based algorithm, prediction-based algorithm, mobicast message-based algorithm and hybrid clustering algorithm [1], [3]. If these algorithms are combined with prediction strategy, their performance will be improved [1]. In prediction-based algorithms, the next location of moving target is predicted based on the measured values of target location up to now, and in every prediction period, only sensors near the prediction location are activated, while the other sensors are in sleep mode to save energy.

Because only some sensors near the tracks of moving target are activated in these algorithms, the larger the prediction error which means the difference between the predicted location and actual location, makes the incorrectly designated sensors activated and makes sensors located in the place of target stay in the sleep mode so that it could bring about the high target missing rate.

Thus prediction-based moving target detection mechanisms, it becomes the most important challenge to search a solution in which less energy will be consumed guaranteeing high quality of target detection [4]. In other words, it becomes an important task to develop an algorithm in which a certain level of detection quality is guaranteed using less energy [5]. And hybrid clustering is a scheme in which on-demand dynamic clustering is integrated into WSN based on scalable static clustering to resolve what is called "boundary problem" in that target missing rate is increased when a target passes the boundary of static clusters or moves along it. In this scheme, the information about the target is shared by sensors of different clusters, thus the quality of target tracking will be guaranteed by tracking the target smoothly according to the movement of the target and trade-off between energy consumption and local sensor collaborations will be well performed [3]. However, hybrid clustering schemes would bring about huge consumption of energy because of overforming and overdismiss of dynamic clusters when the target moves in zigzags between a static cluster and a dynamic cluster in the boundary area or it moves continuously along the boundary area. From this cause, in hybrid clustering scheme, it becomes the most important challenge to search a solution in which a certain level of target tracking quality will be guaranteed while less energy is consuming. To form dynamic clusters deliberately in proper location and proper time, a novel DCH selection algorithm is proposed to prevent the transmission of redundant messages in [6], and in HCMTT [4], the number of redundant dynamic clusters is reduced applying a merging technology instead of selecting DCH properly.

In this paper, a prediction-based hybrid clustering scheme is proposed to prevent consumption of large amount of energy during the transmission of extra control message or target detection data in hybrid clustering due to the frequent forming and dismiss of dynamic clusters such as zigzag movement or movement of the target according to the boundary area.

The rest of this paper is organized as follows. In section 2, reviews of related schemes about moving target tracking are given. In section 3, assumptions of the system and issues in hybrid clustering in some special cases are described. A proposed algorithm is described in section 4 and it is analyzed in section 5. Simulation results are discussed in section 6 and the conclusion of the paper is given in section 7.

2. RELATED WORKS

In this section, reviews of moving target tracking algorithms using WSNs proposed up to now are given with examples.

In tree-based algorithms, a tree structure is set between sensors directly when the target is detected. Each sensor determines its parent and then messages are exchanged between sensors that are needed to remain the existing tree structure or to be dismissed from it. A sensor which detected the movement of target delivers the target information to its parent and it is delivered to the root of the tree finally. A root should be responsible for delivering the sensing report to a base station. There are representative algorithms such as DCTC which tracks the target by forming or dismissing clusters dynamically, DOT [8] in which the sensor closest to a moving target is recognized by exchanging the space-related information in Gabriel graph, a tree-based scheme without statistics of mobility in which the fact is proved that searching the optimal message transmission tree that requires the least amount of sensing report is NP-hard problem [9], etc.

In cluster-based algorithm, the network topology dividing into clusters is constructed before tracking the target or it is constructed directly after detection of moving target. This algorithm often has hierarchical structure in which the sensing information is transmitted from current cluster head to the next head once it is collected in current head from members inside of each cluster. Finally, this information is relayed to the base station by performing this procedure repeatedly. It is difficult to acquire the dynamic characteristics of the target changing according to time when the network topology is static. The representatives of this case are LEACH [10] which is most often used as a standard of comparison between clustering protocols, and RARE [11] which is a pure static clustering target tracking scheme where sensing data is divided into low quality data and redundant data, etc. In dynamic clustering algorithm which overcomes the shortage of static clustering algorithm, dynamic clusters appear around the target when the target enters the monitoring area. The DPT [12] is the typical algorithm which tracks the target purely by forming dynamic clusters which makes the total energy consumption to minimize constructing optimal set of active sensors using prediction.

In prediction-based algorithm, the next location of the moving target is predicted based on the recorded history values of target location measurements, while sensors close to the predicted location of target are active and the others are sleep to save energy in each prediction period [13], [14]. In this scheme, noise is accompanied in measurement, prediction or processing stages, thus it becomes the most important problem to develop a prediction scheme where the deviation of prediction result reaches the Cramer-Rao Lower Bound (CRLB) [15]. Many prediction-based algorithms using different prediction methods has been proposed such as extended Kalman filtering (EKF) using Kalman filters [16], Unscented Kalman filtering (UKF) [17], a scheme using particle filters [18], a scheme combined with maximum likelihood estimation [19], etc. and a simple scheme [20] which evaluates the prediction processing affecting energy consumption as the number of operation as well. All the prediction schemes are prone to estimation error which might lead to activation of the wrong sensors [4].

The hybrid clustering scheme constructs dynamic cluster on the static structure which is already formed and where the target is monitored, switching target tracking tasks between static clusters and dynamic clusters. In this scheme, the dynamic cluster is constructed on the already formed static structure according to the request to track the moving target smoothly between neighboring clusters or on the boundary area of a cluster. Whether the hybrid clustering becomes an effective replacement in case of target missing rate and energy consumption respectively depends on the practical application and resulting overhead should be considered in its usage. There are many schemes such as CODA [21] where new dynamic cluster is appeared when the increment of target size and movement in the boundary is detected, HCTT [3] which is a single target tracking scheme where boundary sensors of static clusters take part in formation of dynamic clusters, HCMTT [4] to which merging technology is applied for multi-target tracking, etc. However, in hybrid clustering schemes such as HCTT and HCMTT, huge amount of energy can be consumed due to the frequent formation and dismiss of redundant dynamic clusters when the target zigzags between a static cluster and a dynamic cluster or when the movement of the target makes overlapped dynamic clusters to be formed continuously.

3. ASSUMPTIONS AND PROBLEM STATEMENT

3.1 ASSUMPTIONS

We assume that the wireless sensor network formed by N sensors randomly deployed in 2-D area is divided into static clusters (SC). We assume that the total tracking duration time of tracking WSN is divided into time intervals and that the target is static in a time interval. Furthermore, we assume that sensors are passive like acoustic sensors and each sensor knows its location by self-localization algorithm and that the time is synchronized in overall WSN to make it possible to give a correct decision for target localization.

Also we assume that each sensor can adjust its transmitting power continuously ranging from minimum to maximum according to the transmission distance. And we call sensors inside of one sensor which has identical communication range of one-hop neighboring sensor. Each sensor has sensing radius and a circle of radius around the target is called the monitoring area. Each sensor can be in one of active state, listening or sensing state and idle state. Sensors in active state can transmit data to and receive data from neighboring sensors, sense the target. Sensors in listening or sensing state can only receive data from neighboring sensors or conduct sensing. Sensors in idle state are sleeping the most of time, so it can conduct neither sensing nor communication. The state that consumes least energy is the idle state. Sensors in idle state are waken periodically after predefined time and change their state from idle to listening. After that, they listen to channels and are converted to active state on receiving target detection message.

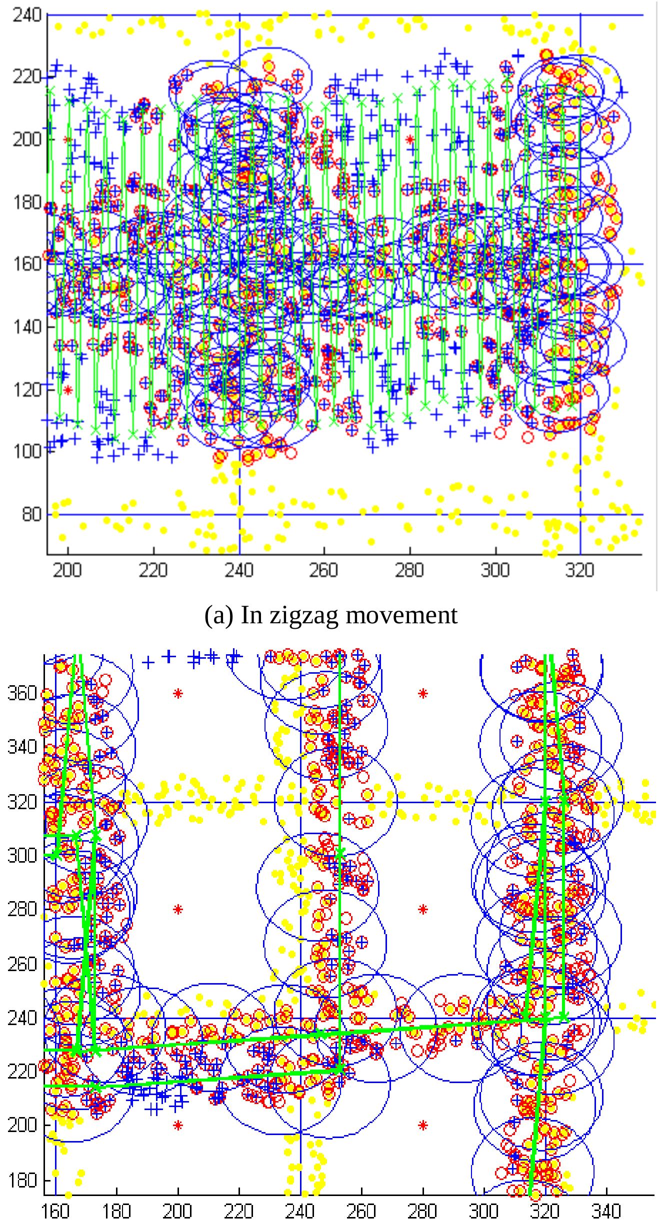
3.2 PROBLEM DESCRIPTION

The boundary problem in target tracking using WSNs occurs when the target passes the boundary between multiple clusters or moves along it. In other words, this problem occurs when the collaboration between local sensors becomes imperfect or unreliable because sensors that can monitor the target are members of different clusters. Once the boundary problem occurs, it can lead to uncertainty of target location detecting, furthermore, the target might be lost. The hybrid clustering is proposed just to resolve this problem. In this scheme, static clusters are responsible for target tracking and collaboration between local sensors when the target moves within the clusters, while dynamic clusters are formed after starting dynamic clustering to solve boundary problems once the target approaches the boundary of clusters. On-demand dynamic clusters are dismissed when the target is out of the boundary far away. In this scheme, static and dynamic clusters track the target efficiently in turn when the target moves within WSN. However, hybrid clustering schemes such as HCTT and HCMTT form and dismiss redundant dynamic clusters frequently when the target moves in zigzags between a static cluster and a dynamic cluster as shown in Fig.1 a), thus it can lead to extra transmission of control message and large consumption of energy.

Furthermore, as shown in Fig.1 b), successive forming and overlapping of dynamic clusters when the target keeps moving along the boundary area can lead to the extra data transmission from sensors to the base station, thus huge amount of energy can be consumed. In the snapshot above, dynamic clusters are organized in grids and cluster head (CH) is located at the center of them. And yellow nodes denote boundary sensors of static clusters, green track denotes the moving track of the target, large blue circles denote dynamic clusters formed according to the moving target, blue crosses denote internal sensors of static clusters which sent sensing report to static cluster head (SCH), red circles filled with yellow color denote members of dynamic clusters originated from boundary sensors of static clusters, red circles with blue cross denote sensors that sent sensing report to dynamic cluster head (DCH) and SCH, red circles filled with white color denote internal sensors of static clusters that sent sensing report to DCH.

4. PROPOSED ALGORITHM

We assume that initial WSN is divided into static clusters (SC) using several clustering algorithms such as LEACH [10] or DHCR [22] that can be properly applied to applications. Then, static cluster heads (SCH) and their members are determined based on multi-criterion such as distance, residual energy, proximity to its neighboring nodes, etc. And the concept of boundary sensor in HCTT or HCMTT is applied without modification when a certain sensor of a static cluster is defined as the boundary sensor of that cluster and when it has sensors of the other static cluster in the range of sensing radius at the same time. That is, boundary sensors can be defined as members of static clusters that have members of the other static cluster as their neighboring sensors. Internal sensors send the sensing data to their SCHs when the target moves in the internal area of static clusters (except the boundary area where boundary sensors are located).



(a) In zigzag movement

(b) In the movement along the boundary of static clusters

Fig.1. Snapshot of hybrid clusters when the target moves along the specific track

When the target approaches boundary sensors of static clusters and dynamic clusters (DC) are formed according to the request before the arrival of the target so that the target tracking can be continued. By doing so, successive tracking of the target will be ensured in collaboration between heads of static and dynamic clusters. Especially, when the construction cost of dynamic clusters are increased due to the zigzag movement of the target between a static cluster and a dynamic cluster or it moves along the boundary of static clusters, predicted clusters (PC) are formed before arriving of the target and the energy consumption is reduced significantly.

In the following part, we describe the plan of processes about prediction-based hybrid clustering proposed in this paper in a more detailed way.

4.1 FORMATION AND DISMISS OF PREDICTED CLUSTERS

As shown in Fig.1(a), once reaching the predefined threshold of the number of zigzag movement of the target between a static cluster and a dynamic cluster is reported from boundary sensors of the dynamic cluster, DCH recognizes the formation of predicted cluster(called predicted cluster in this paper in order to distinguish this dynamic cluster from the case of that prediction is not used, considering that this dynamic cluster is formed by prediction method), that is, the optimal set of active sensors using prediction method.

Meanwhile, when the target keeps moving along the boundary of static clusters, once the number of dynamic clusters exceeds a predefined threshold, the formation of predicted clusters begins. Needless to say, in this case, the number of overlapping of dynamic clusters between DCHs should be exchanged. DCH sends Urgent_PCH_Msg to the boundary sensor closest to the target which has the largest residual energy and elects it as CH of predicted cluster. The elected PCH determine 3 or more CM sensors to form predicted clusters by relatively precise broadcasting Join_Msg as a radius (set 3/4 in the initial forming of PCH) of predicted area using cross layer power control technique proposed in [23] etc. Sensors responding to this message among sensors inside are set as CMs. After that, PCH runs prediction-based clustering algorithm, PCDA described at the following in collaboration with PCM nodes, thus the size and CH and 3 CM nodes of the next predicted cluster, the length of tracking interval time or the length of prediction interval time are determined as the result of it. Furthermore, CH of the next predicted cluster sends Handoff_Req_Msg to DCH and SCH. DCH and SCH respond to this message as Handoff_Conf_Msg, thus they hand over target tracking task to new active predicted cluster officially and dismiss their members out of this predicted cluster and let them in idle state.

Predicted clusters are dismissed when they become useless for target tracking. In other words, it is begun when the target is out of the range of the predicted cluster so its members receive a signal weaker than a predefined threshold power from the target. This phenomenon can be occurred as well when CM nodes miss the target in the measurement stage of the prediction-based clustering algorithm and when the target is missed due to errors in prediction. The recovering mechanism in the case of missing target is described at the following part of the paper. In such two cases, member node of the predicted cluster detected this phenomenon sends Wakeup_Msg to neighboring members of the static cluster instantly so it wakes them. From this moment, the static cluster becomes active and it takes charge of target tracking. In that case, if PCH does not receive sensing report during one tracking interval time (Δt), it sends Resign_Msg to their members, so the predicted cluster is collapsed.

4.2 PREDICTION BASED CLUSTERING ALGORITHM

Proposed algorithm in this subsection is called prediction based clustering algorithm with dynamic adjustment (PCDA) because it adjusts the size of the predicted cluster and the length of the tracking interval time dynamically according to the prediction error and the speed of the target. This algorithm works in three stages such as measurement stage, prediction stage and activation stage. The description of the control messages used in this algorithm is shown in Table.1.

Table.1. Description of the control message

message	description
Measure_Msg	Tuple(node_ID, node_coordinate, node-distotarget)
Predicted_Msg	Tuple(CH_ID, target_nextcoordinate, nextPC_radius)
Utility_Msg	Tuple(node_ID, node_utility)
NextCH_Msg	Tuple(CH_ID, nextPC_radius, nexttracking_interval, target_nextcoordinate, prediction_error)
NextCM_Msg	Tuple(nextCH_ID, nexttracking_interval)

4.2.1 Measurement Stage:

In this stage, CM nodes inside of the current predicted cluster measure the current location of the target in collaboration with CH nodes.

Once CM node detects the target, it makes Measure_Msg including its ID and coordinates, distance to the target and sends it to CH node of the predicted cluster. CH, received Measure_Msg, calculates the location of the target using the following trilateration [21].

$$\begin{cases} \sqrt{(x_t-x_i)^2+(y_t-y_i)^2}=d_{i,t} \\ \sqrt{(x_t-x_{i+1})^2+(y_t-y_{i+1})^2}=d_{i+1,t} \\ \sqrt{(x_t-x_{i+2})^2+(y_t-y_{i+2})^2}=d_{i+2,t} \end{cases} \quad (1)$$

$$(x_t, y_t)^T=(A^TA)^{-1}A^Tb \quad (2)$$

$$A=\begin{bmatrix} 2(x_i-x_{i+2}) & 2(y_i-y_{i+2}) \\ 2(x_{i+1}-x_{i+2}) & 2(y_{i+1}-y_{i+2}) \end{bmatrix} \quad (3)$$

$$b=\begin{bmatrix} x_i^2-x_{i+2}^2+y_i^2-y_{i+2}^2+d_{i+2,t}^2-d_{i,t}^2 \\ x_{i+1}^2-x_{i+2}^2+y_{i+1}^2-y_{i+2}^2+d_{i+2,t}^2-d_{i+1,t}^2 \end{bmatrix} \quad (4)$$

where $d_{i,t}$ is the distance between sensor s_i and the target in t tracking interval time or t prediction period, (x_i,y_i) is the coordinate of sensor s_i, (x_t,y_t) is the coordinate of the target in t prediction period.

4.2.2 Prediction Stage:

In this stage, the next location of the target, the size of the next predicted cluster and the length of the tracking interval time are determined based on the measurement information of the target up to current state. Based on such information, the next predicted cluster is formed for the tracking of the next location of the target.

- **Prediction of the next location of the target**

There are several mobility models of the target [24-27]. In random way point (RWP) model, which is the most general mobility model, M_{n+1} is randomly elected not concerning the history and current location and the size of the next speed V_n is also randomly elected between v_{min} and v_{max} in that way when the target moves from a certain way point M_n inside of the monitoring area to the next way point M_{n+1}. And we assume that the target moves from M_n to M_{n+1} with identical speed V_n. There are two RWP models with or without pause time according to whether the target stops in every way point or not. In RWP model with pause time, the pause time is used before changing the speed and direction of the target, and this model is used in this paper as well.

In random way point and constant acceleration model (RCAM) [20], it is assumed that the target moves in random direction for an arbitrary time with the speed and acceleration in the range of [0, v_{max}] and [0, a_{max}], respectively. In such random way point model, the pause time is used before changing 3 parameters such as speed, direction and acceleration, and the target moves directly with constant acceleration when it moves from M_n to M_{n+1} contrary to the previous RWP model.

Based on such mobility model, CH node calculates the predicted location (x_{t+1}, y_{t+1}) in the next prediction period (t+1) of the target using the following prediction schemes in prediction period t.

In RWP model: The speed and the direction of the target is predicted as follows.

$$v = \frac{\sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2}}{t_t - t_{t-1}} \tag{5}$$

$$\theta = \cos^{-1}\frac{x_t - x_{t-1}}{\sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2}} \tag{6}$$

where (x_{t-1}, y_{t-1}) and (x_t, y_t), t_{t+1} and t_t are coordinates and times of target location measurements of the measured target in (t-1) prediction period and t prediction period, respectively. Once the speed and direction of the target are predicted, the location of the target is calculated as follows.

$$x_{t+1} = x_t + v\Delta t_t \cos\theta \tag{7}$$

$$y_{t+1} = y_t + v\Delta t_t \cos\theta \tag{8}$$

In RCAM: The target moves with constant acceleration, thus the speed of the target in the next prediction period is predicted as follows. The predicted speed and direction of the target in (t+1) prediction period are calculated as follows when the prediction error equals to $\Omega_t \in [-\alpha, \alpha]$.

$$v_{t+1} = 2v_t - v_{t-1} \tag{9}$$

$$\theta = \tan^{-1}\frac{y_{t+1} - y_t}{x_{t+1} - x_t} \tag{10}$$

The next predicted location of the target is as follows.

$$x_{t+1} = v_{t+1}\Delta t_t \cos\theta + x_t \tag{11}$$

$$y_{t+1} = v_{t+1}\Delta t_t \cos\theta + y_t \tag{12}$$

If the prediction error becomes $\Omega_t \notin [-\alpha, \alpha]$, the predicted speed is decomposed into x-axis component and y-axis component. Denoting x-axis component and y-axis component in (t+1)$^{\text{th}}$ prediction period as $v_{x,t+1}$ and $v_{y,t+1}$ respectively, they are calculated as follows.

$$v_{x,t+1} = 2v_{x,t} - v_{x,t-1} \tag{13}$$

$$v_{y,t+1} = 2v_{y,t} - v_{y,t-1} \tag{14}$$

The next predicted location of the target is as follows.

$$x_{t+1} = v_{x,t+1}\Delta t_t + x_t \tag{15}$$

$$y_{t+1} = v_{y,t+1}\Delta t_t + y_t \tag{16}$$

- **Determination of size of the next predicted cluster**

CH calculates the prediction error for determination of size of the next predicted cluster, that is, the predicted area around the target using recorded values of coordinates of the target location such as the current measurement location of the target. First, the prediction error is defined as the deviation between the measured value of the target coordinates in current prediction period and the predicted value of the coordinates of the current target. And it is used as a parameter evaluating the quality of target tracking. And the prediction error of the next prediction period is calculated as the arithmetic mean of the current prediction error and the prediction error of the previous prediction period as follows.

$$\begin{aligned}\Delta d_{next} &= (\Delta d_{current} + \Delta d_{previous})/2 = \\ &(\sqrt{(x_{m,t} - x_{p,t\text{-}1})^2 + (y_{m,t} - y_{p,t\text{-}1})^2} + \\ &\sqrt{(x_{m,t\text{-}1} - x_{p,t\text{-}2})^2 + (y_{m,t\text{-}1} - y_{p,t\text{-}2})^2})/2\end{aligned} \tag{17}$$

where $(x_{m,t}, y_{m,t})$ and $(x_{p,t-1}, y_{p,t-1})$ are measured value and the predicted value of the coordinates of the target in t^{th} prediction period and (t-1)$^{\text{th}}$ prediction period respectively. Based on this prediction error, the size of the next predicted cluster, that is, the radius of the predicted cluster around the location of the target during the next prediction period is calculated as follows.

$$R_{next\_PC} = \Delta d_{next} + \frac{3r_s}{4} \tag{18}$$

where R_{next_PC} is the radius of the next predicted cluster and r_s is the sensing radius of sensors. This formula shows that the size of the next predicted cluster is adjusted dynamically according to the prediction error expressing the quality of target tracking. In other words, the size of the predicted cluster is increased when the quality of target tracking is decreased by large prediction error, so more members can attend target tracking.

- **Determination of the tracking interval time in the next prediction**

The tracking interval time Δt_{t+1} in the next prediction is the time interval between 2 successive tracking point (measurement point or sensing point), and the setting of it affects both of energy consumption and the quality of the tracking. The speed of the target has the most significant influence to the determination of the tracking interval time calculated by CH, thus the tracking interval time is decreased when the speed of the target is fast and it is increased when the speed of the target is low to reduce the energy consumption. In other words, the tracking interval time is not constant and it is adjusted dynamically according to the speed of the target. Using Eq.(5) referred in the above, it is possible to determine the instant speed of the target, so Δt_{t+1} is calculated as follows.

$$\Delta t_{t+1} = \frac{S}{v} \tag{19}$$

where

S is the distance that the target can move during Δt_{t+1} and it is often predefined as a distance that the target can move at maximum speed during 1s.

On the whole, in prediction stages up to now, current CH self-calculates parameters that can be sent to neighboring relaying sensors inside of the communication range for successive tracking of the target such as the coordinates of the target in the next prediction period, the size of the next predicted cluster, tracking interval time of the next prediction, etc.

- **Formation of the next predicted cluster**

In this stage, the next predicted cluster is formed officially based on the information for the next predicted clustering achieved in previous stages.

First, CH makes Predicted_Msg including the coordinates of the next predicted location of the target and the size of the next predicted cluster, and broadcasts it to its neighboring sensors. Each sensor received Predicted_Msg calculates the distance $d_{i,t+1}$ from itself to the target in the next prediction period. Sensors with lower distances than the size of the next predicted cluster evaluate the usability for target tracking using the following formula inspired by [28].

$$w_i = w \times \frac{E_i^{res}}{E_i^{\max}} + (1-w) \times \frac{1}{1+d_{i,t+1}^2} \tag{20}$$

where

w_i is the usability parameter which denotes the extent that can be used to target tracking in the next tracking period,

$E_i^{\max}$ is the initial maximum energy of sensor s_i,

E_i^{res} is the residual energy of sensor s_i.

w is the weight considering the different influence of parameters affecting the calculation of usability and it is elected in the range [0,1].

This weight can be adjusted to be reached the acceptable result through simulations. We will consider it in section 6. Sensors the distance to the target is smaller than the size of the next predicted cluster make Utility_Msg including such usability and its ID, and respond Predicted_Msg from CH.

When CH receives Utility_Msg, it sorts usabilities by size to elect proper sensors for target tracking. Next, it elects sensor with secondly large usability as CH of the next predicted cluster, and expects that next three sensors in order of size are elected as CM nodes.

4.2.3 Activation Stage:

In activation stage, current CH makes NextCH_Msg including the size of the next predicted cluster, tracking interval time in the next prediction, current measured location of the target, current prediction error, etc., and sends it to CH of the next predicted cluster. Besides that behavior, it sends NextCM_Msg including ID and Δt_{t+1} of CH of the next predicted cluster to CM nodes of the next predicted cluster. When CH nodes and CM nodes of the next predicted cluster receive NextCH_Msg and NextCM_Msg respectively, they are in active state in the current prediction stage before the target arrives. CH goes back to idle state finishing its behavior in the current tracking stage.

Activated CH nodes and CM nodes of the next predicted cluster move to the measurement stage again when the target is moving into the area of the cluster, and relay the location of the target to the base station running PCDA algorithm repeatedly, measuring the current location of the target. The pseudo code of PCDA algorithm is shown in Algorithm.

Although the above algorithm is efficient, there are some cases of missing the target including the case that the direction of the target movement is turning into the opposite side, so the recovering mechanism is described briefly. Although there are many recovering schemes, in our case, once the target is missed, the current predicted cluster activates its one-hop neighboring sensors firstly so that they can attend target tracking. That is, it activates all sensors in the communication range of r_c around CH node. If the target is not detected yet after the activation, the base station makes all sensors of WSN to attend the target detection.

Algorithm. Pseudo code of PCDA

Begin (Prediction based Clustering algorithm)

Stage I: Target location measurement by Current Predicted Cluster (CPC)

1. **if** sensor node in CPC detect the target **then**
2. send a *Measure_Msg* to current CH
3. **end**

Stage II: Next Predicted Cluster (NPC) construction

1. **for** CH in CPC receiving *Measure_Msg* **do**
2. Estimate speed and direction of target in next tracking interval and predict the next location of target
3. Determine R_{next_PC}, radius of NPC based on prediction error
4. Determine Δt_{t+1}, the next tracking interval time interval based on target speed
5. Generate a *Predicted_Msg* and broadcast to neighbor nodes
6. **for** each sensor node i receiving *Predicted_Msg* **do**
7. **if** $d_{i,t+1} \le R_{next\_PC}$ **then**

   calculate utility w_j and reply a *Utility_Msg*
8. **end**
9. **for** CH in CPC receiving *Utility_Msg* **do**

   Sort utilities $\{w_i\}$

   Select the sensor node $i = \arg\max_i w_i$ to CH in NPC and three sensor nodes with next orders of the highest utility to CMs in NPC

Stage III: Activation by CH in CPC

1. Generate a *NextCH_Msg* and a *NextCM_Msg*, send to CH and CMs in NPC and activate them

2. After Δt_{t+1}, NPC become CPC and iterate algorithm

End

5. ANALYSIS OF THE PROTOCOL

To analyze the protocol, we assume that n static clusters are randomly deployed throughout the overall monitoring area and all sensors are deployed according to Poisson distribution with the density of λ. Thus, the mean number of neighboring sensors in the case of static and dynamic cluster equals to $\lambda\pi r_c^2 - 1$.

Theorem 1. The computational complexity of the predicted clustering is O(1) and it is much more smaller than the computational complexity $\lambda\pi r_c^2 \times (\lambda\pi r_c^2 - 1)$ of the dynamic clustering.

Proof. The computational complexity of the predicted clustering is due to the computation of the next predicted cluster size produced by prediction error computation in prediction stage and due to the computation of the next tracking interval time produced by prediction of the target speed. Such computations are performed only in PCH nodes. Therefore, the computational complexity of the predicted clustering is O(1). However, in dynamic clustering, the discrimination should be performed for each sensor to determine whether it becomes the boundary node of dynamic cluster for its all neighboring sensors in the worst case, thus the total computational complexity is very high as $\lambda\pi r_c^2 \times (\lambda\pi r_c^2 - 1)$. Such considerations also let us could observe the predicted clustering and dynamic clustering are local processes independent of the size of WSN.

Theorem 2. The overhead complexity of the control message necessary for the predicted clustering is lower $2\lambda\pi r_c^2$ or more than that of control message necessary for the dynamic clustering.

Proof. The predicted clustering is performed through three stages such as measurement stage, prediction stage and activation stage, and the overhead of total control messages for the next predicted cluster formed through these stages is as follows in the worst case.

3(Measure_Msg)+ $(\lambda\pi r_c^2 - 1)$ (Predicted_Msg)+ $\lambda\pi(3r_s/2)^2$ (Utility_Msg)+1(NextCH_Msg)+3(NextCM_Msg)≈ $(\lambda\pi r_c^2 - 1)$ +<2 $(\lambda\pi r_c^2 - 1)$

where X(Y) denotes that the number of Y control messages is X. On the other hand, the dynamic clustering is performed through several stages such as CH election stage, dynamic cluster formation stage, boundary sensor formation stage in dynamic cluster, and one dynamic cluster is formed as a result of it. Then, the overhead of the total control message is as follows.

2$(\lambda\pi r_c^2 - 1)$ (eletion_Msg)+ $(\lambda\pi r_c^2 - 1)$ (recruit_Msg)+ $(\lambda\pi r_c^2 - 1)$ (confirm_Msg)+ some control messages for the replacement≈4 $(\lambda\pi r_c^2 - 1)$

2$(\lambda\pi r_c^2 - 1)$ (eletion_Msg)+ $(\lambda\pi r_c^2 - 1)$ (recruit_Msg)+ $(\lambda\pi r_c^2 - 1)$ (confirm_Msg)+some control messages for the replacement≈4 $(\lambda\pi r_c^2 - 1)$

From the above considerations, we can observe that the communication complexity of the predicted clustering is Θ(1) the same as that of the dynamic clustering and its overhead is lower than that of the dynamic clustering approximately $2\lambda\pi r_c^2$ or more.

Theorem 3. The number of active sensors in the predicted cluster is 16/9 or more times smaller than that of the dynamic cluster.

Proof. The mean number of sensors in one dynamic cluster is $\lambda\pi r_c^2$ and that of the predicted cluster is $\frac{9}{4}\lambda\pi r_s^2$ in maximum when the prediction error is maximized because the size of the predicted cluster is in the range of $[3r_s/4, 3r_s/2]$. The mean number of the dynamic cluster is 16/9 or more times larger than that of the predicted cluster because the relation $r_c \geq 2r_s$ is valid. Moreover, considering that only about four sensors are activated including CH node not all sensors inside of the predicted cluster, saving of the energy due to the activation becomes larger.

Theorem 4. The computation complexity and communication complexity of target tracking by the prediction based hybrid cluster are O(n).

Proof. The computation and communication complexity of target tracking protocol based on the hybrid cluster using request combination of the static and dynamic cluster are O(n) [3]. According to theorem 1 and 2, the computation and communication complexity of the predicted clustering are Θ(1), so that of the prediction based hybrid clustering is also O(n).

6. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed PCDA algorithm through simulations. First, the simulation environment is described, and second, performance measurement parameters are described, and last, the experimental results about performance measurements are given and further analyzed.

6.1 SIMULATION ENVIRONMENT

All simulations are performed using MATLAB. The simulation environment is configured as follows. A total of 2000 sensors are randomly deployed in the area of 400m×400m. Sensors are organized as static clusters according to the LEACH protocol. The transmission range of each node r_c, is set to $r_c/r_s \geq 2$ and r_s is fixed as 20m. Moreover, the energy consumption model is the same as adopted from [10] and [22]. Then, the energy consumed at transmitters is denoted as follows.

$$\begin{cases} E_r = lE_{elec} + l\varepsilon_{emp}d^4 + lE_{DA}, & if \;\; d > d_0 \\ E_r = lE_{elec} + l\varepsilon_{fs}d^2 + lE_{DA}, & if \;\; d \leq d_0 \end{cases} \quad (21)$$

where E_{DA} is the energy for the aggregation of data, E_{elec} is the energy consumed per bit for the circuits at the receiver, ε_{fs} and ε_{amp} are parameters depends on the amplifier model of the transmitter, *d* is a distance, d_0 is the threshold distance, *l* is the length of packets. The energy consumed at the receiver is denoted as follows.

$$E_r = lE_{elec} \tag{22}$$

The parameters of energy consumption model are shown in Table.2.

We assume that the target moves according to random way point models of RWP model and RCAM model considered at the prediction stage of PCDA algorithm. It is assumed that the moving speed of the target varies in the range of 0~15m/s, the maximum acceleration is 3m/s^2 in RCAM, the limit of the error angle of the predicted location is $\alpha = 15°$. Each point of simulated data is achieved from the mean value through one hundred experiments.

Table.2. Parameters used in simulations

Parameter	Value
Network Area	400m×400m
Number of sensors	2000
E_{elec}	50nJ/bit
E_{DA}	5nJ/bit/signal
ε_{fs}	10pJ/bit/m^2
ε_{amp}	0.0013pJ/bit/m^4
d_0	87m
l	500bytes
Initial energy of sensors	1-4J
size of control packets	8bytes

We used the following metrics to evaluate the performance of our proposed algorithm in simulations.

- Number of dead sensors–the number of exhausted sensors in energy during the overall target tracking
- Energy consumption–sum of energy consumed in all sensors of WSN
- Missing probability–the ratio of the total energy of tracking periods where the target is missed to the total target tracking interval time. It equals to the working time of the recovering mechanism and it is an important measurement parameter to evaluate the quality of the tracking.
- Mean distance of activated sensors–the mean distance between actual location of the target and activated sensors. It is used a measurement value to evaluate the quality of target tracking.
- Mean squared distance error of activated sensors–it is the mean squared error of the distance between the activated sensor at the predicted location of the target and the sensor closest to the actual location of it. It is used as the measurement value to evaluate the quality of target tracking.

In our simulations, the performance of the proposed PCDA algorithm in comparison with other schemes such as DPT which is the pure dynamic clustering scheme using prediction, low power prediction mechanism in [20] (denoted as LPPM in this paper), HCTT which is the hybrid clustering scheme for the single target tracking and HCMTT which is the hybrid clustering scheme for multiple target tracking.

6.2 SIMULATION RESULTS

6.2.1 *Effect of the weight w and target tracking interval time Δt to the performance of target tracking*

Effect of the weight *w*

The weight w is set considering the effect to usability of the residual energy and distance to the target and its proper value is chosen through simulations. In general, a lower w can give a higher usability to the sensor closer to the target rather than sensors with higher residual energy contrary to larger w. In our simulation, w is set to 5 values such as 0, 0.25, 0.5, 0.75 and 1.0. Simulation results about the variation of the number of dead sensors versus simulation time and that of missing rate are shown in Fig.2 and Fig.3 respectively.

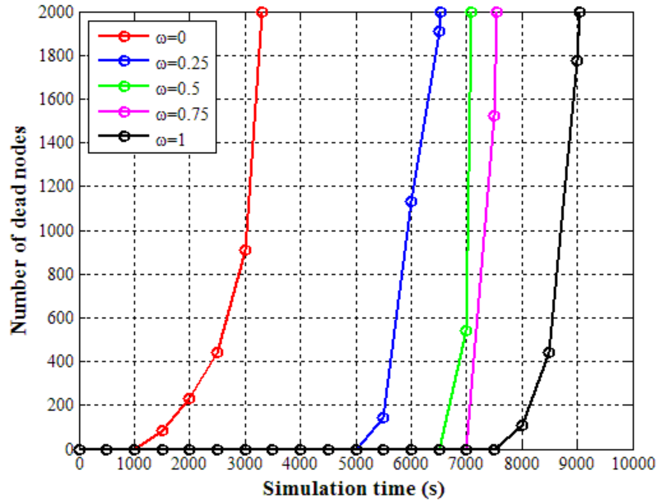


Fig.2. Number of dead sensors versus simulation time in different w

In Fig.2, we can observe that the number of dead sensors is in inverse proportion to w in general. That is, electing w to be approached to 0, the effect of residual energy factor is ignored, while that of distance factor is valued more, thus the number of dead sensors are increased because sensors with low energy closer to the target can attend target tracking.

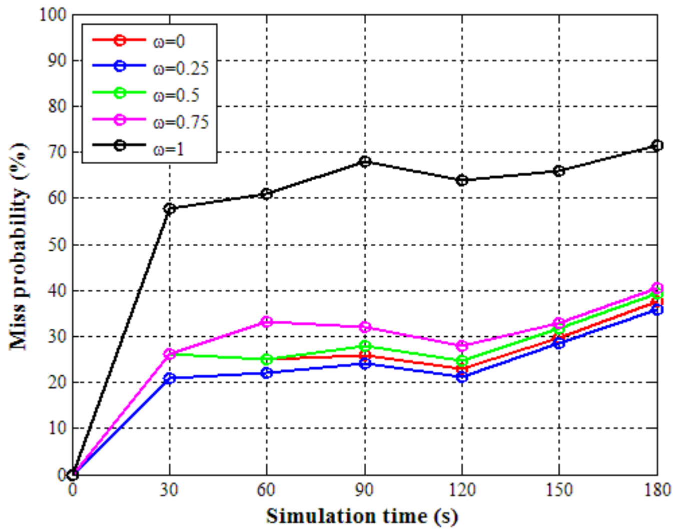


Fig.3. Missing probability versus simulation time in different w

Electing w to be approached to 1, the energy is more valued than the distance to the target, so the numbers of dead sensors are decreased while the missing probability is being increased because sensors with higher residual energy father from the target attend target tracking. This fact shows that sensors with low energy are dead sooner if too low w is used resulting imbalance of energy consumption, on the contrary, if too large w is used, the quality of target tracking is reduced because sensors closest to the target are removed from target tracking due to their low energy.

Observing curves of missing probability in Fig.3, we can see that the higher missing probability is due to the larger w. This is because the distance measurement error of the target and location prediction error depending on it is increased when w is increased reducing the effect of the distance factor, thus the prediction error is increased and missing probability is increased in the end. The increase of prediction error results in the increase of the predicted cluster size in PCDA, so more sensors are activated. Thus, the number of dead sensors is increased due to the increase of energy consumption.

Two curves in Fig.2 and Fig.3 show that the reasonable w is about 0.25~0.5 in the aspect of energy consumption and missing probability. So, the weight is set to 0.4 in the following simulations.

Effect of target tracking interval time Δt

Δt is target tracking interval time and it is the time interval between two neighboring measurement points or two neighboring prediction estimation points. The target has a constant acceleration when it moves according to RCAM, so it can move more distance than S (the distance moved for 1s with maximum speed. In our case, it is 15m) predefined in (19) for Δt. At that time, the prediction error will be increased, if Δt is not decreased. Generally, missing probability is decreased when Δt is decreased, and the energy consumption is decreased when it is increased. In our simulation, Δt is set to 1s, 2s, 3s, 4s, 5s and the energy consumption and missing probability is evaluated versus simulation time. Simulation results are shown in Fig.4 and Fig.5. Fig.4 and Fig.5 show that network lifetime is extended in great extent due to the reduced energy consumption but missing probability is increased when Δt is increased. Thus, we can achieve relatively better compromise between energy consumption and quality of tracking electing Δt in the range of 2-3s in our case.

6.2.2 Quality of Tracking:

In this subsection, the quality of tracking in PCDA is evaluated in comparison with other schemes using several measurement parameters such as mean distance of activated sensors, mean squared distance error of them, missing probability, etc.

First, the actual track of the target is shown in comparison with tracks predicted using PCDA, DPT, LPPM schemes when it in different Δt moves according to RWP and RCAM model. The simulation result shown in Fig.6 shows that the proposed scheme tracks the actual track more precisely than DPT and LPPM schemes. This is just because the size of predicted clusters and tracking interval time is adjusted according to the prediction error and speed of the target in PCDA scheme.

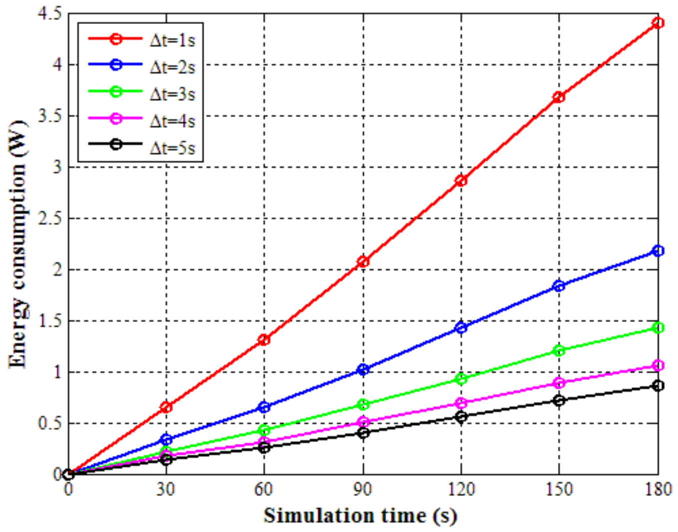


Fig.4. Energy consumption versus simulation time in different Δt

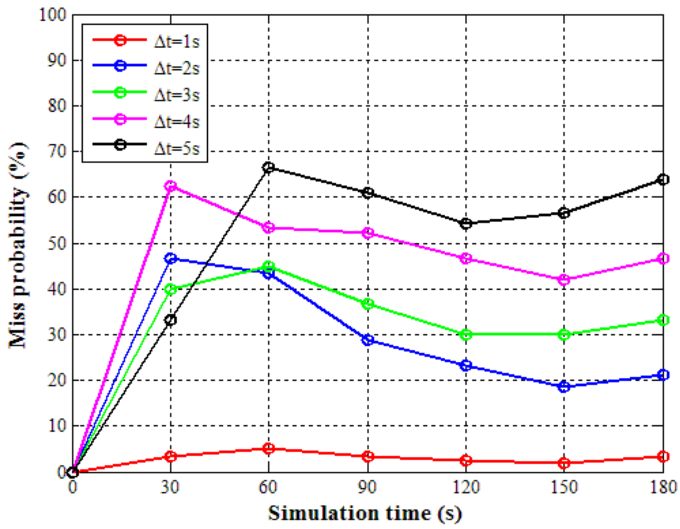
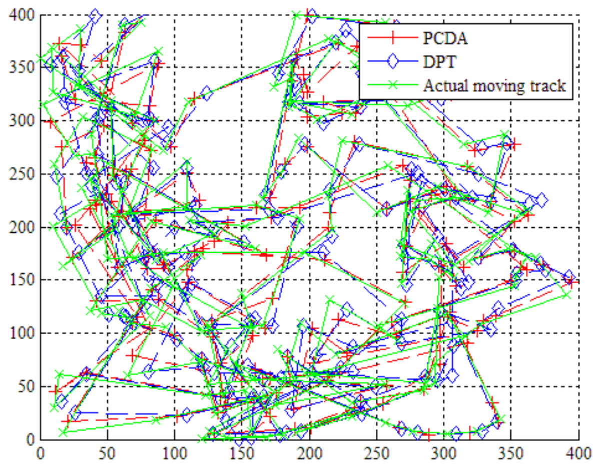
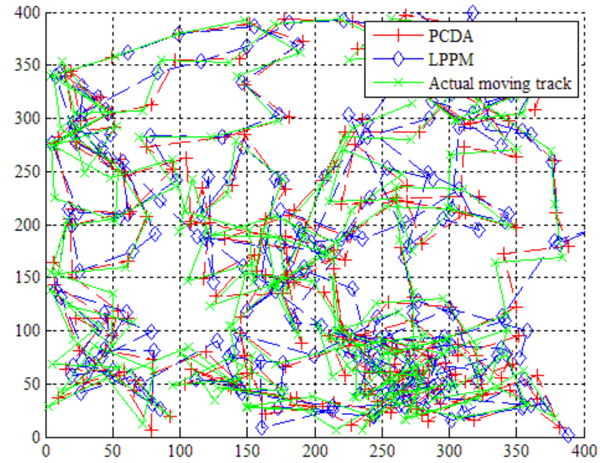


Fig.5. Missing probability versus simulation time



(a) In RWP model



(b) In RCAM model

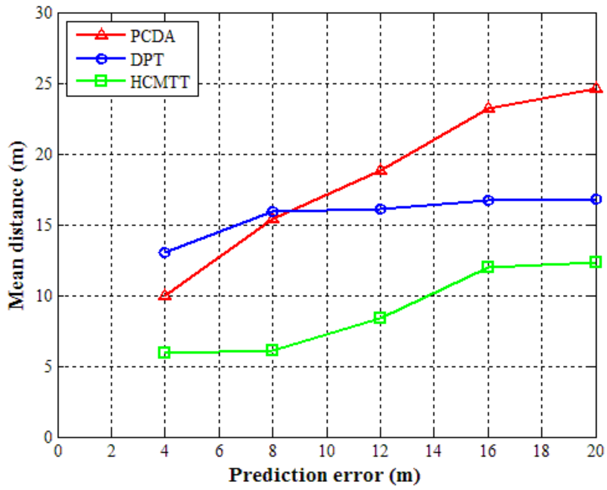
Fig.6. Comparison between actual track of the target and predicted track of it

The simulation results of mean distance of activated sensors, mean squared distance error of them, missing probability according to the variation of prediction error are shown in Fig.7, Fig.8, Fig.9 respectively when the target moves along the track shown in Fig.6 (it is simply called the general track distinguishing it from the track in Fig.1 follows the boundary or from zigzag track).

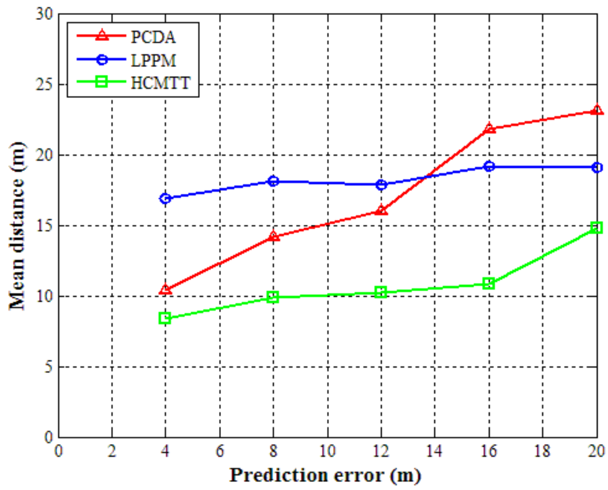
The Fig.7 shows that mean distance of activated sensors is not increased rapidly however the prediction error is increased because the mean distance between actual location of the target and activated sensors are independent to prediction scheme. Furthermore, in PCDA, DPT and LPPM schemes, the mean distance of PCDA is small with small prediction error, but it is increased than DPT because the proposed PCDA increases the size of the predicted cluster dynamically according to the prediction error when it is increased.

The simulation result in Fig.8 shows that mean squared error (MSE) of the distance between activated sensor in predicted location of the target and the sensor closest to the actual location of the target is always 0, but it is increased in DPT or LPPM, PCDA when the prediction error is increased.

The simulation result in Fig.9 shows that the prediction error has no effect to missing probability because sensors for target tracking are not activated based on the prediction scheme in HCMTT. Furthermore, it shows that in DPT and LPPM schemes, predicted cluster with constant size is used and tracking interval time is also constant, thus these schemes have higher missing probability. However, in DPT, we can observe that missing probability is lower than that in the proposed scheme when the size of its predicted cluster is larger than that of PCDA scheme.

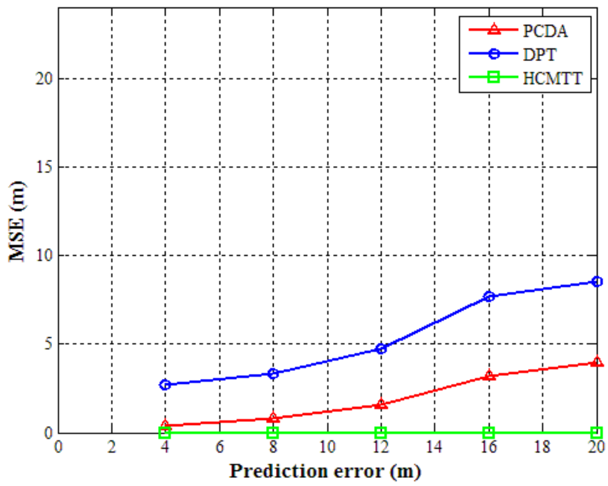


a) In RWP model

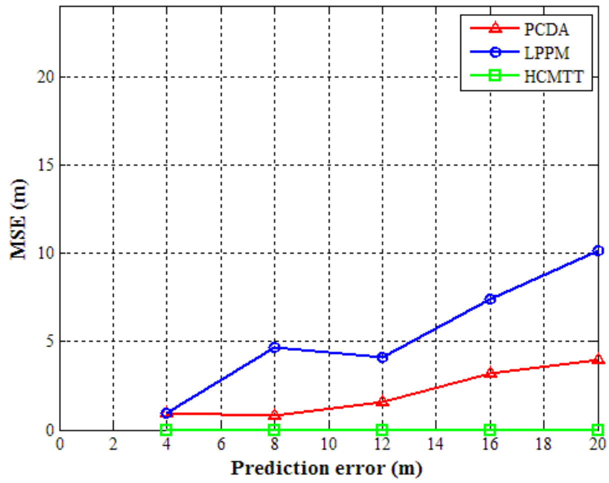


b) In RCAM model

Fig.7. Mean distance of activated sensors versus prediction error (in the case of general track)

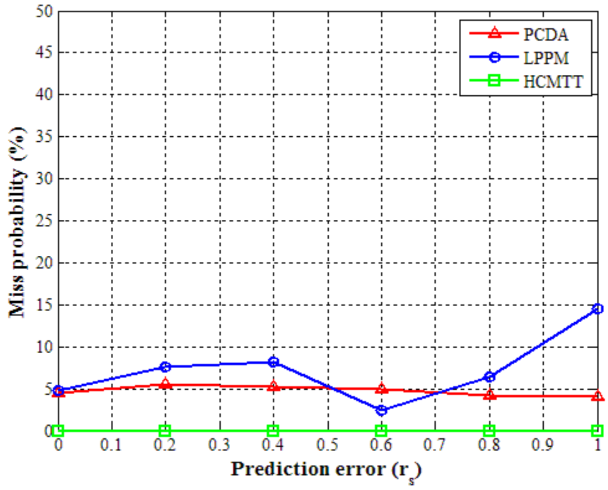


(a) In RWP model



(b) In RCAM model

Fig.8. Mean squared distance error of activated sensors according to prediction error (in the case of general track)



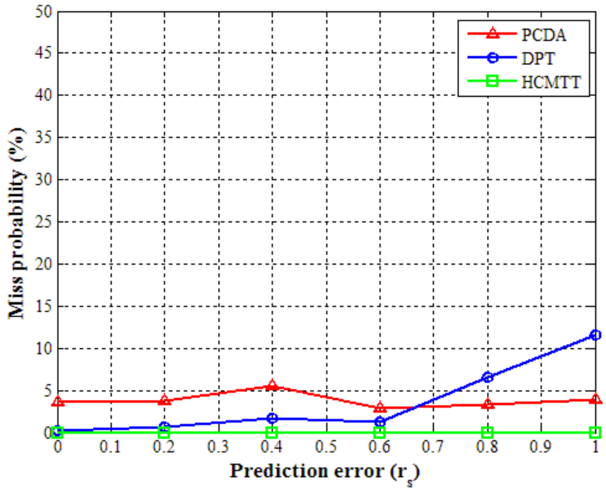
(b) In RCAM model

Fig.9. Missing probability according to the prediction error (in the case of general track)

Curves of missing probability versus simulation time when the target moves along a specific track are shown in Fig.10. This simulation result shows that missing probability of PCDA scheme does not exceed 10% when the target moves along specific tracks such as zigzag track or tracks lied in boundary. Missing probability of hybrid clustering schemes such as HCTT or HCMTT is 0 because they are independent to prediction scheme.

6.2.3 Energy Consumption:

In this subsection, energy consumption performance of PCDA is evaluated in comparison with other schemes. Fig.11 shows the simulation result of the variation of energy consumption when the target moves along the general track shown in Fig.6. Furthermore, Fig.12 and Fig.13 show that the variation of the number of dead sensors and energy consumption in proposed PCDA and other schemes versus simulation time when the target moves along the specific track shown in Fig.1.

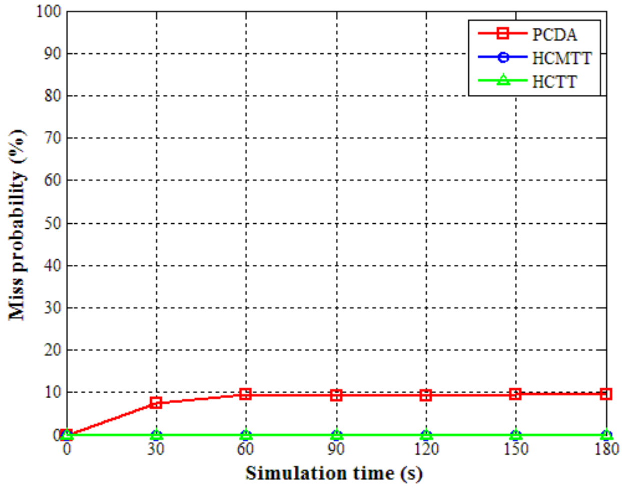


(a) In RWP model

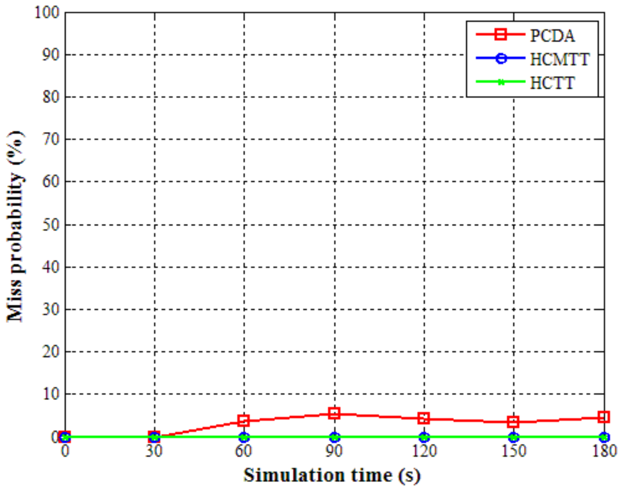
The simulation result in Fig.11 shows that energy consumption of DPT scheme is the lowest. In the simulation of DPT, we assumed that prediction based dynamic cluster is formed using only low beam in this scheme and the usage of recovering mechanism is not considered.

And we can observe that energy consumption of PCDA is higher than that of DPT, but it is lower than other schemes. In HCTT and HCMTT, the communication should be established with all sensors inside of communication range r_c and sensing range r_s, thus energy consumption is increased.

The simulation results in Fig.12 and Fig.13 show that the number of dead sensors increases rapidly due to the overforming and dismiss of dynamic clusters consuming huge amount of energy when the target moves along the specific track in hybrid clustering scheme without prediction. However, in PCDA, network lifetime is increased significantly due to the lower energy consumption.



(a) zigzag track



(b) track lied in boundary

Fig.10. Missing probability versus simulation time in the specific track

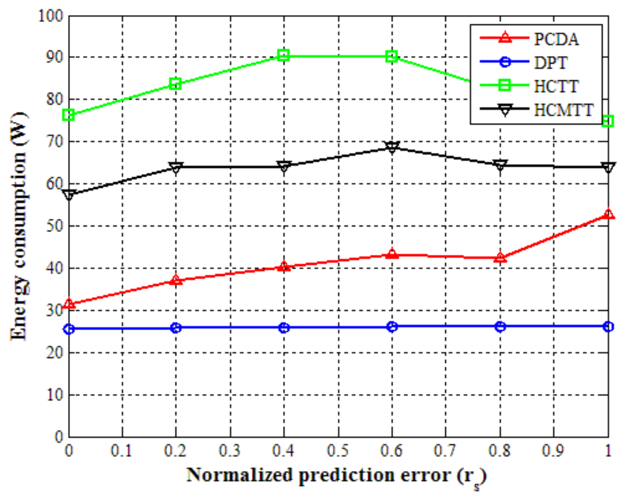
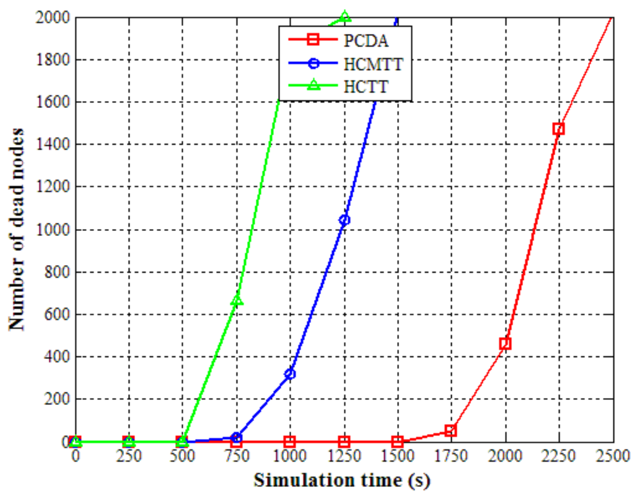
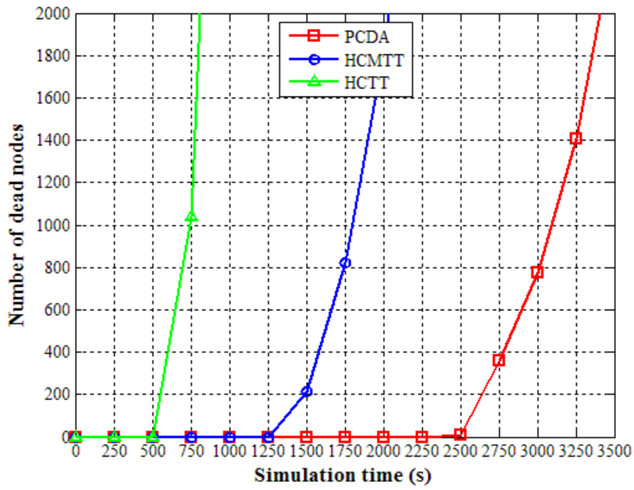


Fig.11. Energy consumption versus prediction error in the case of general track

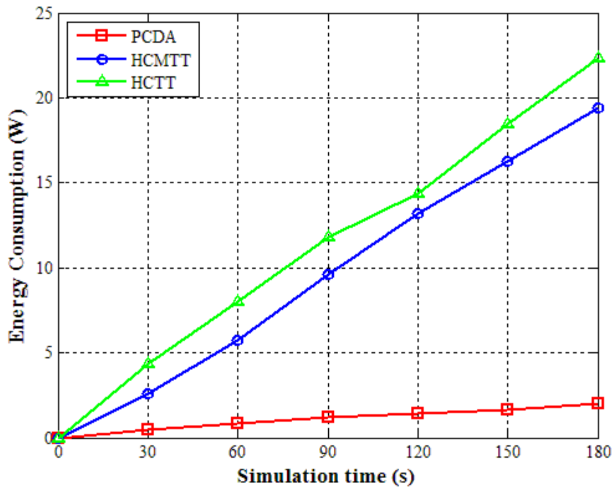


(a) zigzag track

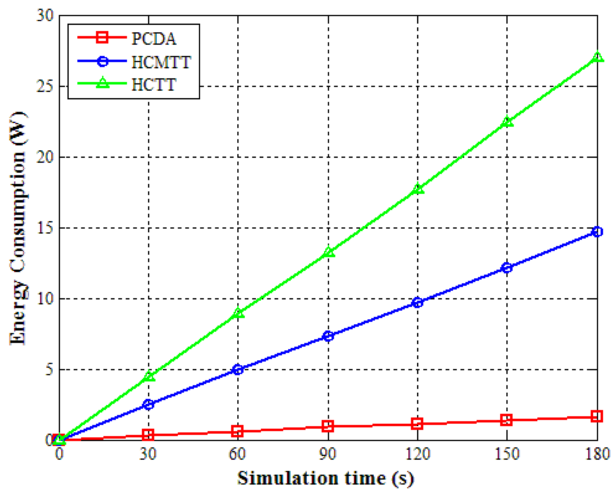


(b) track lied in boundary

Fig.12. Number of dead sensors versus simulation time in the case of specific track



(a) zigzag track



(b) track lied in boundary

Fig.13. Energy consumption versus simulation time in the case of specific track

7. CONCLUSION

In this paper, we proposed an energy efficient hybrid clustering scheme combined with prediction method for target tracking in WSNs. The prediction-based clustering algorithm PCDA is used in this scheme. In this algorithm, the size of the predicted cluster and tracking interval time are adjusted adaptively according to prediction error and the speed of the target, thus achieves the balance between the quality of tracking and energy consumption. Simulation results showed that the proposed scheme can greatly decrease energy consumption and extend network lifetime in great extent guaranteeing a certain level of the quality of tracking when the large amount of energy is consuming due to overforming and dismiss of dynamic clusters.

REFERENCES

[1] M. Naderan, M. Dehghan, H. Pedram and V. Hakami, “Survey of Mobile Object Tracking Protocols in Wireless Sensor Networks: A Network-Centric Perspective”, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 11, No. 1, pp. 34-63, 2012.

[2] S. Bhatti and J. Xu, “Survey of Target Tracking Protocols using Wireless Sensor Networks”, *Proceedings of 5th International Conference on Wireless and Mobile Communications*, pp. 110-115, 2009.

[3] Z. Wang, W. Lou, Z. Wang, J. Ma and H. Chen, “A Hybrid Cluster-Based Target Tracking Protocol for Wireless Sensor Networks”, *International Journal of Distributed Sensor Networks*, Vol. 2013, No. 3, pp. 1-15, 2013.

[4] F. Hajiaghajani, M. Naderan, H. Pedram and M. Dehghan, “HCMTT: Hybrid Clustering for Multi-Target Tracking in Wireless Sensor Networks”, *Proceedings of 4th International Workshop on Sensor Networks and Ambient Intelligence*, pp. 889-894, 2012.

[5] C. Lersteau, A. Rossi and M. Sevaux, “Minimum Energy Target Tracking with Coverage Guarantee in Wireless Sensor Networks”, *European Journal of Operational Research*, Vol. 265, No. 2, pp. 234-245, 2018.

[6] I. Lee, Z. Fu, W. Yang and M. Park, “An Efficient Dynamic Clustering Algorithm for Object Tracking in Wireless Sensor Networks”, *Proceedings of 2nd International Conference on Complex Systems and Applications*, pp. 1484-1488, 2007.

[7] W. Zhang and G. Cao, “DCTC: Dynamic Convoy Tree-based Collaboration for Target Tracking in Sensor Networks”, *IEEE Transactions on Wireless Communications*, Vol. 3, No. 5, pp. 1689-1701, 2004.

[8] H.W. Tsai, C.P. Chu and T.S Chen, “Dynamic Object Tracking in Wireless Sensor Networks”, *Computer Communications*, Vol. 30, No. 4, pp. 1811-1825, 2007.

[9] L. Yen, B. Wu and C. Yang, “Tree-Based Object Tracking Without Mobility Statistics in Wireless Sensor Networks”, *Wireless Networks*, Vol. 16, No. 1, pp. 1263-1276, 2010.

[10] W.B. Heinzelman W.B, Chandrakasan A.P and Balakrishnan H, “An Application-Specific Protocol Architecture for Wireless Microsensor Networks”, *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, pp. 660-700, 2002.

[11] E. Olule, G. Wang, M. Guo and M. Dong, “RARE: An Energy Efficient Target Tracking Protocol for Wireless Sensor Networks”, *Proceedings of International Conference on Parallel Processing Workshops*, pp. 76-81, 2007.

[12] H. Yang and B. Sikdar, “A Protocol for Tracking Mobile Targets using Sensor Networks”, *Proceedings of IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 71-81, 2003.

[13] M.Z.A. Bhuiyan, G.J. Wang, P. Yong and Z. Li, “Prediction-Based Energy-Efficient Target Tracking Protocol in Wireless Sensor Networks”, *Journal of Central South University of Technology*, Vol. 17, No. 4, pp. 340-348, 2010.

[14] A. Keshavarz-Mohammadiyan and H. Khaloozadeh, “Consensus-Based Distributed Unscented Target Tracking in Wireless Sensor Networks with State-Dependent Noise”, *Signal Processing*, Vol. 144, No. 3, pp. 112-120, 2018.

[15] Y. Zhou, J. Li and D. Wang, “Posterior Cramer-Rao Lower Bounds for Target Tracking in Sensor Networks with Quantized Range-Only Measurements”, *IEEE Signal Processing Letters*, Vol. 17, No. 2, pp. 157-160, 2010.

[16] M. Di, E.M. Joo and L.H. Beng, “A Comprehensive Study of Kalman Filter and Extended Kalman Filter for Target Tracking in Wireless Sensor Networks”, *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 2792-2797, 2008.

[17] L. Zhang, Q. Cheng, Y. Wang, and Sherali Zeadally, “A Novel Distributed Sensor Positioning System Using the Dual of Target Tracking”, *IEEE Transactions on Computers*, Vol. 57, No. 2, pp. 246-260, 2008.

[18] Manuel A. Vazquez and Joaquin Miguez, “A Robust Scheme for Distributed Particle Filtering in Wireless Sensors Networks”, *Signal Processing*, Vol. 131, No. 2, pp. 190-201, 2017.

[19] X. Wang, H. Zhang and M. Fu, “Collaborative Target Tracking in WSNs using the Combination Likelihood Estimation and Kalman Filtering”, *Journal of Control Theory and Applications*, Vol. 11, No. 1, pp. 27-34, 2013.

[20] M. Mirsadeghi and A. Mahani, “Low Power Prediction Mechanism For WSN-Based Object Tracking”, *Procedia Technology*, Vol. 17, No. 3, pp. 692-698, 2014.

[21] W.R. Chang, H.T. Lin and Z.Z. Cheng, “CODA: A Continuous Object Detection and Tracking Algorithm for Wireless Ad hoc Sensor Networks”, *Proceedings of 5th IEEE International Conference on Consumer Communications and Networking*, pp. 168-174, 2008.

[22] M. Sabet and H.R. Naji, “A Decentralized Energy Efficient Hierarchical Cluster-Based Routing Algorithm for Wireless Sensor Networks”, *AEU - International Journal of Electronics and Communications*, Vol. 69, No. 1, pp. 790-799, 2015.

[23] Y. Chen, L. Gao, Y. Xing and W. Yi, “Cross-Layer Design for Energy-Efficient Reliable Routing in Wireless Sensor Networks”, *Proceedings of 11th International Conference on Mobile Ad-Hoc and Sensor Networks*, pp. 31-36, 2015.

[24] T. Camp, J. Boleng and V. Davies, “A Survey of Mobility Models for Ad Hoc Network Research”, *Wireless Communications and Mobile Computing*, Vol. 5, No. 5, pp. 483-502, 2002.

[25] C. Bettstetter, H. Hartenstein and P. Xavier, “Stochastic Properties of the Random Waypoint Mobility Model: Epoch

Length, Direction Distribution and Cell Change Rate", *Proceedings of 5th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, pp. 7-14, 2002.

[26] S. Pal Chaudhuri, J.Y. Le Boudec and M. Vojnovic, "Perfect Simulations for Random Trip Mobility Models", *Proceedings of 38th IEEE Annual Simulation Symposium*, pp. 72-79, 2005.

[27] Meeta Singh and A.K. Yadav, "Performance Evaluation of AODV and DSR using Random Way Point Mobility Model", *International Journal of Computer Applications*, Vol. 46, No. 19, pp. 256-261, 2012.

[28] M. Naderan, M. Dehghan and H. Pedram, "Upper and Lower Bounds for Dynamic Cluster Assignment for Multi-Target Tracking in Heterogeneous WSNs", *Journal of Parallel and Distributed Computing*, Vol. 73, No. 3, pp. 1389-1399, 2013.