

DCACT hybrid method in WSN-based target tracking

Morteza Sepahvand^{1*}, Ali Naseri², Meysam Raeesdanaee³, Mohammad Hossein Khanzadeh⁴

1-Ph.D. Candidate, 2-Associate Professor, 3- Assistant Professor, 4-Associate Professor
Imam Hossein Comprehensive University, Tehran, Iran
Corresponding Author's Email: msephvnd@ihu.ac.ir

Abstract - The use of Bayesian location estimation methods (a subset of consensus average tracking methods) is one of the most commonly used tracking methods in Wireless Sensor Networks (WSN) due to high error tolerance, tracking accuracy and scalability. In these networks, due to Quasi-spherical of the transmitting radiation pattern (sensing), it is only possible to estimate the target distance from the received signal level. In this paper, a hybrid method called DCACT based on dynamic clustering and bayesian estimation methods (IMMKF and MMPF), will be presented that tracks the maneuvering target with very good accuracy. The simulation results show DCACT method based on multi-mode particle filter (DCMMPF), with a 50% reduction in active sensors, track the target well only with increasing the location estimation error and velocity estimation error up to 26 and 16 percent respectively.

Keywords: WSN, Tracking, Posterior Cramer-Rao Lower Bound, Multi-mode Kalman Filter, Multi-mode particle filter

1. Introduction

In recent years, much attention has been paid to the use of WSN in target tracking. Common networks are mostly based on radar networks but the advantages and excellence of WSN such as the possibility of sampling within the scope of the operation, lacking LOS, Low cost and low interference have made these systems more effective on the battlefield [1]. Tracking the target in a wireless sensor network is one of the research topics that have been gaining interest in recent years [2-5]. The most important thing in target tracking in a wireless sensor network is to estimate the position and direction of the target (or extracting the target velocity vector) using observed measurements (based on the received signal level). This estimate is used to determine the next header and to wake up sensor nodes that are effective in target tracing. Given that the transmitted radiation pattern (sensing) of the nodes are quasi-spherical and practically it is only possible to estimate the distance from the received signal level, the prediction of the signal direction is not possible and the distance detection method of a single node is limited to this method. In many articles, writers assume that the sensor node can extract the target position [6-8]. But, due to the limitations of the wireless sensor network; this is only possible to measure the distance (range only).

Several papers and research have been conducted in the field of tracing the passive object. Wenjun Tanget al. [7] have been working on a distributed consensus-based distributed particle filter in a wireless sparse sensor network. The main objective of this plan is to provide an optimal way to limit the consensus average error in sparse WSN. In this method, the information is weighted from local particle filters and finally, a consensus of these sensors is considered as an optimal output consensus.

Wang, Xingbo et al. [9] in order to compensate for the sub-optimality of the EKF tracking method, provide an algorithm based on the combination of the ML method and the standard Kalman filter. In this paper, in which additive and multiplicative noises and nonlinear target motion are the purposes of input assumptions, the ML method is used to estimate the initial location of the target and to eliminate nonlinear effects in the distance measurement and then uses the standard Kalman filter algorithm to estimate the target path. This paper then compares the tracking error using the proposed method and the developed Kalman filter in the conclusion section.

Ziyia Jia et al. [10] have tried to provide a distributed algorithm to obtain target path in a binary network with the same distributed sensors. In this paper, the estimation of the moving target velocity (in purely progressive motion) is determined by the time the target is detected by each network sensor. In this method, target tracking is based on the sensing node tracking, which involves several problems, such as fading and Multipath, so this method can be used in high density nodes.

In [11] Atieh Mohammadian Keshavarz and their colleagues have presented a method for tracking Maneuvering target in 3D in WSN. In this method, in a three-dimensional space with scalable and multiplicative noise, sensor clusters based on Posterior Cramer-Rao Lower Band (PCRLB) is selected and then object tracking is performed based on IMMPPF.

Many references in this field, such as [6, 12-16] often do not include two fundamental points:

1) All consensus-based methods, such as the Kalman filter and derivatives, require the initial positioning of the target. As mentioned, each sensor node alone cannot extract the target location without the help of its neighbors. In many of these articles, it is assumed that each node can estimate the position of the target, which is not based on the limitations of the present article.

2) In practice, considering the limited battery and computational power, it is not possible to implement consensus average filtering filters in each sensor node (based on the state-vector combination method).

To fix these problems, an algorithm is selected for this research which combines dynamic clustering based on PCRLB, Multi-lateration and average consensus filters (two different methods, IMMKF and particle filtering). In this algorithm, after sensor selection and forming a PCRLB-based dynamic cluster in the context of the target, by using the multilateration method, the position of the target is extracted from the object observation and the result is sent to the data integration center (Figure 1).

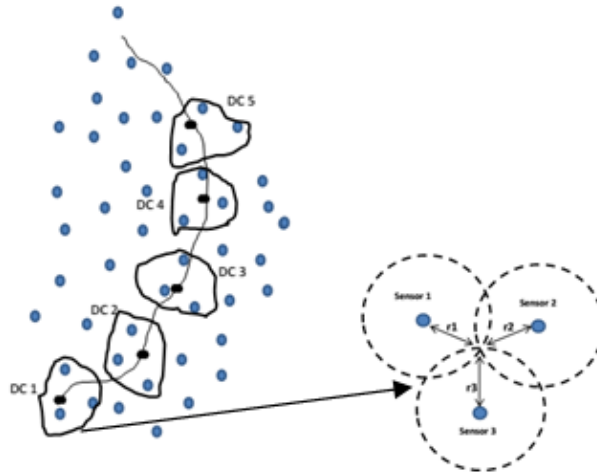


Figure 1. The proposed tracking method scheme

At this center, with the implementation of average consensus tracking algorithm, the future position and trajectory of the target are foreseen. The corresponding new dynamic cluster is determined to awaken the nodes that the target moves towards them. This algorithm will be repeated until the end of the tracking mission (Figure 2).

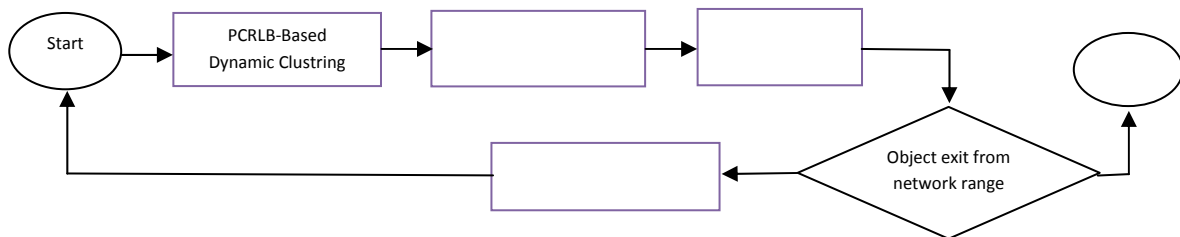


Figure 2. The Proposed Tracking method Block Diagram

The content of this article is organized as follows. In Section 2, the WSN and target dynamics are introduced. In the third section, the formation of the dynamic cluster and the selection of active sensors are described. In Section 4, the target observation and positioning model (which is used as input for the tracker filter) is expressed and the tracking algorithm is described in Section 5. In the final section by simulating the proposed method, the accuracy of the estimation of the target location and target velocity is extracted. Scalability of the DCACT method in the tracking of the target with high percentage variation of active sensors is investigated and the results are presented.

2. Problem formulation

2.1 Network architecture

In WSN, sensors are usually distributed randomly in a uniform distribution due to the lack of access to the sensor trace environment. Based on the principle of WSN, the geographic location of the sensors is determined at the beginning of the network configuration, and the position of each sensor in the network is determined. Due to the fact that the sensors are completely randomly distributed in the network structure, the method of tracking the target in this structure will have the ability to extend and generalize in the same networks (with regular or Gaussian distribution)

2.2. Target Dynamic

The state equations are expressed in the following general form:

$$\mathbf{x}_{k+1} = \mathbf{F}(m) \mathbf{x}_k + \mathbf{w}_k \quad (1)$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k$$

Vectors $\mathbf{v}_k \sim N(0, \mathbf{R})$ and $\mathbf{w}_k \sim N(0, \mathbf{Q})$ are white Gaussian noise zero mean with \mathbf{R} and \mathbf{Q} covariancematrices. The coordinate's vector of the target \mathbf{x}_k is $[x^t(k), v_x^t(k), y^t(k), v_y^t(k)]^T$ and the matrices \mathbf{Q} , \mathbf{G} , \mathbf{F} , and \mathbf{R} in general can also be time-dependent. The element j^{th} is derived from the nonlinear vector mapping $\mathbf{h}(\mathbf{x}_k)$ in terms of the \mathbf{x}_k vector as $\sqrt{(x^t(k) - x_j^s)^2 + (y^t(k) - y_j^s)^2}$.

As a result, j^{th} is the element of the vector of observations \mathbf{z}_k , which means that \mathbf{z}_j is the equivalent of the received signal from the sensor j^{th} , and it is modeled as follows.

$$\mathbf{z}_j(k) = \mathbf{h}_j(\mathbf{x}_k) + \mathbf{v}_j(k) \quad (2)$$

$\mathbf{v}_j(k)$ is the noise of the received observations in the j^{th} sensor at k time. To get enough observations for tracking, at least, the top 3 sensors are selected based on the PCRLB benchmark to estimate the target position [17].

3. Sensor selection and dynamics clustering

One of the major challenges encountered in WSN is the high telecom load because all sensors need to send their observations to a sink. Given the bandwidth and power constraints, the size of sent observations should be minimal. In this regard, the issue of power consumption is important because it determines the life span of the network. As shown in [12], the power consumption level per each transmitted byte 10 times the calculation byte. These values are 400 nJ for each byte of transmission and 40 nJ for each byte of calculations in this reference. Also, broad bandwidth is required to prevent interference between observations from the requirements of sending all observations which it is almost impossible in WSNs. To solve these problems, only the sensors that have better quality observations can be selected. Here, according to event base structures of WSNs, observations can be limited to the scope of the target. In this case, at least 3 sensors should be selected using the Cramer-Rao lower band from the total numbers of sensors that can see the target. Since this band is the low limit of the variance of the non-bias estimation error to estimate the state of an uncertain target [18], we can select the best sensors for providing observations with the least possible error.

The Cramer-Rao lower band for the covariance matrix of the state vector error \mathbf{x} is equal to the inverse of the Fisher information matrix \mathbf{J}_k .

$$\mathbf{C}_k = E \left\{ (\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T \right\} \geq \mathbf{J}_k^{-1} \quad (3)$$

In which $\hat{\mathbf{x}}_k$ is the estimator of the vector \mathbf{x}_k . The calculation of the Fisher information matrix requires high computations. However, it can be calculated recursively [18, 19]:

$$\mathbf{J}_{k+1} = \mathbf{D}_k^{33} - \mathbf{D}_k^{12} (\mathbf{J}_k + \mathbf{D}_k^{11})^{-1} \mathbf{D}_k^{12} + \mathbf{J}_k^z \quad (4)$$

Where

$$\begin{aligned}
 \mathbf{D}_k^1 &= E\left\{-\Delta_x \Delta_x \ln p(\mathbf{x}_{k+1} | \mathbf{x}_k)\right\} = \mathbf{F}^T \mathbf{Q} \mathbf{F} \\
 \mathbf{D}_k^2 &= E\left\{-\Delta_x \Delta_{x+1} \ln p(\mathbf{x}_{k+1} | \mathbf{x}_k)\right\} = \left[\mathbf{D}_k^1\right]^T = -\mathbf{F}^T \mathbf{Q}^{-1} \\
 \mathbf{D}_k^3 &= E\left\{-\Delta_{x+1} \Delta_{x+1} \ln p(\mathbf{x}_{k+1} | \mathbf{x}_k)\right\} = \mathbf{Q}^{-1} \\
 \mathbf{J}_k^E &= E\left\{-\Delta_{x+1} \Delta_{x+1} \ln p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1})\right\} \approx \mathbf{H}^T(\hat{\mathbf{x}}_{k|k-1}) \mathbf{R}^{-1} \mathbf{H}(\hat{\mathbf{x}}_{k|k-1})
 \end{aligned} \tag{5}$$

Here $\mathbf{H}(\mathbf{x}_k)$ is the Jacobin nonlinear vector function of $\mathbf{h}(\mathbf{x}_k)$. Root mean square error (RMSE) of Non-bias estimator $\hat{\mathbf{x}}_k(m)$, $m = 1, 3$ is the m^{th} component state vector \mathbf{x}_k in the following condition applies.

$$\sqrt{E\left[\hat{\mathbf{x}}_k(m) - \mathbf{x}_k(m)\right]^2} \geq b_k(m) \tag{6}$$

Where $b_k(m)$ is the m^{th} component of diagonal matrix \mathbf{J}_k^{-1} .

Since the objective of tracking the target in order to minimize the target location error, a suitable criterion for sensor selection is considered as follows:

$$b_k = \max\{b_k(1), b_k(3)\} \tag{7}$$

That $b_k(1)$ and $b_k(3)$ are the CRLBs of x and y respectively. If the prior distribution of state $p(\mathbf{x}_0)$ is Gaussian with covariance \mathbf{C}_0 , then $\mathbf{J}_0 = \mathbf{C}_0^{-1}$.

Choosing the best sensor with the above criterion is a hybrid problem that requires high volume computing. According to this issue, in order to reduce the computational volume, the following algorithm is proposed.

Flowchart 1- Sensor Selection Algorithm with the PCRLB benchmark

1. Input parameters: B The number of sensors that target in their field of view, M is the maximum number of sensors that are allowed to select and threshold γ_{th} is the acceptable minimum mean square error.
2. Output parameters: N Number of selected sensors, A a set of sensor indexes selected.
3. $S = \{1, \dots, B\}$, $A = \{\emptyset\}$, $N = 0$
4. Calculation of b_k for $N=N+1$ sensors (N of which is related to the selected sensor of the previous step) and sensors of the set S .
5. Selects the sensor that produces the lowest b_k . This sensor is identified by the J index.
6. Remove the selected sensor from S group like $S=S/\{j\}$.
7. $N = N + 1$, $A = A \cup \{j\}$
8. Checking the benchmark for the continuation of the algorithm (if $N < M$ and $b_k < \gamma_{th}$) if yes go to step 4 otherwise stop the algorithm.

4. Observation and positioning model

As noted above, in WSN, since these sensors only have the ability to calculate their distance from the target, they cannot alone extract the position of the target position. Therefore, the target position could be extracted only with the combination of sensors observations. For this subject, at least 3 observations of the sensor should be shared so that the multilateration method can extract the target's Cartesian position. Here, 3 sensors in a dynamic cluster are selected based on the PCRLB criteria participate in the initial positioning process.

$$(x_u - x_i)^2 + (y_u - y_i)^2 = r_i^2, i = 1, 2, 3 \tag{8}$$

Suppose that 3 sensor nodes with specific coordinates. A target with uncertain coordinates in the field of view of these sensors is introduced. The distances are measurable to each of the sensor nodes as follows:

To do this, it's best to write a set of linear equations based on (x_u, y_u) . Here we need to delete the values of x_u^2 and y_u^2 . To do this, it suffices to deduce the third equation from the two previous equations:

$$\begin{aligned} & (x_u - x_1)^2 - (x_u - x_3)^2 + (y_u - y_1)^2 - (y_u - y_3)^2 \\ &= r_1^2 - r_3^2 \\ & (x_u - x_2)^2 - (x_u - x_3)^2 + (y_u - y_2)^2 - (y_u - y_3)^2 \\ &= r_2^2 - r_3^2 \end{aligned} \quad (9)$$

After sorting out the equations we have:

$$\begin{aligned} 2(x_3 - x_1)x_u - 2(y_3 - y_1)y_u &= (r_1^2 - r_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \\ 2(x_3 - x_2)x_u - 2(y_3 - y_2)y_u &= (r_2^2 - r_3^2) - (x_2^2 - x_3^2) - (y_2^2 - y_3^2) \end{aligned} \quad (10)$$

The above equations can easily be expressed as a linear matrix:

$$\begin{aligned} & 2 \begin{pmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{pmatrix} \begin{pmatrix} x_u \\ y_u \end{pmatrix} \\ &= \begin{pmatrix} (r_1^2 - r_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \\ (r_2^2 - r_3^2) - (x_2^2 - x_3^2) - (y_2^2 - y_3^2) \end{pmatrix} \end{aligned} \quad (11)$$

Which can be written as the following linear equation:

$$\mathbf{Ax} = \mathbf{b} \quad (12)$$

The above equation is an overdetermined equation. In this type of linear equation, when the average value of the square error is minimized, the pair (x_u, y_u) will minimize $\|\mathbf{Ax} - \mathbf{b}\|_2^2$ (which is equal to the Euclidean norm). Since $\|\mathbf{v}\|_2^2 = \mathbf{v}^T \mathbf{v}$ exists for each matrix \mathbf{v} , we will have:

$$\begin{aligned} \|\mathbf{Ax} - \mathbf{b}\|_2^2 &= (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) \\ &= \mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - 2\mathbf{x}^T \mathbf{A}^T \mathbf{b} + \mathbf{b}^T \mathbf{b} \end{aligned} \quad (13)$$

Minimizing this value will be the minimization of the average squares. By putting this Polynomial equal to zero, we have:

$$2\mathbf{A}^T \mathbf{Ax} - 2\mathbf{A}^T \mathbf{b} = \mathbf{0} \Leftrightarrow \mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b} \quad (14)$$

By solving the above equations, the value of the \mathbf{x} vector, which is the approximate location of the target, is obtained.

5. Tracking algorithm

In practical scenarios, common goals are similar to those of your own (speed, location, and acceleration) with time. For example, a target that is moving at a steady pace may suddenly enter into its own mode of maneuver. It is impossible to model this kind of behavior in a constant way. Therefore, in addition to estimating the target state, it is also necessary to estimate its fashion.

5.1. Multi-Mode Kalman filtering method

One of the most widely used algorithms for intercepting modes which modalities can be changed by the Markov random variable sequence is the IMM [20].

The current state of the system is assumed to be one of the n possible modes of the $\mathbf{M} = \{\mathbf{M}_1, \dots, \mathbf{M}_n\}$ set. Furthermore, by default, the initial probability is that the target is in \mathbf{M}_j mode specifies and is equal to $\mu_j = p\{\mathbf{M}_j^0\}$. Also, the probability of a change from one mode to another is modeled in the form of the first-order Markov process with the transmission matrix with the following subdivisions:

$$p_{ij} = p\{\mathbf{M}_j^k | \mathbf{M}_i^{k-1}\} \quad (15)$$

The transport mode matrix is defined as follows.

$$\mathbf{P} = \begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & \ddots & \vdots \\ P_{n1} & \cdots & P_{nn} \end{bmatrix} \quad (16)$$

5.1.1. Multi- Mode Kalman Filter

Suppose $M_j(k)$ is the mean of M_j at time k. If the M_j mode happens, the target mode changes with the following model.

$$\mathbf{x}(k+1) = \mathbf{F}_j \mathbf{x}(k) + \mathbf{w}_j(k) \quad (17)$$

And also the received data is as follows.

$$\mathbf{z}(k+1) = \mathbf{H}_j \mathbf{x}(k+1) + \mathbf{v}_j(k+1) \quad (18)$$

Figure 3 shows the general schema of the IMM algorithm. In general, the IMM consists of three steps: 1. Interaction, 2. Filtering and 3. Combination

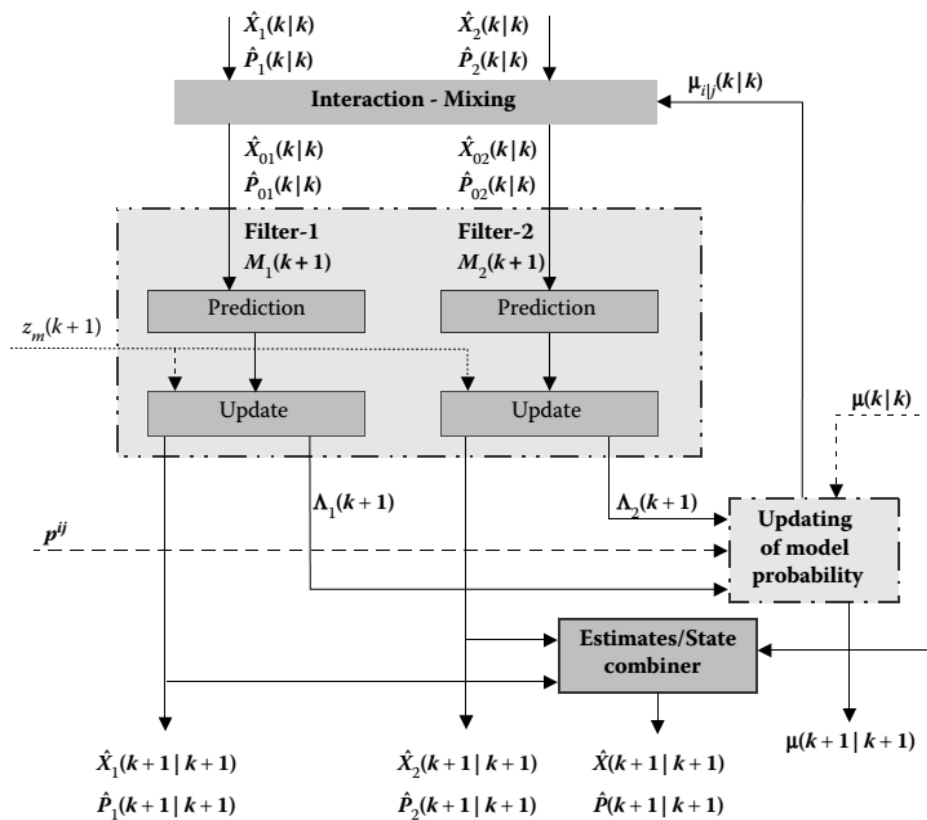


Figure 3. The general scheme of the IMM algorithm[20]

1) Interaction

At this stage, for the mode $M_j(k+1)$, the combined $\hat{\mathbf{x}}_{0j}(k|k)$ estimate and its corresponding covariance matrix $\hat{\mathbf{P}}_{0j}(k|k)$ are calculated from the following equations:

$$\hat{\mathbf{x}}_{0j}(k|k) = \sum_{i=1}^r \mu_{ij}(k|k) \hat{\mathbf{x}}_i(k|k) \quad (19)$$

$$\hat{\mathbf{P}}_{0j}(k|k) = \sum_{i=1}^r \mu_{ij}(k|k) \times \{ \hat{\mathbf{P}}_i(k|k) + [\hat{\mathbf{x}}_i(k|k) - \hat{\mathbf{x}}_{0j}(k|k)] \times [\hat{\mathbf{x}}_i(k|k) - \hat{\mathbf{x}}_{0j}(k|k)]^T \} \quad (20)$$

Where the combinational probabilities $\mu_{ij}(k|k)$ are obtained from the following equation:

$$\hat{\mathbf{x}}_{0j}(k|k) = \sum_{i=1}^r \mu_{ij}(k|k) \hat{\mathbf{x}}_i(k|k) \quad (21)$$

$$\hat{\mathbf{P}}_{0j}(k|k) = \sum_{i=1}^r \mu_{ij}(k|k) \times \{ \hat{\mathbf{P}}_i(k|k) + [\hat{\mathbf{x}}_i(k|k) - \hat{\mathbf{x}}_{0j}(k|k)] \times [\hat{\mathbf{x}}_i(k|k) - \hat{\mathbf{x}}_{0j}(k|k)]^T \} \quad (22)$$

$\mu_j(k+1|k)$ is called the predicted mode probability and it is equal to

$$\mu_j(k+1|k) = \sum_{i=1}^r p_{ij} \mu_i(k|k) \quad (23)$$

The goal change is usually modeled based on the Markov process, which is likely to transfer a mode to another mode as follows

$$p_{ij} = P \{ M_j(k+1) | M_i(k) \} \quad (24)$$

2) Filtering

In this step, the Kalman filter equations with the appropriate model are used to update the hybrid mode in (18) using current data. In order to update, it is necessary to calculate the probabilities of different modes, which are obtained from the following equation.

$$\Lambda_j = \frac{1}{(2\pi)^{n/2} \sqrt{|\mathbf{S}_j|}} e^{-\frac{\mathbf{v}_j^T \mathbf{S}_j^{-1} \mathbf{v}_j}{2}} \quad (25)$$

In which $\mathbf{v}_j = \mathbf{z}(k+1) - \tilde{\mathbf{z}}_j(k+1|k)$ is the innovation vector, n is the dimension and \mathbf{S}_j is the covariance matrix. After this mode has been updated by using $\mathbf{z}(k+1)$ data, Probability of mode $\mu_j(k+1|k+1)$ using probability Λ_j , the predicted probability $\mu_j(k+1|k+1)$ for mode $M_j(k+1)$ is obtained as follows.

$$\mu_j(k+1|k+1) = \frac{1}{c} \mu_j(k+1|k) \Lambda_j$$

$$c = \sum_{i=1}^r \mu_i(k+1|k) \Lambda_i \quad (26)$$

3) Combination

Estimated modes $\hat{\mathbf{x}}_j(k+1|k+1)$ and covariance matrix $\hat{\mathbf{P}}_j(k+1|k+1)$ for each filter by using the predicted mode $\mu_j(k+1|k+1)$ to estimate the final state of $\hat{\mathbf{x}}(k+1|k+1)$, They combine with corresponding matrices of covariance.

$$\hat{\mathbf{x}}(k+1|k+1) = \sum_{j=1}^r \mu_j(k+1|k+1) \hat{\mathbf{x}}_j(k+1|k+1) \quad (27)$$

$$\begin{aligned} \hat{\mathbf{P}}(k+1|k+1) &= \sum_{j=1}^r \mu_j(k+1|k+1) \times \\ &\{ \hat{\mathbf{P}}_j(k+1|k+1) + \\ &[\hat{\mathbf{x}}_j(k+1|k+1) - \hat{\mathbf{x}}(k+1|k+1)] \\ &[\hat{\mathbf{x}}_j(k+1) - \hat{\mathbf{x}}(k+1|k+1)]^T \} \end{aligned} \quad (28)$$

Regarding the nonlinear state of state equations, we can use UKF and EKF. The UKF is rarely used in a multi-mode approach because the covariance matrix cannot be positive Semi-definite by increasing the covariance matrix of non-conformal filters and computational errors [21].

In this case, UKF loses tracking capability. So, the report will only be used by EKF.

5.1.2. Multi-Mode Particle Filter (MMPF)

In this section we study the conventional particle filter algorithm in tracking the maneuverable target. Consider the dynamic equation of a generalized maneuvering target as follows.

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, m_k, \mathbf{v}_k) \\ \mathbf{z}_{k+1} &= \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{w}_k \end{aligned} \quad (29)$$

Where \mathbf{x}_{k+1} is the system state and $m_k \in \{1, \dots, M\}$ denotes the mode of the system and M is the total number of possible target modes. The main idea in the MMPF algorithm is to define the vector of the new random variable, $\mathbf{y}_k = (\mathbf{x}_k, m_k)$, to be estimated. In other words, we want to estimate the lateral density function, $p(\mathbf{y}_k | \mathbf{z}_k)$.

Suppose random samples represent the distribution of the probability function at time $k-1$, $p(\mathbf{y}_{k-1} | \mathbf{z}_{k-1})$ as $\{\mathbf{y}_{k-1}^i\}_{i=1}^{N_s}$. By obtaining new observations at time k , we seek to estimate $p(\mathbf{y}_k | \mathbf{z}_k)$ by random sampling of $\{\mathbf{y}_k^i\}_{i=1}^{N_s}$. Similar to the conventional Bayesian methods, this is done in two stages: predicting and updating.

1) Prediction

The prediction stage is based on the evolutionary model of the $p(\mathbf{y}_k | \mathbf{y}_{k-1})$ mode, which can be displayed as follows.

$$p(\mathbf{y}_k | \mathbf{y}_{k-1}) = p(\mathbf{x}_k | m_k, \mathbf{y}_{k-1}) p(m_k | \mathbf{y}_{k-1}) \quad (30)$$

Also, the second word to the right is simplified as follows.

$$\begin{aligned} p(m_k | \mathbf{y}_{k-1}) &= \\ p(m_k | \mathbf{x}_{k-1}, m_{k-1}) &= p(m_k | m_{k-1}) \end{aligned} \quad (31)$$

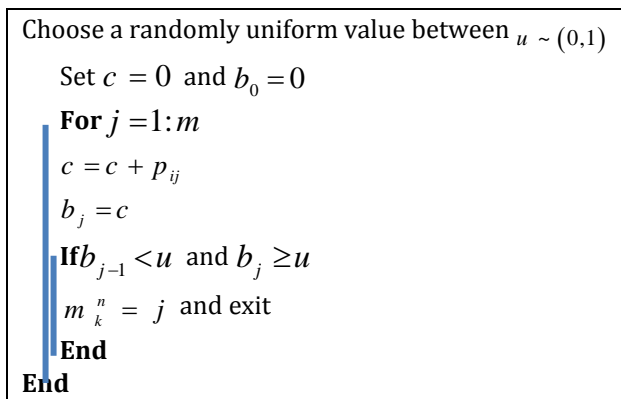
The above relation is the same as the probability of transferring mode of the system. The first term is represented by the equation of system dynamics expressed in the following form.

$$\begin{aligned} p(\mathbf{x}_k | m_k, \mathbf{y}_{k-1}) &= p(\mathbf{x}_k | m_k, m_{k-1}, \mathbf{x}_{k-1}) \\ &= p(\mathbf{x}_k | m_k, \mathbf{x}_{k-1}) \end{aligned} \quad (32)$$

Based on the above-mentioned relationships, having the random samples $\{\mathbf{y}_{k-1}^i\}_{i=1}^{N_s}$, the prediction stage is performed in two steps.

Step 1 - A random sample of indexes $\{m_k^n\}_{n=1}^{N_s}$ is generated from indexes $\{m_k^n\}_{n=1}^{N_s}$. If $m_{k-1}^n = i$ is the m_k^n index, the probability P_{ij} is chosen by j. The m_k^n selection algorithm with the condition $m_{k-1}^n = i$ is executed as follows.

Flowchart 2: Select the target particles in a Multi-Mode Particle Filter



Step 2 - The system mode vector is predicted to be in accordance with the system dynamics. In other words, the particle n of the \mathbf{X}_k^n vector is obtained as follows.

$$\mathbf{x}_{k+1}^n = \mathbf{f}(\mathbf{x}_k^n, m_k^n, \mathbf{v}_k) \quad (33)$$

In this case, the set of predicted particles is $\{\mathbf{y}_k^n\}_{n=1}^{N_s} = \{\mathbf{x}_k^n, m_k^n\}_{n=1}^{N_s}$.

2) Updating

At this stage, a new set of particles is sampled and the estimated latent density function is updated below.

$$p(\mathbf{y}_k | \mathbf{z}_k) \approx \sum_{i=1}^N w_k^i \delta(\mathbf{y}_k - \mathbf{y}_k^i) \quad (34)$$

Where

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k | \mathbf{y}_k^i) = w_{k-1}^i p(\mathbf{z}_k | \mathbf{x}_k^i) \quad (35)$$

6. Simulation

Effectiveness of the proposed tracking approach is validated through some Monte Carlo simulations in terms of tracking accuracy and energy consumption. In these simulations, 800 sensors in 35 hectares (1400 m in 250 m) were distributed randomly with a uniform distribution. Based on the principles of WSN, the geographic location of the sensors is determined at the beginning of the network configuration. Each sensor's visibility is limited and can only see the target in a range of 30 meters radius. In this system, it is assumed that at any moment, up to 10 sensors can be selected. We consider the multi-mode dynamics model in the below form:

$$\mathbf{x}_{k+1} = \mathbf{F}(m) \mathbf{x}_k + \mathbf{G} \times (\mathbf{a}(m) + \mathbf{w}_k) \quad (36)$$

it assumes that the matrix $\mathbf{F}(m)$ is identical for all modes and is equal to [19]:

$$\mathbf{F} = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1 - \cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix} \quad (37)$$

The matrix \mathbf{G} is also defined as:

$$\mathbf{G} = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \quad (38)$$

The state vector and noise vector $\mathbf{v}_k \sim N(0, \mathbf{R})$ and $\mathbf{w}_k \sim N(0, \mathbf{Q})$ are zero-mean white Gaussian noise with \mathbf{R} and \mathbf{Q} covariance matrices as follows:

$$\mathbf{x}_k(t) = \begin{bmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \end{bmatrix}, \mathbf{R} = 10^{-3} \mathbf{I}, \mathbf{Q} = 10^{-3} \mathbf{I} \quad (39)$$

The acceleration of the maneuver's target changes as follows at any time:

$$\mathbf{a} = \begin{cases} [10 \ 0] & k < 20 \\ [-25.25 \ 0] & 20 < k < 34 \\ [-30 \ 0] & 34 < k < 40 \\ [-18 \ 0] & 40 < k < 119 \\ [5 \ -5] & 119 < k < 139 \\ [5 \ 0] & 139 < k < 150 \\ [5 \ 30] & k > 150 \end{cases} \quad (40)$$

Designed filters are based on the following accelerations.

$$\begin{aligned} \mathbf{a}_1 &= [0 \ 0], \mathbf{a}_2 = [20 \ 0] \\ \mathbf{a}_3 &= [0 \ 20], \mathbf{a}_4 = [-20 \ 0] \\ \mathbf{a}_5 &= [0 \ -20], \mathbf{a}_6 = [20 \ 20] \\ \mathbf{a}_7 &= [-20 \ 20], \mathbf{a}_8 = [-20 \ -20] \\ \mathbf{a}_9 &= [20 \ -20], \mathbf{a}_{11} = [0 \ 40] \\ \mathbf{a}_{12} &= [-40 \ 0], \mathbf{a}_{13} = [0 \ -40], \end{aligned} \quad (41)$$

It is also assumed that the target with the probability of 0.9077 remains in the current mode and changes with the probability of 0.0077. Figure.4 shows the simulated and estimated target ($\omega = 0.00005$).

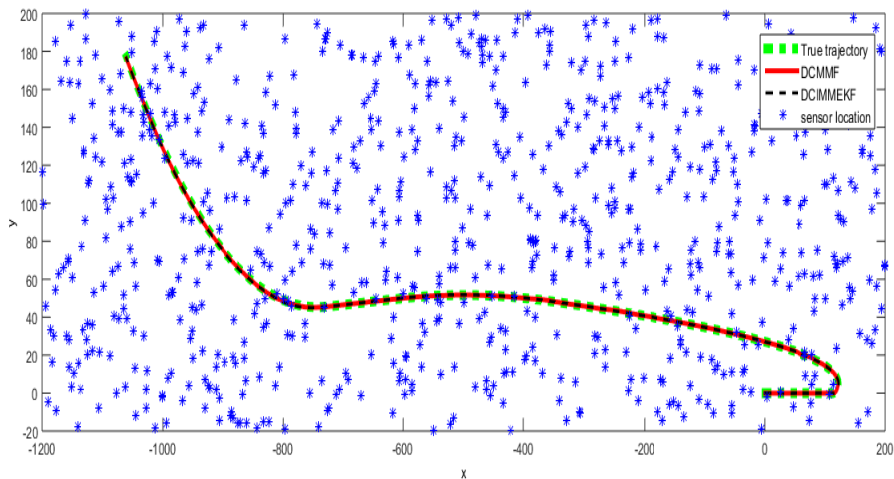


Figure4. Tracing Schema maneuver goal in the DCACT method with Multi-Mode Particle Filter and IMMEKF

The mean square error of the location and the estimated target velocity in the x and y dimensions, for independent Monte Carlo runs, are also shown in Figures 5 and 6.

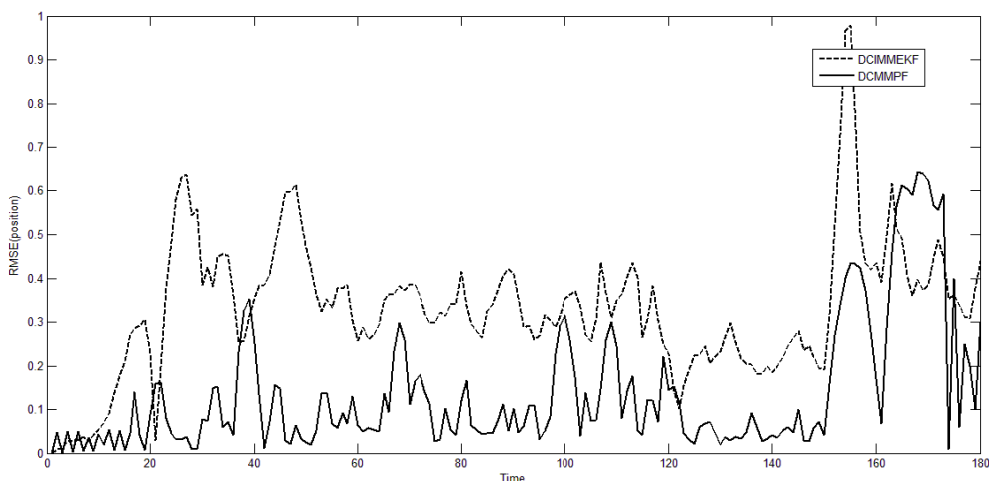


Figure 5. RMSE of target location estimation

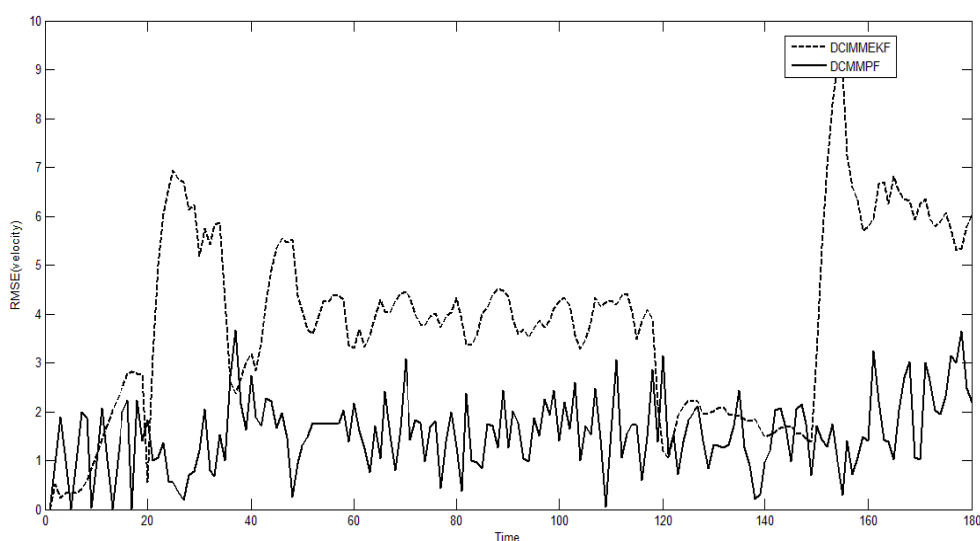


Figure 6. RMSE of velocity estimation

As shown in Table 1, in the DCACT method, the mean error of estimating location and velocity with Multi-Mode extended Kalman filter (DCIMMEKF) is 3 times greater than the multi-mode particle filter (DCMMPF) method.

Table 1: RMSE of location and velocity estimation

Method name	RMSE of location estimation	RMSE of velocity estimation
DCIMMEKF	0.3317m	3.845m/s
DCMMPF	0.1381m	1.553m/s

One of the most important characteristics of the tracking method is its scalability. Scalable tracking algorithm supporting nodal transfer tolerance due to environmental factors, it can tolerate density variations from some nodes to several hundred nodes, without having a particular problem in the algorithm's performance. The issue of scalability is related to the type of method. Some methods may not be scalable, i.e., in a low or limited nodes, number variations can be used. In contrast, scalable methods, in changing the number of nodes in large volumes will be held accountable. Therefore, the design of algorithms that are responsive to any number of sensor nodes and works well under all conditions is one of the most important issues of tracking in WSN. Here is the scalability of the proposed DCACT approach for a reduction of 50% of the active sensor. Tracking scheme, RMSE of target location and speed estimation, respectively, are shown in the figures 7, 8 and 9 for the number of distributed sensors (400 sensors in a space of $1400 \times 220 \text{ m}^2$).

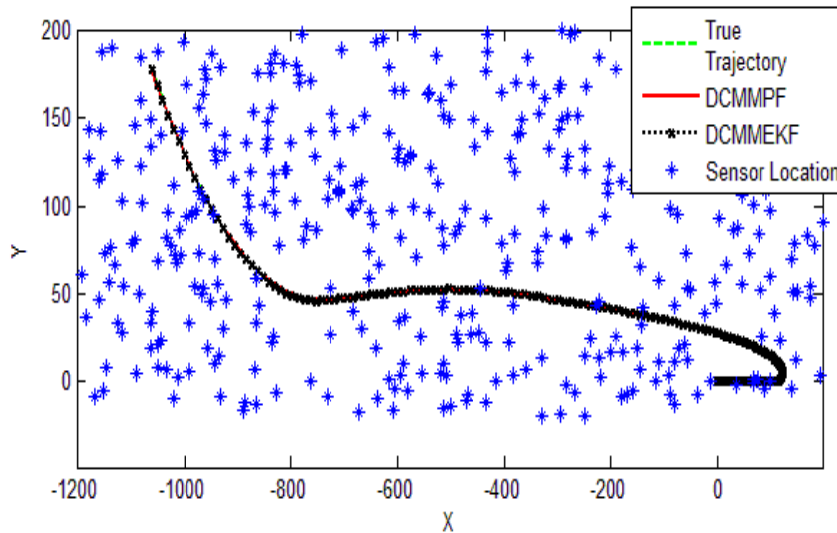


Figure 7. shows the monitored target scene with a 50 percent decrease in active sensors

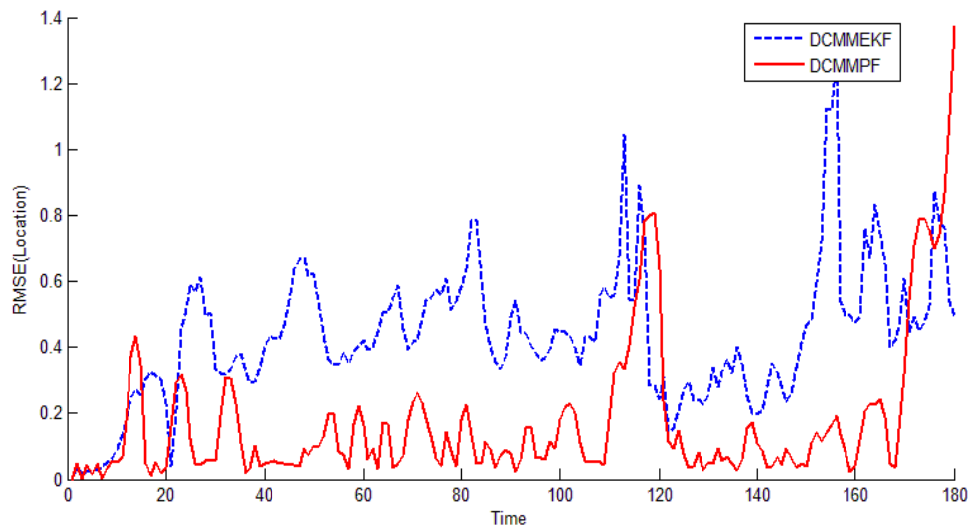


Figure 8. RMSE of target location estimation with 50 percent decrease in active sensors

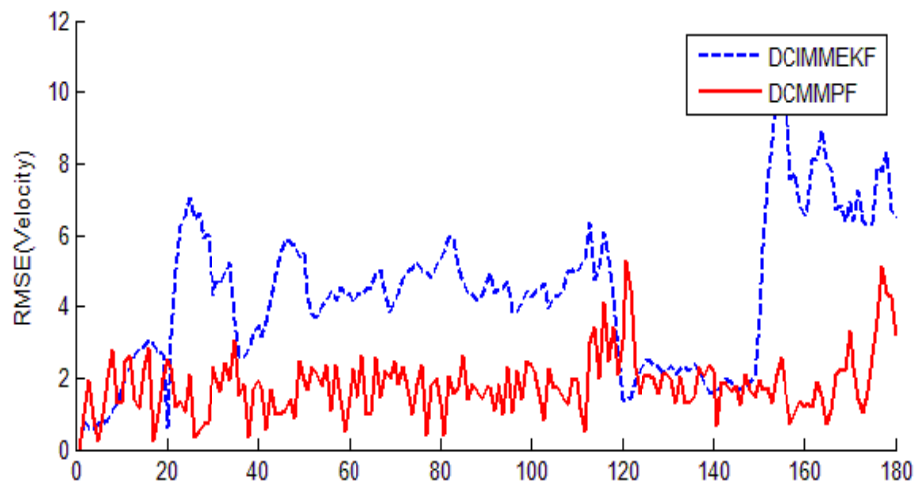


Figure 9. RMSE of target speed estimation with 50% decrease in active sensors

As shown in Tables 2 and 3, the DCACT method is well-scalable and pursues the target with a maximum of 30% error magnification

Table 2: The position RMSE in the scalability test

Method name	800 sensors	400 sensors	Percentage error increase
DCIMMEKF	0.3317m	0.4326m	30.50%
DCMMPF	0.1381m	0.1745m	26.35%

Table 3: The velocity RMSE in the scalability test

Method name	800 sensors	400 sensors	Percentage error increase
DCIMMEKF	3.845m/s	4.358 m/s	13.34%
DCMMPF	1.553m/s	1.814 m/s	16.80%

7. Conclusion

The simulation results show the DCACT tracking sustainability (based on dynamic clustering and the use of multi-mode Kalman filtering methods and multi-mode particle filtering). The DCMMPF tracking method has better results in RMSE of location and speed. The RMSE of location and velocity with Multi-Mode extended Kalman filter (DCIMMEKF) is 2.5 times greater than the multi-mode particle filter (DCMMPF) method. Also, the scalability was tested and the results showed that the DCMMPF method, with a 50 percent decrease in active sensors, present steadily tracking the target with an increase up to 30% in RMSE of location and speed. Also, the RMSE of location and velocity in the DCMMPF method remains about 40% of DCIMMEKF, so it is considered as the preferred method.

8. References

- [1] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, et al., "A line in the sand: a wireless sensor network for target detection, classification, and tracking," *Computer Networks*, vol. 46, pp. 605-634, 2004.
- [2] J. Liang, B. Shen, H. Dong, and J. Lam, "Robust distributed state estimation for sensor networks with multiple stochastic communication delays," *International Journal of Systems Science*, vol. 42, pp. 1459-1471, 2011.
- [3] J. Teng, H. Snoussi, and C. Richard, "Prediction-based cluster management for target tracking in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, pp. 797-812, 2012.
- [4] X. Wang, H. Zhang, and M. Fu, "Collaborative target tracking in WSNs using the combination of maximum likelihood estimation and Kalman filtering," *Journal of Control Theory and Applications*, vol. 11, pp. 27-34, 2013.
- [5] Q. Zhang, C. Zhang, M. Liu, and S. Zhang, "Local node selection for target tracking based on underwater wireless sensor networks," *International Journal of Systems Science*, vol. 46, pp. 2918-2927, 2015.
- [6] A. Nadeau, M. Hassanali, G. Sharma, and T. Soyata, "Energy awareness for supercapacitors using Kalman filter state-of-charge tracking," *Journal of Power Sources*, vol. 296, pp. 383-391, 2015.
- [7] W. Tang, G. Zhang, J. Zeng, and Y. Yue, "Information weighted consensus-based distributed particle filter for large-scale sparse wireless sensor networks," *IET Communications*, vol. 8, pp. 3113-3121, 2014.
- [8] X. Hu, Y.-H. Hu, and B. Xu, "Generalised Kalman filter tracking with multiplicative measurement noise in a wireless sensor network," *Signal Processing, IET*, vol. 8, pp. 467-474, 2014.
- [9] X. Wang, M. Fu, and H. Zhang, "Target tracking in wireless sensor networks based on the combination of KF and MLE using distance measurements," *Mobile Computing, IEEE Transactions on*, vol. 11, pp. 567-576, 2012.
- [10] Z. Jia, M. Chen, and C. Wu, "A Distributed Estimation Algorithm in Binary Sensor Network for Tracking Moving Target," in *Business, Economics, Financial Sciences, and Management*, ed: Springer, 2012, pp. 691-698.
- [11] A. Keshavarz-Mohammadiyan and H. Khaloozadeh, "Interacting multiple model and sensor selection algorithms for manoeuvring target tracking in wireless sensor networks with multiplicative noise," *International Journal of Systems Science*, vol. 48, pp. 899-908, 2017.
- [12] M. Mirsadeghi and A. Mahani, "Energy efficient fast predictor for WSN-based target tracking," *annals of telecommunications-Annales des télécommunications*, vol. 70, pp. 63-71, 2014.
- [13] S. Fan, C. Sun, C. Yang, and B. Ye, "Fast distributed Kalman-Consensus filtering algorithm with local feedback regulation," in *Information and Automation, 2015 IEEE International Conference on*, 2015, pp. 2345-2350.
- [14] P. Chen, H. Ma, S. Gao, and Y. Huang, "Modified Extended Kalman Filtering for Tracking with Insufficient and Intermittent Observations," *Mathematical Problems in Engineering*, vol. 501, p. 981727, 2015.
- [15] G.-r. Bian, H.-h. Zhang, F.-c. Kong, J.-R. Cao, and H.-Y. Shi, "Research on Warehouse Target Localization and Tracking Based on KF and WSN," *Sensors & Transducers (1726-5479)*, 2014.
- [16] J. Read, K. Achutegui, and J. Míguez, "A distributed particle filter for nonlinear tracking in wireless sensor networks," *Signal Processing*, vol. 98, pp. 121-134, 2014.
- [17] E. B. Mazomenos, J. S. Reeve, and N. M. White, "A range-only tracking algorithm for wireless sensor networks," in *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on*, 2009, pp. 775-780.
- [18] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on signal processing*, vol. 46, pp. 1386-1396, 1998.
- [19] M. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, "Multisensor resource deployment using posterior Cramér-Rao bounds," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, pp. 399-416, 2004.
- [20] J. R. Rao, *Multi-Sensor Data Fusion with MATLAB®*: CRC Press, 2009.
- [21] S. Li and P. Ni, "Square-root unscented Kalman filter based simultaneous localization and mapping," in *Information and Automation (ICIA), 2010 IEEE International Conference on*, 2010, pp. 2384-2388.