Target Tracking with Unknown Maneuvers Using Adaptive Parameter Estimation in Wireless Sensor Networks

Morteza Sepahvand, Ali Naseri, Meysam Raeesdanaee, Mohammad Hossein Khanzadeh
Department of Communication and Information Technology, Imam Hossein Comprehensive University, Tehran, Iran
{msephvnd, anaseri, mraeesdanaee, mkhanzade}@ihu.ac.ir
Corresponding author: msephvnd@ihu.ac.ir

***Abstract*- Tracking a target which is sensed by a collection of randomly deployed, limited-capacity, and short-ranged sensors is a tricky problem and, yet applicable to the empirical world. In this paper, this challenge has been addressed by introducing a nested algorithm to track a maneuvering target entering the sensor field. In the proposed nested algorithm, different modules are to fulfil different functions, including sensor selection, adaptive maneuver parameter estimation, and target trajectory extraction. To that end, proposed algorithm combines the auxiliary particle filter with the Liu and West filter and applies them for the first time in the wireless sensor network. Its performance is compared to one of the most common approaches for this kind of problem and the results show the superiority of proposed method in terms of the estimation accuracy. The simulation study also involves evaluating the proposed algorithm based on the scalability criterion and the results are promising since the reduction by 40 percent in the number of active sensors leads to, respectively, 18.2 and 14.3 percent increments in the RMSE of position and velocity estimates.**

***Index Terms*-** Wireless Sensor Network, Tracking, Posterior Cramer-Rao Lower Bound, Auxiliary Particle Filter, Adaptive Parameter Estimation

I. INTRODUCTION

In recent years, much attention has been paid to the use of WSN in target positioning & 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 line of sight, LOS, low cost and low interference have made these systems more effective on the operational field [1]. Tracking a 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 in target tracking in a wireless sensor network is to

Manuscript received 2-June-2018 and revised 18-August-2018, *P- ISSN: 2322-4088*
Accepted on 17- September-2018 *E- ISSN: 2322-3936*

estimate the position and direction of the target using observed measurements. This estimate is used to determine the next header and to wake up sensor nodes that are effective in target tracking. Given that the transmitted radiation pattern of the nodes are quasi-spherical and practically it is only possible to estimate the distance from the received signal level. The distance detection method from a single node is limited to this method and the prediction of the signal direction is not possible. In many articles, Authors assume that the sensor node can extract the target position [6-8]. But, it is possible to estimate the locations of targets by a WSN. Another issue that is important, apart from the discussion of sensor specifications, is the type of target being tracked by wireless sensor networks. If the target can have radiating pattern, it is named an active target. The target with no radiation is referred to as passive target.

Several papers and research articles have been conducted in the field of tracking the passive objects.

Wenjun Tang et al. [7] have been working on a consensus-based distributed particle filter in a sparse wireless sensor network. The main objective of their paper is to provide an optimal way to limit the consensus average error in sparse WSN. In their method, the information is weighted by local particle filters and finally, a consensus of these sensors is considered as the optimal output.

In order to compensate for the sub-optimality of the EKF tracking method, Wang, Xingbo et al.[9] provide an algorithm based on the combination of the ML method and the standard Kalman filter. In their paper, the ML method is used to estimate the initial location of the target and to eliminate nonlinear effects in the range based measurements by using the standard Kalman filter algorithm to estimate a target trajectory. Then, this paper compares the tracking error using the proposed method with the developed Kalman filter in the simulation result section.

In [10], Bajelan and Bakhshi introduced a centralized cluster-based method considering network energy consumption. In their method, information of the current state of nodes (including the remaining energy of nodes, the distance to sink, and between head-clusters) are considered to select the optimal head-clusters for target tracking.

Ziyia Jia et al.[11] 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 sensor of the network. 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 [12], Atieh Mohammadian Keshavarz and their colleagues have presented a method for 3D tracking of a maneuvering target in WSN with additive and multiplicative noise in observation equation. In this method~~,~~ a sensor cluster is selected based on the Posterior Cramer-Rao Lower Band, PCRLB, and then object tracking is performed using the measurements, obtained from the selected sensors, which are then exploited to execute the Interacting Multi Mode Particle Filter, IMMPF.

Nevertheless, contributions made by this paper may be visualized through the following three point which many references, such as [6, 13-17], have left out of consideration:

1) All consensus-based methods, such as the Kalman filter and its nonlinear extensions, 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 average consensus filters in each sensor node (based on the state-vector combination method).

3) In many references, the Kalman tracker filter has been developed (or it has been used together with other methods) [6], [14], [15], [18], [19], [20], [21], [22]. In many cases, the multi mode particle filter has been used [11-14], [23],[24], [25],[26]. These two methods are the most used tracking filters in wireless sensor networks. The methods for tracking a maneuvering target used in these articles are restricted on the availability of the transition probability matrix. In other words, the performance of the tracking algorithm is significantly dependent on this matrix and, if poorly designed, the operation of the algorithm is heavily influenced [14]. On the other hand, the availibility of the transition probability matrix is usually not a realistic expectation in many applications.

This paper contribution to fix these problems, is a proposed hybrid method for tracking the target with unknown maneuvers in WSN. This method, which is called DCAPET[1] hereafter, combines dynamic clustering based on PCRLB, Multi-lateration and a new average consensus tracking algorithm consist of auxiliary particle filter with adaptive parameter estimation called the Liu and West filter.

In this method, after sensor selection and forming a PCRLB-based dynamic cluster in the context of the target, by using the Multilateration method, the initial position of the target is extracted from the object observation and the result is sent to the data integration center, Fig. 1.

At this center, with the implementation of proposed 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 cycle will be repeated until the end of the tracking mission, Fig. 2.

The content of this article is organized as follows. In Section II, the WSN architecture 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 IV, the target observation and positioning model (which is used as input for the tracker filter) is expressed and in the section V, The auxiliary particle filter and

---

[1] Dynamic Clustering Adaptive Parameter Estimation Tracking

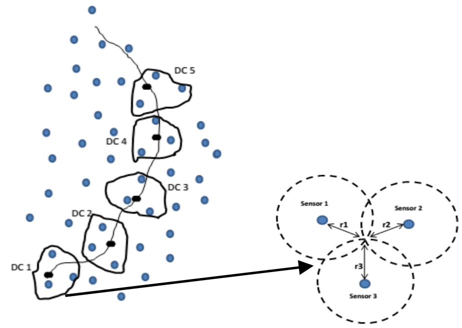

Fig. 1. The proposed tracking method scheme

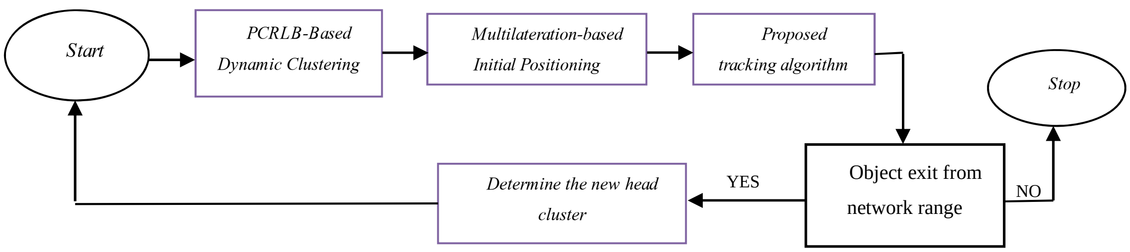

Fig. 2. DCAPET Block Diagram

adaptive parameter estimation method is first intuduced and then, with a combination of these two methods, a hybrid algorithm will be presented for tracking the target with unknown maneuvers.

In the section VI, the accuracy of the proposed hybrid method in the estimation of both the position and velocity of a moving target is evaluated . Furthermore, the scalability of the proposed method in the tracking of the target with high percentage of dropping in the number of active sensors is investigated and the results are presented. Finally, conclusion remarks are given in Sections VII.

II. Problem Formulation

Before proceeding to the problem statement, it is important to give a remark about the notation. The conversation analytic notation system will be used in this paper is as follows. Vector quantities are shown in lower case bold, an example being the state vector, $\boldsymbol{x}$. Matrices are shown as bold and upper case, such as the covariance matrix, $\boldsymbol{C}$.

A. Network Architecture

In WSN, sensors are usually distributed randomly in a uniform distribution in the enviroment. 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.

B. Target Dynamic and measurement's equation

The state equations are expressed in the following general form:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{G} \times (\boldsymbol{a}(m) + \boldsymbol{w}_k) \qquad (1)$$
$$\boldsymbol{z}_k = h(\boldsymbol{x}_k) + \boldsymbol{v}_k$$

Vectors $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R})$ and $\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q})$ are white Gaussian noise zero mean with $\boldsymbol{R}$ and $\boldsymbol{Q}$ covariance matrices. The coordinate's vector of the target $\boldsymbol{x}_k$ is $\left[x^t(k)\ , v_x^t(k), y^t(k), v_y^t(k)\right]^T$ and the matrices Q, G, F, and R can be time dependent. In this equation "m" is the "mode" of target. The mode is varying based on the target accelerations. The jth element $z_j(k)$ is derived from the nonlinear vector mapping $h_j(\boldsymbol{x}_k\ )$in terms of the $\boldsymbol{x}_k$vector as $\left(\left(x^t(k) - x_j^s\right)^2 + \left(y^t(k) - y_j^s\right)^2\right)^{1/2}$.

As a result, jth element of the vector of observations $\boldsymbol{z}_k$, which means that z_j is the equivalent of the received signal from the jth sensor, and it is modeled as follows

$$z_j(k) = h_j(\boldsymbol{x}_k) + v_j(k) \qquad (2)$$

where $\boldsymbol{v}_j(k)$ is the noise of the received observations in the jth sensor at time k. To get enough observations for tracking, at least, the top 3 sensors are selected based on the PCRLB benchmark to estimate the target position [27].

III. SENSOR SELECTION AND DYNAMICS CLUSTERING

One of the major challenges facing the WSN is experiencing a very high communication load as all sensors need to send and receive the network information to and from a sink or a base. This network information includes sensor observations as well as the basecommands or processed results returning to sensors on-demand Given the bandwidth and power constraints, the size of transmitted observations should be minimal. Particularly, as the network information communication widely shares the main part of network energy depletion . As shown in [13], the power consumption level per each transmitted byte is 10 times greater than per byte calculation. These values are 400 nJ for each byte of transmission and 40 nJ for each byte of calculations in this reference.

As a result, devising an effecive strategy to reduce the overal network communcation will prolong the life span of the network. One of the most promising tools to meet this demand is to select only a subset of sensors which have a major impact on the target trajectory estimation accuracy and let other sensors sleep. Up to now, various methods have been proposed for selecting a subset of sensor for target motion estimation, such as the use of information derived from the covariance matrix of the estimated error or the trace of the covariance matrix on its determinant, which is called the GDOP[2]. Recently, Husam, and Havens have also used the GDOP measure to select sensors which are moving [28].The covariance matrix may be derived from different estimation algorithms such as the Kalman filter developed in [29] or the information filter in [30]. In addition to the use of information derived from the various estimation algorithms, another approach independent of the type of estimation algorithm used to select the appropriate sensors is also used which are based on the calculation of the amount of information received by the FIM[3]. This information is independent of the estimation algorithm used to extract them. This approach is called PCRLB based sensor selection because it is based on the CRLB minimization, which itself is a reversal of the FIM. In this article, the selection of sensors is performed on the basis of the PCRLB minimization criterion, due to its advantages. Since the posterior Cramer–Rao expresses the lower bound on the covariance of unbiased estimators of a target state vector [31], one can select those sensors which are providing observations with the least error contamination. As mentioned earlier, the posterior Cramer-Rao lower bound $\mathbf{C}_k$ for the covariance matrix of the state vector error $\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$ is equal to the inverse of the Fisher information matrix (FIM) $\boldsymbol{J}_k$.

$$\mathbf{C}_k = \mathbb{E}\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T\} \geq \boldsymbol{J}_k{}^{-1} \tag{3}$$

In which $\hat{\boldsymbol{x}}_k$ is the estimator of the vector $\boldsymbol{x}_k$. The following Riccati-like recursion giving the sequence of posterior FIMs, $\mathbf{J}_\mathbf{k}$, k > 0, for the unbiased estimation of $\boldsymbol{x}_k$ is provided by [31, 32]

$$\boldsymbol{J}_{k+1} = \boldsymbol{D}_k^{33} - \boldsymbol{D}_k^{12}\left(\boldsymbol{J}_k + \boldsymbol{D}_k^{11}\right)^{-1}\boldsymbol{D}_k^{12} + \boldsymbol{J}_k^z \tag{4}$$

where

$$\begin{aligned}
\boldsymbol{D}_k^{11} &= \mathbb{E}\{-\nabla_x \nabla_x \ln p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)\} = \boldsymbol{F}^T\boldsymbol{Q}^{-1}\boldsymbol{F} \\
\boldsymbol{D}_k^{12} &= \mathbb{E}\{-\nabla_x \nabla_{x+1} \ln p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)\} = \left[\boldsymbol{D}_k^{21}\right]^T = -\boldsymbol{F}^T\boldsymbol{Q}^{-1} \\
\boldsymbol{D}_k^{33} &= \mathbb{E}\{-\nabla_{x+1} \nabla_{x+1} \ln p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)\} = \boldsymbol{Q}^{-1} \\
\boldsymbol{J}_k^z &= \sum_{j=1}^{N} \mathbb{E}\left\{-\nabla_{x+1}\nabla_{x+1} \ln p_j(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1})\right\} \approx \sum_{j=1}^{N} \mathbb{E}[\boldsymbol{H}_j\left(\hat{\boldsymbol{x}}_{k|k-1}\right)\boldsymbol{R}^{-1}\boldsymbol{H}_j\left(\hat{\boldsymbol{x}}_{k|k-1}\right)]
\end{aligned} \tag{5}$$

here $\boldsymbol{H}(\boldsymbol{x}_k)$ is the Jacobin matrix of all first-order partial derivatives of a vector-valued nonlinear function $h(\boldsymbol{x}_k)$ and $\nabla_{x+1}= \frac{d}{dx+1}$, $\nabla_x = \frac{d}{dx}$. Root mean square error (RMSE) of unbiased estimator $\hat{\boldsymbol{x}}_k(m)$, $m = 1, \cdots, 3$ is the mth component state vector $\mathbf{x}_k$ in the following

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

[2] Geometric Dilution Of Precision

[3] Fisher Information Matrix

Flowchart I. Sensor Selection Algorithm with the PCRLB benchmark

1. Input parameters: B is The number of sensors that the target is 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 set of sensor indexes selected.
3. $S = \{1, \ldots, B\}, A = \{\emptyset\}, N = 0$
4. Calculation of bk 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. $N = N + 1$, $A = A \ \mathrm{U}\{j\}$
8. Checking the benchmark for the continuation of the algorithm (if N<M and bk <γ_{th}) if yes go to step 4 otherwise stop the algorithm.

condition applies:

Where $b_k(m)$ is the mth component of diagonal matrix $\boldsymbol{J}_k{}^{-1}$. The prior distribution of state $p(\boldsymbol{x}_0)$ is Gaussian with covariance $\boldsymbol{C}_0$, then $\boldsymbol{J}_0 = \boldsymbol{C}_0 - 1$. Since the objective of tracking the target is minimizing the target location error, a suitable criterion for sensor selection is considered as follows:

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

where $b_k(1)$ and $b_k(3)$ are the CRLBs of x and y respectively.

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 burden, the following algorithm is proposed.

IV. OBSERVATION AND POSITIONING MODEL

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

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

Assume 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:

It's best to write a set of linear equations based on (x_u, y_u). Here one needs to delete the values of $x^2{}_u$ and $y^2{}_u$. 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 one has:

$$\begin{aligned}2(x_3 - x_1)x_u - 2(y_3 - y_1)y_u = (r_1^2 - r_3^2) - (x_1^2 - x_3^3) - (y_1^2 - y_3^2) \\ 2(x_3 - x_2)x_u - 2(y_3 - y_2)y_u = (r_2^2 - r_3^2) - (x_2^2 - x_3^3) - (y_2^2 - y_3^2)\end{aligned} \quad (10)$$

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

$$\begin{aligned}&2\begin{pmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{pmatrix}\begin{pmatrix} x_u \\ y_u \end{pmatrix} \\ &\qquad = \begin{pmatrix} (r_1^2 - r_3^2) - (x_1^2 - x_3^3) - (y_1^2 - y_3^2) \\ (r_2^2 - r_3^2) - (x_2^2 - x_3^3) - (y_2^2 - y_3^2) \end{pmatrix}\end{aligned} \quad (11)$$

Which can be written as the following linear equation:

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

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 $\|\boldsymbol{Ax} - \boldsymbol{b}\|_2^2$ (which is equal to the Euclidean norm)

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

Since $\|\boldsymbol{v}\|_2^2 = \boldsymbol{v}^T\boldsymbol{v}$ exists for each vector $\boldsymbol{v}$, one will have:

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

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

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

V. TRACKING ALGORITHM

In this section we introduce the proposed algorithm on tracking maneuvering target. One of the main assumptions made in the articles of tracking maneuvering targets, is knowing the transition probability matrix. In other words, the performance of the interception algorithm is significantly dependent on this matrix and, if poorly designed, the operation of the algorithm is heavily influenced [14]. On the other hand, properly designed Transition probability matrix depended on knowing of

Flowchart II. Auxiliary particle filter flowchart [33].

Inputs, $\{w_{k-1}^{i}\}_{i=1}^{N_s}$,$\{x_{k-1}^{i}\}_{i=1}^{N_s}$ - Outputs, $\{w_{k}^{i}\}_{i=1}^{N_s}$, $\{x_{k-1}^{i}\}_{i=1}^{N_s}$,$\hat{x}_k$
1- For $i = 1{:}\,N_s$ - Calculate μ_k^i where $\mu^i \sim p(x_k \| x_{k-1}^i)$ - Calculate $w_k^i = p(z_k \| \mu_k^i)$ End
2- Calculate total weight: $t = sum[\{w_k^i\}_{i=1}^{N_s}]$
3- For $i = 1{:}\,N_s$ - Normalized $w_k^i = t^{-1} w_k$ End
4- Resample - $\left[\{\sim,\sim,i^j\}\right] = Resample[\{x_{k-1}^i, w_k^i\}_{i=1}^{N_s}$
5- For $i = 1{:}\,N_s$ - Draw $x_k^j = p(x_k \| x_{k-1}^{i^j})$ - Assign $w_k^j = \frac{p(z_k \| x_k^j)}{p(z_k \| \mu_k^{i^j})}$ End

target statistics condition, that is, in many applications is completely unknown. This is usually the case that the type of target is generally unknown in many operational applications.

In this section, a particle-based filtering technique is presented which is completely independent of the mode transfer matrix, where only knowing the scope of mode variation is sufficient. Prior to introducing the proposed algorithm, an auxiliary particle filter is presented. Then a review of the parameter estimation is given using the Monte Carlo algorithms. Finally, by combination of these two methods (auxiliary particle filter and adaptive parameter estimation), a method for maneuvering target tracking with unknown maneuvers will be presented.

A. Auxiliary particle filter

In a typical particle filter, particles $\left\{\boldsymbol{x}_k^{(i)}\right\}_{i=1}^{N_s}$ of the transfer density function $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ are propagated and then weighed and sampled from the observed observations (propagation-sampling). This method does not function properly in the state estimation, because of the propagated particles are independent from received observations.

It is desirable that the particles are sampled from the density function $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{z}_k)$. Sampling this density function leads to an optimal estimation, since it has been proven that in this case, the variance of the particles is minimized [19]. Calculation of this density function requires a very large computation volume, so that the actual sampling will encounter the calculation problem. The auxiliary particles filter affects the effects of observations in particle propagation [20].

In other words, the particles are sampled, based on the predicted probability $p(\boldsymbol{z}_k|\boldsymbol{x}_{k-1})$

Then they are propagated according to the transfer function (sampling-propagation).The flowchart below shows the implementation steps of the auxiliary particle filter [33].

B. *Deterministic parameter estimation using Monte-Carlo method*

Consider a dynamic system based on the Markov chain and seen by the vector $\boldsymbol{z}_k$ at different times. The function of the probability density of the observations is represented by $p(\boldsymbol{z}_k|\boldsymbol{x}_k,\theta)$in which $\boldsymbol{x}_k$ is the state vector and θ is a constant parameter whose value is unknown. The system is also changed in the form of $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\theta)$ based on the first order Markov chain. In the Monte Carlo method, The estimation of the posterior probability density function $p(\boldsymbol{x}_k,\theta|\boldsymbol{z}_{1:k})$ in a recursive manner is sought. For this purpose, the variables to be estimated, state vector and constant parameter, are combined as a new state vector and then the conventional Monte Carlo methods are used. In this case, the set of new particles can be defined as $\left\{\boldsymbol{x}_k^{(j)},\theta_k^{(j)}: j=1,\ldots,N\right\}$, the weights of which correspond to them as $\left\{w_k^{(j)}: j=1,\ldots,N\right\}$. By using the Bayes rule one has:

$$\begin{aligned} p(\boldsymbol{x}_{k+1},\theta|\boldsymbol{z}_{1:k+1}) &\propto p(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1},\theta)p(\boldsymbol{x}_{k+1},\theta|\boldsymbol{z}_{1:k}) \\ &\propto p(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1},\theta)p(\boldsymbol{x}_{k+1}|\theta,\boldsymbol{z}_{1:k})p(\theta|\boldsymbol{z}_{1:k}) \end{aligned} \tag{15}$$

As can be seen from the above equation, sampling from $p(\boldsymbol{x}_{k+1},\theta|\boldsymbol{z}_{1:k+1})$ requires knowledge of $p(\theta|\boldsymbol{z}_{1:k})$. If θ is known, conventional Monte Carlo methods such as a bootstrap filter or a particle filter can be used. Where θ is completely unknown, there are two general methods that will be explained below.

C. *Artificial evolutionary method*

In this method, for a constant parameter θ, an artificial evolutionary model is considered as follows:

$$\theta_{k+1} = \theta_k + \zeta_{k+1} \tag{16}$$

In which

$$\zeta_{k+1} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{W}_{k+1}) \tag{17}$$

In (17), the matrix $\boldsymbol{W}_{k+1}$ has a certain value. This method, despite its simplicity, is inadequate and and can be causing loss of information between different points [21].

Flowchart III. The Liu & West method [34].

- Inputs $\{\boldsymbol{x}_{k-1}^i, \theta_{k-1}^i\}_{i=1}^{N_s}, w_{k-1}^i$ - Outputs $\{\boldsymbol{x}_k^i, \theta_k^i\}_{i=1}^{N_s}, w_k^i, \widetilde{\boldsymbol{x}}_k, \widetilde{\theta}_k$
1- propagate particles $\mu^i = \mathbb{E}\{\boldsymbol{x}_k \| \boldsymbol{x}_{k-1}^i, \theta_{k-1}^i\}$ Calculate $w_k^i = w_{k-1}^i p(\boldsymbol{z}_k \| \mu_k^i, \boldsymbol{m}_{k-1}^i)$ where $\boldsymbol{m}_{k-1}^{(i)} = \alpha\theta^{(i)} + (1-\alpha)\bar{\theta}$
2- For $i = 1, \dots, N_s$ - Parameters are sampled from the kernel density $\theta^i \sim \mathcal{N}(\theta \| \boldsymbol{m}_{k-1}^i, h^2 \boldsymbol{V}_{k-1})$ where $\boldsymbol{V}_{k-1} = \sum_{i=1}^N w_k^{(i)} (\theta_{k-1}{}^{(i)} - \bar{\theta})(\theta_{k-1}{}^{(i)} - \bar{\theta})^T$ - Propagate state particles $\boldsymbol{x}_k^i \sim p\left(\boldsymbol{x}_k \| \boldsymbol{x}_{k-1}^{(i)}, \theta^i\right)$ - Assign weights $w_k^i = \dfrac{p(\boldsymbol{z}_k \| \boldsymbol{x}_k^i, \theta^i)}{p(\boldsymbol{z}_k \| \mu_k^i, \boldsymbol{m}_{k-1}^i)}$ End
3- $\widetilde{\boldsymbol{x}}_k = \sum_{i=1}^N w_k^i \boldsymbol{x}_k^i, \widetilde{\theta}_k = \sum_{i=1}^N w_k^i \theta_k^i$,state and parameter estimation

D. Kernel smoothing of parameter

As noted, particle sampling will be possible if the density function of the $p(\theta|\boldsymbol{z}_{1:k})$ is specified. To solve this problem it is assumed that it can be approximated as a plurality of Gaussian distributions [21, 22]. In other words:

$$p(\theta|\boldsymbol{z}_{1:k}) \approx \sum_{j=1}^{N_s} w_t^{(j)} \mathcal{N}\left(\theta|\boldsymbol{m}_k^{(j)}, h^2 \boldsymbol{V}_k\right) \tag{18}$$

In which:

$$\begin{aligned} \boldsymbol{m}_k^{(i)} &= \alpha\theta^{(i)} + (1-\alpha)\bar{\theta} \\ \boldsymbol{V}_t &= \sum_{i=1}^{N_s} w_k^{(i)} (\theta^{(i)} - \bar{\theta})(\theta^{(i)} - \bar{\theta})^T \\ \bar{\theta} &= \sum_{i=1}^{N_s} w_k^{(i)} \theta^{(i)} \end{aligned} \tag{19}$$

In the above relations, the parameter h^2 is a smoothing kernel function and α is the shrinkage parameter. This method is known as the Liu & West filter [34]. The Liu & West filter is shown in the algorithm III with auxiliary particle filter.

Using the particle filter with the parameter estimation in the above method is only useful when the predicted parameter is strictly constant. In this method, the parameter distribution function concentrates with increasing the number of observations at a point. Consequently, if the parameter is variable with time, the previously introduced method will be ineffective and cannot follow the sudden changes of the parameter.

E. Proposed tracking algorithm

In the tracking issue, some of the parameters are variable with time so that they can be changed instantaneously. For example, a maneuvering target that has a constant acceleration in any mode which changes with a variation in mode. In the proposed method, one considers these parameters as linear piecewise, which have instantaneous changes at the change points. It is assumed that the goal with the probabilityα at any given moment can change the mode and remains constant with probability $1-\alpha$. At any given time, the constant parameter with probabilityα with the following distribution function changes to a new value γ_t and with probability $1-\alpha$ remains in the value of θ_t.

$$\theta_k = \begin{cases} \theta_{k-1} & with\ probability\ 1-\alpha \\ \gamma_k & with\ probbability\ \ \alpha \end{cases} \tag{20}$$

and γ_k is selected as follows:

$$\gamma_k \sim p_{\theta_{k-1}}(.) \tag{21}$$

The above distribution function is usually considered uniformly, which is selected in the range of minimum and maximum variations of the problem parameters.

If the α value is close to zero, then the probability of modifying the target mode is zero, and the tracking problem is converted to the tracking with constant parameter.

On the other hand, if α is selected large, close to 1, the filter will be in a bid to estimate a new parameter for most of the time, even if there is no change in the parameter. Depending on the level of target maneuver this parameter can be selected.

In summary, by combining the auxiliary particle filter and the Liu-West method, we obatin the proposed tracking algorithm IV.

VI. SIMULATION RESULTS

In order to get closer to real conditions, 2000 sensors were completely randomly distributed with uniform distribution in an area of 9000 hectares (10000 meters in 9000 meters). Based on the principles of wireless sensor networks, the geographic location of the sensors is determined at the beginning of the network configuration. The field of view of each sensor is limited and can only see

Flowchart IV. Proposed tracking algorithm.

1- Inputs $\{\boldsymbol{x}_{k-1}^i, \theta_{k-1}^i\}_{i=1}^{N_s}$, w_{k-1}^i **2-** Outputs $\{\boldsymbol{x}_k^i, \theta_k^i\}_{i=1}^{N_s}$, w_k^i, $\widetilde{\boldsymbol{x}}_k$, $\tilde{\theta}_k$
3- **For** $i = 1, \ldots, N_s$ - Update $\boldsymbol{m}_{k-1}^i = \alpha\theta_{k-1}^i + (1-\alpha)\bar{\theta}_{k-1}$ - Calculate pre-weights $w_{k,1}^i = p\left(\boldsymbol{z}_k \middle\| \mu_k^i, \theta_k^i\right)$ where $\mu_k^i = p\left(\boldsymbol{x}_k \middle\| \boldsymbol{x}_{k-1}^i, \theta_{k-1}^i\right)$ - **End**
4- **For** $i = 1, \ldots, N_s$ - Sample new parameter particle $\gamma_k^i = p_{\theta_{k-1}^i}(.)$ - Calculate pre-weights $w_{k,2}^i = p\left(\boldsymbol{z}_k \middle\| \mu_k^i, \gamma_k^i\right)$ - **End**
5- **For** $i = 1, \ldots, N_s$ - Sample indices k^i from $\{1, \ldots, 2\mathrm{N}_s\}$ with probabilities $\left\{(1-\alpha)w_{k,1}^i\right\}_{i=1}^{N_s}$ and $\left\{\alpha w_{k,2}^i\right\}_{i=N_s+1}^{2N_s}$ - **End**
6- **For** $n^i \in \{1, \ldots, N_s\}$ - Update parameters $\theta_k^i \sim \mathcal{N}\left(. \middle\| \boldsymbol{m}_{k-1}^{n^i}, h^2\boldsymbol{V}_{k-1}\right)$ - Propagate states $\boldsymbol{x}_k^i \sim p\left(\boldsymbol{x}_k \middle\| \boldsymbol{x}_{k-1}^{n^i}, \theta_k^i\right)$ - $w_k^i = \frac{p\left(\mathbf{z}_k \| \mathbf{x}_k^i, \theta_k^i\right)}{w_{k,1}^{n^i}}$ - **End**
7- For $n^i \in \{\mathrm{N}_s + 1, \ldots, 2N_s\}$ 8- Propagate sates $\boldsymbol{x}_k^i \sim p\left(\boldsymbol{x}_k \middle\| \boldsymbol{x}_{k-1}^{n^i}, \gamma_k^{n^i}\right)$ - Set parameters $\theta_k^i = \gamma_k^{n^i}$ - Assign weights $w_k^i = \frac{p\left(\mathbf{z}_k \| \mathbf{x}_k^i, \theta_k^i\right)}{w_{k,2}^{n^i}}$ - **Endfor**
9- Resample particles
10- $\widetilde{\boldsymbol{x}}_k = \sum_{i=1}^N w_k^i \boldsymbol{x}_k^i$, $\tilde{\theta}_k = \sum_{i=1}^N w_k^i \theta_k^i$, state and parameter estimation

the target within a radius of 350 meters. The system assumes that at most 10 sensors (M=10) can be selected at any time. We consider the multi-mode linear dynamical model in the form below

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{G} \times (\boldsymbol{a}(m) + \boldsymbol{w}_k) \tag{22}$$

Assume that the matrix **F** is the same for all modes and is equal to the following value [7]:

$$\boldsymbol{F} = \begin{bmatrix} 1 & \frac{sin\omega T}{\omega} & 0 & -\frac{1-cos\omega T}{\omega} \\ 0 & cos\omega T & 0 & -sin\omega T \\ 0 & \frac{1-cos\omega T}{\omega} & 1 & \frac{sin\omega T}{\omega} \\ 0 & sin\omega T & 0 & cos\omega T \end{bmatrix} \tag{23}$$

The state vector is $\boldsymbol{x}_k(t)$ and the noise vectors $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R})$ and $\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q})$ are white Gaussian noise with zero means with $\boldsymbol{R}$ and $\boldsymbol{Q}$ covariance matrices as follows:

$$\boldsymbol{x}_k = \begin{bmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \end{bmatrix}, G = \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix}, \boldsymbol{R}_{2\times 2} = 10^{-1}\boldsymbol{I}, \boldsymbol{Q}_{2\times 2} = 10^{-3}\boldsymbol{I} \tag{24}$$

The above-mentioned model involves constant velocities ($\mathbf{a} = 0$ and $\omega \to 0$), constant acceleration ($\omega \to 0$and $\mathbf{a} \neq 0$) and constant angular acceleration ($\omega \neq 0$, $\mathbf{a} = 0$). For the simulations of this paper, the fourth mode in which both the truning rate and the acceleration are not zero is considered.The target dynamic model is CT model with known turn rate. we assume that the parameter $\mathbf{a}$ is uncertain and must be estimated. The parameter α and the kernel function are assumed to be 0.5 and 0.2 respectively. $p_{\theta_{k-1}}(.)$ is considered uniformly in the $[-10 \;\; 10]$ interval.

The time step between k and k + 1 is 1 second and the overal simulation time is 100 seconeds in each iteration. In this scenario, accelerations of the target are varying with time as follow:

$$\mathbf{a} = \begin{cases} [5 \quad 10] & k \leq 20 \\ [0 \quad 0] & 20 < k \leq 35 \\ [0 \quad -5] & 35 < k \leq 50 \\ [-5 \quad 0] & 50 < k \leq 70 \\ [-5 \quad 5] & 70 < k \leq 80 \\ [0 \quad -5] & 80 < k < 100 \end{cases} \tag{25}$$

Designed filters for DCIMMEKF[4] are based on the following accelerations

$$\begin{aligned} &\mathbf{a}_1 = [0 \quad 0], \mathbf{a}_2 = [5 \quad 0] \\ &\mathbf{a}_3 = [0 \quad 5], \mathbf{a}_4 = [-5 \quad 0] \\ &\mathbf{a}_5 = [0 \quad -5], \mathbf{a}_6 = [5 \quad 5] \\ &\mathbf{a}_7 = [-5 \quad 5], \mathbf{a}_8 = [-5 \quad -5] \\ &\mathbf{a}_9 = [5 \quad -5], \mathbf{a}_{11} = [5 \quad 10] \\ &\mathbf{a}_{12} = [-5 \quad 10], \mathbf{a}_{13} = [-10 \quad 5] \end{aligned} \tag{26}$$

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.The transition probability matrix for IMMEKF is therefore:

$$\begin{bmatrix} 0.9077 & 0.0077 & \cdots & 0.0077 \\ 0.0077 & 0.9077 & 0.0077 & \vdots \\ \vdots & \vdots & \ddots & 0.0077 \\ 0.0077 & \cdots & 0.0077 & 0.9077 \end{bmatrix}_{13\times 13} \tag{27}$$

[4] Dynamic Clustering Interaction Multi Mode Extended Kalman Filter

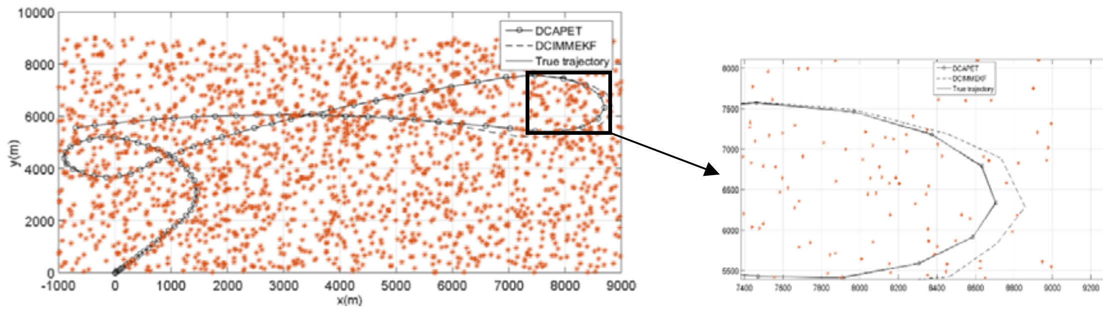

Fig. 3. Tracing Schema maneuver goal in the DCAPET method with adaptive acceleration estimation and DCIMMEKF.

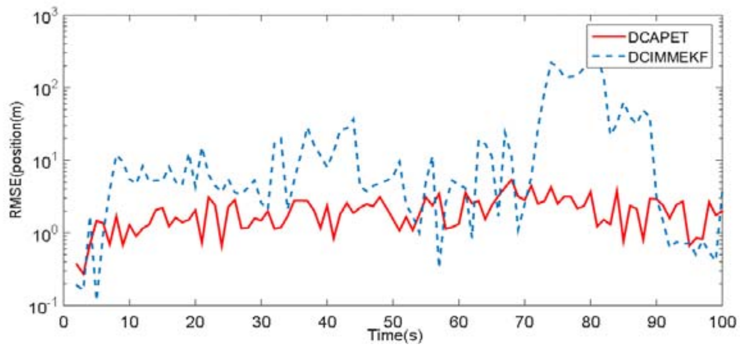

Fig. 4. RMSE of target position estimate.

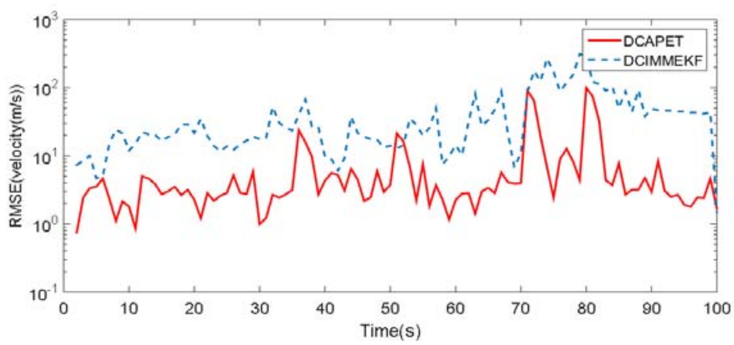

Fig. 5. RMSE of target velocity estimate.

Dynamic clustering is done based on PCRLB for each methods in the same way. Fig. 4 shows the simulated and estimated target (ω = 0.00005). It is notable that the tables (and all simulation results hereafter) are made on the basis of averaging over 100 independent Monte Carlo runs with different random seeds.

The root-mean-square error (RMSE) of position and velocity estimates in the x and y dimensions are also shown in Figs. 4 and 5.

Table.I. RMSE of position and velocity estimates

Method name	RMSE of position estimation	RMSE of velocity estimation
DCIMMEKF	26.43m	45.23 m/s
DCAPET	2.007 m	7.80m/s

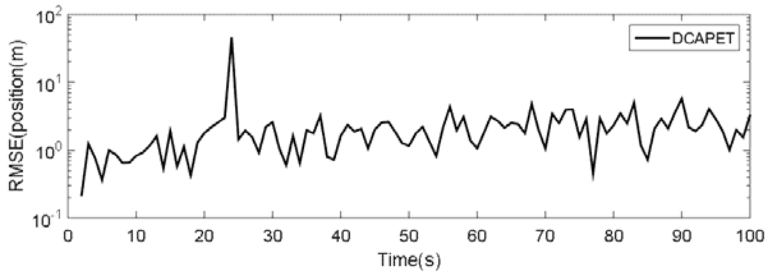

Fig. 6. RMSE of target location estimation with 40 percent decrease in the number of active sensors.

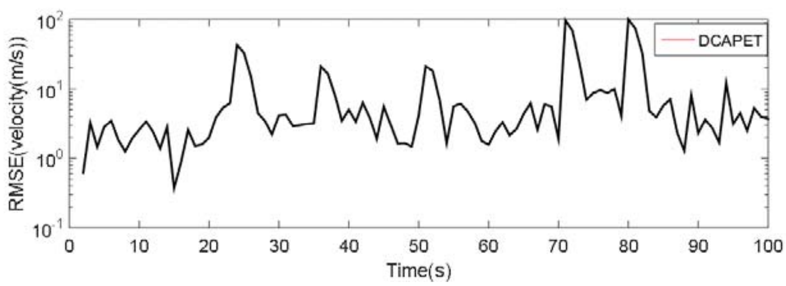

Fig. 7. RMSE of target speed estimation with 40 percent decrease in the number of active sensors.

As shown in Table 1, the time average RMSE of position and velocity estimates with Multi-Mode extended Kalman filter (DCIMMEKF) is about 13 and 5.8 times greater than those of the prorosed DCAPET method respectively.

One of the most important characteristics of any tracking method is its scalability to the possible failure of nodes in WSN. Althoung a system is considered scalable if it is capable of handling an increasing amount of load when new resources (typically new sensors) are added, here scalable tracking algorithm is meant a system which should also tolerate reduction in the number of sensors without experiencing detrimental effects on the algorithm's general performance.

To check this aspect of the proposed algorithm, the performance has been evalouated in terms of RMSE of the position and velocity estimates after a random reduction of 40 percent in the number of total available sensors in the field is implemented.

The RMSE of the position and velocity estimates, respectively, are shown in the Figs. 6 and 7 for 1200 sensorswhich has been uniformly deployed in a space of 10000×9000 m^2.

Table II. The position and velocity RMSE in the scalability test.

DCAPET Method	2000 sensors	1200 sensors	Percentage error increase
Position RMSE	2.007 m	2.373m	18.2%
velocity RMSE	7.80m/s	8.909m/s	14.3%

As shown in Tables II and III, the DCAPET method is well-scalable and pursues the target, while experiencing a rise of, at most, 18.2 percent in the RMSE of the position estimates.

VI. CONCLUSION

In this paper, a target tracking algorithm has been presented based on the combination of dynamic sensor selection and advanced auxiliary particle filtering which is equipped with the Liu-West method to estimate the target maneuver parameter while the target is moving through the sensor field. The presented method is called DCAPET algorithm and achieves a higher level of robustness and resource utilization when it is compared with DCIMMEKF which is one of the conventional methods used for tracking a target in the WSN. In both arget position and target velocity estimation, there are gains to be earned (in terms of RMSE's) from using the DCAPET procedure over the use of the DCIMMEKF. The DCIMMEKF filtering yielded the RMSE of position and velocity estimates 13 and 5.8 times greater than those of the DCAPET method respectively. Also, the scalability of the DCAPET was tested and the results showed that the DCAPET method, with a 40 percent decrease in the number of active sensors, maintains target tracking by experiencing about 18.2% and 14.3% increases in its RMSE values of position and velocity estimates, respectively. This result may allow the DCAPET method to serve as a highly-scalable tracking solution.

REFERENCES

[1] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, "A line in the sand: a wireless sensor network for target detection, classification, and tracking," Computer Networks, vol. 46, no. 5, pp. 605-634, Dec. 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, no. 9, pp. 1459-1471, Jan. 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, no. 9, pp. 797-812, June 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, no. 1, pp. 27-34, Feb. 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, no. 16, pp. 2918-2927, Feb. 2015.

[6] A. Nadeau, M. Hassanalieragh, 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, Nov. 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, no. 17, pp. 3113-3121, Nov. 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, no. 5, pp. 467-474, July 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, no. 4, pp. 567-576, April 2012.

[10] M. Bajelan and H. Bakhshi, "An Adaptive LEACH-based clustering algorithm for wireless sensor networks," *Journal of communication engineering,* vol. 2, no. 4, pp. 351-365, Autumn 2013.

[11] 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, AISC 143, pp. 691-698, 2012.

[12] 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, no. 5, pp. 899-908, April 2017.

[13] 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, March 2014.

[14] S. Fan, C. Sun, C. Yang, and B. Ye, "Fast distributed Kalman-Consensus filtering algorithm with local feedback regulation," I*EEE International Conference on Information and Automation,* August 2015, pp. 2345-2350.

[15] 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. 2015, article ID 981727, June 2015.

[16] 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,* vol. 163, no. 1, pp. 255-261, Jan. 2014.

[17] 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.

[18] Q. Wen, Y. Zhou, L. Hu, J. Li, and D. Wang, "Comparison of filtering techniques for simultaneous localization and tracking," *International Conference on Estimation, Detection and Information Fusion (ICEDIF),* Jan. 2015, pp. 387-392.

[19] S. Wen, Z. Cai, and X. Hu, "Constrained Extended Kalman Filter for Target Tracking in Directional Sensor Networks," *International Journal of Distributed Sensor Networks,* vol. 2015, pp. 1-13, May 2015.

[20] 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,* vol. 163, p. 255, Jan. 2014.

[21] X. Hu, Y.-H. Hu, and B. Xu, "Generalised Kalman filter tracking with multiplicative measurement noise in a wireless sensor network," *IET Signal Processing,* vol. 8, no. 5, pp. 467-474, June 2013.

[22] W. Zhu, W. Wang, and G. Yuan, "An improved interacting multiple model filtering algorithm based on the cubature Kalman filter for maneuvering target tracking," *Sensors,* vol. 16, no. 6, p. 805, June 2016.

[23] J. M. Pak, C. K. Ahn, P. Shi, Y. S. Shmaliy, and M. T. Lim, "Distributed hybrid particle/FIR filtering for mitigating NLOS effects in TOA-based localization using wireless sensor networks," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 6, pp. 5182-5191, June 2017.

[24] S. Mahfouz, F. Mourad-Chehade, P. Honeine, J. Farah, and H. Snoussi, "Non-parametric and semi-parametric RSSI/distance modeling for target tracking in wireless sensor networks," *IEEE Sensors Journal,* vol. 16, no. 7, pp. 2115-2126, April 2016.

[25] A. Keshavarz-Mohammadiyan and H. Khaloozadeh, "Adaptive IMMPF for bearing-only maneuvering target tracking in Wireless Sensor networks," *4th International Conference on Control, Instrumentation, and Automation (ICCIA),* Jan. 2016, pp. 6-11.

[26] A. Keshavarz-Mohammadiyan and H. Khaloozadeh, "Logarithm based adaptive Particle Filter for maneuvering target tracking in Wireless Sensor Networks with multiplicative noise," *4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, Jan. 2016, pp. 1-5.

[27] E. B. Mazomenos, J. S. Reeve, and N. M. White, "A range-only tracking algorithm for wireless sensor networks," *International Conference on Advanced Information Networking and Applications Workshops, WAINA'09,* May 2009, pp. 775-780.

[28] H. Sweidan and T. Havens, "Sensor Relocation for Improved Target Tracking," *IET Wireless Sensor Systems*, vol. 8, no. 2, pp. 76-88, March 2018.

[29] L. M. Kaplan, "Global node selection for localization in a distributed sensor network," *IEEE Transactions on Aerospace and Electronic systems,* vol. 42, no. 1, pp. 113-135, Jan. 2006.

[30] M. Zoghi and M. Kahaei, "Adaptive sensor selection in wireless sensor networks for target tracking," *IET Signal Processing,* vol. 4, no. 5, pp. 530-536, Oct. 2010.

[31] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on signal processing,* vol. 46, no. 5, pp. 1386-1396, May 1998.

[32] 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, no. 2, pp. 399-416, April 2004.

[33] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174-188, Feb. 2002.

[34] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," *Sequential Monte Carlo methods in practice*, ed: Springer, 2001, pp. 197-223.