A New Adaptive Target Tracking Protocol in Wireless Sensor Networks

Elham Ahmadi
Computer Engineering Department
Islamic Azad University
South Tehran Branch
Tehran, Iran

Masoud Sabaei
Computer Engineering and Information Technology department
Amirkabir University
Tehran, Iran

ABSTRACT
In wireless sensor networks sampling time interval and the number of nodes involved in each stage of tracking are important factors which have high effect on the efficiency of target tracking applications. In this paper a new target tracking method has been proposed which at each time step employs two helpful tools. First, an extended Kalman filter (EKF)-based estimation technique to predict the tracking error and second, an energy consumption model to estimate energy consumption based on different number of nodes and sampling time intervals. By using these estimations, this method selects the best number of nodes and sampling time interval according to an objective function which is defined based on tracking accuracy and energy consumption.

Keywords
Wireless sensor networks; target tracking; energy consumption; tracking error

1. INTRODUCTION
A wireless sensor network (WSN) is composed of a large number of sensor nodes and deployed either inside the phenomenon or very close to it. Wireless sensor networks are expected serve as a key infrastructure for a broad range of applications including precision agriculture, surveillance highway systems, emergent disaster response and recovery. One of the important application issues for sensor networks is utilized to track mobile object [1].

In wireless sensor networks generally the accuracy of target tracking strongly depends on two important factors. First, the number of nodes which are involved in target tracking operations, second, the sampling time intervals. In other words when we use more sensor nodes and smaller tracking time interval, it causes an increase in the accuracy of tracking a moving object. However, this increases the power consumption significantly.

Until now a lot of researches on adaptive target tracking protocols have been done. Some of them have focused on adaptive sampling time intervals and some of them tried to adapt the size of wakening up areas. For example, in paper [2] a new method that is called Target Tracking Based on Mobility Model (T^2BM^2) has been proposed, which uses Semi-Dynamic clustering (SDC) structure. The main idea of this method is that when the target moves randomly, to increase the accuracy of tracking it is better to use bigger size of clusters. On the other hand, when the target's motion is predictable, in order to decrease the energy consumption it is better to decrease the size of clusters. In *SDC* structure, at first, the network backbone is formed from small static clusters then during the tracking period based on the number of nodes needed to participate in tracking operation, some of these backbone clusters are merged and form a bigger cluster. In paper [3] a new classification algorithm has been introduced that by using easy classification based on velocity estimate provides a way to reduce the uncertainly in movement of target and thus the error on the estimate of target's position is reduced. In [4], a protocol for Prediction Accuracy-based Tracking Energy Saving (PATES) is well developed to conserve energy of WSNs. In [5], an energy efficient adaptive tracking algorithm called Predict-and-Mesh (PaM) is proposed, which consists of two prediction models and a failure recovery process. Paper [6] has proposed the AEC algorithm. In this algorithm, tracking time interval is modified based on averaging the *Error*, which is the difference between the measured and predicted locations of the moving object, after some given iterations, periodically. In distributed structure approach, in paper [7] an energy efficient adaptive sensor scheduling approach has been proposed that jointly selects tasking sensors and determines their associated sampling intervals according to the predicted tracking accuracy and tracking energy cost. In [8], based on the uniform sampling intervals, a covariance control framework is presented where the tracking accuracy is defined as an expected covariance matrix, and multiple tasking sensors for the next time step are selected such that the updated estimation covariance will be within the desirable covariance at all times. In general, most of the previous adaptive target tracking protocols(like AEC in [6], PATES in[4], T^2BM^2 in[2]), have focused on the adaptive sampling time interval or adaptive wakening up areas separately, but in this paper a new protocol has been presented that have pained attention on both of the subjects. In other words, in this paper a new Adaptive Target Tracking Protocol (ATTP) has been proposed which at each time step, employs two helpful tools. First, an extended Kalman filter (EKF)-based estimation technique to predict the tracking error and second, an energy consumption model to estimate energy consumption based on different number of nodes and sampling time intervals. After that by using these estimations selects the best number of nodes and sampling time interval according to an objective function which is defined based on tracking accuracy and energy consumption.

The rest of the paper is organized as follows. In section 2, the proposed network model (including the network structure and adaptive prediction-based tracking) is described. In section 3, the new adaptive target tracking protocol is described. Also simulation results are given in section 4. Meanwhile, the results are compared with two existing prediction-based tracking schemes (i.e. AEC and T^2BM^2). Finally, conclusion and future work are introduced in Section 5.

2. PROPOSED NETWORK MODEL

In this section, we describe the proposed network model and argue about our network structure and adaptive prediction-based tracking briefly.

2.1 Network Structure

In ATTP protocol, the network structure is based on SDC[1]structure. In this structure, at first the network backbone is defined in static clustering form and then according to the number of nodes needed for target tracking some of these backbone clusters are merged and form the bigger cluster. One of the main points in this structure is that, it includes two level cluster heads. The cluster heads of backbone clusters which are called the first-level cluster heads (CH1) and a node named second-level cluster head that is selected among CH1 nodes and is generally in the center of the big cluster. After receiving information from member nodes in backbone clusters, CH1 nodes transmit their integrated information to CH2. Subsequently, after receiving this information from all of CH1 nodes, CH2 transmits its integrated information to the base station [2]. Fig. 1 illustrates this structure briefly.

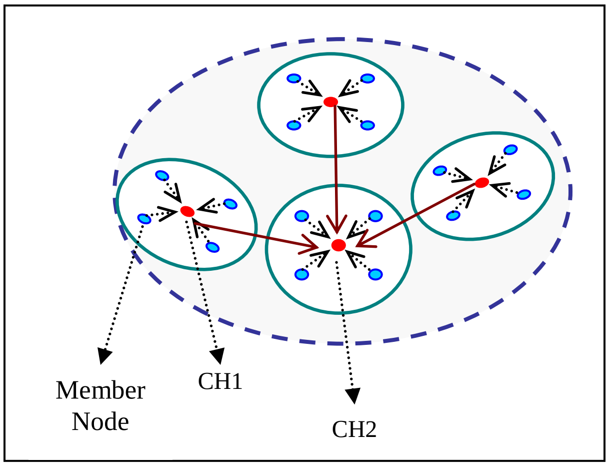

Figure 1. A scheme of SDC structure

2.2 Prediction-Based Tracking

In the proposed protocol, the prediction of the next target's location is based on the tracking algorithm in paper [9]. This protocol uses linear prediction to estimate the target's next location. Suppose according to Fig.2 the location of the mobile object at the time instance of t+1 is approximately predicted by estimating the target's velocity will move during the time [t-1, t]. Let us suppose the target's locations at the time instants T_{t-1} and T_t are $(x(t-1), y(t-1))$ and $(x(t), y(t))$ .Then the target's velocity can be estimated as:

$$v_{pre}(t+1) = \frac{\sqrt{(x(t)-x(t-1))^2 + (y(t)-y(t-1))^2}}{(t)-(t-1)} \quad (1)$$

Also let us express the prediction error by the angle between the actual location and the previously predicted, denoted by $\theta'(t)$ as shown in Fig. 2, we have:

[1] Semi-Dynamic Clustering

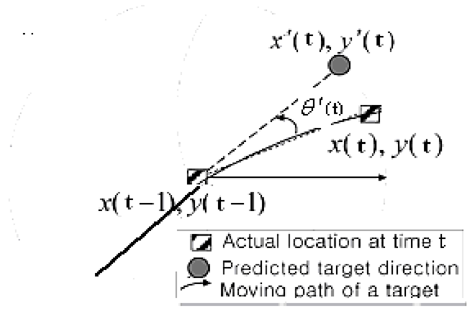

Figure 2. Tracking of the mobile target in WSNs

$$\cos\theta'(t) = \frac{x(t)-x(t-1)}{\sqrt{[x'(t)-x(t-1)]^2 + [y(t)-y(t-1)]^2}} - \frac{x'(t)-x(t-1)}{\sqrt{[x'(t)-x(t-1)]^2 + [y'(t)-y(t-1)]^2}} \quad (2)$$

And the target's direction can be estimated as:

$$\cos\theta(t+1) = \frac{x'(t)-x(t-1)}{\sqrt{[x'(t)-x(t-1)]^2 + [y'(t)-y(t-1)]^2}} - \cos\theta'(t) \quad (3)$$

Finally the target's location at the time instance (t+1) is predicted as:

$$x(t+1) = x(t) + v_{pre}(t)\cos(\theta(t+1)) \quad (4)$$

$$y(t+1) = y(t) + v_{pre}(t)\sin(\theta(t+1)) \quad (5)$$

3. ADAPTIVE TARGET TRACKING PROTOCOL

As mentioned before, most of the previous adaptive target tracking protocols have focused on the adaptive sampling time interval or adaptive wakening up areas separately, but our proposed protocol have considered both of these subjects concurrently. In other words, in this protocol, at first the tracking error and the amount of energy consumption based on various wakening up areas (different numbers of nodes) and different sampling time intervals are predicted. After that the number of nodes needed to be waked up and the amount of sampling time interval for the next tracking operation are selected according to an objective function. In the following we describe this protocol in detail.

3.1 Estimation of the total wakening up area

In this section, we want to estimate the total area which the target probably would appear there. In addition to the predicted velocity which is $v_{pre}(t+1)$, we consider maximum velocity, $v_{max}(t+1)$ and minimum velocity, $v_{min}(t+1)$, then based on these supposes, we can estimate the total wakening up area. But this area expands toward up or down regarding to prediction error $(\theta'(t))$. In other words, if the target's location in the past sampling time interval has been placed upper than the real location then the area would be stretched toward up, otherwise, it would be stretched down. For example regarding to Fig. 2 the area in this example would stretch toward down. If we denote the total wakening up area at time instance (t+1) by $Area_{Total}(t+1)$, then in mathematical approach, this area can be described as:

$$Area_{Total}(t+1)=\int_{\theta(t+1)-\beta}^{\theta(t+1)+\alpha}\int_{v_{\min}\Delta t_k}^{v_{\max}\Delta t_k} dRd\theta \qquad (6)$$

Where

- V_{max}: the target's maximum velocity
- V_{min} : the target's minimum velocity
- Δt_k: the *kth* sampling time interval $(\Delta t_k = t_{k+1} - t_k)$
- $\theta(t+1)$: the predicted direction for target mobility
- α: the maximum angle which the target probably would bend toward up
- β: the maximum angle which the target probably would bend toward down

3.2 The candidate wakening up areas

We always don't consider the $Area_{Total}(t+1)$ as the waken-ing up area, in other words, respect to energy consumption it is better to select a smaller area as wakening up area. So we describe a series of *candidate wakening up areas* which are grown up hierarchically from the target's next predicted loc-ation in time instance $(t+1)$, $(x(t+1), y(t+1))$ .The main point about the candidate waking up areas is that similar to $Area_{Total}(t+1)$ these areas expand according to α and β angles. In this paper, we consider these areas as discrete areas $\{Area_i\}_1^N$. Fig. 3 illustrates these concepts better.

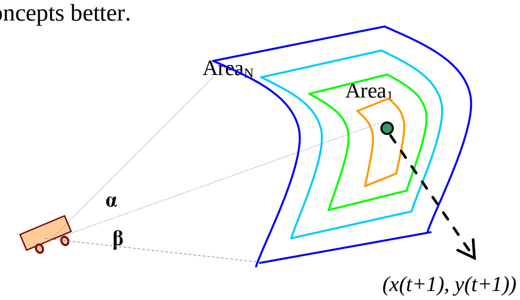

Figure 3. The discrete candidate wakening up areas

Also in mathematical view we can define the *ith* candidate area as:

$$Area_i = \int_{\theta(t+1)-\frac{i\beta}{n}}^{\theta(t+1)+\frac{i\alpha}{n}} \int_{(v_{pre}-\frac{i(v_{pre}-v_{\min})}{n}\Delta t)}^{(v_{pre}+\frac{i(v_{\max}-v_{pre})}{n}\Delta t)} dRd\theta \qquad (7)$$

Supposing that the density of nodes in network is λ then the number of sensor nodes in the $Area_i$, is denoted by N_i and is defined as:

$$N_i = \lambda A_i \qquad (8)$$

where A_i is supposed to be the total area of the *ith* candidate area and is defined as:

$$A_i = \int_{\theta(t+1)-\frac{i\beta}{n}}^{\theta(t+1)+\frac{i\alpha}{n}} \int_{(v_{pre}-\frac{i(v_{pre}-v_{\min})}{n}\Delta t)}^{(v_{pre}+\frac{i(v_{\max}-v_{pre})}{n}\Delta t)} \frac{R^2\theta}{2} dRd\theta \qquad (9)$$

3.3 The estimation of tracking error

By taking the idea from paper [7], in this section we want to predict the tracking error in each candidate wakening up area in the next tracking operation. In paper [7], the tracking operation is done just by using one sensor node in addition to the network structure is supposed to be a distributed structure, but as mentioned before, we have supposed in the ATTP protocol the *SDC* structure as network structure. Furthermore, we use more than three sensor nodes for target tracking in each stage. We assume a linear target motion model and a non-linear measurement model, both with Gaussian noise distributions. EKF is used as the estimation algorithm. The target motion is modeled by the following state equation:

$$X(k+1) = F(\Delta t_k)X(k) + w(k,\Delta t_k) \qquad (10)$$

where *X (k)* is the state of the target at the *k*th time step that happens at t_k and $(\Delta t_k = t_{k+1} - t_k)$ is the *k*th sampling interval. *F*(Δt_k) is the transition matrix dependent on Δt_k. $w(k,\Delta t_k)$ is the process noise, which is also dependent on Δt_k. If the candidate area *i*th, ($Area_i$), is used to obtain the *k*th measurement $Z_i(k)$ of the target at t_k, the measurement model is given by

$$Z_i(k) = h_i(X(k)) + v_i(k) \qquad (11)$$

where h_i is a (generally non-linear) measurement function depending on *X(k)*, the measurement characteristic, and the parameters (e.g., the central location of cluster) of $Area_i$. $v_i(k)$ is the measurement noise in $Area_i$. Both *w* and v_i are independent and assumed to be with zero-mean, white, Gaussian probability distributions. The covariance matrices of $w(k,\Delta t_k)$ and $v_i(k)$ are $Q(\Delta t_k)$ and $R_i(k)$, respectively.

EKF operates in the following way: Given the estimate $\hat{X}(k|k)$ of X(k) at t_k with covariance *P(k|k)* and assuming that $Area_j$ is used for measurement at t_{k+1}, the predicted state $\hat{X}(k+1|k)$ of $Area_j$ at t_{k+1} can be calculated with the predicted state covariance

$$P(k+1|k) = F(\Delta t_k)P(k|k)F^T(\Delta t_k) + Q(\Delta t_k) \qquad (12)$$

The predicted measurement of $Area_j$ is

$$\hat{Z}_j(k+1|k) = h_j(\hat{X}(k+1|k)) \qquad (13)$$

Then, the innovation, i.e., the difference between the mea-surement $Z_j(k+1)$ of $Area_j$ and the predicted measurement of $Area_j$ at t_{k+1}, is given by

$$\gamma_j(k+1) = Z_j(k+1) - \hat{Z}_j(k+1|k) \qquad (14)$$

with the covariance

$$S_j(k+1) = H_j(k+1)P(k+1|k)H_j^T(k+1) + R_j(k+1) \qquad (15)$$

where H_j (k + 1) is the Jacobian matrix of the measurement function h_j at t_{k+1} with respect to the predicted state $\hat{X}(k+1|k)$

The EKF gain is given by

$$K(k+1) = P(k+1|k)H_j^T(k+1)S_j^{-1}(k+1) \qquad (16)$$

And the state estimation will be updated with the covariance matrix

$$P(k+1|k+1) = P(k+1|k) - K(k+1)S_J(k+1) \times K^T(k+1) \qquad (17)$$

Particularly, for a 2-dimensional constant velocity model with X(k) =(x(k), $x_v(k)$, y(k), $y_v(k))^T$ where x(k) and y(k) are the *x*- and *y*-coordinates of the target at time step k and $x_v(k)$ and $y_v(k)$ are the velocities of the target along *x*- and *y*-directions at t_k, the matrix $F(\Delta t_k)$ is given by

$$F(\Delta t_k)=\begin{bmatrix}1 & \Delta t_k & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & \Delta t_k\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (18)$$

In this paper, the matrix $Q(\Delta t_k)$ is assumed to be

$$Q(\Delta t_k)=q\begin{bmatrix}\frac{\Delta t_k^3}{3} & \frac{\Delta t_k^2}{2} & 0 & 0\\ \frac{\Delta t_k^2}{2} & \Delta t_k & 0 & 0\\ 0 & 0 & \frac{\Delta t_k^3}{3} & \frac{\Delta t_k^2}{2}\\ 0 & 0 & \frac{\Delta t_k^2}{2} & \Delta t_k\end{bmatrix} \qquad (19)$$

Where, q is a known scalar that determines the intensity of the process noise.

The tracking error $\Phi(k)$ at time step k is defined as the trace of the covariance matrix $P(k|k)$, i.e.,

$$\varphi(k)=Trace(P(k\,|\,k)) \qquad (20)$$

The definition of tracking error threshold Φ_0 is predefined. The tracking accuracy is considered satisfactory if the tracking error $\Phi(k)$ is not greater than Φ_0 ; otherwise, it is considered to be unsatisfactory. Also if we set tracking adjustments each T seconds in order to decrease the communication and calculation overhead, the term $\Phi^T_{j,\Delta t}$ means the predicted error until T seconds later using:

1) The sampling time interval Δt
2) The wakening up areas in form $area_j$.

Also it is important to mention that, the value of $\Phi^T_{j,\Delta t}$ is calculated using the values of Φ in previous sampling times.

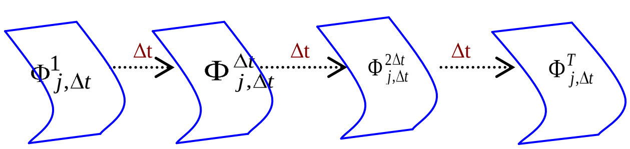

Figure 4. The calculation of $\Phi^T_{j,\Delta t}$ by using the past values of Φ

3.4 The estimation of energy consumption

In this section we estimate the energy consumption in the volunteer area *i*th, $Area_i$, which involves N_i sensor nodes. We used a simple radio model to estimate the consumed energy at the receiver and the transmitter. Suppose the network backbone is consisted of small clusters which every one has M sensor nodes. The consumed energy by a member node to transmit l bit message to a cluster head based on [10] is:

$$E_{NCH}=l(E_{elec}+E_{fs}d^2_{toch})T_{nchtx} \qquad (21)$$

where E_{elec} is the energy of electronic transmission/recepti-on, E_{fs} is the amplification factor, d_{toch} is the distance betwe-en each member node and its cluster head, T_{nchtx} is supposed to be the transmitting time for each member node to transmit l bit message to its cluster head.

The energy consumption in each *CH1* node to receive data from member nodes, integrate it and transmit to *CH2* node is denoted by E_{CH1} and is defined as:

$$E_{CH1}=MlE_{elec}T_{CHRX}+MlE_{DA}+ (lE_{elec}+lE_{fs}d^2_{CH1-CH2})T_{CH1-CH2} \qquad (22)$$

where $MlE_{elec}T_{CHRX}$ is the consumed energy by each CH1 node to receive l bit information from M member node, T_{CHRX} is the number of seconds for which each CH1 node listens to each member node, MlE_{DA} is the energy used for data aggregation. $(lE_{elec}+lE_{fs}d^2_{CH1-CH2})T_{CH1-CH2}$ is the consumed energy for transmitting the integrated information to CH2. Where $d_{CH1-CH2}$ is the distance between each CH1 and CH2. $T_{CH1-CH2}$ is supposed to be the transmitting time for each CH1 node to transmit l bit information to CH2 node.

As it was mentioned before, the CH2 node is selected among the CH1 nodes and according to paper [2], if assume the neighborhood degree of each CH1 node is q, then the number of message interchanges between CH1 nodes for introduction of CH2 node, *Num*, is estimated as:

$$Num=kq \qquad (23)$$

where k is the number of backbone clusters (the number of CH1 nodes), which are going to be merged, and is given as

$$k=\frac{N_i}{M} \qquad (24)$$

So the consumed energy for introducing *CH2* to the other CH1 nodes is estimated as:

$$E_{CH2-Intro}=Num(2lE_{elec}+lE_{fs}d^2_{CH1-CH1}) \qquad (25)$$

where $d_{CH1-CH1}$ is the distance between two CH1 nodes. The expression for the energy consumption by a CH2 node is given by:

$$E_{CH2}=klE_{elec}T_{toCH1}+klE_{DA}+ (lE_{elec}+lE_{mp}d^4_{BS})T_{CH2-BS} \qquad (26)$$

where $klE_{elec}T_{toCH1}$ is the consumed energy by the *CH2* node to receive l bits information from *CH1* nodes, T_{toCH1} is the number of seconds for which *CH2* node listens to each *CH1* node, and klE_{DA} is the energy used for data aggregation. Also $(lE_{elec}+lE_{mp}d^4_{BS})T_{CH2-BS}$ is consumed energy to transmit integrated information to base station where T_{CH2-BS} is supposed to be the transmitting time for *CH2* node to transmit l bit information to the base station.

Finally, the total energy consumption in $Area_i$ is denoted by $E_{Total}(i)$ and estimated as:

$$E_{Total}(i) = (N_i - 1)E_{NCH} + k(E_{CH1}) + E_{CH2-Intro} + E_{CH2} \quad (27)$$

If we suppose that during *T* seconds we use the same sampling time intervals and same size of wakening up areas so the term $E^T_{j,\Delta t_k}$ means the predicted energy consumption until *T* seconds later if we use

1) The sampling time interval Δt_k
2) The wakening up areas in size of $area_j$.

And is estimated as:

$$E^T_{j,\Delta t_k} = \frac{T}{\Delta t_k} E_{Total}(j) \quad (28)$$

3.5 Adaptive target tracking protocol

Suppose the current time step is *k* and the current tasking area is $Area_i$ with state estimation $\hat{X}(k|k)$ and estimation covariance matrix $P(k|k)$. Adaptive target tracking is used to select the optimal next tasking $Area_j$' and the associated sampling interval Δt_k during *T* seconds, such that the $Area_j$ can undertake the sensing task at time $t_{k+1} = t_k + \Delta t_k$. We suppose Δt_k is in the range $[T_{min}, T_{max}]$ where T_{min} and T_{max} are the given minimal and maximal sampling intervals. We divide the interval $[T_{min}, T_{max}]$ into N − 1 equally spaced sub-intervals by *N* discrete values $\{T_t\}_1^N$ where $T_1 = T_{min}$,

$T_N = T_{max}$, and they satisfy $T_{t_1} < T_{t_2}$ if $t_1 < t_2$

After predicting the tracking error and energy consumption for different candidate wakening up areas and different sampling time intervals, the optimum sampling time interval and volunteer wakening up area for the next T seconds are selected as:

$$(Area_j^*, \Delta t_k^*) = \arg\min_{j\in A, \Delta t_k \in [T_{min}, T_{max}]} \{Cost\_F(Area_j, \Delta t_k)\} \quad (29)$$

with

$$Cost\_F(Area_j, \Delta t_k) = w\frac{\Phi^T_{j,\Delta t_k}}{\Phi_0} + (1-w)\frac{E^T_{j,\Delta t_k}}{E_0} \quad (30)$$

If the predicted error meets this condition

$$\Phi^T_{j,\Delta t_k}(k+1) < \Phi_{Threshold} \quad (31)$$

where *A* is the set of candidate wakening up areas, $Cost\_F(Area_j, \Delta t_k)$ is the object function if $Area_j$ is selected with the sampling time interval Δt_k. $\Phi^T_{j,\Delta t_k}(k+1)$ is the predicted tracking error obtained from the update covariance $p(k+1|k+1)$ by (20) if $Area_j$ is used as the tasking area. Also $\frac{\Phi^T_{j,\Delta t_k}}{\Phi_0}$ is the normalized tracking error and $\frac{E^T_{j,\Delta t_k}}{E_0}$ is the normalized total energy consumption over *T* seconds. ($E^T_{j,\Delta t_k}$ is calculated by (28)).

$w \in [0,1]$ is a weighting parameter used to balance the tracking accuracy and the energy consumption. Also we used (31) to select the areas which their predicted error is less than $\Phi_{Threshold}$. Finally table.1 shows our proposed protocol briefly.

Table 1. Adaptive Target Tracking Algorithm

```
1: estimate the next location of target using (4, 5)
2: estimate the total area where the target would probably appear
   there using (6)
3: Determine the candidate wakening up areas by using (7)
        Set Area = {the candidate wakening up areas}
4: Determine the candidate sampling time intervals
        Set T = {the candidate sampling time intervals}
5: For j = 1 to |Area|
     For t = 1 to |T|
       Calculate Φ^T_{j,Δt_k}(k+1) by using Δt_k = T_t (20)
        If Φ^T_{j,Δt_k}(k+1) < Φ_Threshold
          Then
            Calculate E^T_{j,Δt_k} according to (28)
            Calculate Cost_F(Area_j, Δt_k) by using (30)
            End
       End
   End
6: Calculate (Area_j*, Δt_k*) according to (29)
```

4. PERFORMANCE EVALUATION OF THE PROPOSED PROTOCOL

In this section we compare our proposed protocol with the proposed methods in papers [2, 6]. In paper [2], the T^2BM^2 method has been proposed. In this method the size of waking up areas are adapted according to tracking errors in the previous times. In other words, in this method when the error of tracking increases, subsequently, the size of wakening up areas increases. On the other hand, the *AEC* method proposed in paper [6], adjusts the sampling time intervals according to tracking error in the past tracking periods. We will apply the proposed adaptive target tracking algorithm to the tracking of a moving sound source (the target) using a network of acoustic amplitude sensors. Using computer simulations, we evaluate the proposed algorithm from accuracy, power consumption points of view. The simulations are performed using Matlab. The measurement model for $Area_j$ is:

$$z_j(k) = \frac{a}{\|(x(k), y(k)) - (x_s(j), y_s(j))\|} + V_j(k) \quad (32)$$

where $a \in R$ is the assumed known amplitude of the sound source, $(x(k), y(k))$ is the center of $Area_{Total}$ and $(x_s(j), y_s(j))$ is known as the center of $Area_j$ and $V_j(k)$ is the zero-mean Gaussian measurement noise of $Area_j$ with variance σ_j^2. This measurement model corresponds to the nonlinear function h_j

$$h_j(X(k)) = \frac{a}{\sqrt{(x(k) - x_s(j))^2 + (y(k) - y_s(j))^2}}$$

with Jacobian matrix

$$H_j(k) = \begin{bmatrix} -\dfrac{a(x(k)-x_s(j))}{((x(k)-x_s(j))^2+(y(k)-y_s(j))^2)^{\frac{3}{2}}} \\ 0 \\ -\dfrac{a(y(k)-y_s(j))}{((x(k)-x_s(j))^2+(y(k)-y_s(j))^2)^{\frac{3}{2}}} \\ 0 \end{bmatrix}$$

Simulations are done to validate and characterize the performance of the proposed adaptive target tracking algorithm. The monitored field is $400m \times 400m$ and covered by 4000 randomly placed sensors. The sound source produces sound with constant amplitude a= 40. We assu-me σ_j^2 = 0.001 for any $Area_j$ and q = 10 in the covariance matrix of the process noise. The parameters listed in Table.2 are used in the energy model.

Table 2. Simulation settings of energy consumption

Parameter	Description	Setting
E_{fs}	The amplification factor for transmitting information to a near distance	10 nJoules/bit
E_{mp}	The amplification factor for transmitting information to a far distance	0.0013 pJoules/bit
E_{elec}	The electronic energy	50 nJoules/bit
E_{da}	The necessary energy for data aggregation	5 nJoules/bit
T_{nchtx}	The transmitting time for each member node to transmit *l* bit message to cluster head	0.5 sec
$T_{CH1\text{-}CH2}$	The transmitting time for each *CH1* node to transmit *l* bit information to *CH2* node	0.5sec
T_{toCH1}	The number of seconds for which *CH2* node listens to each *CH1* node	1sec
$T_{CH2\text{-}BS}$	The transmitting time for *CH2* node to transmit the information to base	1sec
l	Message size	6bit

For the sampling time intervals, we suppose N=10, T_{min}=0.1s and T_{max}=1s. Also we suppose target travels at a constant speed *v=5* m/s and it can vary its direction as a Normal distribution, *N(0,150)*.Also we suppose maximum 40 nodes and minimum 4 nodes participate in each tracking operation. For implementation of T^2BM^2 method we assume the sampling time interval to be *0.9* seconds. According to paper [6], in AEC method the number of nodes which participate in tracking is supposed to be 4 nodes.

Fig.5 shows the effect of size of wakening up areas on the tracking error for various sampling time intervals. It is obvious, when the size of wakening up areas increases, at first the amount of tracking error decreases rapidly but after a while, increasing the size of wakening up areas doesn't have a large effect on tracking error.

Also Fig. 6 illustrates the effect of size of wakening up areas on objective function. This figure shows that for all values of *w* when the number of nodes participating in tracking increases, at first, the amount of objective function decreases but after a while which is dependant on the amount of *w*, it gradually increases.

Also Fig.7 illustrates the variation of objective function regarding to different tracking errors. For all values of *w*, when the tracking error increases by decreasing the number of nodes, at first the amount of objective function decreases but after a while dependent on the amount of *w* it gradually increases.

Also in Fig. 8 and Fig. 9 our proposed protocol including two different values of *w* (*w*=0.1, *w*=0.2) has been compared with the T^2BM^2 and AEC protocols to evaluate the accuracy and energy consumption. By the way, the threshold error is supposed to be 14 cm. it is clear from Fig. 8, that the ATTP protocol has less tracking error compared to the other protocols. But when we take more amounts of *w* we can catch better accuracy. However, in Fig. 9 it has been shown that the energy consumption of the T^2BM^2 method is less than the other methods. Also the ATTP method including *w*=0.1 has energy consumption similar to the AEC method but when the amount of *w* increases, its energy consumption is more than the other methods.

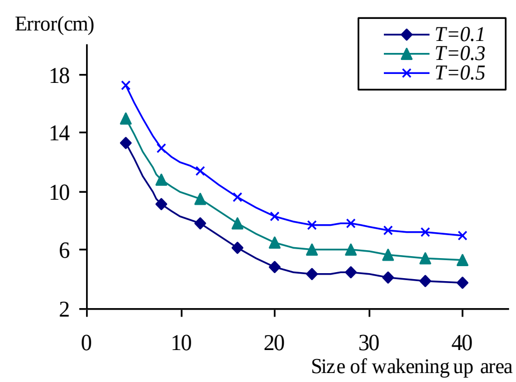

Figure 5. The effect of size of wakening up area on tracking error

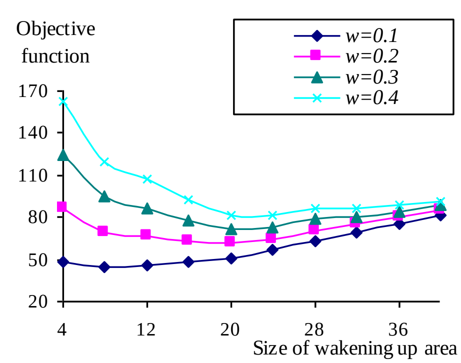

Figure 6. The effect of size of wakening up area on objective function (Δt=0.1).

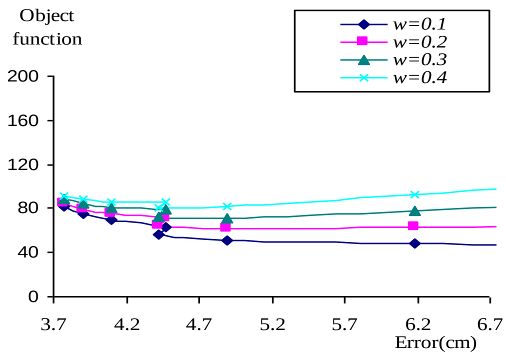

Figure 7. The effect of error on objective function (Δt=0.1).

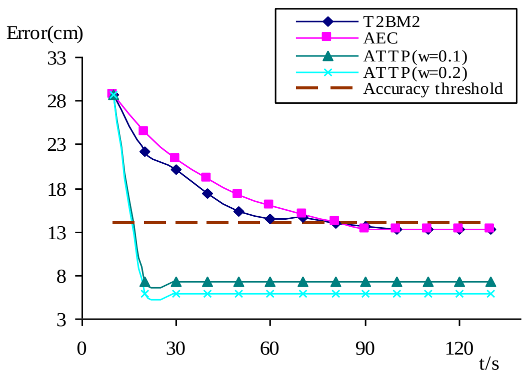

Figure 8. Tracking error in different tracking schemes.

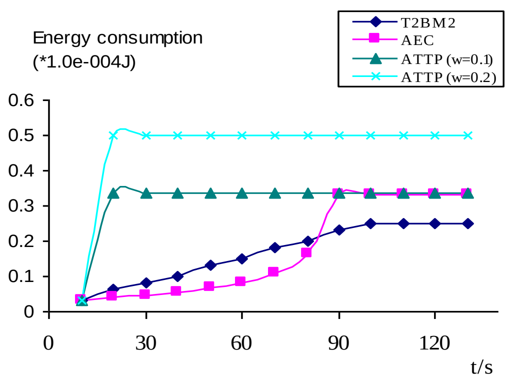

Figure 9. Energy consumption in different tracking schemes.

5. CONCLUSION

This paper has presented a new adaptive target tracking protocol in *WSN*s. In this method, at first the total area that the target would appear is estimated, after that we define some candidate wakening up areas based on this estimation and consider some volunteer sampling time intervals. Then we use two helpful tools. First, an extended Kalman filter (EKF)-based estimation technique to predict the tracking error. Furthermore, we use an energy consumption model to estimate the amount of energy consumption based on different number of nodes and sampling time intervals. After that, the optimum number of nodes and the sampling time interval are selected according to an objective function which is defined based on tracking accuracy and energy consumption. Simulation results show that, compared to the other previous adaptive tracking methods, our method has better tracking accuracy with similar energy consumption. There are many issues remaining for future study. This paper adopts EKF as the estimation and prediction algorithms; however, EKF can only deal with noises with Gaussian distributions and more advanced techniques (such as particle filter) are required for adaptive sensor scheduling with more general (non-Gaussian) noises. Multi-step, adaptive motion model based target tracking and multi-target tracking are other challenging problems for further investigations.

6. REFERENCES

[1] H. W. Tsai, C. P. Chu, T. S. Chen, “Mobile object tracking in wireless sensor networks,” Computer Communication, 2007, 30(8): 1811-1825.

[2] E. Ahmadi, M. Sabaei, M. H. Ahmadi “A New Adaptive Method for Target Tracking in Wireless Sensor Networks”, International Journal of Computer Applications, volume22, No.9, 2011, pp.21-29.

[3] Sara Pino-Povedano, Francisco-Javier Gonzalez Serrano, “Distributed Tracking and Classification of Targets with Sensor Networks,” IEEE 2009.

[4] Z. Guo, M. Zhou and L. Zakrevski, "Optimal Tracking Interval for predictive Tracking in wireless Sensor Networks," IEEE Communication Letters, vol. 9, No.9, Sept 2005.

[5] L. Yang, C. FENG, J. W. Rozenblit , and H. Qiao, "Adaptive Tracking in distributed Wireless Sensor Networks," in proc. Of 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems, pp. 9, Mar. 2006.

[6] H. Jamali Rad, B. Abolhassani and M. Abidzadeh, "A new Adaptive Prediction–based Tracking Scheme for Wireless Sensor Networks ,"Proc. Of 7th Annual conference on communication Networks and System Research(CNSR09) , pp. 335-341, May, 2009.

[7] W. Xiao, Sen. Zhang, J. Lin, C. K.Tham, "Energy-efficient adaptive sensor scheduling for target tracking in wireless sensor networks", J Control Theory Appl, 2010, vol. 8, No. 1, pp.86-92.

[8] M. Kalandros, L. Pao. Covariance control for multisensor systems[J] .IEEE Transactions on Aerospace and Electronic Systems, 2002, 38(4): 1138 − 1157.

[9] HyunSook, K; Eunhwa, K; Kijun,H. An Energy Efficient Tracking Method in Wireless Sensor Networks. In Next Generation Teletraffic and Wired/Wireless Advanced Networking, Publisher: Springer Berlin / Heidelberg, 2006;4003, pp. 278- 286.