

Energy efficient fast predictor for WSN-based target tracking

M. Mirsadeghi · A. Mahani

Received: 12 July 2013 / Accepted: 26 February 2014 / Published online: 28 March 2014
© Institut Mines-Télécom and Springer-Verlag France 2014

Abstract Power source replacement of the sensor nodes, which are once deployed in the network area, is generally difficult. So, energy saving is one of the most important issues for object tracking in wireless sensor networks. To reduce the consumed energy and prolong the network lifetime, the nodes surrounding the mobile object should be responsible for sensing the target. The number of participant nodes in target tracking can be reduced by an accurate prediction of the object location. In this paper, we present a fast energy efficient with high-accuracy target tracking scheme which is based on location prediction. The missing rate of proposed predictor is very low in comparison with other predictors especially in a random waypoint mobility model in which after pause time, the three main parameters direction, velocity and, acceleration would be changed. The accuracy of predictor has a direct effect on missing rate and so strongly reduces the consumed energy. Additionally, a new node selection criterion is proposed in which minimum nodes surrounding the object are wakened and track the object. Simulation results show that our proposed predictor has low consumed energy and complexity in comparison with Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) predictors.

Keywords Target tracking · Wireless sensor networks · Energy efficient · Kalman filter · Random waypoint

1 Introduction

Rapid advances in miniaturization in computing and sensor technologies and advent of low-power short-range radios

recently have given rise to strong interest in wireless sensor networks [1]. Each sensor node has a capability of sensing the environment, processing data, and communicating with the base station. All these tasks are done without human involvement, and this makes such networks have a lot of applications in both military and civilian fields [2, 3].

Tracking moving objects are emerging as applications such as wild animal habit monitoring, military and civilian scenarios, environment monitoring, etc. [2]. The sensor nodes are used to find mobile object position which is a complicated scenario and is accompanied with collaborative works between nodes.

But, the use of sensor networks for object tracking faces a number of main issues. These issues contain limited energy supply and communication bandwidth, distributed algorithms and control, and handling the fundamental performance limits of sensor nodes, especially as the size of the network becomes large. Unlike traditional networks, a wireless sensor network (WSN) has resource constraints. Resource constraints contain a limited amount of energy, short communication range, low bandwidth, and limited processing and storage in each node [1, 4]. Tracking of the mobile objects should solve main problems such as object detection, localization, and prediction [5].

In the localization problem, excessive sensors may join in detection and tracking for only a few objects. And, if all nodes have to always wake up to detect a mobile target, there are a lot of waste of battery power and channel utilization. Actually, power conservation is one of the most critical issues in target tracking since it would be difficult to replace the battery of the sensor nodes that are once deployed in the network area [3].

Many research works about object tracking are focused on finding the optimum point of energy consumption, tracking accuracy, and calculation time [6]. So, many researchers try to reduce the number of active nodes during tracking process [5,

M. Mirsadeghi · A. Mahani (✉)
Department of Electrical Engineering, Shahid Bahonar University of
Kerman, 76169133, Kerman, Iran
e-mail: amahani@uk.ac.ir

7–9]. The tracking methods are classified into three categories:

1. Tree-based methods [10–12]
2. Cluster-based methods [9, 13, 14]
3. Prediction-based methods [7, 15, 16]

In prediction based methods, the next location of a moving object is predicted. Then, during each defined time step, only some nodes near the predicted location are activated and other nodes stay in sleep mode as energy save state. For example, in [15], advantages of prediction mechanism in a cellular network lead to a limited search space of object tracking and so strongly reduces the paging overheads. In [7], authors use prediction mechanism to decrease the number of active nodes. So, in each time step, just one hop-surrounding nodes of the next predicted location of the object would be activated. In [16], by using the past sensing history and spatial and temporal knowledge of sensors, the future location can be predicted, and just a few number of nodes are activated.

In most target tracking applications, tracking and prediction mechanisms are based on object motion model. For example [5, 7, 17–21], assume that the target has a linear mobility with constant velocity. In [7] the regression based prediction and Kalman filter are used. The linear predictor is used to predict the next position of an object in which the current location and the previous location of the object are taking account [5, 17]. Authors in [18] assume that the object just sometimes move nonlinearly, so they use a moving average estimator with a proposed correction mechanism. In [19–21], Kalman filter-based methods such as extended Kalman filter (KF) and SOI-KF was proposed.

In [22, 23], particle filter is proposed to predict the location of the object, and in [8], a new adaptive prediction mechanism is introduced in which the position of target for the next t_0 seconds would be predicted. t_0 is denoted as escape period and is proportional with the distance between target and node.

Gauss Markove mobility model is considered in [15], and authors propose a simple prediction model in which the object mobility stays constant for a certain period of time. In [24], authors use an extended Kalman filter to track the moving target with nonlinear mobility model.

So far, there are no research works about prediction-based WSN tracking object with variable velocity. So, the previous methods lost the variable speed object during tracking process. It is clear that the missing rate of the object leads to a high total energy consumption in the network. In WSN object tracking mechanisms, the energy consumption is dependent on the missing rate, since the nodes must find the lost object via recovery mechanism as a high-energy consumption procedure [25]. In an accurate predictor, less nodes would be activated for collaborative tracking and lead to negligible consumed energy and transmission overheads.

In this paper, we define a deferent kind of mobility model for a moving object and a new prediction-based energy efficient and high-accurate tracking method in a clustered sensor network is introduced. This method is suitable for tracking more than one object coming into the considered area simultaneously and are far enough from each other. If they move close to each other and there are uncertainties in the association of observed measurements, we should apply multi-target tracking algorithms [2] which are outside the scope of this paper.

According to our proposed method, in each time instance, only some nodes with the following characteristics will be selected for tracking and other nodes go to the power saving mode:

- Near to the target
- Have more energy
- Less distance to their head node

Despite of the previous works [22–24, 26, 27], we assume a nonlinear measurement from the object with variable speed. In addition, a new recovery mechanism is introduced to find the lost object with high accuracy.

The results show that the proposed predictor has good accuracy with low missing rate which leads to an energy efficient object tracking method. Also, our proposed predictor reduces the number of operations without losing the prediction accuracy.

2 Preliminaries

2.1 Environment

In this paper, a WSN with large number of stationary sensors is considered in which nodes are capable of sensing, computing, and communicating. Each node knows its own location by using GPS or other location awareness techniques. They can sense the environment, detect the target, and track it. The understudy network is a cluster-based heterogeneous sensor network which consists of two types of nodes: cluster head and ordinary nodes in which just head nodes send the final tracking information to the sink.

Here, all cluster head nodes work in two modes, idle mode and active mode, and all ordinary nodes work in three modes, sleep, idle, and active modes. So, in our energy efficient prediction mechanism, in each time step, just a few numbers of nodes nearby the predicted location of target will be activated as tracker nodes and the others will stay in power-saving mode.

2.2 Energy consumption model

In this section, the mathematical model of consumed energy in target tracking sensor network is introduced. During sleep and

idle modes, the power consumption of a node is ignored. Also, we do not consider the energy overhead for switching between operating modes of nodes. For active nodes, all of their components are active and the energy consumption of sensing and communication components is considered.

Each node that wakes up senses the environment for a certain period of time t_{sense} and consumes energy E_{sense} , then sends its data packets to the head node. The energy consumption model for transmitter is given by [28]:

$$\left\{ \begin{array}{l} \text{if } d > d_0 E_t = l E_{\text{elec}} + l \varepsilon_{\text{mp}} d^4 + l E_{\text{DA}} \\ \text{if } d \leq d_0 E_t = l E_{\text{elec}} + l \varepsilon_{\text{fs}} d^2 + l E_{\text{DA}} \end{array} \right\}$$

where E_{DA} is the energy for data aggregation in head nodes, E_{elec} is the dissipated per bit to run the transmitter or receiver circuit, ε_{fs} and ε_{mp} depend on the transmitter amplifier model, d is distance, and l is the packet length. The consumed energy for receiver nodes E_r is as follows:

$$E_r = l E_{\text{elec}}$$

3 Energy efficient tracking scheme

3.1 Target tracking mechanism

First, all head nodes are in idle mode. Once received, an alarm message from sink wakes up and senses the environment. Each head node, which finds the object, stays active, and others go back to idle mode in which they just can listen to communication channel to check messages. The head node should carry out the necessary computation and find the best sensor nodes for object tracking. Then, it sends a wake-up message to ordinary nodes which are initially in sleep mode and changes their state to idle mode periodically and synchronously to check the messages. If they receive a wake-up message, they become active; otherwise, they return to sleep mode. If the object is going to leave the current head node’s cluster, it informs the next head node by sending a wake-up message. Also, the current position of the object would be sent to the next head node. Head nodes send their information to the sink for a period of length T seconds.

3.2 Measurements and localization model

It is assumed that each sensor node can find the approximate angle of the target in its sensing range. So, the nonlinear measurement model for sensor $i \in \{1, 2, \dots, n\}$ at time instant t is as follow:

$$\theta_i = \text{Arctan} \left(\frac{y_t^i - S_y^i}{x_t^i - S_x^i} \right) + v_i \tag{1}$$

where (S_x^i, S_y^i) is the position of the sensor, $i, (x_t^i, y_t^i)$ is the real position of the target, and v_i is the sensing error, which is zero mean, Gaussian distribution with constant standard deviation of σ_θ .

There are various works that have been done on the bearing-only localization. Among them, least-square algorithm has a closed form solution and does not need to solve nonlinear problem which make it computationally easy. So, it can be suitable for object localization in the network. LS algorithm is based on assumption that the node-sensing errors are sufficiently small. Therefore, the head node can adopt the least-square algorithm to determine the target location, with node sensing collaboratively [29]. The true value of an angle can be written as

$$\tan(\theta_i) = \frac{\sin(\theta_i)}{\cos(\theta_i)} = \left(\frac{y_t^i - S_y^i}{x_t^i - S_x^i} \right) \tag{2}$$

In presence of noise, we can write

$$FX = h \tag{3}$$

Where X is the location of the target and the matrices, F and h , are given by

$$F = \begin{bmatrix} \sin\theta_1 & \cos\theta_1 \\ \vdots & \vdots \\ \sin\theta_n & \cos\theta_n \end{bmatrix}, \quad h = \begin{bmatrix} S_x^1 \cdot \sin\theta_1 - S_y^1 \cdot \cos\theta_1 \\ \vdots \\ S_x^n \cdot \sin\theta_n - S_y^n \cdot \cos\theta_n \end{bmatrix}$$

The least square solution of Eq. (3) is

$$\hat{X}_{\text{LS}} = (F^T F)^{-1} F^T h \tag{4}$$

3.3 Target mobility model

There are many mobility models for moving objects such as models in [8, 15, 18, 22, 23]. For modeling the motion of some mobile targets like a vehicle, a mobile target moves for a random period of time in a random direction with variable speed and acceleration that are in ranges $[0, v_{\text{max}}]$ and $[0, a_{\text{max}}]$. This vehicle cannot change its direction suddenly. It should stop for a short random period of time and then starts moving and changing its direction. Also the congestion leads to variable speed for the vehicle most of the time. In fact, we use random waypoint and constant acceleration model (RCAM) for target mobility in which random waypoint model uses pause times before changing three parameters: speed, direction, and acceleration; the state equations for the RCAM contain acceleration component

Table 1 Parameters for energy consumption model

E_{elec}	50nJ/bit
E_{DA}	5nJ/bit/signal
E_{sense}	106μJ/s
ϵ_{mp}	0.0013pJ/bit/m ⁴
ϵ_{fs}	100pJ/bit/m ²
d_0	100√2
t_{sense}	4.81 ms
Data packet size	4000 bits
Signal packet size	64 bits

model state transition matrix, w_k is the process noise that is white Gaussian noise with zero mean and variance matrix $Q(k)$, and $G(k)$ is the process noise input matrix. The related matrices are given by:

$$A = \begin{bmatrix} 1 & 0 & T & 0 & \frac{T^2}{2} & 0 \\ 0 & 1 & 0 & T & 0 & \frac{T^2}{2} \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & T \end{bmatrix} \quad w_k = \begin{bmatrix} w_x \\ w_y \end{bmatrix}$$

which is time invariant. The mobility equations are as follows:

$$x(t) = x(t-1) + v(t) * \Delta t \tag{5}$$

$$v(t) = v(t-1) + a(t) * \Delta t \tag{6}$$

$$a(t) = a(t-1) \tag{7}$$

By considering the process noise for target movement, dynamic model of object under tracking can be defined as:

$$X_k = AX_k + Gw_k \tag{8}$$

where k is the time index, $X_k = [x(k) y(k) v_x(k) v_y(k) a_x(k) a_y(k)]^T$ is the state vector which contains relative positions, velocities, and accelerations of the target in two dimension plans, A is the

3.4 Proposed predictor

Proposed predictor is based on prediction error described by the angle between the actual location and previously predicted location, denoted by θ_n . Since it is assumed that a moving object has constant acceleration, so after predicting the speed of the object in the next time step, we try to find the next position of the object based on prediction error. If θ_n is in range $[-\alpha, \alpha]$, the velocity of the moving object is given by:

$$v(t + 1) = 2 * v(t) - v(t-1) \tag{9}$$

Then the next location of the object is predicted as follows:

$$\begin{aligned} x(t + 1) &= v(t + 1) * T * \cos\Phi + x(t) \\ y(t + 1) &= v(t + 1) * T * \sin\Phi + y(t) \end{aligned} \tag{10}$$

Fig. 1 Number of dead nodes vs. simulation time for two different methods (without predictor and prediction-based)

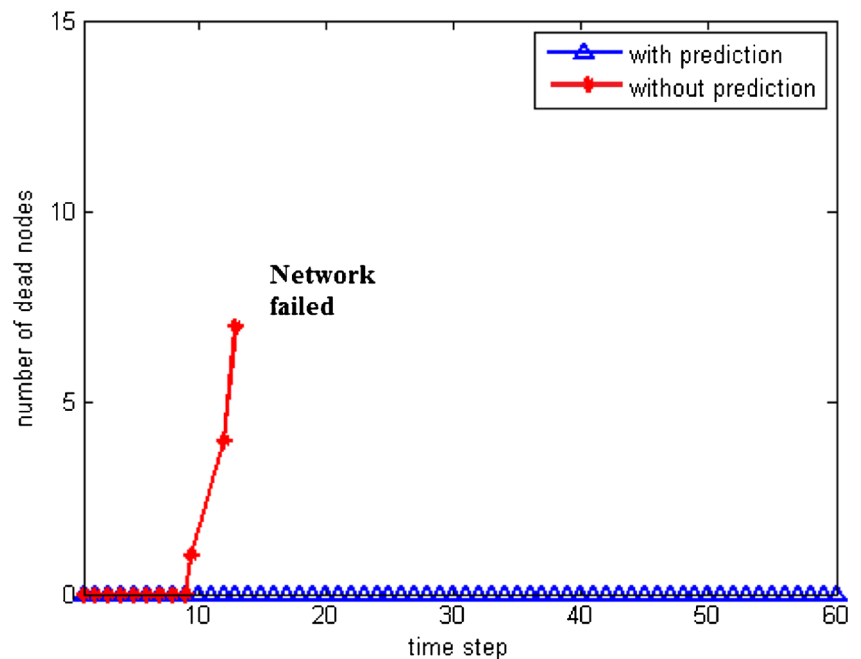
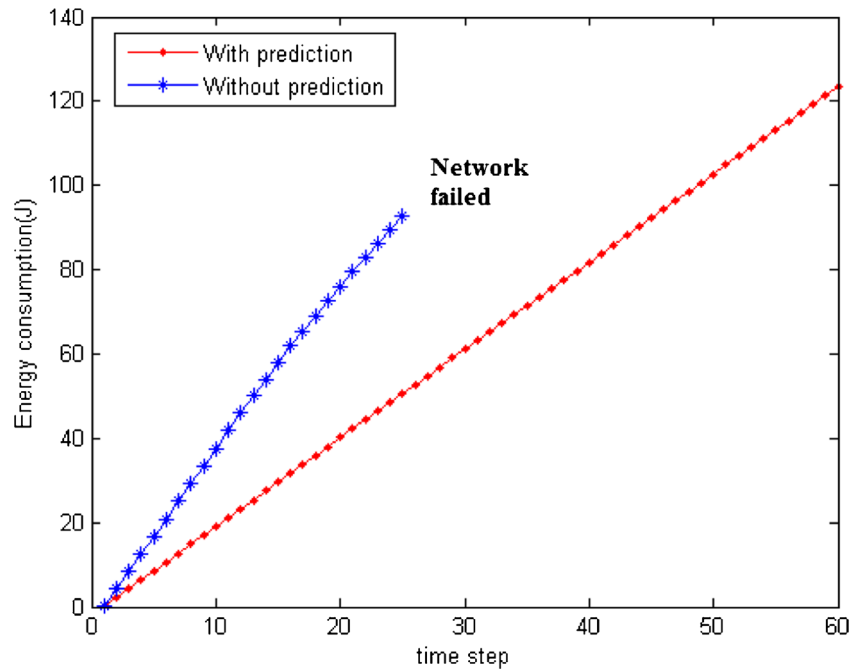


Fig. 2 Energy consumption of two different methods (without predictor and prediction-based) vs. simulation time



where $\Phi = \arctan\left(\frac{y(t+1)-y(t)}{x(t+1)-x(t)}\right)$
 but if θ_n is not in the range $[-\alpha, \alpha]$, so we have

$$\begin{aligned} v_x(t+1) &= 2*v_x(t)-v_x(t-1) \\ v_y(t+1) &= 2*v_y(t)-v_y(t-1) \\ x(t+1) &= v_x(t+1)*T + x(t) \\ y(t+1) &= v_y(t+1)*T + y(t) \end{aligned} \tag{11}$$

3.5 Node selection algorithm

The node selection algorithm is the next step after prediction mechanism. This algorithm finds and

activates three suitable sensor nodes to guarantee the accuracy of the next prediction and also prolong the network lifetime. Current prediction-based methods consider nearest nodes to the predicted location such as [14, 15] and in [5]; remaining energy of nodes is considered too. In the proposed algorithm, three parameters are used as selection criterion:

- Distance
- Remaining energy of nodes
- Energy consumption for sending a packet to the head node

Fig. 3 Prediction error of different predictors

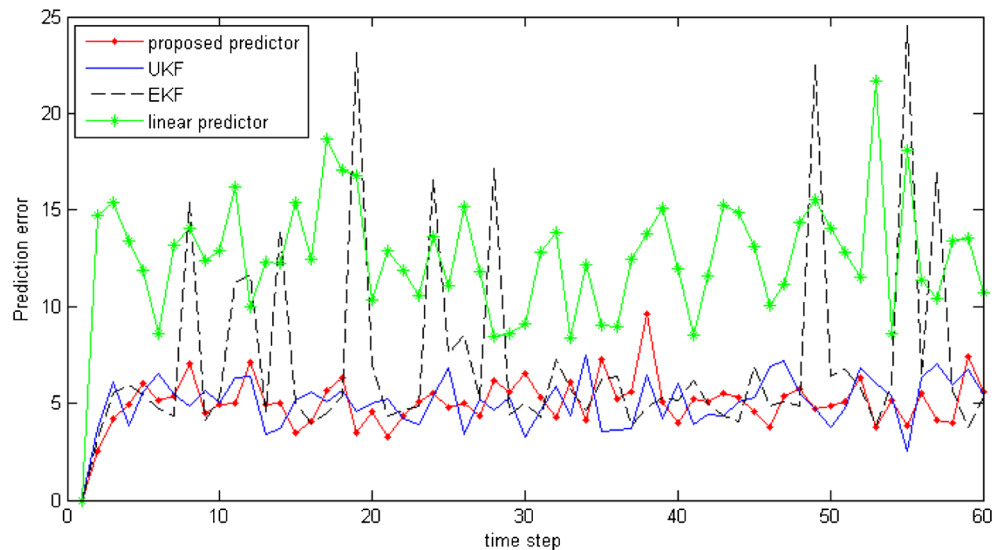
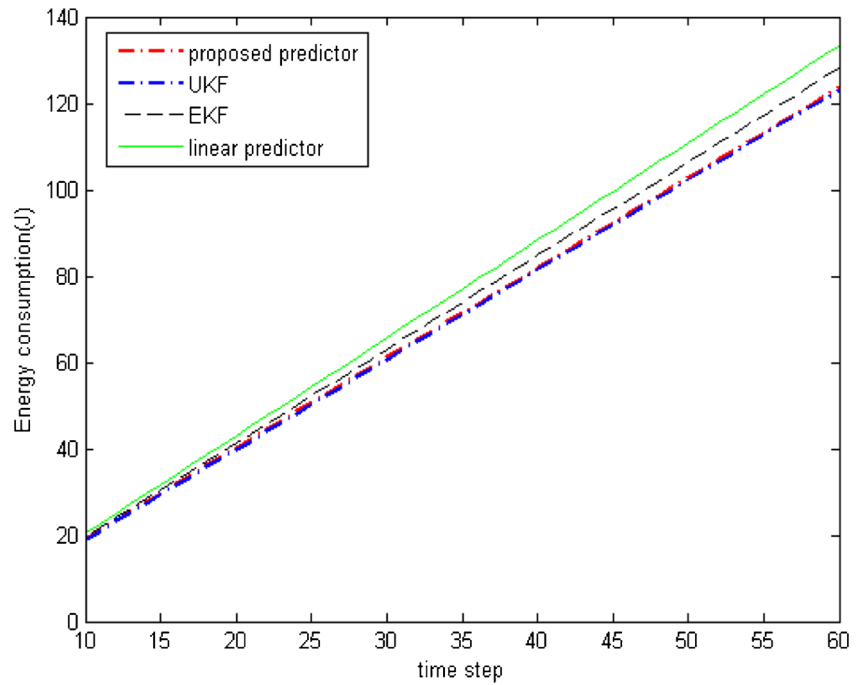


Fig. 4 Consumed energy of different predictors vs. simulation time



The active head node calculates the selection criterion as follows:

$$\text{Selection}_i = \frac{E_{\text{remain}_i} - E_{\text{send}_i}}{d_i^2}$$

where d_i is the distance of node i from the object, E_{remain_i} is the remaining energy of node i , and E_{send_i} is the consumed energy of node i for sending a packet to the head node. Then head node chooses three nodes with maximum selection parameter for tracking purpose. Hence, in each time step, the closer nodes to the object with more energy and shorter distance from the head node will be selected. The proposed selection criterion with three mentioned parameter helps us to prolong the lifetime of target tracking system.

3.6 Recovery

There is a possibility of getting a lost object in the presence of link failures, node failures, or prediction errors. So, a recovery procedure is needed to find the lost object. In this procedure, when a head node discovers the moving object, it sends a packet to the previous head node to inform the location of the object. If the previous head node does not receive the packet, it means that the object is lost and so informs the sink node. The head nodes surrounding the cluster in which the object is lost are awakened by sink to sense the environment and try to find the lost object.

4 Simulation results

In this section, we try to analyze the effect of prediction mechanism on energy efficiency and lifetime of target tracking system. Also, the accuracy of predictors and the effect of their accuracy on missing rate and energy consumption are considered. So, the linear predictors [5], a modified version of Kalman filter named extended KF (EKF) which was introduced in [26] and another modified version of Kalman filter named unscented KF (UKF) [27], are considered to have a fair comparison. Finally, the impact of node selection algorithm on network lifetime is investigated.

4.1 Simulation environment

The network area is 400×400 which is divided into 16 equal-size clusters, and so, 16 head nodes with initial energy of 50 J are placed at the center of these clusters. The sink node is located out of the area ($x=200, y=450$). Three hundred sensor nodes are uniformly distributed in the network area with sensing range $R_{\text{sensing}}=60m$ and initial energy of 0.5 J. For

Table 2 Comparison of speed and missing rate of different predictors

Predictor	Number of Operations	Missing rate
Proposed predictor	28,440	0.69%
UKF	702,017	0.88%
EKF	107,734	3.92%
Linear	14,193	4.14%

Table 3 Network lifetime comparison for different methods of node selection

Node selection method	Lifetime
Proposed method	11,946 s
Distance and remaining energy	10,061 s
Distance-based selection	5,339 s

each sensor node, the standard deviation of measurement noise is $\sigma_\theta=1$.

During target tracking process, the sink receives information about the moving object, every $T=0.5s$. For the target, we have the maximum acceleration $a_{max}=15m/s^2$, the maximum velocity $v_{max}=30m/s$, and the variance matrix of process noise $Q=5^2 \times I_{2 \times 2}$. For the predictor, the bound of the prediction location error is $a=15$ (see Sections 3–4). The parameters of energy consumption model are shown in Table 1.

4.2 Advantages of prediction-based methods

In this section, the effect of prediction mechanism on network performance metrics is discussed. We compare two target tracking methods: prediction-based method which uses proposed predictor and nonprediction method. Then, the energy consumption, number of dead nodes, and lifetime of two understudy tracking methods are compared. As shown in Fig. 1, the number of dead nodes in prediction-based method is zero during simulation time, but the network without prediction is failed in the 15th time step of simulation.

Figure 2 shows that in nonprediction method energy consumption of system is too high so that the network failed (the

connected network graph becomes at least two split graph), very soon in comparison with prediction based method. The reason is that in each time instant T , the head node has to wake all of the nodes up to track the target, but in prediction mechanism, only three most suitable nodes are woke up for tracking purpose.

4.3 Accuracy and missing rate

First, we compare the accuracy of four mentioned predictors that is determined by square root of the difference between the predicted and actual location of the object in two dimensions. The prediction error is given by:

$$pre_error^t = \sqrt{(x_{pre}^t - x_{real}^t)^2 + (y_{pre}^t - y_{real}^t)^2}$$

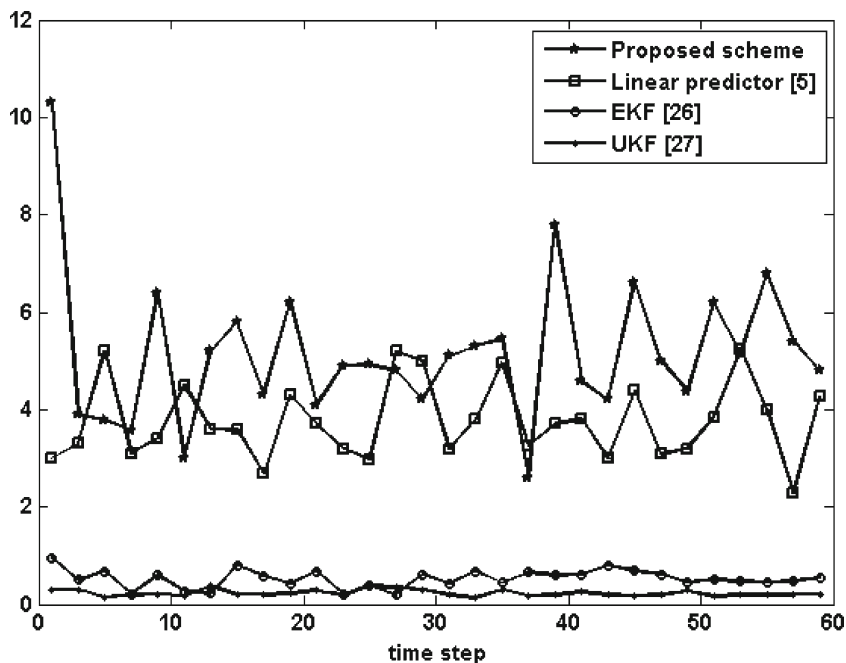
where the pre_error^t is the prediction error of predictor at time t .

As shown in Fig. 3, the UKF predictor and our proposed predictor have the lowest prediction error and so are more accurate than the others. But, as we expect, the linear current predictor has the greatest prediction error and EKF predictor is in the middle. EKF predictor has good accuracy most of the time, except in some peak points. The EKF predictor uses a first-order linearization that sometimes leads to instabilities during simulation time.

4.4 Energy consumption

When the object is missed, a recovery mechanism is needed to find the lost object, and this leads to excessive energy

Fig. 5 The defined performance factor α vs. simulation time



consumption in the network. After missing the object, all nodes in a cluster should wake up, sense the environment, and transmit their packet to head nodes. Figure 4 shows the total energy consumption of network from 10 to 60min. As we expect, the predictors with lower missing rates have less energy consumption than the others. We point out that consumed energy of prediction procedure is neglected in Fig. 4.

Also, another main evaluation metric for target tracking algorithm by WSN is the number of operations. Table 2 shows a comparison between accuracy and number of operations in these four predictors in a 1-h target tracking. As shown in Table 2, the linear predictor has lower accuracy with the fastest run time because of its low number of operation. But, UKF predictor with the high accuracy has a large number of operations while computational capacities of sensor nodes are constraint. The proposed predictor gives an optimum point between accuracy and run time speed. So, our proposed predictor has good accuracy that can be equal to the UKF

predictor and the number of operations about 24.7 times less than UKF. The number of operations is a man-evaluating factor of predictors especially in WSNs with processing resource constraints.

4.5 Effect of node selection algorithm on network lifetime

Finally, we try to evaluate our proposed selection criterion. Table 3 shows the effect of proposed method on network lifetime. As depicted in Table 3, when considering three-element selection criterion, the lifetime of network is increased about 10 %. Since in each time step, three nodes with highest level of energy and lowest distance to the CH are selected.

To have a fair comparison between the predictors, the following factor is defined in which the normalized number of operations, predictor error rate, and consumed energy are considered.

$$\alpha = \frac{1}{(\text{Normalize number of operations}) * (\text{Error rate}) * (\text{Energy consumption ratio})}$$

So, as shown in Fig. 5, our proposed predictor gives the best α , as defined factor, during simulation time. The linear predictor has the lowest number of operation and so leads to great α , in comparison with UKF and EKF.

5 Conclusion

Tracking of the mobile objects should solve main problems such as object detection, localization, and prediction. And, if all nodes have to always wake up to detect a mobile target, there are a lot of waste of battery power and channel utilization.

In this paper, we propose an energy efficient tracking method to reduce the number of nodes participating in object tracking. Simulations results show that our method leads to saving energy and thus prolong the network lifetime as well regardless of mobility pattern of the mobile object including random waypoint model.

The proposed predictor causes lower missing rate than linear, EKF and UKF predictors, and the number of operations which is the main determinant factor in the speed of the predictor is 24.7 times less than UKF. Also against the moving average estimator [18], a correction step is necessary and leads to extra computation. But, this step is removed from our proposed method and so, the numbers of operations are reduced. So, the proposed predictor helps the network to

increase the speed of prediction and decrease the energy consumption.

References

1. Boukerche MA, Richard W, Pazzi N (2008) Energy-aware and quality of service-based routing in wireless sensor networks and vehicular ad hoc networks. Institut TELECOM and Springer France, pp669–681
2. Rezaei TY, Tinati MA (2011) Distributed multi-target tracking using joint probabilistic data association and average consensus filter. Institut Télécom and Springer, pp553–566
3. Pathan AK, Hong CS (2008) SERP: secure energy-efficient routing protocol for densely deployed wireless sensor networks. Institut TELECOM and Springer, France, pp 529–541
4. Monowar MM, Alam MM, Obaidur Rahman Md, Hong CS, Lee S (2010) A load-aware energy-efficient and throughput-maximized asynchronous duty cycle MAC for wireless sensor networks. Institut Télécom and Springer, pp777–794
5. Deldar F, Yaghmaee MH (2011) Designing a prediction-based clustering algorithm for target tracking in wireless sensor networks. International symposium on computer networks and distributed systems (CNDS), pp 199–203
6. Nandhini M, Sarma Dhulipala VR (2012) Energy-efficient target tracking algorithms in wireless sensor networks: an overview. Int J Comput Sci Technol IJCST 3(1):66–71
7. Hsua JM, Chenb CC, Li CC (2012) POOT: an efficient object tracking strategy based on short-term optimistic predictions for

- face-structured sensor networks. Elsevier Int J Comput Math Appl 63(2):391–406
8. Yang L, Feng C, Rozenblit JW, Qiao H (2006) Adaptive Tracking in Distributed Wireless Sensor Networks. Proceedings of the 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems, pp 9–11
 9. Olule E, Wang G, Gu M, Dong M (2007) RARE: an energy efficient target tracking protocol for wireless sensor networks. In: Intl conf on parallel processing workshops (ICPP 2007), pp 76–81
 10. Kung HT, Vlah D (2003) Efficient Location Tracking Using Sensor Networks. Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), pp 1–8
 11. Zhang W, Cao G (2004) DCTC: dynamic convoy tree-based collaboration for target tracking in sensor networks. IEEE Trans Wirel Commun 3(5):1689–1701
 12. Tsai HW, Chu CP, Chen TS (2007) Mobile object tracking in wireless sensor networks. Comput Commun 30(8):1811–1825
 13. Chen W-P, Hou JC, Sha L (2004) Dynamic clustering for acoustic target tracking in wireless sensor networks. IEEE Trans Mob Comput 3(3):258–271
 14. Yang WC, Fu Z, Kim JH, Park MS (2007) An adaptive dynamic cluster-based protocol for target tracking in wireless sensor networks. Adv Data Web Manag pp 157–167
 15. Xu Y, Winter J, Lee WC (2004) Prediction-based strategies for energy saving in object tracking sensor networks. Proceedings of the Fifth IEEE International Conference on Mobile Data Management (MDM'04), pp 346–357
 16. Goel S, Imielinski T (2001) Prediction-based monitoring in sensor networks: taking lessons from MPEG. ACM Comput Commun Rev 31(5):82–98
 17. Bhuiyan MZA, Wang GJ, Yong P, Zhang L (2010) Prediction-based energy-efficient target tracking protocol in wireless sensor networks. J Cent S Univ Technol 17:340–348
 18. Kim H, Kim E, Han K (2006) An energy efficient tracking method in wireless sensor networks, next generation teletraffic and wired/wireless advanced networking. LNCS 4003:278–286
 19. Di M, Joo EM, Beng LH (2008) A comprehensive study of Kalman filter and extended Kalman filter for target tracking in wireless sensor networks. IEEE Int Conf Syst Man Cybern pp 2792–2797
 20. Yadav A, Naik N, Ananthasayanam MR, Gaur A, Singh YN (2012) A constant gain Kalman filter approach to target tracking in wireless sensor networks. IEEE Int Conf Ind Inf Syst pp 1–7
 21. Ribeiro A, Schizas ID, Roumeliotis SI, Giannakis GB (2010) Kalman filtering in wireless sensor network. IEEE Control Syst Mag 30(2): 66–86
 22. Wang X, Ma JJ, Wang S, Bi DW (2007) Prediction-based dynamic energy management in wireless sensor networks sensor networks. Sensors J 7(3):251–266
 23. Wang X, Ma JJ, Wang S, Bi DW (2007) Cluster-based dynamic energy management for collaborative target tracking in wireless sensor networks. Sensors J 7(7):1193–1215
 24. Medeiros H, Park J, Kak AC (2008) Distributed object tracking using a cluster-based Kalman filter in wireless camera networks. IEEE J Sel Top Sign Process 2(4):448–463
 25. Jin G, Lu X, Park M-S (2006) Dynamic clustering for object tracking in wireless sensor networks”, ubiquitous computing system 06. LNCS 4239:200–209
 26. Pant B, Alkin O (2012) Correlated movement mobility model and constant acceleration model for EKF-based tracking applications. IEEE Wirel Mob Comput Netw Commun pp 869–874
 27. Gadsden SA, Dunne D, Habibi SR, Kirubarajanb T (2009) Comparison of extended and unscented Kalman, particle, and smooth variable structure filters on a bearing only target tracking problem. Signal Data Proc Small Targets Proc SPIE vol. 7445
 28. Heinzelman WB, Chandrakasan AP, Balakrishnan H (2002) An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wirel Commun 1(4):660–670
 29. Vaghefi RM, Gholami MR, Strom EG (2010) Bearing-only target localization with uncertainties in observer position. IEEE Pers Indoor Commun Work pp 238–242