Sensor Scheme for Target Tracking in Mobile Sensor Networks

Hao Dong[1,2](✉) and Qiang Liu[1,2]

[1] School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, Sichuan, China
`donghao@std.uestc.edu.cn`, `liuqiang@uestc.edu.cn`
[2] Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China

Abstract. Wireless sensor networks is an important component of Internet of everything, and can be deployed in many applications, such as search and rescue, border patrols, environmental monitoring, and combat scenarios. In these applications, target tracking is a crucial difficulty. Compared with the traditional static wireless sensor networks (WSN), the mobile sensor networks (MSN) has the advantages of strong robustness, flexibility, energy saving, etc., and has been widely deployed. For target tracking applications in mobile wireless sensor networks, this paper investigates an extended Kalman filter (EKF) algorithm in a dynamic scenario, and proposes a low-power, high-accuracy sensor scheduling strategy based on the extend kalman filter algorithm. The properly sensors selection and path planning at each sample time of target tracking can make the EKF algorithm in dynamic scenarios complete target trajectory prediction more efficiently. Simulation results show that the proposed sensor scheduling strategies have better performances in power consumption and tracking accuracy, compared with the static network extend Kalman filter algorithm.

Keywords: Mobile sensor networks · Target tracking · Extend Kalman filter · Sensor schedule

1 Introduction

Wireless sensor networks (WSN) have been widely used in the fields of regional monitoring, environmental protection, rescue and urban construction [1]. In recent years, people have begun to study mobile sensor networks (MSN). Compared with the static sensor network, the MSN is composed of static sensors and mobile sensors, which cannot achieve full coverage in the monitoring area. On the other hand, the mobility of the network simplifies many complex issues, such as area coverage, network connectivity, and energy consumption [1]. Target tracking is an important application of WSN. Compared with static WSN, MSN needs to consider the heterogeneous networks, the sensors' movement paths and some other questions. Taking into account low energy consumption

Y. Chen et al. (Eds.): BICT 2020, LNICST 329, pp. 275–285, 2020.
https://doi.org/10.1007/978-3-030-57115-3_24

and high accuracy of target tracking, the design and optimization of the joint scheduling algorithm of mobile nodes is extremely important.

Two key issues are the selection of mobile nodes and the path planning of mobile nodes. Since the MSN is not full coverage, the selection of mobile nodes is not only a simple wake-up and sleep mechanism, but also tracking path planning. MSN can use traditional target tracking algorithms, such as: using time of arrival (TOA) for target location and node selection for tracking [2]; using hierarchical Markov decision process for prediction [3], they model the target's motion trajectory as a Markov process, and use the properties of Markov chain to make trajectory prediction; using interactive multi-model filtering algorithm (IMM) for prediction [4], etc. In MSN, the EKF algorithm is the most popular prediction algorithm for target state. The EKF algorithm is mainly used by MSN to handle non-linear measurement models, while the target model is usually linear [5]. In [6], Ren et al. used a distance-to-target measurement model, and in [7] Wu et al. used a range-bearing sensor model. Martínez and Bullo [8] use a general target model in their derivation and, in simulation, use an 8-shaped movement for the target. Masazade E et al. [9] proposed an improvement EKF algorithm, they give a new real meaning to the Kalman gain matrix: the *i*-th column of the Kalman gain represents the motion vector of the *i*-th sensor. They modify the original objective function of the EKF algorithm, introduce a penalty term, and then solve the new objective function by using the ADMM algorithm to make the Kalman gain appear as many zero columns as possible, which means that keep as many sensors sleep as possible. Generally, all these papers above use a centralized implementation of the EKF. Sensor scheduling strategy is usually a distributed node structure. La H M and Sheng W [10] proposed a cluster node group control method. W. Yuan and N. Ganganath [11] proposed an improved semi-group control method. This type of sensor scheduling strategy will form a center by a cluster node and the cluster node controls the surrounding nodes for target positioning [12, 13]. The whole process is completed by only moving the cluster nodes, while other nodes keep at a fixed position. Another type of sensor scheduling algorithm uses heuristic algorithms, such as J Liang and X YuH [1] proposed a multi-node joint movement algorithm based on neural network and a fuzzy genetic tree algorithm. The two algorithms find the current time's optimal node selection and path planning scheme by training the historical samples.

Compared with the heuristic algorithm used in some of the above articles, the EKF algorithm has less computation cost and better real-time performance. However, EKF algorithm is usually used in static WSN. In this paper, we apply the EKF algorithm to MSN, and design two strategies to schedule sensors for target tracking, which makes the performance of EKF algorithm better.

The remaining structure of this article is arranged as follows: Sect. 2 introduces the EKF prediction algorithm. Section 3: Introduce the target tracking system. This section firstly introduces the deployment of the system, and then theoretically models the system used in this article. Finally, based on the EKF algorithm, two sensor scheduling strategy are proposed for target tracking. Section 4: Simulation analysis. In this section, three kind of trajectory simulation analysis will be performed on the scheduling strategies proposed in this paper. Section 5: Conclusion. Summarize the contents of this article.

2 System Model

2.1 Extend Kalman Filter Structure

Based on the assumption of linear Gaussian distribution, Kalman filter uses the minimum mean square error as the object function to obtain the target's state estimation. It performs well for systems where the process and measurement are linear and the error follows the Gaussian distribution. However, in practice, many systems are non-linear, which means the state equation is non-linear or the measurement equation is non-linear. In this case, Kalman filter can't perform well, and we need to linearly approximate this non-linear system into a linear system. The EKF algorithm solves this problem by linearizing the non-linear process and ignoring or approximating the higher-order terms.

The EKF this article used is a non-linear measurement model but a linear the target model. This model can be written as:

$$x_k = Fx_{k-1} + w_k \tag{1}$$

$$z_k = h(x_k) + v_k \tag{2}$$

Where F is the transition matrix, and $h(x_k)$ is a non-linear measurement function. w_k is the system noise, and satisfies the Gaussian distribution of zero mean and covariance matrix Q_k. v_k is the measurement noise and obeys the Gaussian distribution which satisfies the zero mean and the covariance matrix is R_k.

The predict equations are:

$$\hat{x}_{k|k-1} = F\hat{x}_{k-1|k-1} \tag{3}$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q_k \tag{4}$$

$$S_k = H_k P_{k|k-1} H_k^T + R_k \tag{5}$$

The update equations are:

$$\bar{y}_k = z_k - h\left(\hat{x}_{k|k-1}\right) \tag{6}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \bar{y}_k \tag{7}$$

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \tag{8}$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \tag{9}$$

Where,

$$H_k = \frac{\partial h}{\partial x}\bigg|_{\hat{x}_{k|k-1}} \tag{10}$$

2.2 System Deployment

In this paper, the monitoring area we used for target tracking is a rectangular area of size b × b (m^2), and assuming a total of n mobile sensor nodes in the area, their initial deployment positions are uniform, as shown in Fig. 1. In fact, this initial deployment assumption is arbitrary, and it has no effect on the target tracking algorithm proposed in this paper. More generally, this article considers heterogeneous networks, that is, sensors in the area are of different types, such as: optical sensors, acoustic sensors, radar sensors, infrared sensors, and so on. Suppose there are n sensors $\{S_1, S_2, \ldots, S_n\}$, and their four main parameters are considered in this network: 1. Remaining power $\{b_1, b_2, \ldots, b_n\}$. 2. Sensing range $\{r_1, r_2, \ldots, r_n\}$. 3. Moving rate $\{v_1, v_2, \ldots, v_n\}$. 4. The current position of the sensor (x, y). We assume that a moving target appears in the monitoring area with a certain trajectory, and the first value of the trajectory is used as the known initial state value. The target will be detected its position information (x_t, y_t) by some sensors at each discrete time point t during the movement. These sensor nodes will send their measurement to the data center, and then the data center uses EKF algorithm to estimate the target trajectory, which means to predict the position and speed of the target at the next moment. At the same time, the data center will select three optimal nodes according to the current sensor node distribution, plan the movement path of each node, and then perform target tracking.

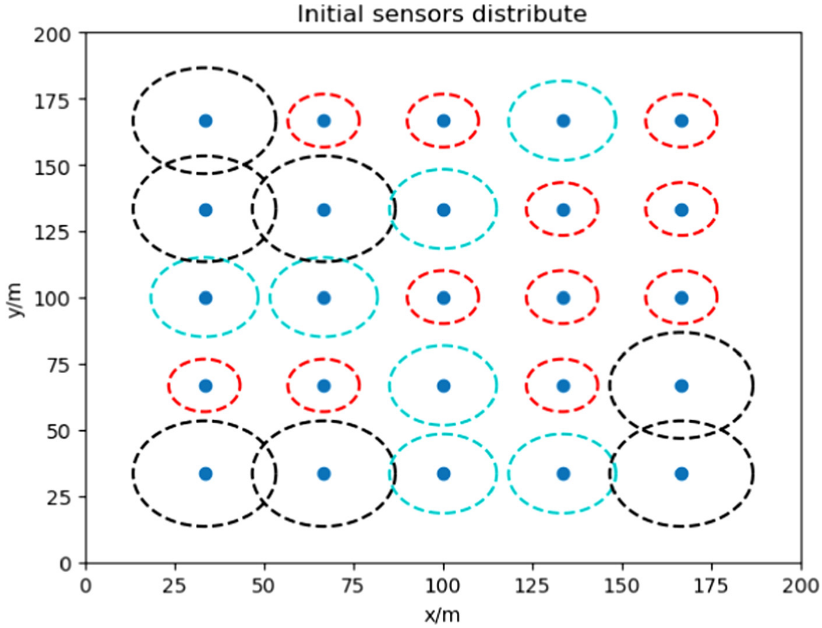

Fig. 1. Initial sensors distribute. The dotted line circles represent sensor's sense range.

At time t, the motion state of the target can be represented by a 4 × 1 column vector:

$$\mathrm{x}_t = \left[x_t, y_t, v_x, v_y\right]^T \tag{11}$$

As a result, F is a 4 × 4 state transition matrix, and Q is a 4 × 4 matrix. The set values of F and Q are:

$$F = \begin{bmatrix} 1 & 0 & \mathrm{dt} & 0 \\ 0 & 1 & 0 & \mathrm{dt} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, Q = \gamma \begin{bmatrix} \frac{\mathrm{dt}^3}{3} & 0 & \frac{\mathrm{dt}^2}{2} & 0 \\ 0 & \frac{\mathrm{dt}^3}{3} & 0 & \frac{\mathrm{dt}^2}{2} \\ \frac{\mathrm{dt}^2}{2} & 0 & 0 & 0 \\ 0 & \frac{\mathrm{dt}^2}{2} & 0 & 0 \end{bmatrix}$$

Where dt is the sampling interval of sensor node, and γ is the system noise parameter.

At time t, it is assumed that the measurement vector after fusion of all sensor data is $\mathrm{z}_t = \left[z_{t,1}, z_{t,2}, \ldots, z_{t,N}\right]^T$, and z_t satisfies (2), where $R = \sigma^2 E$ (E representation an identity matrix of size $n \times n$). For each component $z_{t,i}$, which is the measurement value corresponding to the i-th sensor, can be expressed as:

$$z_{t,i} = \sqrt{\frac{P_0}{1 + \left(d_{t,i}\right)^n}} + v_{t,i} \tag{12}$$

Where P_0 represents the signal power transmitted by the target, and n is the attenuation index of the signal power. Generally, $n = 2$. $d_{t,i}$ is the distance between the i-th sensor and the target. Assuming the position of the i-th sensor is (x_i, y_i), then:

$$d_{t,i} = \sqrt{(x_t - x_i)^2 + (y_t - y_i)^2} \tag{13}$$

Now, the basic model in the EKF system has been completed, and the target trajectory prediction can be achieved.

3 Sensor Scheduling

In order to achieve target trajectory prediction, another issue we need to consider is how to select the appropriate sensor nodes and how to make these sensor nodes move to complete the target location. [9] directly uses node scheduling into the original EKF algorithm process and changes the original algorithm structure. In fact, such node selection and path planning are not reliable. The EKF algorithm is a relatively sensitive algorithm. When the nonlinearity of the system increases, the EKF algorithm is easy to diverge. This approach will cause the EKF algorithm not obtain the minimum value of the trajectory prediction, but to obtain the minimum value of the objective function constructed in [9], which may also cause the EKF algorithm to diverge. Therefore, we need to perform mobile node selection and path planning independently from the EKF algorithm.

3.1 Selecting Strategy

For sensor node S_i, its state can be represent by $\{(x_i, y_i), v_i, b_i, r_i\}$. We assume that the node's energy consumption is proportional to the displacement of the node. We score all sensor nodes based on **Strategy 1** at each decision moment, and then select three sensor nodes with the highest scores for target tracking. If the current energy of a sensor is no longer sufficient for the next mobile tracking task, we consider it to be exhausted and set the score to the lowest. The scoring rules are:

Strategy 1 Sensor selecting

```
Input: energy scale factor $k$; select number $num$; target's predict position $(\hat{x}_{t|t-1}, \hat{y}_{t|t-1})$; sensors' state $\{(x_i, y_i), v_i, b_i, r_i\}$, $i \in \{1,2,\ldots,n\}$.
Output: selected sensors.
for all $i \in \{1,2,\ldots,n\}$ do
    if $b_i - k * v_i * dt \leq 0$ then
        $b_i = 0$
        $score_i = -1$
    else
        $dist_i = \sqrt{(\hat{x}_{t|t-1} - x_i)^2 + (\hat{y}_{t|t-1} - y_i)^2}$
        $score_i = (r_i + v_i * dt - dist_i) / dist_i$
    end if
end for
Sort all scores from high to low
Select the first $num$ sensors
```

The absolute maximum detectable distance r_{max} of a sensor is defined as:

$$r_{max} = r_i + v_i * dt \tag{14}$$

The idea of scoring is to judge whether the sensor can capture the target at next moment by the difference between the absolute maximum detectable distance of the sensor and the target distance at the current moment, and the sensor that can be captured is superior. In order to evaluate the sensors with different absolute maximum detectable distances, we need to normalize them. The normalized minimum score is -1.

3.2 Moving Strategy

After nodes are selected according to the strategy **Strategy 1**, the **Strategy 2** will be introduced next. **Strategy 2** makes the selected nodes move properly so that the target location can be completed and the location is used as a feedback to EKF algorithm.

Strategy 2 Sensor moving

Input: energy scale factor k; select number *num*; target's current predict position $(\hat{x}_{t|t-1}, \hat{y}_{t|t-1})$; target's last update position $(\hat{x}_{t-1|t-1}, \hat{y}_{t-1|t-1})$; sensors' state $\{(x_i, y_i), v_i, b_i, r_i\}, i \in \{1,2,\ldots,n\}$.

Output: the selected sensors after moving.

```
for all $i \in \{1,2,\ldots,n\}$ do
    $dist_i = \sqrt{(\hat{x}_{t|t-1} - x_i)^2 + (\hat{y}_{t|t-1} - y_i)^2}$
    if $r_i > 2 * dist_i$ then
        don't move and detect the target at current position
    else if $r_i > \frac{3}{2} dist_i$ then
        $\alpha = (\hat{x}_{t|t-1}, \hat{y}_{t|t-1}) - (\hat{x}_{t-1|t-1}, \hat{y}_{t-1|t-1})$
        $\alpha \leftarrow \alpha / \|\alpha\|$
        $(x_i, y_i) \leftarrow (x_i, y_i) + v_i * dt * \alpha$
        $b_i \leftarrow b_i - k * v_i * dt$
    else
        $\alpha = (\hat{x}_{t|t-1}, \hat{y}_{t|t-1}) - (x_i, y_i)$
        $\alpha \leftarrow \alpha / \|\alpha\|$
        $(x_i, y_i) \leftarrow (x_i, y_i) + v_i * dt * \alpha$
        $b_i \leftarrow b_i - k * v_i * dt$
    end if
end for
```

The idea of the moving strategy is that if the target appears within the $\frac{1}{2}r$ range of the sensor, the sensor can remain stationary to achieve energy saving. If the target is within $\frac{1}{2}r \sim \frac{2}{3}r$, the sensor's moving direction is the moving direction of the target's current time and the previous time. By doing so, the detection coverage area of the selected sensor nodes can be kept largely. If the target exceeds the $\frac{2}{3}r$ range, the sensor node will be in danger of not being able to detect the target at next time, so must move towards the target at the next moment.

Then, we can construct the entire target tracking system, as shown in the Fig. 2.

4 Simulation Analysis

This section we will show the utility of our strategies by the simulation results. Considering a real scenario, we need to deploy an MSN to monitor an area. Any people entering this area is considered as a target, and each sensor is regarded as a mini-car. The task of the MSN is to track the intruding people in real time. We use the area of 200 m × 200 m and $n = 25$ sensors. We initial these parameters in EKF system as following: sample time $dt = 0.2$ s, energy scale factor $k = 0.1$, system noise $\gamma = 0.01$, measurement

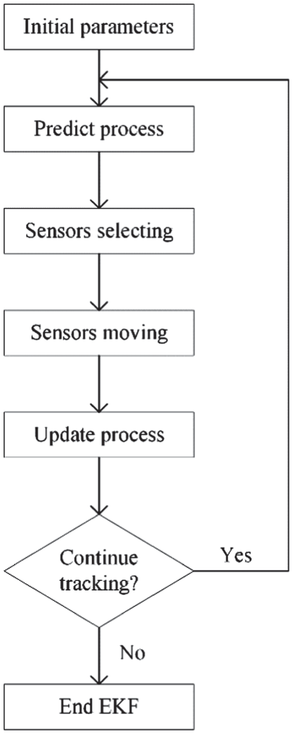

Fig. 2. The target tracking system based on EKF algorithm.

noise $\sigma^2 = 0.01$, target transmit power $P_0 = 1000$. We assume that for all the sensors, as a mini-car, their velocity randomly takes a value from $\{2, 3, 4\}(m/s)$, and their detect range randomly takes a value from $\{10, 15, 20\}(m)$, and their initial battery randomly takes a value from [0, 100]. In general, We select three sensors to complete the tracking task. We will obtain the simulation results by comparing the tracking error between our methods with EKF in static WSN which is fully covered by sensors. We use the root mean squared error (RMSE) at time t to represent current tracking error as:

$$\mathrm{RMSE(t)} = \sqrt{\left(x(t) - \hat{x}(t)\right)^2 + \left(y(t) - \hat{y}(t)\right)^2} \tag{15}$$

Totally we use three kind of target trajectory to simulate. Figure 3 shows the power trajectory results. Figure 4 shows the square trajectory results. Figure 5 shows the circle trajectory results. From the three kind of target trajectory's simulation results, we can see that although static EKF system is fully covered, dynamic EKF system performs much better. These results reveal that our strategies make the EKF system obtain a higher accuracy, while it only schedule three sensors to move to track the target which is more energy-saving compared to static WSN.

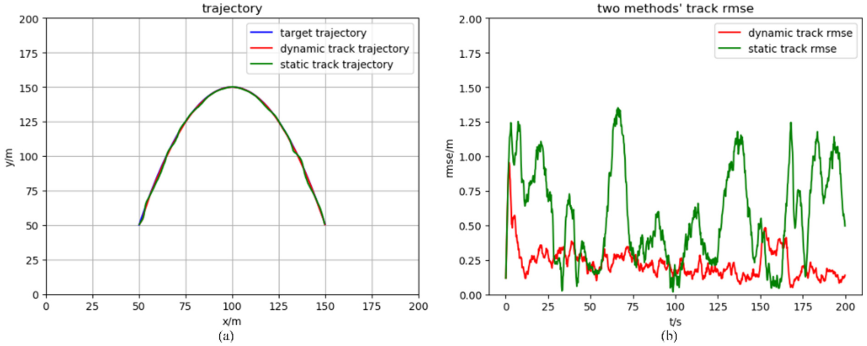

Fig. 3. (a) Target's power trajectory and the tracking trajectory. (b) Power trajectory's track rmse of dynamic EKF system and static EKF system.

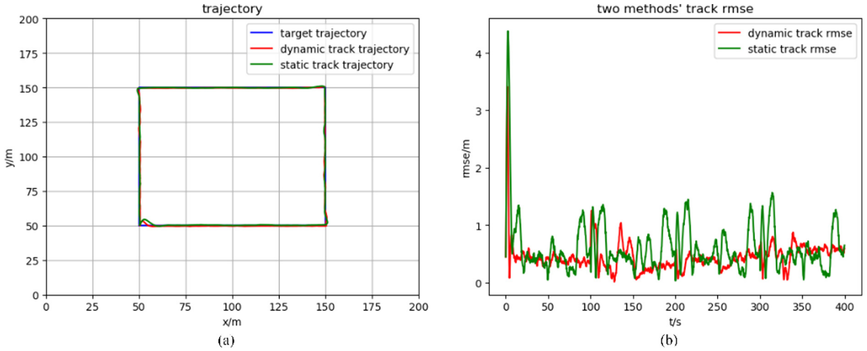

Fig. 4. (a) Target's square trajectory and the tracking trajectory. (b) Square trajectory's track rmse of dynamic EKF system and static EKF system.

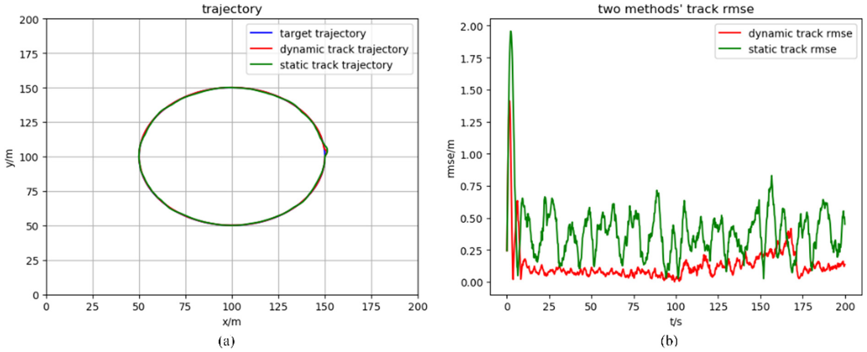

Fig. 5. (a) Target's circle trajectory and the tracking trajectory. (b) Circle trajectory's track rmse of dynamic EKF system and static EKF system.

5 Conclusion

In this article, we have solved the problem of target tracking in MSN. Different from static WSN, we need to select the optimal sensors and schedule their moving path to track the target so that system can obtain a high accuracy. Then we proposed two strategies aiming for the above two problems. The first strategy uses the absolute maximum detectable distance of a sensor and normalize it for different sensors' score, so we can judge at current time which sensor is more qualified for the next time's tracking. The second strategy decides the selected sensors' moving path by a sensor' detect range and the distance between target and this sensor. Simulation results show that using the two strategies significantly improves the EKF algorithm's performance in MSN compared with that in static WSN.

Acknowledgments. This work is supported in part by the Zhongshan City Team Project under Grant No. 180809162197874 and National Natural Science Foundation of China under Grants No. 61731006.

References

1. Liang, J., Yu, X., Li, H.: Collaborative energy-efficient moving in internet of things: genetic fuzzy tree vs. neural networks. IEEE Internet Things J. **6**(4), 6070–6078 (2018)
2. Xu, E., Ding, Z., Dasgupta, S.: Target tracking and mobile sensor navigation in wireless sensor networks. IEEE Trans. Mob. Comput. **99**(1), 177–186 (2011)
3. Yeow, W.L., Tham, C.K., Wong, W.C.: A novel target movement model and energy efficient target tracking in sensor networks. In: IEEE Vehicular Technology Conference. IEEE (2005)
4. Liu, Y., Xu, B., Feng, L.: Energy-balanced multiple-sensor collaborative scheduling for maneuvering target tracking in wireless sensor networks. Control Theor. Technol. **9**(1), 58–65 (2011). https://doi.org/10.1007/s11768-011-0249-2
5. Anders, A., Lars, I.: Survey: mobile sensor networks for target searching and tracking. Cyber-Phys. Syst. **4**(2), 57–98 (2018)
6. Zhang, G., Lian, F., Han, C., et al.: Two novel sensor control schemes for multi-target tracking via delta generalised labelled multi-Bernoulli filtering. IET Sig. Process. **12**(9), 1131–1139 (2018)
7. Hoang, H.G.: Control of a mobile sensor for multi-target tracking using multi-target/object multi-bernoulli filter. In: 2012 International Conference on Control, Automation and Information Sciences (ICCAIS), Ho Chi Minh City, pp. 7–12 (2012)
8. Gostar, A.K., Hoseinnezhad, R., Babhadiashar, A.: Sensor control for multi-object tracking using labeled multi-bernoulli filter. In: International Conference on Information Fusion. IEEE (2014)
9. Masazade, E., Fardad, M., Varshney, P.K.: Sparsity-promoting extended Kalman filtering for target tracking in wireless sensor networks. IEEE Sig. Process. Lett. **19**(12), 845–848 (2012)
10. La, H.M., Sheng, W.: Flocking control of a mobile sensor network to track and observe a moving target. In: IEEE International Conference on Robotics & Automation. IEEE Press (2009)
11. Yuan, W., Ganganath, N., Cheng, C.T., et al.: Semi-flocking-controlled mobile sensor networks for dynamic area coverage and multiple target tracking. IEEE Sens. J. **18**(21), 8883–8892 (2018)

12. Wu, B., Feng, Y.P., Zheng, H.Y., et al.: Dynamic cluster members scheduling for target tracking in sensor networks. IEEE Sens. J. **16**(19), 1 (2016)
13. Yuan, W., Ganganath, N., Cheng, C., et al.: Energy-efficient semi-flocking control of mobile sensor networks on rough terrains. IEEE Trans. Circuits Syst. II Exp. Briefs **66**(4), 622–626 (2019)