Pers Ubiquit Comput (2014) 18:987–996
DOI 10.1007/s00779-013-0713-8

ORIGINAL ARTICLE

Square-root unscented Kalman filtering-based localization and tracking in the Internet of Things

Junqi Guo · Hongyang Zhang · Yunchuan Sun · Rongfang Bie

Received: 10 November 2012 / Accepted: 10 April 2013 / Published online: 30 August 2013

Abstract The Internet of Things (IoT), which is usually established over architectures of wireless sensor networks, provides an actual platform for various applications of personal and ubiquitous computing. Recently, moving target localization and tracking in an IoT environment have been paid more and more attention. This paper proposes a square-root unscented Kalman filtering (SR-UKF)-based algorithm to discover real-time location of a moving target in an IoT environment where there exist quantities of sensors. The data generated from wireless sensor nodes of the IoT make contributions to localization and tracking of the moving target. First, a least-square (LS) criterion-based mathematical model is proposed for localization initialization in an IoT scenario. Next, we employ an SR-UKF idea for the further localization and tracking. By using the data coming from sensor nodes near the target, real-time location of the moving target can be estimated by implementation of SR-UKF in an iterative fashion so as to achieve target status tracking. Simulation results show that the proposed algorithm achieves good performance in estimation of both position and velocity of the target with either uniform linear motion or variable-speed curve motion. Compared with some existing conventional extended Kalman filtering (EKF) or UKF-based methods, the proposed algorithm shows lower location/velocity estimation error under the same computational complexity, which demonstrates its potential significance in ubiquitous computing applications for an IoT environment.

Keywords Internet of Things (IoT) · Wireless sensor network (WSN) · Localization · Tracking · Square-root unscented Kalman filtering (SR-UKF)

1 Introduction

The Internet of Things (IoT) [1–8], which has gained widespread acceptance recently, represents a technological revolution in the field of future networks, mobile computing, and wireless communications. Future IoT may establish over the architecture of wireless sensor network (WSN) [9–14], which consists of hundreds of sensor nodes performing distributed sensing and collaborative computing. By deploying sensors on or around physical objects, the IoT seamlessly integrates a world of networked smart objects, makes their information be shared on a global scale, and provides an ability of intelligent computing and information processing, such as reporting status, position, and surrounding condition of each sensor node. There are various application domains which have already benefited from the IoT, especially in areas of industrial production, logistics, management, and civilian life. As a new type of Internet application, the IoT brings a new ubiquitous computing era, which will extremely change people's life [15].

Localization and tracking of moving targets, which is considered as a conventional and classical problem, have been already studied for several years in many fields, including wireless cellular network-based mobile user localization [16], simultaneous localization and mapping (SLAM) systems [17], global positioning systems (GPS) [18], Ad Hoc network-based localization [19] and the

J. Guo · H. Zhang · R. Bie
College of Information Science and Technology, Beijing Normal University, Beijing, People's Republic of China
e-mail: guojunqi@bnu.edu.cn

Y. Sun (✉)
Business School, Beijing Normal University,
100875 Beijing, People's Republic of China
e-mail: yunch@bnu.edu.cn

ultrasonic-based object localization [20], etc. However, solutions in the above fields may not directly suit an IoT scenario where large quantities of sensor nodes which perform distributed sensing and collaborative information processing tasks are interconnected together over a wireless channel [21].

With the rapid development of WSN, sensor-based localization and tracking have attracted many research efforts in recent years [21–25], most of which transform a localization and tracking problem into target state estimation by utilizing Kalman filtering [26–28]—a prevailing method for numerous nonlinear estimation, advanced signal processing and machine learning applications. As a candidate of improved Kalman filtering schemes, the extended Kalman filtering (EKF) is employed in [29], in which the EKF has successfully combined different sensor data to achieve suboptimal estimation of target states. However, the EKF has two inherent flaws [30]: on one hand, it uses only the first-order linearization to update the mean and covariance of state variables, while ignoring higher order terms derived from Taylor-series expansion, which results in only the first-order accuracy; on the other hand, the EKF needs to calculate the Jacobian matrix, which has proven to be a complicated task. Some subsequent studies [30–35] employ the unscented Kalman filtering (UKF) as an alternative to the EKF for solving target localization and tracking problem. By utilizing a deterministic "sampling" approach to calculate mean and covariance terms with higher accuracy, the UKF outperforms the EKF in terms of estimation and prediction with no burden of calculation of Jacobian matrix [36]. Although the UKF performance is far more promising for target localization and tracking, there still exist some numerical problems such as loss of positive definite of the state covariance matrix, which derives from rounding errors and may lead to filtering failure [37]. To solve this problem, square-root UKF (SR-UKF) is introduced for target localization and tracking [37, 38], in which the square root of covariance matrix is used in propagating process so as to ensure the nonnegative definite of the state covariance matrix.

Based on the studies above, we propose an SR-UKF-based localization and tracking algorithm to discover real-time location of a moving target in an IoT environment where there exist quantities of sensors. First, a least-square (LS) criterion-based mathematical model is proposed for localization initialization in an IoT scenario. Next, we employ an SR-UKF idea for the further localization and tracking. By using the data coming from neighboring sensor nodes of the target, real-time location of the moving target can be estimated by implementation of SR-UKF in an iterative fashion so as to achieve target status tracking. Simulation results show that the proposed algorithm achieves good performance in estimation of both position and velocity of the target with either uniform linear motion or variable-speed curve motion. Compared with some existing conventional EKF- or UKF-based methods, the proposed algorithm shows lower localization error under the same computational complexity, which demonstrates its potential significance in ubiquitous computing applications for an IoT environment.

The rest of this paper is organized as follows: Section II presents a localization initialization model in the IoT. The SR-UKF-based localization and tracking algorithm are proposed in Section III. Performance evaluations of our proposed algorithm are presented and discussed in Section IV. Finally, Section V presents our conclusions.

2 Localization initialization model in the IoT

In an IoT scenario, hundreds or thousands of networked sensor nodes are located at the sensing layer of IoT, performing distributed sensing and collaborative data processing tasks. Each sensor node has an ability to sense and report its own state and surrounding conditions. In this work, we uniformly deploy N "location-aware" sensor nodes in an IoT scenario, in which each of them has a priori knowledge of its own location.

Assume that the sensing ability of each sensor node is limited in a round region with a radius R. All sensors periodically detect their own surroundings, and the detection period is usually set to be much smaller than the velocity of moving target. Note that, there are only two possible detection results for each sensor node at a certain moment: 1 or 0. Here, "1" represents that the target has been detected at this moment, while "0" denotes target undetected. When detection result of a sensor node at one moment is different from its result at the adjacent moment before, this sensor node can be considered as an available sensor node at this moment for localization initialization, and the distance between this sensor node and moving target approximates to R at this moment. As is shown in Fig. 1, the sensor node is available for localization initialization at t_2 and t_5.

If there are three available sensor nodes for localization initialization, we can determine initial location area of moving target by utilizing a localization initialization model, as shown in Fig. 2. In this figure, the shadow area represents initial location region of the target. However, we still expect to obtain its precise location information. Here, we employ a least-square (LS) mathematical criterion [39] to calculate initial location point of the moving target.

We assume that there are n ($n \geq 3$) available sensor nodes for localization initialization, as is shown in Fig. 3. Since each sensor node has a priori knowledge of its own

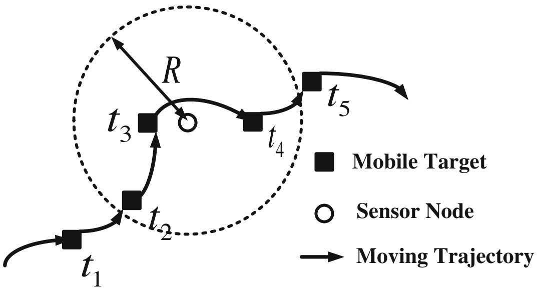

Fig. 1 Available sensor node (at t_2 and t_5)

location, we have already known location information of these available sensor nodes, which can be represented as (X_1, Y_1), (X_2, Y_2), …, (X_n, Y_n), while the unknown initial location information of moving target is denoted as (x, y). Since the distance between the moving target and an available sensor node approximates to its sensing radius R as shown in Fig. 1, we can establish the following equations:

$$\begin{bmatrix} (X_1 - x)^2 + (Y_1 - y)^2 \\ (X_2 - x)^2 + (Y_2 - y)^2 \\ \vdots \\ (X_n - x)^2 + (Y_n - y)^2 \end{bmatrix} = \begin{bmatrix} R_1^2 \\ R_2^2 \\ \vdots \\ R_n^2 \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \tag{1}$$

where R_i denotes the maximum sensing radius of the i-th ($i = 1, 2, \ldots, n$) available sensor node; w_i represents the i-th measurement noise. Subtract the last row of Eq. (1) from other rows, then we get the following equations represented in matrix form:

$$\mathbf{U} \times \mathbf{S} = \mathbf{V} \tag{2}$$

where **U**, **S** and **V** are given by:

$$\mathbf{U} = -2 \times \begin{bmatrix} (X_1 - x_n) & (Y_1 - y_n) \\ (X_2 - x_n) & (Y_2 - y_n) \\ \vdots & \vdots \\ (X_{n-1} - x_n) & (Y_{n-1} - y_n) \end{bmatrix} \tag{3}$$

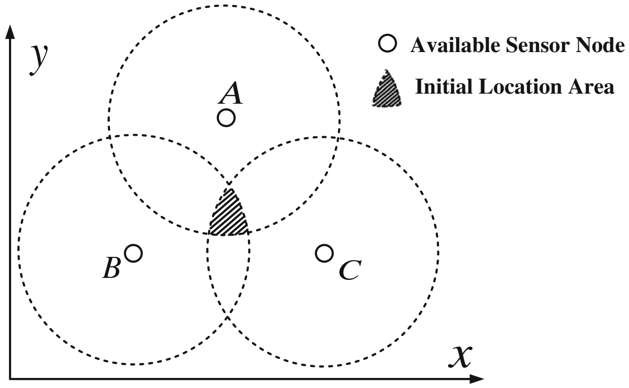

Fig. 2 Localization initialization model in an IoT scenario

$$\mathbf{S} = \begin{bmatrix} x \\ y \end{bmatrix} \tag{4}$$

$$\mathbf{V} = \begin{bmatrix} \left(R_1^2 - R_n^2\right) + \left(X_n^2 - X_1^2\right) + \left(Y_n^2 - Y_1^2\right) + (w_1 - w_n) \\ \left(R_2^2 - R_n^2\right) + \left(X_n^2 - X_2^2\right) + \left(Y_n^2 - Y_2^2\right) + (w_2 - w_n) \\ \vdots \\ \left(R_{n-1}^2 - R_n^2\right) + \left(X_n^2 - X_{n-1}^2\right) + \left(Y_n^2 - Y_{n-1}^2\right) + (w_{n-1} - w_n) \end{bmatrix} \tag{5}$$

Since Eq. (2) contains measurement noise components which may affect accuracy of its solution, here we use the LS estimator to obtain the estimate of **S**. The optimal solution (or estimate) of Eq. (2) based on the LS criterion is given by:

$$\mathbf{S}_{LS} = \left(\mathbf{U}^T\mathbf{U}\right)^{-1}\mathbf{U}^T\mathbf{V} \tag{6}$$

From Eqs. (1–6), we obtain initial location information of the moving target, which is given by $\mathbf{S}_{LS}$. After localization initialization discussed above, we should implement further localization and tracking. This gives rise to an SR-UKF-based algorithm, as discussed in the next section.

3 SR-UKF-based localization and tracking algorithm in the IoT

Kalman filters (KF) and their variants are promising solutions for target localization and tracking. However, the standard linear KF methods are usually not suitable for target tracking with a nonlinear observation equation. The extended KF (EKF) [29] methods, which apply the standard linear KF methodology to linearization of a nonlinear system [37], show improved filtering performance. However, the EKF implements only the first-order approximation to a nonlinear system and ignores higher order terms derived from Taylor-series expansion, which degrades its performance when it is used in a nonlinear system. Although the unscented KF (UKF) [30–35] is an effective alternative to the EKF for a nonlinear system, it still suffers some numerical problems such as loss of positive definite of the state covariance matrix which derives from rounding errors and may lead to filtering failure [37]. Since the square-root UKF (SR-UKF) [37, 38] outperforms standard KF, EKF and UKF in nonlinear

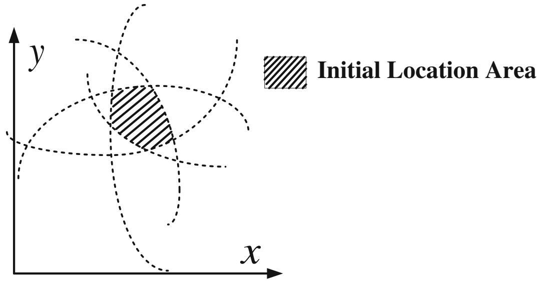

Fig. 3 Extended localization initialization model in an IoT scenario

filtering, here we employ the SR-UKF to design localization and tracking algorithm in an IoT environment.

3.1 SR-UKF Principle

For a nonlinear dynamic system, we define the following dynamic state and observation equations for SR-UKF implementation, which are given by:

$$\begin{cases} \mathbf{a}_{k+1} = \mathbf{F}(\mathbf{a}_k) + \mathbf{v}_k, & \text{State Equation} \\ \mathbf{b}_k = \mathbf{H}(\mathbf{a}_k) + \mathbf{w}_k, & \text{Observation Equation} \end{cases} \tag{7}$$

where $\mathbf{F}(\cdot)$ is defined as the state function, which shows the recursive relationship of parameter $\mathbf{a}$ at the current moment (k) and the next moment ($k + 1$). $\mathbf{H}(\cdot)$ denotes the observation function, which provides the relationship between $\mathbf{a}_k$ and $\mathbf{b}_k$. $\mathbf{F}(\cdot)$ may be linear or nonlinear depending on application scenarios, whereas $\mathbf{H}(\cdot)$ is usually highly nonlinear in SR-UKF. Here, $\mathbf{a}_k$ and $\mathbf{b}_k$ are the state and observation vectors at the k-th moment (also at the k-th iteration), respectively. The value of $\mathbf{b}_k$ can be obtained by observation or measurement, while the value of $\mathbf{a}_k$ is unknown. $\mathbf{v}_k$ and $\mathbf{w}_k$ are state and observation noise vectors, respectively. Usually, the aim of SR-UKF is to estimate the value of $\mathbf{a}_k$ at each k-the moment ($k = 1, 2, 3, 4,\ldots$) under the given value of $\mathbf{b}_k$.

The SR-UKF principle for state estimation (or prediction) of $\mathbf{a}_k$ ($k = 1, 2, 3, 4,\ldots$) can be summarized as follows:

1. Initialization

$$\hat{\mathbf{a}}_0 = E\{\mathbf{a}_0\}, \quad \mathbf{S}_0 = chol\left\{E\left[(\mathbf{a}_0 - \hat{\mathbf{a}}_0)(\mathbf{a}_0 - \hat{\mathbf{a}}_0)^T\right]\right\} \tag{8}$$

where $E\{\cdot\}$ denotes the expectation operator; $chol\{\mathbf{A}\}$ is the original Cholesky factor of $\mathbf{A}$ [37].

2. The k-th iterative estimation, $k \in \{1,2,\ldots,\infty\}$

(a) Sigma point calculation:

$$\boldsymbol{\chi}_{k-1} = \left[\hat{\mathbf{a}}_{k-1}, \hat{\mathbf{a}}_{k-1} + \sqrt{L+\lambda}\mathbf{S}_{k-1}, \hat{\mathbf{a}}_{k-1} - \sqrt{L+\lambda}\mathbf{S}_{k-1}\right] \tag{9}$$

where L is dimension of the state vector $\mathbf{a}_k$; $\lambda = L(\alpha^2 - 1)$ is a compound scaling parameter, in which α is a primary scaling factor and is usually set between 10^{-4} and 1.

(b) Time update:

$$\boldsymbol{\chi}_{k|k-1} = \mathbf{F}(\boldsymbol{\chi}_{k-1}) \tag{10}$$

$$\hat{\mathbf{a}}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \boldsymbol{\chi}_{i,k|k-1} \tag{11}$$

$$\mathbf{S}_{\mathbf{a}_k} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\boldsymbol{\chi}_{1,k|k-1} - \hat{\mathbf{a}}_k^-\right), \ldots, \sqrt{W_{2L}^{(c)}}\left(\boldsymbol{\chi}_{2L,k|k-1} - \hat{\mathbf{a}}_k^-\right), \sqrt{\mathbf{Q}^{\mathbf{v}}}\right]\right\} \tag{12}$$

$$\mathbf{S}_{\mathbf{a}_k} = cholupdate\left\{\mathbf{S}_{\mathbf{a}_k}, \boldsymbol{\chi}_{0,k} - \hat{\mathbf{a}}_k^-, W_0^{(c)}\right\} \tag{13}$$

where $\mathbf{Q}^{\mathbf{v}}$ denotes covariance matrix of the state noise vector $\mathbf{v}_k$; $\{W_i\}$ is a set of scalar weights, which is given by:

$$W_0^{(m)} = \lambda/(L+\lambda); W_0^{(c)} = \lambda/(L+\lambda) + \left(1 - \alpha^2 + \beta\right) \tag{14}$$

$$W_i^{(m)} = W_i^{(c)} = 1/[2(L+\lambda)]; \quad i = 1,\ldots,2L \tag{15}$$

where β is a secondary scaling factor. Generally, $\beta = 2$ is optimal for Gaussian distribution; $qr\{\mathbf{A}\}$ represents the upper triangular part of matrix $\mathbf{R}$ ($\mathbf{R}$ results from the QR decomposition of matrix $\mathbf{A}$); $cholupdate\{\mathbf{A},\mathbf{B},c\}$ is available in Matlab as "*cholupdate*". If $\mathbf{A} = chol\{\mathbf{D}\}$ is the original Cholesky factor of $\mathbf{D}$, then $cholupdate\{\mathbf{A},\mathbf{B},c\}$ returns the Cholesky factor of $\mathbf{D}$ + c$\mathbf{BB}^{\mathrm{T}}$.

(c) Measurement update:

$$\gamma_{k|k-1} = \mathbf{H}\left(\boldsymbol{\chi}_{k|k-1}\right) \tag{16}$$

$$\hat{\mathbf{b}}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \gamma_{i,k|k-1} \tag{17}$$

$$\mathbf{S}_{\mathbf{b}_k} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\gamma_{1,k|k-1} - \hat{\mathbf{b}}_k^-\right), \ldots, \sqrt{W_{2L}^{(c)}}\left(\gamma_{2L,k|k-1} - \hat{\mathbf{b}}_k^-\right), \sqrt{\mathbf{Q}^{\mathbf{w}}}\right]\right\} \tag{18}$$

$$\mathbf{S}_{\mathbf{b}_k} = \mathrm{cholupdate}\left\{\left[\mathbf{S}_{\mathbf{b}_k}, \gamma_{0,k} - \hat{\mathbf{b}}_k^-, W_0^{(c)}\right]\right\} \tag{19}$$

$$\mathbf{P}_{\mathbf{a}_k\mathbf{b}_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\boldsymbol{\chi}_{i,k|k-1} - \hat{\mathbf{a}}_k^-\right]\left[\gamma_{i,k|k-1} - \hat{\mathbf{b}}_k^-\right]^T \tag{20}$$

$$\boldsymbol{\kappa}_k = \mathbf{P}_{\mathbf{a}_k\mathbf{b}_k}\left(\mathbf{S}_{\mathbf{b}_k}\mathbf{S}_{\mathbf{b}_k}^T\right)^{-1} \tag{21}$$

$$\hat{\mathbf{a}}_k = \hat{\mathbf{a}}_k^- + \boldsymbol{\kappa}_k\left(\mathbf{b}_k - \hat{\mathbf{b}}_k^-\right) \tag{22}$$

$$\mathbf{U}_k = \boldsymbol{\kappa}_k\mathbf{S}_{\mathbf{b}_k} \tag{23}$$

$$\mathbf{S}_k = \mathrm{cholupdate}\{\mathbf{S}_{\mathbf{a}_k}, \mathbf{U}_k, -1\} \tag{24}$$

where $\mathbf{Q}^{\mathbf{w}}$ denotes covariance matrix of the observation noise vector $\mathbf{w}_k$.

The SR-UKF performs the above procedure in an iterative fashion so as to achieve improved performance of state estimation of $\mathbf{a}_k$ ($k = 1, 2, 3, 4,\ldots$).

3.2 SR-UKF-based localization and tracking algorithm

Here, we employ the above SR-UKF procedure to solve localization and tracking problems in an IoT scenario. In order to describe movement state of a moving target in a sensor-based IoT environment, we define the following dynamic state and observation equations.

3.2.1 State equation

$$\begin{aligned}\mathbf{a}_{k+1} &= \mathbf{F}(\mathbf{a}_k) + \mathbf{v}_k \\ &= \Phi\mathbf{a}_k + \mathbf{v}_k\end{aligned} \tag{25}$$

where $\mathbf{a}_k$ denotes location and velocity information of moving target at the k-th moment; $\mathbf{v}_k$ represents the state noise vector; $\mathbf{\Phi}$ is a state transition matrix. Parameters in Eq. (25) should be defined in accordance with motion pattern of the target, while the target may move in any fashion. Without loss of generality, here we consider two situations: uniform linear motion and variable-speed curve motion.

When the target moves in a uniform linear motion (ULM), parameters in Eq. (25) are given by:

$$\mathbf{a}_k = [x_k \quad y_k \quad x_k' \quad y_k']^T \tag{26}$$

$$\mathbf{\Phi}_{ULM} = \begin{bmatrix} \mathbf{I}_{2\times2} & T\mathbf{I}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{I}_{2\times2} \end{bmatrix} \tag{27}$$

where $[x_k, y_k]$ is location coordinates of target at the k-th moment; $[x_k', y'_k]$ represents the derivative of $[x_k, y_k]$, which also stands for the velocity in the x- and y-axis direction; $\mathbf{I}_{2\times2}$ and $\mathbf{0}_{2\times2}$ denotes the 2 × 2 unit matrix and all-zero matrix, respectively; T is time interval of sensor detection.

When the target moves in a variable-speed curve motion (VCM), parameters in Eq. (25) are given by:

$$\mathbf{a}_k = [x_k \quad y_k \quad x_k' \quad y_k' \quad x_k'' \quad y_k'']^T \tag{28}$$

$$\mathbf{\Phi}_{VCM} = \begin{bmatrix} \mathbf{I}_{2\times2} & T\mathbf{I}_{2\times2} & \frac{1}{2}T^2\mathbf{I}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{I}_{2\times2} & T\mathbf{I}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & \mathbf{I}_{2\times2} \end{bmatrix} \tag{29}$$

where x_k'' and y_k'' represents the second-order derivative of x_k and y_k, which also stands for the acceleration in the x- and y-axis direction.

3.2.2 Observation equation

$$\begin{aligned}\mathbf{b}_k &= \mathbf{H}(\mathbf{a}_k) + \mathbf{w}_k \\ &= \begin{bmatrix} \sqrt{(X_1 - x_k)^2 + (Y_1 - y_k)^2} \\ \sqrt{(X_2 - x_k)^2 + (Y_2 - y_k)^2} \\ \vdots \\ \sqrt{(X_M - x_k)^2 + (Y_M - y_k)^2} \end{bmatrix} + \mathbf{w}_k\end{aligned} \tag{30}$$

where (X_1, Y_1), (X_2, Y_2), …, (X_M, Y_M) represent location coordinates of M sensor nodes, which are available for localization and tracking at the k-th moment, respectively; $\mathbf{b}_k = [R_1, R_2, \ldots, R_M]^{\mathrm{T}}$ represents distances between the target and available sensor nodes at the k-th moment, where R_i denotes the maximum sensing radius of the i-th ($i = 1, 2, \ldots, M$) available sensor node; $\mathbf{w}_k$ is the observation noise vector.

The SR-UKF-based localization and tracking algorithm can be implemented as follows:

1. Initialization

At the initialization stage, we use the proposed localization initialization model proposed in Section II to obtain initial location coordinates of target $[x_0, y_0]$. For simplicity, the initial velocity $[x'_0, y'_0]$ is given when the target moves in a uniform linear motion, while the initial acceleration $[x_0'', y_0'']$ is also given for a variable-speed curve motion. So the initial state vector is given by:

$$\mathbf{a}_0 = [x_0 \quad y_0 \quad x_0' \quad y_0']^T \quad \text{(see Eq. (26)for ULM)} \tag{31}$$

$$\mathbf{a}_0 = [x_0 \quad y_0 \quad x_0' \quad y_0' \quad x_0'' \quad y_0'']^T \quad \text{(see Eq. (28) for VCM)} \tag{32}$$

Based on several initialization tests, we acquire a group of initial values of $\mathbf{a}_0$, and then, we can calculate $\hat{\mathbf{a}}_0$ and $\mathbf{S}_0$ by using Eq. (8).

2. The k-th iterative Estimation, $k \in \{1, 2, \ldots, \infty\}$

(a) Sigma point calculation

Substitute $\hat{\mathbf{a}}_{k-1}$ and $\mathbf{S}_{k-1}$ into Eq. (9), then we get $\boldsymbol{\chi}_{k-1}$.

(b) Time update

By utilizing Eqs. (10–13), we get $\boldsymbol{\chi}_{k|k-1}$, $\hat{\mathbf{a}}_k^-$ and $\mathbf{S}_{\mathbf{a}_k}$. Note that, $\mathbf{F}(\cdot)$ in Eq. (10) should be replaced by Eq. (25).

(c) Measurement update

According to Eqs. (16–24), we get the estimate $\hat{\mathbf{a}}_k$ and $\mathbf{S}_k$ at the k-th moment. Note that, $\mathbf{H}(\cdot)$ in Eq. (16) should be replaced by Eq. (29).

Based on the above procedure which performs in an iterative fashion, we get the estimates of location and velocity information of the moving target at any moment, which indicates realization of target localization and tracking in an IoT scenario. Figure 4 shows the whole procedure of the proposed SR-UKF-based localization and tracking algorithm.

3.3 Computational complexity

Assume that the dimension of the state vector $\mathbf{a}_k$ and the observation vector $\mathbf{b}_k$ in Eqs. (25, 29) is L and M, respectively. In the whole algorithm, its computational complexity mainly comes from two complicated operations: QR decomposition and *cholupdate,* which are given by Eqs. (12, 13, 18 and 19). For simplicity and without loss of generality, here we only consider computational complexity, which comes from these two operations. The QR decomposition of matrix in Eqs. (12, 18) brings the complexity of $O((3L + 1)L^2)$; the function of *cholupdate* in Eqs. (13, 19) brings the complexity of $O(2LM^2)$. Generally, L is larger than

M, so the complexity of SR-UKF-based localization and tracking algorithm is $O(L^3)$, which is equal to the complexity of both the EKF- and UKF-based methods [37].

4 Simulation results and analysis

4.1 Simulation parameters

We consider a square sensor field (1,000 m × 1,000 m) which has (−500, −50 m), (−500, 500 m), (500, −500 m) and (500, 500 m) as its four corners. Fifty sensor nodes which are used for target detection are uniformly and randomly deployed in this field. We assume that the sensing ability of each sensor node is limited in a round region whose radius is 300 m. $T = 1\mathrm{s}$ is the time interval of sensor detection. Processing requirements in terms of time for the whole IoT architecture can be satisfied by using common processors such as MSP430 (Mixed Signal Processor 430), where it is used as 16 MHz clock with 16 bits CPU.

4.2 Performance assessment

4.2.1 Case 1: The target moves in a uniform linear motion (ULM)

(a) Localization Initialization

We assume that the initial location coordinate and velocity of the moving target are (−400, 400 m) and ($x_0' = 25$ m/s, $y_0' = -25$ m/s), respectively, which is given by:

$$\begin{aligned}\mathbf{a}_0 &= [\,x_0 \quad y_0 \quad x_0' \quad y_0'\,]^T \\ &= [\,-400\ \mathrm{m} \quad 400\ \mathrm{m} \quad 25\ \mathrm{m/s} \quad -25\ \mathrm{m/s}\,]^T\end{aligned} \tag{33}$$

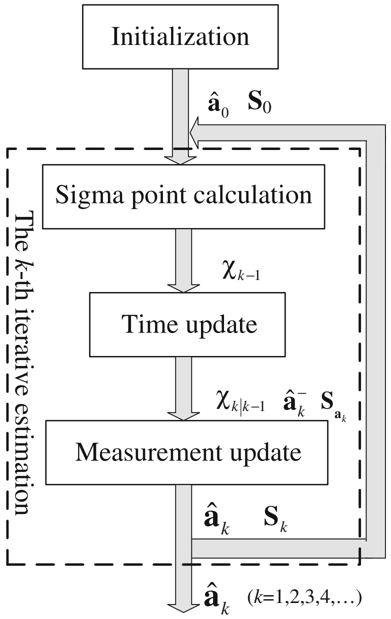
Initialization
$\hat{\mathbf{a}}_0$ $\mathbf{S}_0$
Sigma point calculation
χ_{k-1}
Time update
$\chi_{k|k-1}$ $\hat{\mathbf{a}}_k^-$ $\mathbf{S}_{\mathbf{a}_k}$
Measurement update
$\hat{\mathbf{a}}_k$ $\mathbf{S}_k$
$\hat{\mathbf{a}}_k$ (k=1,2,3,4,…)
The k-th iterative estimation

Fig. 4 Flow chart of the proposed SR-UKF-based algorithm

(b) Estimation of target trajectory

Figure 5 compares target trajectories obtained from the three algorithms, which are the EKF [29], the UKF [30–35] and the proposed SR-UKF-based localization and tracking algorithms, respectively. The black line in the figure means the real moving trajectory of the target. The lines with marks are trajectory estimates by the three algorithms. From this figure, we find that both the UKF and the proposed algorithms show identical and better trajectory estimation performance when the target moves in a uniform linear motion. The EKF performs poorly compared with the UKF and the proposed, which verifies the superiority of UKF-based methods to the EKF-based methods. The reason is that the linearization approach of the EKF results only in the first-order accuracy, while approximations of the UKF-based methods are accurate to at least the second order [37].

(c) Estimation of location point coordinates in the X-/Y-axis direction

Figures 6 and 7 compare location estimation errors of the three algorithms in the X-axis and Y-axis directions, respectively. Obviously, the EKF exhibits a worse result in localization, and its location estimation error is limited to about 20 m, while errors of the UKF-based and the proposed algorithms are less than about 10 and 3 m, respectively. Moreover, our algorithm shows slightly better estimation performance for target position than the UKF-based algorithm in Figs. 6 and 7, in which we find that the positioning accuracy of the UKF-based algorithm is less than 10 m, while that of the proposed algorithm is less than 3 m. Since the square-root (SR) filter employed in our algorithm ensures a nonnegative definite characteristic of the state covariance matrix, the proposed algorithm shows its superiority.

(d) Estimation of velocity in the X-/Y-axis direction

Figure 8 and 9 compare velocity estimation errors of the three algorithms in the X- and Y-axis directions, respectively. From the figure, we find that velocity estimates of all the three algorithms achieve convergence only after several seconds. However, the proposed algorithm shows the best velocity tracking since its velocity estimation error limits to less than about 0.2 m/s in both X- and Y-axis directions.

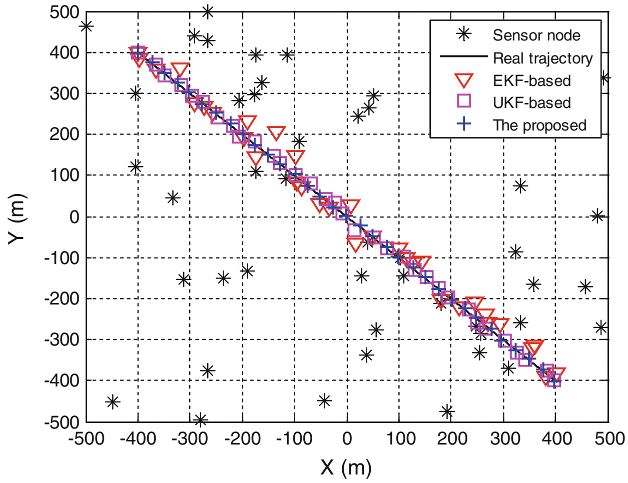

Fig. 5 Comparison of linear trajectory estimation

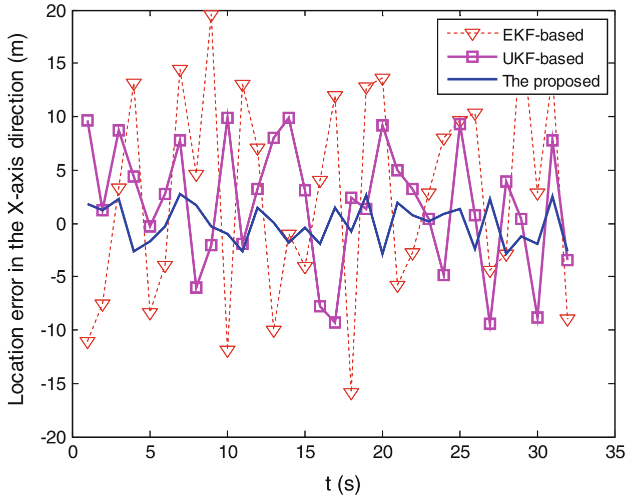

Fig. 6 Comparison of location estimation errors in the X-axis direction

4.2.2 *Case 2: The target moves in a variable-speed curve motion (VCM)*

(a) Localization Initialization

According to the basic physical laws of the movement of objects, difference of acceleration values between the X- and Y-axis directions results in a variable-speed curve motion. Here, we assume that the initial location coordinate, velocity and acceleration of the moving target are (–400, 400 m), ($x_0' = 25$, $y_0' = -25$ m/s) and ($x_0'' = 0$, $y_0'' = -1$ m/s^2), respectively. Note that, acceleration values here (x_0'' and y_0'') are different in the X- and Y-axis directions, so that the target moves in a variable-speed curve motion. The initial location information is summarized by:

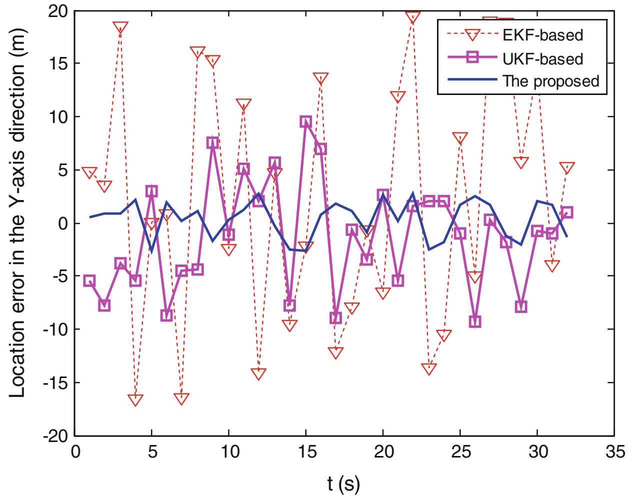

Fig. 7 Comparison of location estimation errors in the Y-axis direction

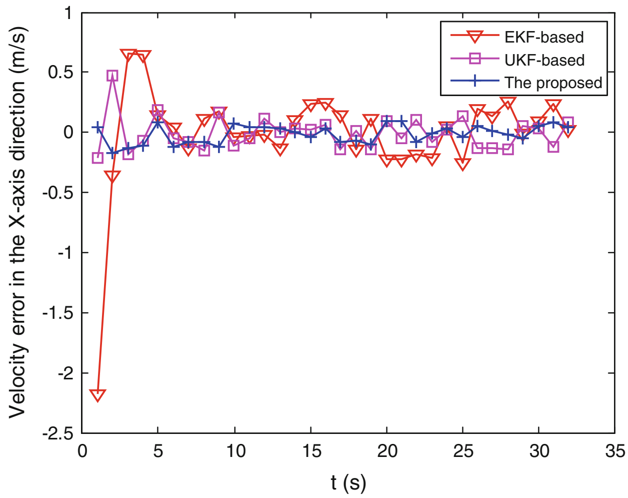

Fig. 8 Comparison of velocity estimation errors in the X-axis direction

$$\begin{aligned}\mathbf{a}_0 &= [x_0 \quad y_0 \quad x_0' \quad y_0' \quad x_0'' \quad y_0'']^T \\ &= \left[-400\text{ m} \quad 400\text{ m} \quad 25\text{ m/s} \quad 0\text{ m/s} \quad 0\text{m/s}^2 \quad -1\text{ m/s}^2\right]^T\end{aligned} \tag{34}$$

(b) Estimation of target trajectory

Figure 10 compares target trajectories recovered from the three algorithms. Obviously, all the three recovered curves presented with marks are close to the real trajectory curve, which means all the three algorithms perform well in target localization and tracking when the target moves in a variable-speed curve motion. However, both the UKF and the proposed algorithms show identical and superior trajectory estimation performance compared with the EKF-

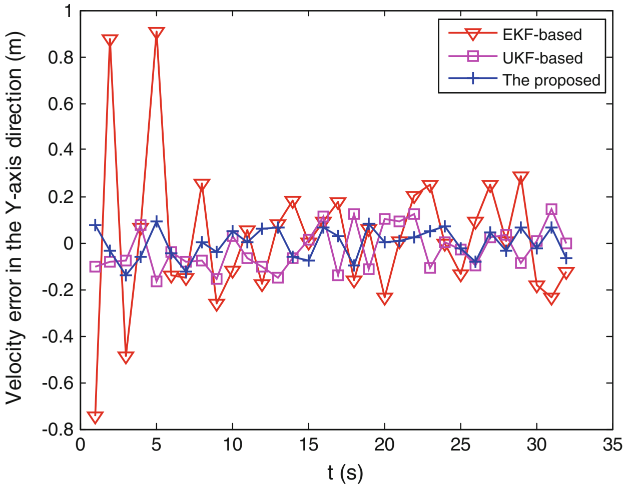

Fig. 9 Comparison of velocity estimation errors in the Y-axis direction

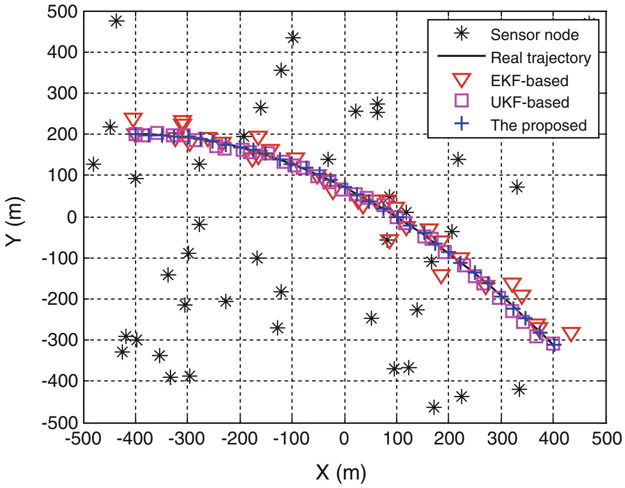

Fig. 10 Comparison of curve trajectory estimation

based algorithm, which is similar to the previous conclusion in a uniform linear motion. This demonstrates that the proposed algorithm can be applied for various occasions where the target moves in different motions.

(c) Estimation of location point coordinates in the X-/Y-axis direction

Figures 11 and 12 compare positioning errors of different algorithms in the X- and Y-axis directions, respectively. From the two figures, we may find that the proposed algorithm keeps on the best positioning performance among the three. Its positioning errors remain the same as those in Figs. 6 and 7 (that is: less than 3 m), which also demonstrates movement states of objects have little influence on positioning estimation performance of the proposed algorithm.

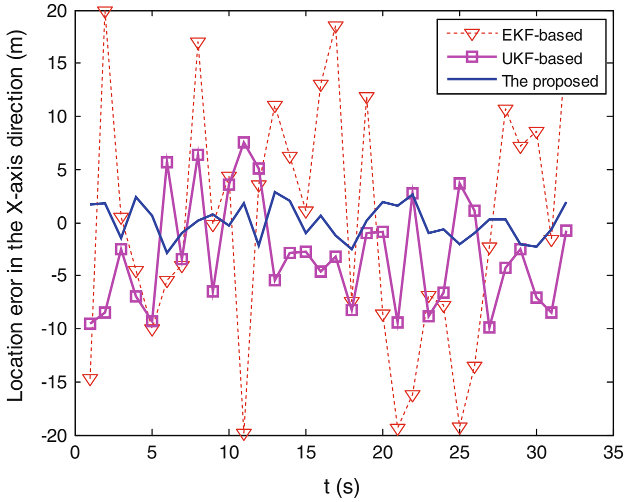

Fig. 11 Comparison of location estimation errors in the X-axis direction

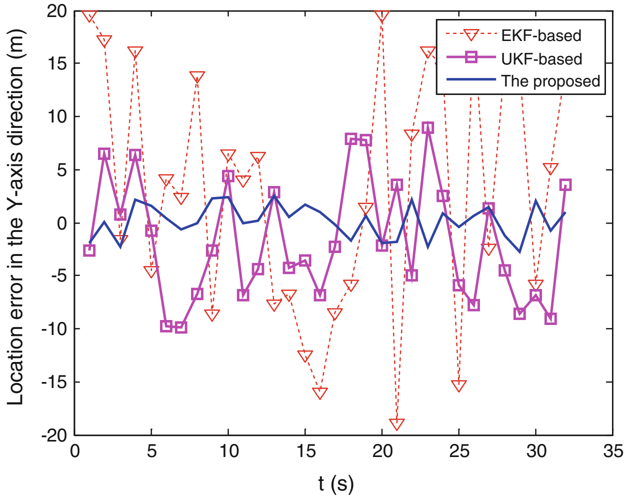

Fig. 12 Comparison of location estimation errors in the Y-axis direction

(d) Estimation of velocity in the X-/Y-axis direction

Figures 13 and 14 compare velocity estimation errors of the three algorithms in the X-axis and Y-axis directions, respectively. From the figures, we find that both the velocity estimation convergence and estimation errors of the proposed algorithm in a variable-speed curve motion case are similar to those in a uniform linear motion case, which demonstrates velocity states of objects have little influence on tracking performance of the proposed algorithm. Although there exist acceleration in the X- and Y-axis directions for the variable-speed motion, velocity estimation accuracy of the proposed algorithm still remain the best among three algorithms, which is less than about 0.2 m/s.

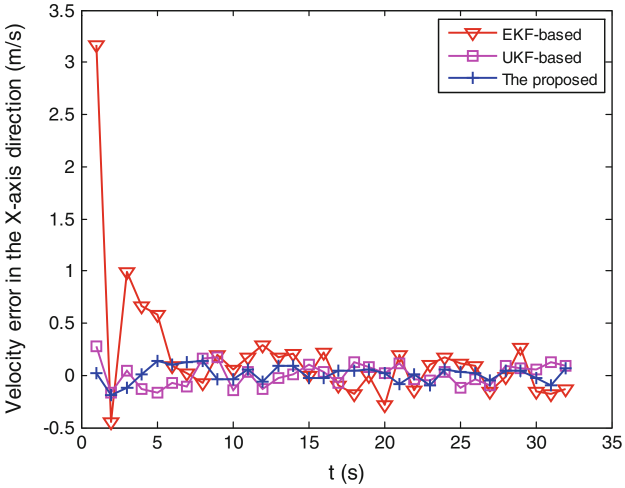

Fig. 13 Comparison of velocity estimation errors in the X-axis direction

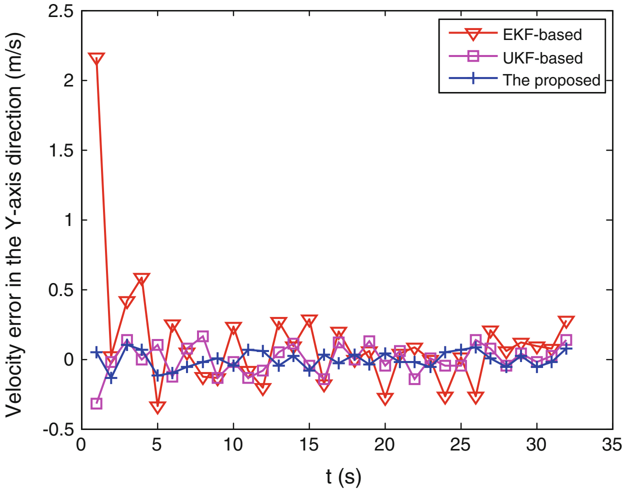

Fig. 14 Comparison of velocity estimation errors in the Y-axis direction

5 Conclusions

In order to achieve real-time localization and tracking of a moving target, we have proposed a square-root unscented Kalman filtering (SR-UKF)-based algorithm, which is suitable for an IoT environment where hundreds or thousands of networked sensor nodes are located at the sensing layer for distributed sensing and collaborative data processing. The data generated from wireless sensor nodes of the IoT make contributions to localization and tracking of the moving target. Based on the architecture of wireless sensor network (WSN), we first present a least-square (LS) criterion-based mathematical model for localization initialization in the IoT scenario. Then, we employ an SR-UKF idea for the further localization and tracking. By using the data coming from neighboring sensor nodes near the target, real-time location and velocity of the moving target can be estimated by implementation of SR-UKF in an iterative fashion so as to achieve target status tracking. Simulation results show that the proposed algorithm achieves good performance in estimation of both position and velocity of the target with either uniform linear motion or variable-speed curve motion. Compared with some conventional extended Kalman filtering (EKF) or UKF-based methods, the proposed algorithm shows lower location/velocity estimation error under the same computational complexity. Take our simulation for example, it achieves more precise localization and tracking. Under our simulation parameter setting, the errors of its location and velocity estimation can be limited to less than about 3 m and 0.2 m/s, respectively, which has already satisfied the demands for conventional and daily localization in campus, buildings and other public regions. In summary, this work is of significance for sensor-based localization and tracking applications.

Acknowledgments This work is supported by National Natural Science Foundation of China (61171014) and the Fundamental Research Funds for the Central Universities.

References

1. Iera A, Floerkemeier C, Mitsugi J, Morabito G (2010) The internet of things [Guest Editorial]. IEEE Wirel Commun 17(6):8–9
2. Guinard D, Trifa V, Karnouskos S, Spiess P, Savio D (2010) Interacting with the SOA-based internet of things: discovery, query, selection, and on-demand provisioning of web services. IEEE Trans Serv Comput 3(3):223–235
3. Zheng J, Simplot-Ryl D, Bisdikian C, Mouftah HT (2011) The internet of things [Guest Editorial]. IEEE Commun Mag 49(11):30–31
4. Florian M, Stephan K, Albrecht S (2010) What can the internet of things do for the citizen? IEEE Pervasive Comput 9(4): 102–104
5. Welbourne E, Battle L, Cole G, Gould K, Rector K, Raymer S, Balazinska M, Borriello G (2009) Building the internet of things using RFID: the RFID ecosystem experience. IEEE Internet Comput 13(3):48–55
6. Ning HS, Wang ZO (2011) Future internet of things architecture: like mankind neural system or social organization framework? IEEE Commun Lett 15(4):461–463
7. Zorzi M, Gluhak A, Lange S, Bassi A (2010) From today's INTRAnet of things to a future INTERnet of things: a wireless- and mobility-related view. IEEE Wirel Commun 17(6):44–51
8. Kranz M, Holleis P, Schmidt A (2010) Embedded interaction: interacting with the internet of things. IEEE Internet Comput 14(2):46–53
9. Choi S-H, Kim B-K, Park J, Kang C-H, Eom D-S (2004) An implementation of wireless sensor network. IEEE Trans Consum Electron 50(1):236–244
10. Li M, Li M, Koutsopoulos I, Poovendran R (2010) Optimal jamming attack strategies and network defense policies in wireless sensor networks. IEEE Trans Mob Comput 9(8):1119–1133

11. Hong S, Kim D, Ha M, Bae S, Park SJ, Jung W, Kim J-E (2010) SNAIL: an IP-based wireless sensor network approach to the internet of things. IEEE Wirel Commun 17(6):34–42
12. Wang C, Ma HD, He Y, Xiong SG (2012) Adaptive approximate data collection for wireless sensor networks. IEEE Trans Parallel Distrib Syst 23(6):1004–1016
13. Luo RC, Chen O (2012) Mobile sensor node deployment and asynchronous power management for wireless sensor networks. IEEE Trans Industr Electron 59(5):2377–2385
14. Tsai C-H, Tseng Y-C (2012) A path-connected-cluster wireless sensor network and its formation, addressing, and routing protocols. IEEE Sens J 12(6):2135–2144
15. Koshizuka N, Sakamura K (2010) Ubiquitous ID: standards for ubiquitous computing and the Internet of Things. IEEE Pervasive Comput 9(4):98–101
16. Mayorga CLF et al (2007) Cooperative positioning techniques for mobile localization in 4G cellular networks. IEEE international conference on pervasive services, pp 39–44
17. Gutmann JS et al (2012) Vector field SLAM: localization by learning the spatial variation of continuous signals. IEEE Trans Rob 99:1–18
18. Ouyang RW et al (2010) GPS localization accuracy improvement by fusing terrestrial TOA measurements. 2010 IEEE International conference on communications (ICC), pp 1–5
19. Huang R, Zaruba GV (2007) Incorporating data from multiple sensors for localizing nodes in mobile Ad Hoc networks. IEEE Trans Mob Comput 6(9):1090–1104
20. Guo B, Imai M (2007) Home-explorer: search, localize and manage the physical artifacts indoors. 21st International conference on advanced information networking and applications (AINA), pp 378–385
21. Zhang LQ et al (2008) A novel distributed sensor positioning system using the dual of target tracking. IEEE Trans Comput 57(2):246–260
22. Wang LJ et al (208) Location estimation of mobile user in wireless sensor network based on unscented Kalman filter. International conference on microwave and millimeter wave technology (ICMMT), vol 1, pp 96–99
23. Sun CJ et al (2010) A sensor based indoor mobile localization and navigation using unscented Kalman filter. 2010 IEEE on Position Location and Navigation Symposium (PLANS), pp 327–331
24. Ji X, Zha HY (2004) Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling. INFOCOM 2004, 23rd Annual joint conference of the IEEE computer and communication societies, vol 4, pp 2652–2661
25. Patwari N et al (2003) Relative location estimation in wireless sensor networks. IEEE Transactions on signal processing, special issue on signal processing in networking, vol 51, No. 8, pp 2137–2148
26. Kalman RE (1960) A new approach to linear filtering and prediction theory. Trans ASME J Basic Eng 82(1):35–45
27. Sinopoli B, Schenato L, Franceschetti M, Poolla K, Jordan MI, Sastry SS (2004) Kalman filtering with intermittent observations. IEEE Trans Autom Control 49(9):1453–1464
28. Gandhi MA, Mili L (2010) Robust Kalman filter based on a generalized maximum-likelihood-type estimator. IEEE Trans Signal Process 58(5):2509–2520
29. Chatterjee A, Matsuno F (2007) A neuro-fuzzy assisted extended Kalman filter-based approach for simultaneous localization and mapping (SLAM) problems. IEEE Trans Fuzzy Syst 15(5): 984–997
30. Julier SJ, Uhlmann JK (2004) Unscented filtering and nonlinear estimation. Proc IEEE 92(3):401–422
31. Zhan RH, Wan J (2006) Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation. IEEE Signal Process Lett 13(7):445–448
32. Angrisani L, Baccigalupi A, Schiano Lo Moriello R (2006) Ultrasonic time-of-flight estimation through unscented Kalman filter. IEEE Trans Instrum Meas 55(4):1077–1084
33. Zhan R, Wan J (2007) Iterated unscented Kalman filter for passive target tracking. IEEE Trans Aerosp Electron Syst 43(3): 1155–1163
34. Dini DH, Mandic DP, Julier SJ (2011) A widely linear complex unscented Kalman filter. IEEE Signal Process Lett 18(11): 623–626
35. Valverde G, Terzija V (2011) Unscented kalman filter for power system dynamic state estimation. IET generation, transmission & distribution, pp 29–37
36. Gustafsson F, Hendeby G (2012) Some relations between extended and unscented Kalman filters. IEEE Trans Signal Process 60(2):545–555
37. Van der Merwe R, Wan EA (2001) The square-root unscented Kalman filter for state and parameter-estimation. 2001 IEEE International conference on acoustics, speech, and signal processing (ICASSP), vol 6, pp 3461–3464
38. Holmes SA et al (2009) A O(N2) square root unscented Kalman filter for visual simultaneous localization and mapping. IEEE Trans Pattern Anal Mach Intell 31(7):1251–1263
39. Cattivelli FS, Lopes CG, Sayed AH (2008) Diffusion recursive least-squares for distributed estimation over adaptive networks. IEEE Trans Signal Process 56(5):1865–1877