Interacting Multiple Sensor Unscented Kalman Filter for Accelerating Object Tracking

Zhigang Liu, Jinkuan Wang, and Wei Qu

Abstract— **Due to limited sensing range for sensor nodes, moving object tracking has to be realized by relaying from one node to the other in a cluster. By taking object tracking in a fixed cluster as a Markov jump nonlinear system, the interacting multiple sensor unscented Kalman filter(IMSUKF) algorithm is designed to deal with distributed tracking. The proposed method can be divided into two parts: one-step unscented Kalman filter for object tracking and the fusion of the information provided by all the nodes. Finally, simulation results show the effectiveness of the proposed method.**

I. Introduction

DISTRIBUTED tracking is one of the most fundamental collaborative information processing problems in wireless sensor networks(WSN)[1]-[7]. The tracking problem in sensor networks is how to fuse observations from spatially distributed nodes for estimating the state of the dynamic process. There exist two kinds of fusion architectures: centralized, where fusion center receives all measurements from the local nodes to carry out filtering operation and the resulting state estimates are optimal in the mean squared sense, and distributed (or decentralized)in which local nodes can perform filtering operation on their own observations and send the local state estimates to the fusion center. Then fusion center combines local estimates to generate overall global state estimate[1].

The limited energy, computational power and communication resources of a sensor node require the use of a large number sensor nodes in a wider region. Collaboration among these nodes also allows the sensor network to determine the parameters of interest at a given location with greater accuracy. R. Olfati-Saber proposed distributed Kalman filter(DKF) with embedded consensus filters[3] that a central Kalman filter for homogeneous sensor networks can be decomposed into n microfilters with inputs that are provided by two types of consensus filters. By introducing the novel microfilter architecture with identical high-pass consensus filters, R. Olfati-Saber extended his algorithm for heterogeneous sensor networks[4]. Due to system modelling uncertainties, A. Ahmad et al designed the decentralized robust Kalman filtering scheme for uncertain stochastic systems over heterogeneous sensor networks[5]. But these existing algorithms are only valid for linear dynamic systems.

The cost of transmitting a bit is higher than that of computation and hence it may be beneficial to organize the sensor nodes into groups(or clusters) to reduce the communication overhead and to save energy resources of sensor nodes. In this paper we will discuss distributed tracking problems for fixed cluster, in which the cluster head is the fusion center. Motivated by the Interacting Multiple Model algorithm[10][11], we consider object tracking in a fixed cluster as a Markov jump nonlinear system, and design the interacting multiple sensor (IMS) algorithm for object tracking in a cluster, in conjunction with solving the nonlinear problem by unscented Kalman filter (UKF)[8][9]. Therefore, the proposed algorithm is named as the interacting multiple sensor unscented Kalman filter (IMSUKF).

II. Problem Formulation

Consider a cluster of n sensors at positions, $s_1, s_2, \cdots, s_n$. At any time k, the target state evolves according to

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{x}(k-1)) + \mathbf{w}(k) \tag{1}$$

where $\mathbf{w}(k)$ is zero-mean white Gaussian noise(WGN) vector of variance $\mathbf{Q}$ caused by disturbances and modelling errors. The sensing model of the sensor s_i is given by

$$\mathbf{z}_i(k) = \mathbf{h}(\mathbf{x}(k), s_i) + \mathbf{v}_i(k) \tag{2}$$

where $\mathbf{z}_i(k)$ is the observation vector for node i, and $\mathbf{v}_i(k)$ is the zero-mean additive WGN of variance $\mathbf{R}$ which is expressed in units of dB and typically as low as four and as high as 12.

Let d be the distance between the sensor node i and the target, its received signal strength(RSS) is typically modelled as

$$\mathbf{h}(d) = P_0 - 10 n_p \log \frac{d}{d_0} \tag{3}$$

where P_0 is the received power(dB) at a short reference distance d_0, and n_p is the path-loss exponent, typically between two and four [7]. Noted that RSS can be measured by each node's received signal strength indicator(RSSI) circuit during normal data communication without presenting additional bandwidth or energy requirements.

Assume that each node only communicates messages with the head node in the cluster. The proposed IMS algorithm can be divided into two parts: one-step unscented Kalman filter for object tracking and the fusion of the information provided by all the nodes. The former runs in each node, and the latter works in the head node.

This work was supported by the National Natural Science Foundation of China, under Grant No. 60874108.

Z. Liu is with the Department of Automation Engineering, Northeastern University, Qinhuangdao 066004, P. R. China (zliu@mail.neuq.edu.cn).

J. Wang is with the Department of Automation Engineering, Northeastern University, Qinhuangdao 066004, P. R. China (wjk@mail.neuq.edu.cn).

W. Qu is with the Department of Automation Engineering, Northeastern University, Qinhuangdao 066004, P. R. China (quweineu@yahoo.com.cn).

978-1-4244-6453-1/10/$26.00 ©2010 IEEE

III. ONE-STEP UNSCENTED KALMAN FILTER

Unscented transformation(UT) is designed by virtue of an observation that it is easier to approximate a gaussian distribution than an arbitrary nonlinear function. In contrast with extended Kalman filter (EKF), UT can provide higher order approximations without calculating any derivatives for nonlinear functions, by using the sigma points,

$$\{\mathbf{x}_k, W_k : k = 0, \cdots, 2D_x\}$$

where $\mathbf{x}_k$ are D_x-dimensional vectors and W_k are weights associated with each $\mathbf{x}_k$. Unscented Kalman Filter(UKF) embeds UT into the KF's recursive prediction and update structure[6]. Noted that we have dropped the subscript so that it can represent any sensor in a cluster. Then, we give the procedure of one-stepped UKF for each sensor node in a cluster as follows.

Firstly, we initialize with the mean of the state vector $\mathbf{x}(0)$ and its variance $K(0)$. Then we construct an approximation of the state estimation to get the sigma points with

$$\{\mathbf{x}_j(0), W_j : j = 0, \cdots, 2D_x\}.$$

Notice that the subscript indexes the sigma points for the augmented state variables.

Secondly, we compute the predictive means and variances for the state and observation vectors based on the current measurement as follows:

$$\begin{aligned} \mathbf{x}_j(k) &= \mathbf{f}(\mathbf{x}_j(k-1)) \\ \mathbf{x}(k|k-1) &= \sum_{j=0}^{2D_x} W_j \mathbf{x}_j(k) \end{aligned} \tag{4}$$

$$K_x(k|k-1) = \sum_{j=0}^{2D_x} W_j (\mathbf{x}_j(k) - \mathbf{x}(k|k-1))(\mathbf{x}_j(k) - \mathbf{x}(k|k-1))^T \tag{5}$$

$$\begin{aligned} \mathbf{z}_j(k) &= \mathbf{h}(\mathbf{x}_j(k)) \\ \mathbf{z}(k|k-1) &= \sum_{j=0}^{2D_x} W_j \mathbf{z}_j(k) \end{aligned} \tag{6}$$

$$K_z(k|k-1) = \sum_{j=0}^{2D_x} W_j (\mathbf{z}_j(k) - \mathbf{z}(k|k-1))(\mathbf{z}_j(k) - \mathbf{z}(k|k-1))^T \tag{7}$$

$$K_{xz}(k|k-1) = \sum_{j=0}^{2D_x} W_j (\mathbf{x}_j(k) - \mathbf{x}(k|k-1))(\mathbf{z}_j(k) - \mathbf{z}(k|k-1))^T \tag{8}$$

Thirdly, the update step uses the predictive means and variances with the new measurement $\mathbf{z}(k)$ to compute the new state mean and variance as follows:

$$\Phi(k|k-1) = K_{xz}(k|k-1)(K_z(k|k-1))^{-1} \tag{9}$$

$$\tilde{\mathbf{z}}(k) = \mathbf{z}(k) - \mathbf{z}(k|k-1) \tag{10}$$

$$\mathbf{x}(k) = \mathbf{x}(k|k-1) + \Phi(k|k-1)\tilde{\mathbf{z}}(k) \tag{11}$$

$$K_x(k) = K_x(k|k-1) - \Phi(k|k-1) K_z(k|k-1) \Phi^T(k|k-1) \tag{12}$$

In the above equations, $\Phi(k|k-1)$ is a gain matrix. Now, we have obtained the innovation $\tilde{\mathbf{z}}(k)$, the innovation variance $K_z(k|k-1)$, the updated state $\mathbf{x}(k)$ and the updated state variance $K_x(k)$. Then these parameters computed by node i are packed in the message as follows

$$\mathrm{msg}_i = \{\mathbf{x}^i(k), K_x^i(k), \tilde{\mathbf{z}}^i(k), K_z^i(k|k-1)\}, i = 1, \cdots, n \tag{13}$$

which are send to the head node in a cluster.

IV. INTERACTING MULTIPLE SENSOR ALGORITHM

A. Data Fusion

Due to limited sensing range for sensor nodes, moving Object tracking has to be realized by relaying from one node to the other in a cluster. Such the tracking process can be modelled as Markov stochastic process with

$$\mathrm{Prob}\{m(k+1) = s_j | m(k) = s_i\} = \pi_{ij}, \forall i, j \in n \tag{14}$$

After receiving the messages from all the nodes in the cluster, the cluster head node fuse these messages and further compute the sensor likelihood, the sensor probability, the overall state estimate and the overall covariance in order. For sensor node i, at any time k, its node likelihood is given by

$$\begin{aligned} L^i(k) &= p[\tilde{z}^i(k) | m^i(k), z^{k-1}] \\ &= \mathcal{N}(\tilde{z}^i(k); 0, K_z^i(k|k-1)) \end{aligned} \tag{15}$$

and the corresponding node probability is obtained as follows

$$c^i(k) = \frac{\tilde{c}^i(k) L^i(k)}{\sum_j \tilde{c}^j(k) L^j(k)} \tag{16}$$

where $\tilde{c}^i(k)$ is the predicted node probability for node i by the Markov switch matrix(14).

Using the equations (15) and (16), the overall estimate and covariance in the cluster are computed according to the following two equations

$$\hat{\mathbf{x}}(k) = \sum_{i=1}^{n} \mathbf{x}^i(k) c^i(k) \tag{17}$$

and

$$K(k) = \sum_{i=1}^{n} [K_x^i(k) + (\hat{\mathbf{x}}(k) - \mathbf{x}^i(k))(\hat{\mathbf{x}}(k) - \mathbf{x}^i(k))^T] c^i(k). \tag{18}$$

To compute the initialization parameters of new cycle, we give the following four equations based on the Markov chain method.

$$\tilde{c}^i(k+1) = P[m^i(k+1) | z^k] = \sum_j \pi_{ji} c^i(k), \tag{19}$$

$$\tilde{c}^{j|i}(k) = P[m^i(k) | m^i(k+1), z^k] = \pi_{ji} c^i(k) / \tilde{c}^i(k+1), \tag{20}$$

$$\bar{x}^i(k) = E[\mathbf{x}(k)|m^i(k+1), z^k] = \sum_j x^j(k)\tilde{c}^{j|i}(k), \quad (21)$$

and

$$\bar{K}^i(k) = \sum_{j=1}^{n}[K_x^j(k)+(\bar{x}^i(k))-\mathbf{x}^j(k))(\bar{x}^i(k)-\mathbf{x}^j(k))^T]\tilde{c}^{j|i}(k). \quad (22)$$

Using the above equations, we can get the mixing estimate $\bar{x}^i(k)$ and mixing covariance $\bar{K}^i(k)$ for node i in the cluster, and these values are include in the message as

$$\text{msg}_h = \{\bar{\mathbf{x}}^1(k), \bar{K}^1(k); \cdots; \bar{\mathbf{x}}^n(k), \bar{K}^n(k)\}. \quad (23)$$

Finally, the cluster head will broadcast this message to start the new cycle of object tracking.

B. Summary

Once that one object enters into a cluster, the cluster head firstly broadcasts the initial state and covariance to begin with the tracking process. On receiving the starting message, each node computes the innovation $\tilde{\mathbf{z}}(k)$, the innovation variance $K_z(k|k-1)$, the updated state $\mathbf{x}(k)$ and the updated state variance $K_x(k)$ by one-step unscented Kalman filter, and sends these results to the head node.

After collecting the messages from all the nodes, the head node fuses these messages into the overall state estimate and covariance. On the other hand, these messages are mixed into the mixing estimate $\bar{x}(k)$ and mixing covariance $\bar{K}(k)$ for each node in the cluster by use of the equations (19)-(22). Finally, this head node encapsulates these mixing results into a message packet and broadcasts this packet to start new cycle in the cluster.

While this object moves from the cluster to the other, the cluster head on one hand sends the current state estimate and covariance to the other cluster head, and on the other hand notifies all the nodes of the end of object tracking.

V. SIMULATION

Consider an accelerating target with dynamics

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} x_1(k-1) + x_2(k-1) \\ x_2(k-1) + 0.01 \end{bmatrix} + \mathbf{w}(k) \quad (24)$$

and the measurement equation can be displayed as follows

$$z_i(k) = -20 - 23\log[(x_1(k) - x_i)^2 + y_i^2] + v_i(k) \quad (25)$$

where (x_i, y_i) denotes the position of node i in a cluster. In the following simulation experiments, the state noise variance is

$$\mathbf{Q} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.0001 \end{bmatrix}$$

and the measurement noise variance $\mathbf{R} = 100$. In addition, the initial conditions are $\mathbf{x}(0) = [1, 1]^T$ and $P(0) = \mathbf{I}$, and

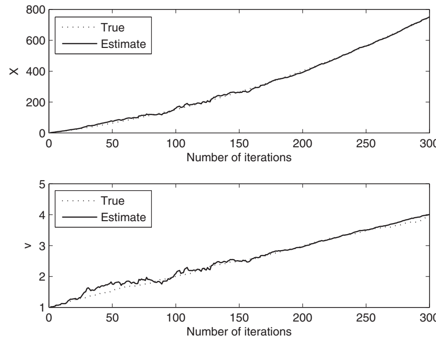

Fig. 1. The target's position versus and velocity versus number of iterations for a cluster with six nodes.

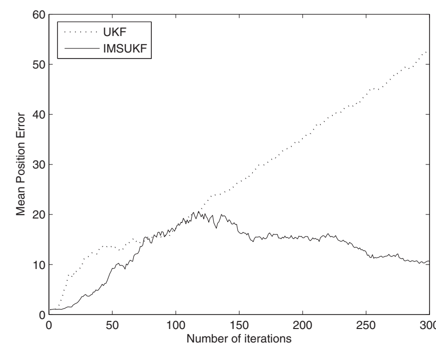

Fig. 2. Comparison of the performance of IMSUKF and UKF.

the Markov transition matrix is given as

$$\pi = \begin{bmatrix} 0.95 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.95 & 0.01 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.95 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.95 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.01 & 0.95 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.95 \end{bmatrix}.$$

Assume that six nodes in a cluster locate at (-1, 20), (150, 20), (300, 20), (450, 20), (600, 20) and (750, 20) respectively, and a target accelerates along the x-axis.

Figure 1 shows the tracking performance of the proposed IMSUKF algorithm. To show the performance of the proposed method further, we compare it with the UKF algorithm for one node at (-1, 20), and the obtained result with 50 independent runs is depicted in Figure 2, where the dotted line denotes the UKF algorithm and the solid line the IMSUKF algorithm. Obviously, the mean position error of the UKF algorithm is increasing with the number of iterations for the RSS-baed range estimates have variance proportional to their actual range. But the proposed method has a better performance than the UKF algorithm when the tracking procedure happens in the cluster. When the iteration time is greater than 120, the position error of the IMSUKF algorithm decreases because of collaborative tracking among

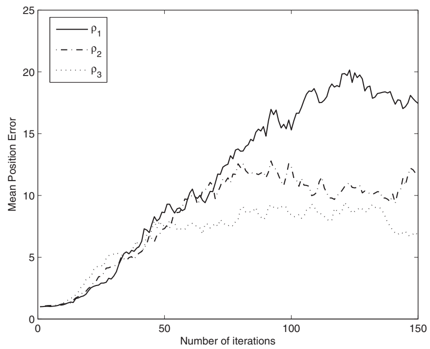

Fig. 3. The mean position error versus node density in a cluster.

all the nodes in the cluster.

The mean position error of the proposed method is affected by the node density ρ in Figure 3, where $\rho_1 < \rho_2 < \rho_3$. This implies that the performance of the proposed method gets better while the node density increases in a cluster. Noted that the position error is higher with density rising, when the iteration time is less than 40. This comes from the uncertainty of the beginning phase of the iteration process.

VI. Conclusion

Now that object tracking in a cluster is taken as a Markov jump nonlinear dynamic system, the IMSUKF algorithm is presented by introducing the Markov transition matrix and solving the nonlinear problem by UKF. Assume that each node only communicates messages with the head node in the cluster. The proposed IMSUKF algorithm can be divided into two parts: one-step unscented Kalman filter for object tracking and the fusion of the information provided by all the nodes. Simulation results show that this algorithm has a better performance than the UKF algorithm for one node because of the information fusion from all the nodes in the cluster. In addition, the performance of the proposed method gets better with node density increasing.

References

[1] T. Vercauteren, and X. Wang. Decentralized sigma-point information filters for target tracking in collaborative sensor networks. IEEE Trans. on Signal Processing, vol.53, pp.2997-3009, Aug. 2008.

[2] R. Olfati-Saber, and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays,IEEE Trans. on Automatic Control, vol.49, pp.1520-1533, Sep. 2004.

[3] R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference, pp. 8179C 8184, Dec. 2005.

[4] R. Olfati-Saber. Distributed Kalman filtering for sensor networks, Proc. of the 46th IEEE Conference on Decision and Control,pp.5492-5498, Dec. 2007.

[5] A. Ahmad, M. Gani, and F. Yang. Decentralized robust Kalman filtering for uncertain stochastic system over heterogeneous sensor networks. Signal Processing, vol.88, pp.1919-1928. Aug. 2008.

[6] L. Zhang, Q. Cheng, Y. Wang, and S. Zeadally. Novel distributed sensor positioning system using the dual of target tracking. IEEE Trans. on Computers, vol.57, pp.246-260, Feb. 2008.

[7] N. Patwari, J. Ash, S. Kyperountas, A. Hero III, R. Moses, and N. Correal. Locating the nodes: cooperative localization in wireless sensor networks, IEEE Signal Processing Magazine, vol.22, pp.54-69, Jul. 2005.

[8] S.J. Julier, J.K. Uhlmann, and H. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. IEEE Trans. on Automatic Control, vol.45, pp.477-482, Mar. 2000.

[9] S.J. Julier, and J.K. Uhlmann. Unscented filtering and nonlinear estimation. Proc. of the IEEE, vol.92, pp.401-422, Mar. 2004.

[10] E. Mazor, A. Averbuch, Y. Bar-shalom, and J. Dayan. Interacting multiple model methods in target tracking: a survery. IEEE Trans. on Aerospace and Electronic Systems, vol. 34, pp.103-123, Jan. 1998.

[11] X.R. Li, and V. P. Jilkov. Survey of maneuvering target tracking, Part V: Multiple-model methods. IEEE Trans. on Aerospace and Electronic Systems, vol. 41, pp.1255-1321, Oct. 2005.