Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Distributed Sigma-Point Kalman Filtering for Sensor Networks: Dynamic Consensus Approach

Yan Zhou, *Student Member, IEEE*
Department of Automation
Shanghai Jiao Tong University
Shanghai, P.R. China
yanzhou@sjtu.edu.cn

Jianxun Li, *Member, IEEE*
Department of Automation
Shanghai Jiao Tong University
Shanghai, P.R. China
lijx@sjtu.edu.cn

***Abstract*—A scalable Sigma-Point Kalman filter (DSPKF) is proposed for distributed target tracking in a sensor network in this paper. The main idea is to use dynamic consensus strategy to the information form sigma-point Kalman filter (ISPKF) that derived from weighted statistical linearization perspective. Each node estimates the global average information contribution by using local and neighbors' information rather than by the information from all nodes in the network. Therefore, the proposed DSPKF algorithm is completely distributed and applicable to large-scale sensor network. A novel dynamic consensus filter is proposed, and its asymptotical convergence performance and stability are discussed. Finally, a numerical example is given to illustrate the proposed scheme.**

Keywords—sigma-point Kalman filtering, average consensus, weighted statistical linearization, target tracking, sensor network

I. INTRODUCTION

With the advance in the fabrication technologies that integrate the sensing and the wireless communication technologies, tiny sensor motes can be densely deployed in the desire field to form a large-scale wireless sensor network (WSN). Distributed estimation and tracking through sensor networks is a problem with a large spectrum of applications [1-3], such as surveillance, rescue, traffic monitoring, pursuit evasion games, etc. The well-known strategy concerning estimation and tracking is decentralized Kalman filtering, which involves state estimation using a set of local Kalman filters that communicate with all other nodes [4-5]. The information flow in the traditional decentralized Kalman filtering or sigma-point Klaman filtering scheme [6] is all-to-all with communication complexity of $O(n^2)$ which is not scalable for sensor network [7]. Usually the energy cost related to communication between sensor nodes and computation in each node is significant when using such an algorithm in sensor networks. Reducing the energy cost in communication and computation can significantly increase the node life span [8].

Average consensus algorithms have proven to be effective tools for performing network-wide distributed computation task ranging from flocking to robot rendezvous as in the papers [9-10] and the references therein. Recently in [11-12], Olfati-Saber introduces a distributed Kalman filtering (DKF) algorithm that uses dynamic consensus strategy. The DKF algorithm consists of a network of micro-Kalman filters each embedded with a high-gain high-pass consensus filter. The role of consensus filters is fusion of sensor and covariance data obtained at each node. Very recently, the problem of estimating a simpler scenario with a scalar state of a dynamical system from distributed noisy measurements based on consensus strategies is considered in [13], the focuses are with the interaction between the consensus matrix, the number of messages exchanged per sampling time, and the Kalman gain for scalar systems. However, to the best the authors' knowledge, the distributed estimation problem in a sensor network so far is mainly focused on the state estimation problem for linear Gaussian noises case, little effort is devoted to the distributed state estimation problem in the case of nonlinear system dynamic or non-Gaussian noises.

In this paper, we focus on scalable or distributed sigma-point Kalman filtering algorithms (DSPKF) based on the information form SPKF, which is derived from the weighted statistical linear regression property of SPKF. Moreover, each node in the network only communicates message with its neighbors and then estimates the global average information contribution by using local and neighbors' information using a dynamic consensus strategy. Therefore, unlike the decentralized sigma-point Kalman filter (see e.g. [6]), the proposed filter is completely distributed and applicable to the large-scale network.

II. PROBLEM STATEMENT

Consider a sensor network with N sensors that are interconnected via an overlay network. As is well known the sensor network can be modeled by using algebraic graph theory [14]. A vertex of the graph corresponds to a node and edges of the graph capture the dependence of interconnections. Formally, a graph $G=(V,E)$ consists of a set of vertices $V=\{v_1,v_2,\cdots v_N\}$ indexed by nodes in the network, and a set of edges $E=\{(v_i,v_j)\in V\times V\}$, containing unordered pairs of distinct vertices. The set of neighbors of node i on graph G is defined as $\aleph_i=\{j:(i,j)\in E\}$. The degree of vertex i is defined as $d_i=|\aleph_i|$ and maximum degree is $d_{\max}=\max_i d_i$. Let Δ be the degree matrix, $\Delta=diag(d_i)$. The adjacency matrix J is the integer matrix with rows and columns indexed by the vertices, such as the *ij*-entry of J is equal to the number of edges from i to j. Following [14], Laplacian matrix of a graph G is defined as $L=\Delta-J$.

Considering the following nonlinear discrete-time system modeling a moving object in the sensor network

$$x(k+1) = f(x(k),k) + \omega(k) \tag{1}$$

$$z_i(k) = h_i(x(k),k) + \upsilon_i(k), \quad i = 1, 2, 3, \cdots, N \tag{2}$$

where $x(k) \in \Re^n$, $z_i(k) \in \Re^m$ are the state of the system and the ith measurement at the time step k, respectively; f and h_i are known, possibly nonlinear function of the state $x(k)$ and time step k. The process noise $\omega(k) \in \Re^p$ is assumed to be zero mean with covariance $Q(k)$, $\upsilon_i(k) \in \Re^m$ is additive measurement noise of the ith sensor. It is assumed that $\upsilon_i(k)$ is zero mean with covariance $R_i(k)$, and independent of the process noise $\omega(k)$. To ease the analysis, we assume that all sensors are synchronized and have the same measurement rate.

This paper focuses on design an appropriate filter capable of performing the target state estimation and tracking tasks with less communication and computational load. Note that the information flow in the traditional decentralized sigma-point Kalman filtering is all-to-all with communication complexity of $O(n^2)$ which is not scalable for large-scale sensor networks. Usually the energy cost related to communication between sensor nodes is significant when using such an algorithm.

III. THE INFORMATION FORM SPKF

In this section, the so-called information form sigma-point Kalman filter is derived from the weighted statistical linearization perspective.

A. Weighted Statistical Linearization

Now, consider a nonlinear function $u = g(x)$ which is evaluated in r sigma-points ($\chi_j, j = 1,2,...,r$), i.e.

$$\mu_j = g(\chi_j), j = 1,2,...,r \tag{3}$$

where the points χ_j are chosen such that the mean and covariance are consistent with the prior information $\bar{x} = \hat{x}$ and $\overline{P}_{xx} = P_{xx}$, with $\bar{x} = \sum_{j=1}^{r} \omega_j \chi_j$, $\overline{P}_{xx} = \sum_{j=1}^{r} \omega_j (\chi_j - \bar{x}) \cdot (\chi_j - \bar{x})^T$, and $\sum_{j=1}^{r} \omega_j = 1$. The statistical linearization is to find the linear regression $\mu = g(x) \approx Ax + b$ that minimizes the weighted sum of squared errors [15]

$$\{A, b\} = \arg\min \sum_{j=1}^{r} \omega_j \varepsilon_j^T \varepsilon_j \tag{4}$$

where the point-wise linearization error is defined as $\varepsilon_j = \mu_j - (A\chi_j + b)$. If we further define the following estimate of the posterior (Gaussian approximate) statistics of the propagated regression points μ_j

$$E(u) \approx \bar{u} = \sum_{j=1}^{r} \omega_j \mu_j \tag{5}$$

$$\mathrm{Var}(u) \approx \overline{P}_{uu} = \sum_{j=1}^{r} \omega_j (\mu_j - \bar{u})(\mu_j - \bar{u})^T \tag{6}$$

$$\mathrm{Cov}(x,u) \approx \overline{P}_{xu} = \sum_{j=1}^{r} \omega_j (\chi_j - \bar{x})(\mu_j - \bar{u})^T \tag{7}$$

the weighted statistical linear regression solution to is the usual weighted least square fitting: $A = \overline{P}_{xu}^T \overline{P}_{xx}^{-1}$ and $b = \bar{u} - A\bar{x}$. The mean and covariance of the linearization error ε is

$$\bar{\varepsilon} = \sum_{j=1}^{r} \omega_j \varepsilon_j = \bar{u} - A\bar{x} - b = 0 \tag{8}$$

$$\overline{P}_{\varepsilon\varepsilon} = \sum_{j=1}^{r} \omega_j \varepsilon_j \varepsilon_j^T = \overline{P}_{uu} - A\overline{P}_{xx}A^T \tag{9}$$

Once the statistical linearization has been determined we now can approximate the nonlinear function $u = g(x)$ as

$$g^{lin}(x) = Ax + b + \varepsilon \tag{10}$$

where ε is assumed to be zero-mean with covariance matrix $\overline{P}_{\varepsilon\varepsilon}$ and uncorrelated with x. Note that the statistical linearization in (18) provides the same approximation as (5)-(7):

$$E(g^{lin}(x)) = A\bar{x} + (\bar{u} - A\bar{x}) + 0 = \bar{u} \tag{11}$$

$$\mathrm{Var}(g^{lin}(x)) = A\overline{P}_{xx}A^T + (\overline{P}_{uu} - A\overline{P}_{xx}A^T) = \overline{P}_{uu} \tag{12}$$

$$\mathrm{Cov}(x, g^{lin}(x)) = \mathrm{Var}(x)A^T = \overline{P}_{xx}(\overline{P}_{xu}^T \overline{P}_{xx}^{-1})^T = \overline{P}_{xu} \tag{13}$$

B. Information form Sigma-Point Kalman Filtering

From (10), the discrete-time system (1)-(2) can be linearized into the following formulations

$$x(k+1) = \overline{F}(k)x(k) + b^x(k) + \overline{\omega}(k) \tag{14}$$

$$z_i(k) = \overline{H}_i(k)x(k) + b^{z_i}(k) + \overline{\upsilon}_i(k) \tag{15}$$

where the components corresponding to the state and the noise are separated. To obtain the linearization (14), (2n+1) sigma-points $\{\chi_j(k|k), \omega_j\}$ are generated according to

$$\chi_0(k|k) = \hat{x}(k|k), \; \omega_0 = \frac{\kappa}{n+\kappa}$$

$$\chi_j(k|k) = \hat{x}(k|k) + \left(\sqrt{(n+\kappa)P_{xx}}\right)_j, \; \omega_j = \frac{1}{2(n+\kappa)}, j = 1,2,..,n$$

$$\chi_{j+n}(k|k) = \hat{x}(k|k) - \left(\sqrt{(n+\kappa)P_{xx}}\right)_j, \; \omega_{j+n} = \frac{1}{2(n+\kappa)} \tag{16}$$

where κ is a scaling parameter usually chosen as 0 or 3-n and $\left(\sqrt{P}\right)_j$ denotes the j-th row of the Cholesky decomposition of P. Then, each of the sigma-point is processed through the nonlinear transition equation (1). Moreover, to obtain the linearization (15), each of those (2n+1) sigma-points $\{\chi_j(k+1|k), \omega_j\}$ generated by (16) in a similar way is propagated through the nonlinear observation equation (2).

Based on the propagation of these sigma-points, the corresponding matrices in (14) and (15) can be computed from (10) as follows

$$\overline{F}(k) = \overline{P}_{xx}^{T}(k+1, k+1 \mid k)\overline{P}_{xx}^{-1}(k \mid k),$$
$$b^{x}(k) = \overline{x}(k+1 \mid k) - \overline{F}(k)\overline{x}(k \mid k)$$

$$\overline{H}(k) = \overline{P}_{xz_i}^{T}(k+1 \mid k)\overline{P}_{xx}^{-1}(k+1 \mid k),$$
$$b^{z_i}(k) = \overline{z}_i(k \mid k-1) - \overline{H}_i(k)\overline{x}(k \mid k-1) \qquad (17)$$

Furthermore, the total process noise $\overline{\varpi}(k) = \omega(k) + \varepsilon^{x}(k)$ and the total observation noises $\overline{\upsilon}_i(k) = \upsilon_i(k) + \varepsilon_i^{z_i}(k)$ of ith sensor are zero-mean with covariance respectively

$$\begin{aligned} \mathrm{Var}(\overline{\varpi}(k)) &= \overline{Q}(k) = Q(k) + \overline{P}_{\varepsilon\varepsilon}^{x}(k) \\ &= Q(k) + \overline{P}_{xx}(k+1 \mid k) - \overline{F}(k)\overline{P}_{xx}(k \mid k)\overline{F}^{T}(k) \end{aligned}$$

$$\begin{aligned} \mathrm{Var}(\overline{\upsilon}_i(k)) &= \overline{R}_i(k) = R_i(k) + \overline{P}_{\varepsilon\varepsilon}^{z_i}(k) \\ &= R_i(k) + \overline{P}_{z_i z_i}(k \mid k-1) - \overline{H}_i(k)\overline{P}_{xx}(k \mid k-1)\overline{H}_i^{T}(k) \end{aligned} \qquad (18)$$

where in second equation of above two derivations, covariance of the linearization error in (9) is used.

Then, by define $Y_i(k \mid k)$ and $Y_i(k \mid k-1)$ the information matrices of the local estimator and applying the standard information form Kalman filter to the linearized state-space (14)-(15), the information form sigma-point Kalman filter for the nonlinear system (1)-(2) can be derived as

PREDICTION

$$\hat{y}_i(k \mid k-1) = Y_i(k \mid k-1)\left[\overline{F}(k)\hat{x}_i(k-1 \mid k-1) + b^{x}(k)\right] \qquad (19)$$
$$Y_i(k \mid k-1) = \left(\overline{F}(k)Y_i^{-1}(k-1 \mid k-1)\overline{F}^{T}(k) + \overline{Q}(k)\right)^{-1} \qquad (20)$$

UPDATE

$$\hat{y}_i(k \mid k) = \hat{y}_i(k \mid k-1) + \overline{H}_i^{T}(k)\overline{R}_i^{-1}(k)\left[z_i(k) - b^{z_i}(k)\right] \qquad (21)$$
$$Y_i(k \mid k) = Y_i(k \mid k-1) + \overline{H}_i^{T}(k)\overline{R}_i^{-1}(k)\overline{H}_i(k) \qquad (22)$$

As is well known, the sigma-point Kalman filter is a usual Kalman filter running on the very same linearized state-space. On this linear state-space, the information filter and the Kalman filter are strictly equivalent. Therefore, the information form sigma-point Kalman filter is equivalent to a sigma-point Kalman filter.

Remark 1: Since the similar structural simplicity to the standard information filter is preserved, the sigma-point information Kalman filter can be easily extended to achieve decentralized estimation as described in [6]. However, as stated above, the information flow in the decentralized sigma-point Kalman filtering is all-to-all with communication complexity of $O(n^2)$ which is not scalable. Usually the energy cost related to computation in each node and communication between sensor nodes is significant when using such an algorithm. This motivates the research on the scalable sigma-point Kalman filtering for sensor networks in Section IV.

IV. DISTRIBUTED SIGMA-POINT KALMAN FILTER

We now consider the distributed sigma-point filtering based on average consensus. The main idea is to calculate global information contribution of the entire network from the local (neighbors') information based on dynamic consensus strategy. First, we define the following average inverse-covariance matrix

$$U(k) = \frac{1}{N}\sum_{i=1}^{N} U_i(k) = \frac{1}{N}\sum_{i=1}^{N} \overline{H}_i^{T}(k)\overline{R}_i^{-1}(k)\overline{H}_i(k) \qquad (23)$$

and the average measurements

$$u(k) = \frac{1}{N}\sum_{i=1}^{N} u_i(k) = \frac{1}{N}\sum_{i=1}^{N} \overline{H}_i^{T}(k)\overline{R}_i^{-1}(k)\left[z_i(k) - b^{z_i}(k)\right] \qquad (24)$$

These averages can be estimated in each sensor node by using an average consensus filter proposed in [9] and [16]. Each node exchanges the local information contribution with its neighbors and estimates the global information contribution based on neighbor's local ones through the consensus filter. Specifically, in a sensor node i, let $\hat{U}_i(k)$ denote the estimate of global average inverse-covariance matrix $U(k)$, and $\hat{u}_i(k)$ denote the estimate of average measurement $u(k)$. A discrete-time form consensus filter is designed as follows

$$\mathrm{y}_i(k) \leftarrow \mathrm{y}_i(k) + \delta\left[\beta\sum_{j\in\aleph_i}\left(\mathrm{y}_j(k)\text{-}\mathrm{y}_i(k)\right) + \left(\mathrm{x}_i(k)\text{-}\mathrm{x}_i(k-1)\right)\right] \qquad (25)$$

where δ is the sampling time-step, the gain $\beta > 0$ is relatively large ($\beta \sim O(1/\lambda_2)$ where λ_2 is the second smallest eigenvalue of the Laplacian matrix L) for randomly generated ad-hoc topologies that are rather sparse [12]. $\mathrm{y}_i(k)$ is the consensus filter state of node i on time step k, which estimates the filter input $\mathrm{x}_i(k)$. As can be proofed, $\mathrm{y}_i(k)$ asymptotically converges to the average of the local input $\mathrm{x}_i(k)$ (see subsection *B* for details)

$$\mathrm{y}_i(k) \to \frac{1}{N}\sum_{i=1}^{N}\mathrm{x}_i(k) \qquad (26)$$

Note that the output $\mathrm{y}_i(k)$ represents either $\hat{U}_i(k)$ or $\hat{u}_i(k)$, and the input $\mathrm{x}_i(k)$ represents either $U_i(k)$ or $u_i(k)$, we have

$$\hat{U}_i(k) \to U(k) = \frac{1}{N}\sum_{i=1}^{N} U_i(k) \qquad (27)$$

$$\hat{u}_i(k) \to u(k) = \frac{1}{N}\sum_{i=1}^{N} u_i(k) \tag{28}$$

Remark 2: In the continuous-time form, (25) can be reformulated as

$$\dot{y}_i = \beta \sum_{j \in \aleph_i} \left(y_j - y_i \right) + \dot{x}_i \tag{29}$$

We notice the consensus filter proposed in [12] has the following form:

$$\frac{d}{dt}\left(y_i - x_i\right) = \beta \sum_{j \in \aleph_i} \left(y_j - y_i \right) + \beta \sum_{j \in \aleph_i} \left(x_j - x_i \right) \tag{30}$$

The difference between (29) and (30) is that the consensus filter in (30) does not need to have neighbor's input $x_{j,k} (j \in \aleph_i)$, which certainly reduces the cost of communication. Moreover, in [16], the equation of the dynamic consensus algorithm is given as $\dot{y}_i = \sum_{j \in \aleph_i} \left(y_j - y_i \right) + \dot{x}_i$, which is a special case when assuming $\beta = 1$.

A. Procedure of the Distributed Sigma-Point Kalman Filter

In terms of the above discuss, procedure of the proposed DSPKF can be summarized as follows.

Algorithm 1. Distributed sigma-point Kalman filtering

1: Initialization: $U_i(0)$, $u_i(0)$, $\hat{U}_i(0)$, $\hat{u}_i(0)$.

2: Update the local information contribution in terms of

$$U_i(k) = \overline{H}_i^T(k)\overline{R}_i^{-1}(k)\overline{H}_i(k)$$

$$u_i(k) = \overline{H}_i^T(k)\overline{R}_i^{-1}(k)\left[z_i(k) - b^{z_i}(k)\right].$$

3: Consensus filter:

Each node calculates the estimated global information contribution using (25).

4: Update the local predictive information state and matrix according to (19)-(20).

5: Estimate the local information state and matrix

$$\hat{y}_i(k \mid k) = \hat{y}_i(k \mid k-1) + \hat{u}_i(k)$$

$$Y_i(k \mid k) = Y_i(k \mid k-1) + \hat{U}_i(k).$$

Remark 3: Note in Algorithm 1 the message broadcasted only to the neighbor nodes in the set $\aleph_i$ for ith sensor. In the contrary, the decentralized estimation algorithm proposed in [6] involves broadcasting the message to all the nodes in the network, which will congest the link. Especially in the large-scale network or tiny sensor motes densely deployed wireless sensor network, we believe there are significant scalability advantages for the proposed sigma-point Kalman filtering strategy compared with the traditional schemes [6]. Moreover, since the dynamic consensus filter (25) is convergent provided that the network is connected (see the following subsection for details), this makes the distributed sigma-point filter more robust in the case of switching topology and link failure.

B. Performance Analysis of the Consensus Filter

In this subsection, we will prove that the proposed dynamics (26) is a high-pass filter and tracks the dynamic consensus with zero steady-state error. First, by stacking all node states y_i and input x_i into vectors y and x, respectively, we get a matrix form from (26)

$$\dot{y} = -\beta L y + \dot{x} \tag{31}$$

The corresponding MIMO transfer function is given by

$$H(s) = \frac{Y(s)}{X(s)} = \left(sI + \beta L\right)^{-1} s \tag{32}$$

It can be seen from (32) easily that $\lim_{s \to \infty} H(s) = I$. Therefore, given the graph G is connected, the proposed consensus filter (25) is a high-pass filter.

Furthermore, for a connected undirected graph G, $\sum_j L_{ij} = 0$. That implies that $Ly = 0$. Thus, we have the following conservation property [16] for the above dynamics consensus

$$\frac{d}{dt}\left(\sum_{i=1}^{N} y_i\right) = \frac{d}{dt}\left(\sum_{i=1}^{N} x_i\right) \tag{33}$$

Therefore, the instantaneous sum of the estimate variables y_i is equal to the instantaneous sum of the input variables x_i. Intuitively, this is precisely the property we would expect in order to track the time-varying average consensus. This intuition is formalized in the following theorem.

Theorem 1: Consider the dynamic system (26) with the MIMO transfer function (40). Suppose the input $x(s)$ has all its poles in the left half-plane, and at most one pole at s=0. Then, for all i,

$$\lim_{t \to \infty}\left(y_i(t) - \frac{1}{N}\sum_{i=1}^{N} x_i(t) \right) = 0 \tag{34}$$

That is, each agent tracks the dynamic consensus with zero steady-state error.

Proof: Consider the error signal $e(t)$, and its Laplace transform

$$E(s) = Y(s) - \frac{1}{N}11^T X(s) \tag{35}$$

where 1 is the vector composed of N entries of scalar 1. The error $e(t)$ is the vector of deviations between the $y_i(t)$

estimates, and the instantaneous average of the $\mathrm{x}_i(t)$ terms. Based on (35) and (32), we have the following MIMO transfer function from $\mathrm{X}(s)$ to $\mathrm{E}(s)$

$$H_{\mathrm{ex}}(s)=\frac{\mathrm{E}(s)}{\mathrm{X}(s)}=(sI+\beta L)^{-1}s-\frac{1}{N}11^T \quad (36)$$

Since the Laplacian is a symmetric matrix, it admits a spectral decomposition as

$$L=\sum_{i=1}^{N}\lambda_i P_i \quad (37)$$

where the λ_i terms are real eigenvalues, and the P_i terms are orthogonal projections onto mutually orthogonal eigenspaces. It is a fact from graph theory that connectedness of G implies the following properties: 1) $\lambda_1=0$; 2) $P_1=11^T/N$; and 3) $\lambda_1>0$ for all i>1. Then (36) can be reformulated as

$$H_{\mathrm{ex}}(s)=\left(\frac{1}{N}11^T+\sum_{i=2}^{N}\frac{s}{s+\beta\lambda_i}P_i\right)-\frac{1}{N}11^T=\sum_{i=2}^{N}\frac{s}{s+\beta\lambda_i}P_i \quad (38)$$

This transfer function has a single zero at s=0, and all the terms in the summation are stable. Thus, for an arbitrary stable input signal $\mathrm{X}(s)$ with at most one pole at s=0, the Final Value Theorem implies that $\mathrm{e}(t)\to 0$ as $t\to\infty$. This completes the proof. □

V. SIMULATION EXAMPLE

The proposed sigma-point filter is applied to tracking a target moving on noisy circular trajectories. Consider N=50 sensors randomly deployed in the query area, which is $50\mathrm{m}\times 50\mathrm{m}$ with the coordinate from (-25, -25) to (25, 25). The layout of the network is illustrated in Fig. 1, where a 'o' stands for the location of a sensor. In the particular setup of Fig. 1, there are 230 links with $d_{\max}=8$. Consider a target with the following dynamics adopted from [12]

$$\dot{x}=F_c x+G_c\omega$$

with $A_c=[0\ -2;2\ 0]$, and $G_c=I_2$. We use the discrete-time model of the target with parameters

$$F=I_2+\delta F_c+\frac{\delta^2}{2}F_c^2+\frac{\delta^3}{6}F_c^3,\ G=\delta G_c$$

The step-size is $\delta=0.025$. We adopt the following measurement model for the ith sensor [18]

$$y_i(k)=a/\|(x(k),y(k))-(x_s(i),y_s(i))\|+\upsilon_i(k)$$

where $a=40$ is the assumed known amplitude of the sound source, $\|(x(k),y(k))-(x_s(i),y_s(i))\|$ denoting the distance between the target and the ith sensor, and the covariance of $\upsilon_i(k)$ is $R_i(k)=\sqrt{i}$ for i= 1, 2, …, 50.

The simulation is performed 100 Monte Carlo runs each with 200 time steps. The Root of Mean Square Error (RMSE) is adopted to evaluate the performance. We compared the proposed algorithm (referred *DisSPKF* thereafter) with

1) centralized sigma-point Kalman filter (*CenSPKF*), in which the original measurement of each node is transmitted to a fusion center, then the SPKF is adopted to estimate the target state; and

2) decentralized sigma-point Kalman filter with a track fusion center (*DeFusion*), in which each node in the network estimate the target's track by SPKF in terms of local measurement, then the fusion center combines the local estimate according to weighted average approach.

Simulation results for RMSE in x-direction are compared in Fig. 2 and Fig. 3. In Fig. 2, for *DisSPKF*, the i=49 node with degree $d_{49}=6$ is shown; for *CenSPKF* and *DeFusion*, the measurements or the estimation of all nodes are sent to the fusion center, respectively. In Fig. 3, RMSE in x-direction by *DisSPKF* from node i=40 ($d_{40}=1$) is compared with those by *CenSPKF* and *DeFusion*. Obviously, the proposed algorithm yields identical, if not better, performance compared to the *DeFusion* strategy. Furthermore, both *DisSPKF* and *DeFusion* performance very closely to the *CenSPKF*, which is minimum mean squared error estimation (MMSE) in the case of Gaussian white noise. The RMSE in y-direction performs similarly, thus omitted for space reason. Of course, the difference among *DisSPKF*, *DeFusion*, and *CenSPKF* is that the latter two algorithm need the communication between all the local nodes and the fusion center, hence are not scalable for large-scale sensor network. While the proposed *DisSPKF* is scalable due to that it estimates the global average information contribution by using information from its neighbors rather than from all nodes in the network. The results from other nodes have a similar pattern and are omitted here.

Estimation error by itself is no longer the only measure of performance in distributed estimation in sensor networks. In a peer-to-peer estimation architecture, no particular fusion centers exist and every node is supposed to known the estimate of the target state. The agreement of estimate of every node permits the query on any node in the network about the estimation. In Fig. 4, the consecutive snapshots of estimates of all nodes are shown. The estimates appear as a cohesive set of particles that move around the position of the target.

VI. CONCLUSIONS

The distributed sigma-point Kalman filtering problem for target tracking in sensor networks has been investigated based on average consensus strategy. Each node estimated the global average information contribution by using local and neighbors' information. A novel dynamic consensus algorithm has been proposed, with its asymptotical convergence performance explored. Furthermore, the proposed DSPKF is applicable to the large-scale sensor network. A numerical example has been given to illustrate the effectiveness of the proposed scheme.

ACKNOWLEDGMENT

This work is partly supported by the National Natural Science Foundation of China (60874104), the Key Foundation for Basic Research from Science and Technology Commission of Shanghai (08JC1411800), the Aeronautic Science Foundation of China (20075157007), and 973 Project from Science and Technology Ministry of China (2009CB824900).

REFERENCES

[1] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration for tracking applications," *IEEE Signal Proces. Mag.*, vol.19, no. 2, pp. 61-72, Mar. 2002.

[2] M. Han, W. Xu, H. Tao, and Y.Gong, "An algorithm for multiple object trajectory tracking," In: *Proc. CVPR*, 2004.

[3] S. Oh, and S. Sastry, "Tracking on a graph," In: *Proc. Int. Workshop on Information Processing in Sensor Networks*, 2005.

[4] X. R. Li, and J. Wang, "Unified optimal linear estimation fusion – Part II: Discussions and examples," in *Proc. 3th ISIF Conf. On Information Fusion*, MoC2/18- MoC2/25 vol.1, July 2000.

[5] S.L. Sun and Z.L. Deng, "Multi-sensor optimal information fusion Kalman filter", *Automatica*, vol. 40, pp. 1017-1023, 2004.

[6] T. Vercauteren, and X. Wang, "Decentralized sigma-point information filters for target tracking in collaborative sensor networks," *IEEE T. Signal Processing*, vol. 53, no. 8, pp. 2997-3009, Aug. 2005.

[7] J.L. Speyer, "Computation and transmission requirements for a decentralized linear-quadratic-gaussian control problem," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp.1453-1464, Sep. 2004.

[8] P. Gupta, and P.R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 4, no.3, pp. 388-404, 2000.

[9] R. Olfati-Saber, and R.M. Murry, "Consensus problems in network of agents with switching topology and time-delays," *IEEE Trans. Automat. Control*, vol. 49, no.9, pp. 101-115, Sep. 2004.

[10] W. Ren, and R.W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Automat. Control*, vol. 50, no.5, pp. 655-661, May, 2005.

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

[12] R. Olfati-Saber, Distributed Kalman filtering for sensor networks. *46th IEEE Conf. on Decision and Control*, New Orleans, UAS, Dec., 2007.

[13] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distriubted Kalman filtering based on consensus strategies," *IEEE J. Selected Areas in Communications*, vol. 26, no. 4, pp. 622-632 May, 2006.

[14] C. Godsil and G. Royle, *Algebraic graph theory*. Springer-Verlag, 2001.

[15] R. Van der Merwe and E. Wan, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," in *Proc. Workshop Adv. In Machine Learning*, Montreal, QC, Canada, Jun. 2003.

[16] D. Spanos, R. Olfati-Saber, and R.M. Murray, "Consensus tracking in mobile networks," *The 16th IFAC World Congress*, Prague, Czech, 2005.

[17] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next contrary challenges: Scalable coordination in sensor networks," *Mobicom '99*, Seattle Washington, UAS, 1999.

[18] A. Ribeiro, G.B. Giannakis, and S.E. Rounmeliotis, "SOI-KF: Distributed Kalman filtering with low-cost communications using the sign of innovations," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4782-4795, Dec. 2006.

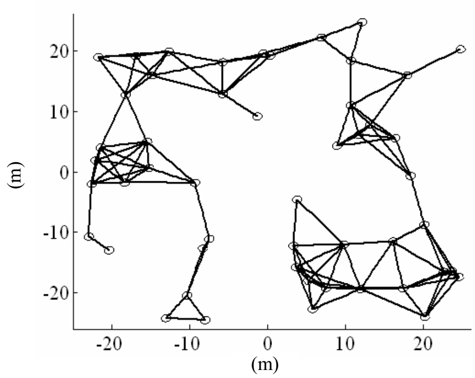

Fig. 1. A sensor network with 50 nodes and 230 links.

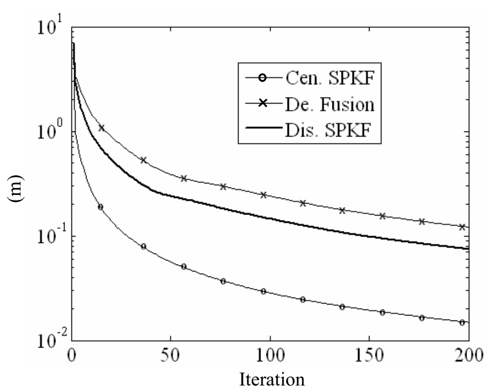

Fig. 2. Compared RMSE in *x*-direction by *Dis. SPKF* from node *i*=49 (degree=6) with by *Cen. SPKF* and *De.Fusion*.

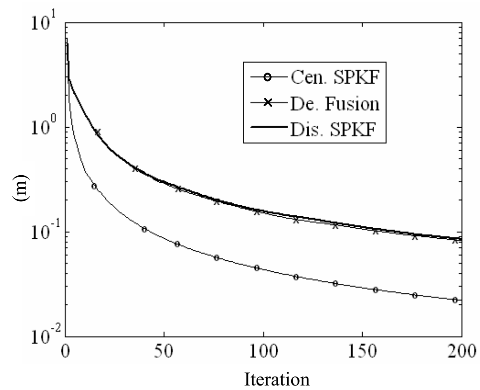

Fig. 3. Compared RMSE in *x*-direction by *Dis. SPKF* from node *i*=40 (degree=1) with by *Cen. SPKF* and *De.Fusion*.

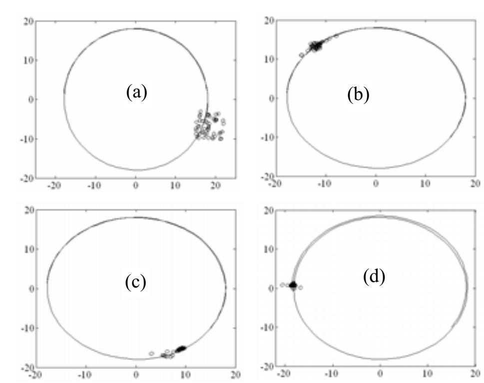

Fig. 4. Snapshots of the location estimates from all nodes on the 1, 60, 120, and 200th iteration in (a)-(d), respectively.