INTERLACED SIGMA-POINT INFORMATION FILTERING FOR DISTRIBUTED STATE ESTIMATION OF MULTI-AGENT SYSTEMS

Wenyun Gao

Department of Electronic Engineering
Tsinghua University, Beijing

Xi Chen, Menglu Wang

Space Center
Tsinghua University, Beijing

ABSTRACT

This paper presents a new interlaced sigma-point information filtering (ISPIF) algorithm for distributed state estimation of multi-agent systems (MAS). The ISPIF is derived by first p-resenting an interlaced information filtering (IIF) algorithm for linear MAS and then embedding the sigma-point transformation (SPT) that used in the sigma-point Kalman filters into the IIF architecture through a *statistical linear regression* methodology. The ISPIF enjoys both the effectiveness brought by the interlacement technique and the accuracies and flexibilities brought by the SPTs. Performance comparison of the ISPIF with the interlaced extended information filter is demonstrated through network localization simulations.

***Index Terms*—** distributed state estimation, multi-agent system, Kalman filter, information filtering, interlacement technique

1. INTRODUCTION

Distributed state estimation (DSE) of multi-agent systems (MAS) has attracted much attention in novel application-s such as network (or cooperative) localization [1–3], and multi-robot control and exploration [4–6]. Usually, for a MAS involving $|M|$ networked agents, the DSE aims at distributing the centralized state estimation into a set of $|M|$ decentralized procedures each corresponding to an agent, so that each agent only needs to estimate its own state on the basis of local observations and inter-agent communications. Compared with the centralized algorithms, the DSE can provide more flexibility, achieve more robustness, and require less computation and communication overheads [7, 8].

As a classical technique, Kalman filtering (KF) has been studied for DSE of MAS for many years [9–15]. In [9], an interlaced extended Kalman filter (IEKF) has been developed—which was lately used for self-localization of wireless networks [11]—by neglecting any coupling terms in the covariance matrix of the state estimation error and counteracting the errors so introduced by suitably "increasing" the noise covariance matrices. After that, an interlaced extended information filter (IEIF), which is essentially an IEKF expressed in terms of the inverse of the covariance matrix, has also been proposed in [12], due to the advantages of information filtering (IF) over the KF; the structure of the information estimation is computationally simpler than the KF update equations, and the IF is easily initialized compared to the KF without knowing *a priori* information of the state of the systems [16].

This paper proposes an interlaced sigma-point information filtering (ISPIF) algorithm for DSE of MAS. The motivation behind this paper comes from the fact that the sigma-point transformation (SPT) based *statistical linearization* methods adopted in the sigma-point Kalman filters (SP-KFs)[1] are more accurate and easier to implement than the first-order Taylor series expansion based *deterministic linearization* method adopted in the EKF. The remainder of this paper is organized as follows. Section II describes the system model of MAS. Section III derives the interlaced information filtering (IIF) for linear MAS. Section IV generalizes the IIF into nonlinear MAS by a *statistical linear regression* methodology. Section V presents the simulation results.

2. SYSTEM MODEL

Consider a generic nonlinear networked multi-agent system involving $|\mathcal{M}|$ agents, with the indexes set of all agents denoted by $\mathcal{M} = \{1, 2, \cdots, |\mathcal{M}|\}$ ($|\cdot|$ represents the cardinality of the set). The state of agent $m \in \mathcal{M}$ evolves as

$$\mathbf{x}_k^m = \mathbf{f}_k^m \left(\boldsymbol{\chi}_{k-1}^{\mathbf{f},m}, \boldsymbol{\xi}_{k-1}^m \right) \tag{1}$$

where $\mathbf{x}_k^m$ represents the state of agent m; $\mathbf{f}_k^m(\cdot)$ is the nonlinear process function; $\boldsymbol{\chi}_k^{\mathbf{f},m} = \{\mathbf{x}_k^{\mathbf{f},m,1}, \mathbf{x}_k^{\mathbf{f},m,2}, \cdots \mathbf{x}_k^{\mathbf{f},m,N_m^{\mathbf{f}}}\} \subseteq \boldsymbol{\chi}_k^{\mathcal{M}}$, in which $\boldsymbol{\chi}_k^{\mathcal{M}} = \{\mathbf{x}_k^m \mid m \in \mathcal{M}\}$ is the set consisting of the states of all $|\mathcal{M}|$ agents at time k, $\mathbf{x}_k^{\mathbf{f},m,1} \triangleq \mathbf{x}_k^m$, and the number of agent in $\boldsymbol{\chi}_k^{\mathbf{f},m}$ is denoted as $N_m^{\mathbf{f}} \geqq 1$; $\boldsymbol{\xi}_k^m \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}^m)$ is the corresponding process noise. We denote $\boldsymbol{\chi}_k^{\Omega,m} = \{\boldsymbol{\chi}_k^{\mathbf{f},n} \mid \boldsymbol{\chi}_k^{\mathbf{f},n} \ni \mathbf{x}_k^m, n \in \mathcal{M}\}$.

This work was supported in part by National Natural Science Foundation of China (61321061, 91338108).

[1] Besides the algorithms mentioned in [17], we extend the coverage to embrace the recently proposed Gauss-Hermite quadrature filter [18], cubature Kalmal filter [19], sparse-grid quadrature filter [20], and their variations.

At time k, there are $|\Phi_k^m|$ observations related to $\mathbf{x}_k^m$, we call these observations the local observations of agent m, of which the jth observation is described by

$$\mathbf{z}_k^{\mathbf{h},m,j} = \mathbf{h}_k^{m,j}\left(\boldsymbol{\chi}_k^{\mathbf{h},m,j}, \boldsymbol{\nu}_k^{m,j}\right) \tag{2}$$

where $\mathbf{h}_k^{m,j}(\cdot)$ is the corresponding observation function; $\boldsymbol{\chi}_k^{\mathbf{h},m,j} = \{\mathbf{x}_k^{\mathbf{h},m,j,1}, \mathbf{x}_k^{\mathbf{h},m,j,2}, \cdots \mathbf{x}_k^{\mathbf{h},m,j,N_m^{\mathbf{h},j}}\} \subseteq \boldsymbol{\chi}_k^{\mathcal{M}}$ is the states set consisting of all states that related to the jth observation, in which $\mathbf{x}_k^{\mathbf{h},m,j,1} \triangleq \mathbf{x}_k^m$ and $N_m^{\mathbf{h},j} \geqq 1$; $\boldsymbol{\nu}_k^{m,j} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}^{m,j})$. We denote $\boldsymbol{\chi}_k^{\Phi,m} = \{\boldsymbol{\chi}_k^{\mathbf{h},m,j} \mid j = 1, 2, \cdots, |\Phi_k^m|\}$. We say that two agents are connected if they can communicate directly with each other. The set of agents connected to a certain agent $m \in \mathcal{M}$ is called the *neighborhood* of agent m, with the corresponding states set denoted by $\boldsymbol{\chi}_k^{\mathcal{M},m}$. We assume that $\boldsymbol{\chi}_k^{\Omega,m} \cup \boldsymbol{\chi}_k^{\Phi,m} \subseteq \boldsymbol{\chi}_k^{\mathcal{M},m}$, and all process and observation noise vectors are uncorrelated.

At each time slot k, each agent needs to locally estimate the mean $\hat{\mathbf{x}}_{k|k}^m$ and covariance $\hat{\mathbf{p}}_{\mathbf{xx},k|k}^m$ of its state $\mathbf{x}_k^m$.

3. IIF FOR DSE OF LINEAR MAS

We first consider a linear MAS corresponding to (1) and (2):

$$\mathbf{x}_k^m = \sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \mathbf{a}_k^{m,i} \mathbf{x}_{k-1}^{\mathbf{f},m,i} + \mathbf{b}_{k-1}^m + \bar{\boldsymbol{\xi}}_{k-1}^m, \tag{3}$$

$$\mathbf{z}_k^{m,j} = \sum\nolimits_{i=1}^{N_m^{\mathbf{h},j}} \mathbf{c}_k^{m,j,i} \mathbf{x}_k^{\mathbf{h},m,j,i} + \mathbf{d}_k^{m,j} + \bar{\boldsymbol{\nu}}_k^{m,j} \tag{4}$$

where $\mathbf{a}_k^{m,i}$ and $\mathbf{c}_k^{m,j,i}$ are known matrices, $\mathbf{b}_k^m$ and $\mathbf{d}_k^{m,j}$ are deviations; $\bar{\boldsymbol{\xi}}_k^m \sim \mathcal{N}(\mathbf{0}, \bar{\mathbf{Q}}^m)$ and $\bar{\boldsymbol{\nu}}_k^{m,j} \sim \mathcal{N}(\mathbf{0}, \bar{\mathbf{R}}^{m,j})$.

We can see that the process and observation functions are not only related to the state of agent m of time k, but also related to the states of some of its neighbors of time k, thus, conventional Kalman/information filters cannot be straightforwardly adopted by agent m to estimate its state locally.

3.1. IKF

The interlacement technique was first developed in [9] to reduce the computational load of a nonlinear filter. It can be applied to DSE of linear MAS that modeled by (3)-(4), taking advantage of inter-agent communications.

3.1.1. State Prediction

Each agent $m \in \mathcal{M}$ broadcasts its state estimate of time $k-1$ to its neighbors, receives state estimates from its neighbors and uses them to predict its state as

$$\begin{aligned}\hat{\mathbf{x}}_{k|k-1}^m &= \sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \mathbf{a}_k^{m,i} \hat{\mathbf{x}}_{k-1|k-1}^{\mathbf{f},m,i} + \mathbf{b}_{k-1}^m \\ \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m &= \sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \mathbf{a}_k^{m,i} \hat{\mathbf{p}}_{\mathbf{xx},k-1|k-1}^{\mathbf{f},m,i} (\mathbf{a}_k^{m,i})^T + \bar{\mathbf{Q}}^m.\end{aligned} \tag{5}$$

3.1.2. Interlaced Observation Update

Each agent $m \in \mathcal{M}$ broadcasts its state predict to its neighbors, receives state predicts from its neighbors and uses them along with its local observations to update its state as

Start with

$$\begin{aligned}\hat{\mathbf{x}}_{k|k}^m &\leftarrow \hat{\mathbf{x}}_{k|k-1}^m \\ \hat{\mathbf{p}}_{\mathbf{xx},k|k}^m &\leftarrow \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m,\end{aligned} \tag{6}$$

Then, for $j = 1, 2, \cdots, |\Phi_k^m|$, **repeat:**

$$\begin{aligned}\mathbf{k}_k^m \leftarrow\ & \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m (\mathbf{c}_k^{m,j,1})^T \\ & \cdot \left(\mathbf{c}_k^{m,j,1} \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m (\mathbf{c}_k^{m,j,1})^T + \tilde{\mathbf{R}}_k^{m,j}\right)^{-1} \\ \hat{\mathbf{x}}_{k|k}^m \leftarrow\ & \hat{\mathbf{x}}_{k|k}^m + \mathbf{k}_k^m (\tilde{\mathbf{z}}_k^{m,j} - \mathbf{c}_k^{m,j,1} \hat{\mathbf{x}}_{k|k-1}^m) \\ \hat{\mathbf{p}}_{\mathbf{xx},k|k}^m \leftarrow\ & \hat{\mathbf{p}}_{\mathbf{xx},k|k}^m - \mathbf{k}_k^m \mathbf{c}_k^{m,j,1} \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m\end{aligned} \tag{7}$$

where $\tilde{\mathbf{R}}_k^{m,j} = \sum_{i=2}^{N_m^{\mathbf{h},j}} \mathbf{c}_k^{m,j,i} \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^{\mathbf{h},m,j,i} (\mathbf{c}_k^{m,j,i})^T + \bar{\mathbf{R}}^{m,j}$, and $\tilde{\mathbf{z}}_k^{m,j} = \mathbf{z}_k^{m,j} - \sum_{i=2}^{N_m^{\mathbf{h},j}} \mathbf{c}_k^{m,j,i} \hat{\mathbf{x}}_{k|k-1}^{\mathbf{h},m,j,i} - \mathbf{d}_k^{m,j}$.

The interlacement technique is introduced in (7), where agent m's local observation noise covariance matrices are "suitably" increased using the predicted state error covariance matrices of its neighbors.

Proof: The above update can be derived by: 1) augmenting all agent's state predicts into a vector with its mean $\hat{\mathbf{X}}_k = [(\hat{\mathbf{x}}_k^1)^T, (\hat{\mathbf{x}}_k^2)^T, \cdots, (\hat{\mathbf{x}}_k^{|\mathcal{M}|})^T]^T$ and covariance $\hat{\mathbf{P}}_{\mathbf{XX},k|k-1} = \mathrm{diag}\{\hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^1, \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^2, \cdots, \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^{|\mathcal{M}|}\}$, 2) using a KF to update it, and 3) decoupling this update into m parallel interlaced ones each corresponding to an agent.

3.2. IIF

Using the matrix inversion lemma, the IKF can be rewritten into an equivalent information form [8, 21], the interlaced information filtering (IIF), in terms of the information state vector $\boldsymbol{\psi} = \hat{\mathbf{p}}^{-1}\hat{\mathbf{x}}$ and the information matrix $\boldsymbol{\lambda} = \hat{\mathbf{p}}^{-1}$. Where, the state prediction equation (5) can be rewritten as

$$\begin{aligned}\boldsymbol{\lambda}_{k|k-1}^m &= \left(\sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \mathbf{a}_k^{m,i} \hat{\mathbf{p}}_{\mathbf{xx},k-1|k-1}^{\mathbf{f},m,i} (\mathbf{a}_k^{m,i})^T + \bar{\mathbf{Q}}^m\right)^{-1} \\ \boldsymbol{\psi}_{k|k-1}^m &= \boldsymbol{\lambda}_{k|k-1}^m \left(\sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \mathbf{a}_k^{m,i} \hat{\mathbf{x}}_{k-1|k-1}^{\mathbf{f},m,i} + \mathbf{b}_{k-1}^m\right),\end{aligned} \tag{8}$$

and the observation update (6)-(7) can be rewritten as

$$\begin{aligned}\boldsymbol{\lambda}_{k|k}^m &= \boldsymbol{\lambda}_{k|k-1}^m + \sum\nolimits_{j=1}^{|\Phi_k^m|} (\mathbf{c}_k^{m,j,1})^T (\tilde{\mathbf{R}}^{m,j})^{-1} \mathbf{c}_k^{m,j,1} \\ \boldsymbol{\psi}_{k|k}^m &= \boldsymbol{\psi}_{k|k-1}^m + \sum\nolimits_{j=1}^{|\Phi_k^m|} (\mathbf{c}_k^{m,j,1})^T (\tilde{\mathbf{R}}^{m,j})^{-1} \tilde{\mathbf{z}}_k^{m,j}.\end{aligned} \tag{9}$$

The architecture of the IIF is similar to that of IEIF [8], and a detailed discussion and performance analysis of the interlaced IF over the interlaced KF can be seen in [8].

4. ISPIF FOR DSE OF NONLINEAR MAS

This section develops the ISPIF by embedding a SPT-based statistical linear regression methodology into the IIF structure.

4.1. statistical linear regression

4.1.1. SPT

Consider a random variable $\mathbf{x}$ with mean $\bar{\mathbf{x}}$ and variance $\bar{\mathbf{p}}_{\mathbf{xx}}$, and a generally nonlinear function $\mathbf{g}(\cdot)$. Firstly, a set of N_{sp} weighted points $\{\mathcal{X}_l, \mathcal{W}_l\}_{l=1}^{N_{sp}}$ are deterministically chosen such that the mean $\hat{\mathbf{x}} = \sum_{l=1}^{N_{sp}} \mathcal{W}_l \mathcal{X}_l$ and variance $\hat{\mathbf{p}}_{\mathbf{xx}} = \sum_{l=1}^{N_{sp}} \mathcal{W}_l(\mathcal{X}_l - \hat{\mathbf{x}})(\mathcal{X}_l - \hat{\mathbf{x}})^T$ are equal to $\bar{\mathbf{x}}$ and $\bar{\mathbf{p}}_{\mathbf{xx}}$ (Details in the realization of SPT are different among the available SPKFs). These points are then propagated through the nonlinear function $\mathbf{g}(\cdot)$, yielding a set of transformed points $\mathcal{Y}_l = \mathbf{g}(\mathcal{X}_l)$. The weighted sigma points $\{\mathcal{X}_l, \mathcal{Y}_l, \mathcal{W}_l^m, \mathcal{W}_l^{(c)}\}_{l=1}^{N_{sp}}$ are then used to approximate the second-order statistics of $\mathbf{x}$ and $\mathbf{y}$ as:

$$\begin{aligned}\hat{\mathbf{y}} &= \sum\nolimits_{l=1}^{N_{sp}} \mathcal{W}_l \mathcal{Y}_l \\ \hat{\mathbf{p}}_{\mathbf{yy}} &= \sum\nolimits_{l=1}^{N_{sp}} \mathcal{W}_l [\mathcal{Y}_l - \bar{\mathbf{y}}][\mathcal{Y}_l - \bar{\mathbf{y}}]^{\mathbf{T}} \\ \hat{\mathbf{p}}_{\mathbf{xy}} &= \sum\nolimits_{l=1}^{N_{sp}} \mathcal{W}_l [\mathcal{X}_l - \bar{\mathbf{x}}][\mathcal{Y}_l - \bar{\mathbf{y}}]^{\mathbf{T}}.\end{aligned} \tag{10}$$

From the view of statistical linear regression [21], the SPT can be seen as propagating the mean $\bar{\mathbf{x}}$ and variance $\bar{\mathbf{p}}_{\mathbf{xx}}$ through the linearization

$$\mathbf{g}^{\text{lin}}(\mathbf{x}) = \mathbf{a}\mathbf{x} + \mathbf{b} + \mathbf{e}, \tag{11}$$

where $\mathbf{a} = \hat{\mathbf{p}}_{\mathbf{xy}}^T \hat{\mathbf{p}}_{\mathbf{xx}}^{-1}$, $\mathbf{b} = \hat{\mathbf{y}} - \mathbf{a}\hat{\mathbf{x}}$ and $\mathbf{e}$ is the linearization noise with mean $\hat{\mathbf{e}} = \mathbf{0}$ and variance $\hat{\mathbf{p}}_{\mathbf{e}} = \hat{\mathbf{p}}_{\mathbf{yy}} - \mathbf{a}\hat{\mathbf{p}}_{\mathbf{xx}}\mathbf{a}^{\mathbf{T}}$.

4.1.2. State-Space Linearization

We use the SPT to linearize (1) and (2). Noted that the expressions of (1) and (2) are similar, detailed linearization procedures are only demonstrated for (1).

Denoting $\mathbb{X}_{k-1}^m = [(\mathbf{x}_{k-1}^{\mathbf{f},m,1})^T, \cdots, (\mathbf{x}_{k-1}^{\mathbf{f},m,N_{\mathbf{f}}^m})^T]^T$ with mean and covariance

$$\begin{aligned}\hat{\mathbb{X}}_{k-1}^m &= [(\hat{\mathbf{x}}_{k-1|k-1}^{\mathbf{f},m,1})^T, \cdots, (\hat{\mathbf{x}}_{k-1|k-1}^{\mathbf{f},m,N_{\mathbf{f}}^m})^T]^T \\ \hat{\mathbf{P}}_{\mathbb{XX},k-1|k-1}^m &= \text{diag}\{\hat{\mathbf{p}}_{\mathbf{xx},k-1|k-1}^{\mathbf{f},m,1}, \cdots, \hat{\mathbf{p}}_{\mathbf{xx},k-1|k-1}^{\mathbf{f},m,N_{\mathbf{f}}^m}\},\end{aligned}$$

and $\mathbb{X}_{k-1}^a = [(\mathbb{X}_{k-1}^m)^T, (\boldsymbol{\xi}_{k-1}^m)^T]^T$ with mean and covariance

$$\begin{aligned}\hat{\mathbb{X}}_{k-1}^a &= [(\hat{\mathbb{X}}_{k-1}^m)^T, \mathbf{0}]^T \\ \hat{\mathbf{P}}_{\mathbb{XX},k-1|k-1}^a &= \text{diag}\{\hat{\mathbf{P}}_{\mathbb{XX},k-1|k-1}^m, \mathbf{Q}^m\},\end{aligned}$$

(1) can be rewritten into

$$\mathbf{x}_k^m = \mathbf{f}_k^m(\mathbb{X}_{k-1}^a). \tag{12}$$

Then, based on (11), (12) can be linearized as

$$\begin{aligned}\mathbf{x}_k^m &= [\check{\mathbf{a}}_{\mathbb{X},k-1}^m, \check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m]\mathbb{X}_{k-1}^a + \check{\mathbf{b}}_{k-1}^m + \check{\mathbf{e}}_{k-1}^m \\ &= \check{\mathbf{a}}_{\mathbb{X},k-1}^m \mathbb{X}_{k-1}^m + \check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m \boldsymbol{\xi}_{k-1}^m + \check{\mathbf{b}}_{k-1}^m + \check{\mathbf{e}}_{k-1}^m\end{aligned} \tag{13}$$

where

$$[\check{\mathbf{a}}_{\mathbb{X},k-1}^m, \check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m] = (\check{\mathbf{P}}_{\mathbf{x}\mathbb{X},k|k-1}^{m,a})(\hat{\mathbf{P}}_{\mathbb{XX},k-1|k-1}^a)^{-1} \tag{14}$$

$$\check{\mathbf{b}}_{k-1}^m = \hat{\mathbf{x}}_{k|k-1}^m - \check{\mathbf{a}}_{\mathbb{X},k-1}^m \hat{\mathbb{X}}_{k-1}^m. \tag{15}$$

Defining the total process noise

$$\check{\boldsymbol{\xi}}_{k-1}^m = \check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m \boldsymbol{\xi}_{k-1}^m + \check{\mathbf{e}}_{k-1}^m \tag{16}$$

with its corresponding variance given by

$$\begin{aligned}\check{\mathbf{Q}}_{k-1}^m &= \check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m \mathbf{Q}^m (\check{\mathbf{a}}_{\boldsymbol{\xi},k-1}^m)^T + \hat{\mathbf{P}}_{\mathbf{e},k-1}^m \\ &= \hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m - \check{\mathbf{a}}_{\mathbb{X},k-1}^m \hat{\mathbf{P}}_{\mathbb{XX},k-1|k-1}^m (\check{\mathbf{a}}_{\mathbb{X},k-1}^m)^T,\end{aligned} \tag{17}$$

a linearized form of (1) can be finally obtained, as:

$$\begin{aligned}\mathbf{x}_k^m &= \check{\mathbf{a}}_{\mathbb{X},k-1}^m \mathbb{X}_{k-1}^m + \check{\mathbf{b}}_{k-1}^m + \check{\boldsymbol{\xi}}_{k-1}^m \\ &= \sum\nolimits_{i=1}^{N_m^{\mathbf{f}}} \check{\mathbf{a}}_{\mathbb{X},k-1}^{m,i} \mathbf{x}_{k-1}^{\mathbf{f},m,i} + \check{\mathbf{b}}_{k-1}^m + \check{\boldsymbol{\xi}}_{k-1}^m.\end{aligned} \tag{18}$$

Where, in (14),(15), and(17), $\check{\mathbf{P}}_{\mathbf{x}\mathbb{X},k|k-1}^{m,a}$, $\hat{\mathbf{x}}_{k|k-1}^m$ and $\hat{\mathbf{p}}_{\mathbf{xx},k|k-1}^m$ are obtained by SPT as in (10).

Similarly, the observation function (2) can be linearized into the form of

$$\mathbf{z}_k^{m,j} = \sum\nolimits_{i=1}^{N_m^{\mathbf{h},j}} \check{\mathbf{c}}_{\mathbb{X},k}^{m,j,i} \mathbf{x}_k^{\mathbf{h},m,j,i} + \check{\mathbf{d}}_k^{m,j} + \check{\boldsymbol{\nu}}_k^{m,j} \tag{19}$$

4.2. ISPIF for DSE of MAS

At this point, naively substituting (1)-(2) with (18)-(19), the interlaced sigma-point information filtering (ISPIF) for DSE of MAS can be derived, as summarized in Algorithm 1.

Algorithm 1 : The ISPIF for DSE of Nonlinear MAS

For $m = 1, \cdots, |\mathcal{M}|$, **initialize**
  $\boldsymbol{\lambda}_{0|0}^m = (\hat{\mathbf{p}}_{0|0}^m)^{-1}$, $\boldsymbol{\psi}_{0|0}^m = (\hat{\mathbf{p}}_{0|0}^m)^{-1}\hat{\mathbf{x}}_{0|0}^m$.
End
For time $k = 1, 2, \cdots$
  For $m = 1, \cdots, |\mathcal{M}|$, **do in parallel**
    Broadcasts its last state estimate to its **neighbors**;
    Receive state estimates from its **neighbors**;
    linearizes (1) into (18) using the SPT;
    Predicts the state information using (8).
  End
  For $m = 1, \cdots, |\mathcal{M}|$, **do in parallel**
    Broadcasts its state predict to its **neighbors**;
    Receive state predicts from its **neighbors**;
    linearizes (2) into (19) using the SPT;
    Updates the state information using (9).
  End
End

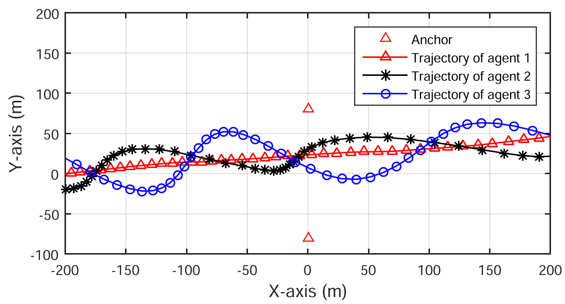

Fig. 1: Agent movements and anchor deployments.

5. AN APPLICATION EXAMPLE

Consider a 2 dimensional network localization scenario involving $|\mathcal{M}|$ mobile agents with the agent indexes set $\mathcal{M} = \{1, 2, \cdots, |\mathcal{M}|\}$. The state of agent $m \in \mathcal{M}$ consists of position and velocity, i.e., $\mathbf{x}_k^m \triangleq ((\mathbf{p}_k^m)^T, (\mathbf{v}_k^m)^T)^T$ with $\mathbf{p}_k^m \triangleq (x_k^m, y_k^m)^T$ and $\mathbf{v}_k^m \triangleq (v_{x,k}^m, v_{y,k}^m)^T$. We assume that agent 1 is a leader, whose positional states are used as a reference of other agent's movements. At time k, the state evolution of agent m is described by

$$\begin{pmatrix}\mathbf{p}_k^m\\ \mathbf{v}_k^m\end{pmatrix} = \begin{pmatrix}\mathbf{p}_{k-1}^m + \mathbf{v}_{k-1}^m + \frac{1}{2}\boldsymbol{\xi}_{k-1}^m\\ \mathbf{v}_{k-1}^m + \boldsymbol{\xi}_{k-1}^m\end{pmatrix}, \text{ if } m=1; \text{ else}$$
$$\begin{pmatrix}\mathbf{p}_k^m\\ \mathbf{v}_k^m\end{pmatrix} = \begin{pmatrix}(1-\alpha)\mathbf{p}_{k-1}^m + \alpha\mathbf{p}_{k-1}^1 + \mathbf{v}_{k-1}^m + \frac{1}{2}\boldsymbol{\xi}_{k-1}^m\\ \mathbf{v}_{k-1}^m + \alpha(\mathbf{p}_{k-1}^1 - \mathbf{p}_{k-1}^m) + \boldsymbol{\xi}_{k-1}^m\end{pmatrix} \quad (20)$$

where α is a positive scalar for velocity adjustment, and in all our simulations, $\alpha = 0.1$ and $\boldsymbol{\xi}_k^m \sim \mathcal{N}(\mathbf{0}, 0.1 \cdot \mathbf{I}_{2\times 2})$. From time $k = 1$ to 50, every node can communicate and range with its neighbors, and the distance measurement of agent $m \in \mathcal{M}$ relative to its neighboring agent (or anchor) n is given by $z_k^{m,n} = \|\mathbf{p}_k^m - \mathbf{p}_k^n\| + \nu_k^{m,n}$, in which $\nu_k^{m,n} \sim \mathcal{N}(0, 10)$.

In the first simulation, we compared the root mean square errors (RMSEs) and the corresponding error cumulative distribution functions (c.d.f.s) of the position estimates obtained by 4 algorithms, as shown in Fig.2, where CEIF and CUIF are centralized algorithms based on augmenting all states into a state vector and centralized implementing the EIF [16] or the UIF [21], while IEIF [12] and IUIF are decentralized algorithms, in which the IUIF is one of proposed ISPIF algorithms that realized using unscented transformation [22]. The position estimate error of agent m at time k is given by

$$e_k^m = \|\hat{\mathbf{p}}_k^m - \mathbf{p}_k^{m,\text{true}}\|,$$

and all RMSEs are obtained by averaging all agents and 1000 simulation runs, while the c.d.f.s are determined over states of all nodes, all simulation runs, and all time slot between 4 and 50. In all simulation runs, $|\mathcal{M}| = 3$ and two anchors are placed at (0,80) and (0 -80); all trajectories are generated according to (20) with the initial states $\mathbf{x}_k^{1,\text{true}} = (-210, 0, 7.88, 0)^T$, $\mathbf{x}_k^{2,\text{true}} = (-200, -20, 7.69, 0.38)^T$ and $\mathbf{x}_k^{3,\text{true}} = (-220, 30, 8, -0, 57)^T$; all agents can communicate and range with each other and the anchors; the initial state estimate of agent $m \in \mathcal{M}$ at time $k = 1$ is given by $\hat{\mathbf{x}}_1^m = \mathcal{N}(\mathbf{x}_1^{m,\text{true}}, \text{diag}\{60, 60, 10, 10\})$; and an illustration of agent movements and anchor deployments is given in Fig.1.

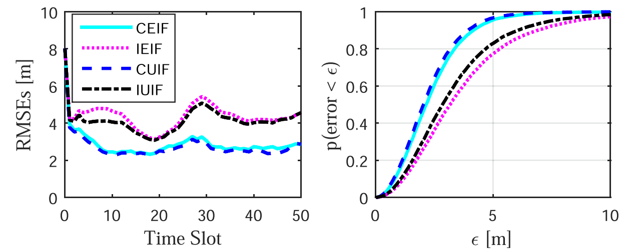

Fig. 2: The simulated RMSEs and c.d.f.s.

Table 1: Time cost for different agent numbers (seconds)

$\vert\mathcal{M}\vert$	5	10	20	40	80
CEIF	0.10	0.52	4.73	16.09	135.16
CUIF	0.28	1.02	7.22	23.80	208.54
IEIF	0.24	0.59	1.56	4.87	19.18
IUIF	0.32	0.80	1.85	6.77	25.45

In the second simulation, we compared the computation complexities under different agent numbers, where all simulations are carried out in MATLAB on an Intel(R) Core(TM)2 Duo P8400 CPU, and each agent can only communicate with no more than 5 nearest neighbors (including anchors, where 30 anchors are randomly scattered in the simulated region). Table.1 shows the total time costs of these algorithms.

We can see that in our simulations, 1) the IUIF outperforms the first Taylor series expansion-based IEIF in accuracy, while its computation complexity is similar to that of IEIF; 2) compared with their centralized counterparts, the decentralized IUIF and IEIF work at the cost of sacrificing accuracies, however, as the number of agents grows, the computation effectiveness of the IUIF and IEKF will gradually show up, not to mention the communication efficiencies.

6. CONCLUSION

This paper proposes a new ISPIF algorithm for DSE of MAS. The ISPIF is achieved by deriving an IIF for linear MAS and embedding the SPT based *statistically linearization* method that used in SPKFs into the IIF architecture. Thus, the proposed ISPIF can enjoy both the effectiveness of the interlacement technique and the accuracies and flexibilities of SPKF-s. What is more, since it inherits the advantages of SPKFs which outperform EKF in performance, it can outperform the IEIF in performance, which is partly validated by network localization simulations. The ISPIF can be applied to multi-robots/sensors networks, or other systems of large scale.

7. REFERENCES

[1] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, 2009.

[2] M. Win, A. Conti, S. Mazuelas, Y. Shen, W. Gifford, D. Dardari, and M. Chiani, "Network localization and navigation via cooperation," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 56–62, May 2011.

[3] F. Cattivelli and A. Sayed, "Distributed nonlinear kalman filtering with applications to wireless localization," in *Proc. 35th IEEE Int. Conf. Acoust., Speech, Signal Process*, Mar. 2010, pp. 3522–3525.

[4] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proc. IEEE*, vol. 94, no. 7, pp. 1325–1339, Jul. 2006.

[5] S. Bandyopadhyay and S.-J. Chung, "Distributed estimation using Bayesian consensus filtering," in *Proc. IEEE Amer. Contr. Conf.*, Portland, Oregon, USA, Jun. 2014, pp. 634–641.

[6] M. Bryson and S. Sukkarieh, "Architectures for cooperative airborne simultaneous localisation and mapping," *J. Intell. Robot. Syst.*, vol. 55, no. 4-5, pp. 267–297, 2009.

[7] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed kalman filtering based on consensus strategies," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 4, pp. 622–633, May 2008.

[8] F. Cattivelli and A. Sayed, "Diffusion strategies for distributed Kalman filtering and smoothing," *IEEE Trans. Autom. Control*, vol. 55, no. 9, pp. 2069–2084, Sep. 2010.

[9] L. Glielmo, R. Setola, and F. Vasca, "An interlaced extended Kalman filter," *IEEE Trans. Autom. Control*, vol. 44, no. 8, pp. 1546–1549, Aug. 1999.

[10] S. Roumeliotis and G. A. Bekey, "Distributed multirobot localization," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 781–795, Oct. 2002.

[11] A. Ribeiro, G. Giannakis, and S. Roumeliotis, "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.

[12] A. Gasparri and F. Pascucci, "An interlaced extended information filter for self-localization in sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 10, pp. 1491–1504, Oct. 2010.

[13] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *Proc. 46th IEEE Conf. Decision Control*, Dec 2007, pp. 5492–5498.

[14] J. Hu, L. Xie, and C. Zhang, "Diffusion kalman filtering based on covariance intersection," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 891–902, Feb 2012.

[15] U. Khan and J. Moura, "Distributing the Kalman filter for large-scale systems," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4919–4935, Oct. 2008.

[16] A. G. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. Boca Raton, FL: CRC press, 1998.

[17] R. Van Der Merwe, "Sigma-point kalman filters for probabilistic inference in dynamic state-space models," Ph.D. dissertation, OGI School of Science & Engineering, Oregon Health & Science University., Portland, OR, USA., 2004.

[18] I. Arasaratnam, S. Haykin, and R. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature," *Proc. IEEE*, vol. 95, no. 5, pp. 953–977, May 2007.

[19] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1254–1269, Jun. 2009.

[20] B. Jia, M. Xin, and Y. Cheng, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327–341, 2012.

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

[22] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.