DISTRIBUTED OPTIMAL CONSENSUS-BASED KALMAN FILTERING AND ITS RELATION TO MAP ESTIMATION

Shengdi Wang, Henning Paul, Armin Dekorsy

Department of Communications Engineering
University of Bremen, Germany
Email: {wang, paul, dekorsy}@ant.uni-bremen.de

ABSTRACT

In this paper, we address the problem of distributed state estimation, where a set of nodes are required to jointly estimate the state of a linear dynamic system based on sequential measurements. In our distributed scenario, all the nodes 1) are interested in the full state of the observed system and 2) pursue a consensus-based state estimate with high accuracy. We exploit the equivalent relation between the maximum-a-posteriori (MAP) estimation and the Kalman filter (KF) in the minimum mean square error (MMSE) sense under the Gaussian assumption. Utilizing this relation, a distributed Kalman filtering algorithm is derived, which ensures consensus-based state estimates among nodes and converges to the optimal central KF solution.

Index Terms— Distributed state estimation, consensus-based, Kalman filtering, MAP

1. INTRODUCTION

Distributed estimation is one of the most fundamental applications of cooperative networks like sensor networks. Using a network of nodes to collect discrete-time samples and to estimate a non-stationary state of a dynamic system is a popular research topic, as demanded by distributed tracking or control system. In this work, we assume that a dynamic system is monitored by a sensor network and each sensor intends to estimate the global system state making full use of all available measurements. Due to the different local noise levels, sensing capabilities and possibly types of sensors, the observations on the state will be totally different from sensor to sensor. To ensure an accurate state estimate and avoid the disadvantages of centralized processing, e.g., a single point of failure, an alternative distributed signal processing scheme is preferred, where each node performs peer-to-peer communication with its neighbors and processes data locally.

In previous works, many distributed Kalman filtering schemes were proposed in [1]-[7] to estimate the state of a linear dynamic system. The algorithms in [1]-[5] are all sub-optimal in the sense not achieving the central Kalman filter (KF) solution [8]. Nevertheless, the concept of micro-KF [1] is a good starting point to decompose the centralized scheme inspiring the subsequent works such as [6] and [7], in which different iterative consensus schemes, i.e., alternating direction method of multipliers and average consensus, are used to ensure the algorithms converge to the optimal solution, but the consensus is not enforced on state estimate directly.

Different from the other works on distributed Kalman filtering, we firstly consider the problem from another point of view by solving an equivalent maximum-a-posteriori (MAP) estimation problem. Next, to ensure a consensus-based solution, consensus constraints on state estimate are introduced. This kind of constrained optimization problem can be solved by numerical approaches in a distributed way, e.g., the method of multipliers. Analytical and experimental results show that our proposed algorithm asymptotically approaches the optimal centralized performance. Even with little consensus iterations, simulations show that our algorithm is still robust and provides a more accurate and consensus-based solution compared to other schemes using average consensus.

2. SYSTEM MODEL AND PROBLEM STATEMENT

We consider the following state space model of a discrete-time linear dynamic system

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{w}_k, \tag{1}$$

where $\mathbf{x}_k \in \mathbb{R}^m$ is the state vector of the system, $\mathbf{w}_k \in \mathbb{R}^m$ denotes the process noise and $\mathbf{A}_k \in \mathbb{R}^{m\times m}$ is the system matrix. Here the index k is the discrete time instant.

Now the state vector in (1) is observed by J sensors, which are connected through assumed error-free links, forming a time-invariant network topology described by a geometric graph $\mathcal{G} = \{\mathcal{J}, \mathcal{E}\}$. Here, $\mathcal{J} := \{1, \cdots, J\}$ and $\mathcal{E}$ denote the set of nodes and edges, respectively. These sensor nodes are assumed to have sensing, processing and communication capabilities. At each time k, a local measurement $\mathbf{y}_{j,k} \in \mathbb{R}^n$ is obtained by each node $j \in \mathcal{J}$, following

$$\mathbf{y}_{j,k} = \mathbf{H}_{j,k} \mathbf{x}_k + \mathbf{v}_{j,k}, \tag{2}$$

where $\mathbf{H}_{j,k} \in \mathbb{R}^{n\times m}$ and $\mathbf{v}_{j,k} \in \mathbb{R}^n$ denote the measurement matrix and measurement noise, respectively. $\mathbf{w}_k$ and $\mathbf{v}_{j,k}$ are

assumed as zero mean white Gaussian noise with covariances $\mathbf{Q}_k$ and $\mathbf{R}_{j,k}$, respectively, with the following relation

$$\mathrm{E}\left\{\begin{bmatrix}\mathbf{w}_k\\ \mathbf{v}_{j,k}\end{bmatrix}\begin{bmatrix}\mathbf{w}_p\\ \mathbf{v}_{i,p}\end{bmatrix}^{\mathrm{T}}\right\}=\begin{bmatrix}\mathbf{Q}_k & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_{j,k}\end{bmatrix}\delta_{ji}\delta_{kp}.$$

Here, δ_{cd} is the Kronecker delta, i.e., $\delta_{cd}=1$ only if $c=d$.

Then we further define the collective sensor measurements of the entire sensor network as $\mathbf{y}_k=[\mathbf{y}_{1,k}^{\mathrm{T}},\cdots,\mathbf{y}_{J,k}^{\mathrm{T}}]^{\mathrm{T}}\in\mathbb{R}^{nJ}$, the stacked block measurement matrix as $\mathbf{H}_k=[\mathbf{H}_{1,k}^{\mathrm{T}},\cdots,\mathbf{H}_{J,k}^{\mathrm{T}}]^{\mathrm{T}}\in\mathbb{R}^{nJ\times m}$ and the global measurement noise as $\mathbf{v}_k=[\mathbf{v}_{1,k}^{\mathrm{T}},\cdots,\mathbf{v}_{J,k}^{\mathrm{T}}]^{\mathrm{T}}\in\mathbb{R}^{nJ}$. The global measurement model of the whole network can be formulated as $\mathbf{y}_k=\mathbf{H}_k\mathbf{x}_k+\mathbf{v}_k$ by stacking all local measurements $\mathbf{y}_{j,k}$. The global measurement noise has covariance $\mathbf{R}_k=\mathrm{blkdiag}\,[\mathbf{R}_{1,k},\cdots,\mathbf{R}_{J,k}]\in\mathbb{R}^{nJ\times nJ}$. Let $\hat{\mathbf{x}}_{k|k-1}$ be the a-priori estimate, which denotes the estimate of the true state $\mathbf{x}_k$ given measurements up to time $k-1$, and let $\hat{\mathbf{x}}_{k|k}$ be the a-posteriori estimate containing the new measurement at time k. Then, we define the estimation errors $\mathbf{e}_{k|k-1}=\hat{\mathbf{x}}_{k|k-1}-\mathbf{x}_k$, $\mathbf{e}_{k|k}=\hat{\mathbf{x}}_{k|k}-\mathbf{x}_k$ and further calculate the corresponding error covariance matrices $\mathbf{P}_{k|k-1}=\mathrm{E}[\mathbf{e}_{k|k-1}\mathbf{e}_{k|k-1}^{\mathrm{T}}]$ and $\mathbf{P}_{k|k}=\mathrm{E}[\mathbf{e}_{k|k}\mathbf{e}_{k|k}^{\mathrm{T}}]$ related to $\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\mathbf{x}}_{k|k}$, respectively.

The main objective of this work is to make every node $j\in\mathcal{J}$ in the network achieve an accurate and consensus-based estimate $\hat{\mathbf{x}}_{j,k}$ on the entire true state $\mathbf{x}_k$ at each time k based on a set of all collective sensor measurements $\{\mathbf{y}_1,\cdots,\mathbf{y}_k\}$ from time 1 to time k, denoted by $\boldsymbol{\mathcal{Y}}_k$.

3. KALMAN FILTER

Based on the system model introduced in Section 2, the KF [8] is an optimal filter in the minimum mean-square-error (MMSE) sense by minimizing the trace of $\mathbf{P}_{k|k}$. It obtains the optimal estimate of state $\mathbf{x}_k$ given all measurements from 1 to k. The KF has two steps, namely, measurement update and time update. When initial values on $\hat{\mathbf{x}}_{0|-1}$ and $\mathbf{P}_{0|-1}$ are given, the algorithm runs recursively. Referring to [1], [5], [9], an equivalent information form of the conventional KF will be discussed here by defining the inverses of $\mathbf{P}_{k|k-1}$ and $\mathbf{P}_{k|k}$ as information matrices. The KF iterations in the information form are summarized as

Measurement update:

$$\mathbf{P}_{k|k}^{-1}=\mathbf{P}_{k|k-1}^{-1}+\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{H}_k \tag{3}$$

$$\hat{\mathbf{x}}_{k|k}=\hat{\mathbf{x}}_{k|k-1}+\mathbf{P}_{k|k}\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}(\mathbf{y}_k-\mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}) \tag{4}$$

Time update:

$$\hat{\mathbf{x}}_{k+1|k}=\mathbf{A}_k\hat{\mathbf{x}}_{k|k} \tag{5}$$

$$\mathbf{P}_{k+1|k}=\mathbf{A}_k\mathbf{P}_{k|k}\mathbf{A}_k^{\mathrm{T}}+\mathbf{Q}_k. \tag{6}$$

For the remaining of the paper, (3)-(6) can be regarded as the centralized KF (CKF) approach.

4. DISTRIBUTED CONSENSUS-BASED KALMAN FILTER (DCKF)

In this section, we firstly decompose the measurement update of CKF in a distributed way using the equivalent relation between KF and MAP estimation. Next the time update steps are involved considering the state dynamics.

4.1. Centralized MAP

According to [10], the KF can be also derived using Bayesian framework, leading to a MAP solution, i.e.,

$$\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}}=\arg\max_{\mathbf{x}_k}\ p(\mathbf{x}_k|\boldsymbol{\mathcal{Y}}_k). \tag{7}$$

This means we like to find the $\mathbf{x}_k$ which maximizes the probability density function (pdf) $p(\mathbf{x}_k|\boldsymbol{\mathcal{Y}}_k)$. Because the noise vectors $\mathbf{w}_k$ and $\mathbf{v}_k$ are both Gaussian, $\mathbf{x}_k$ and $\mathbf{y}_k$ can also be assumed Gaussian. Using Bayes rule and reconstructing the corresponding pdfs in Gaussian domain illustrated in [10], we can reformulate the general criterion (7) by minimizing the cost function with two Mahalanobis norms[1]:

$$\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}}=\arg\min_{\mathbf{x}_k}\left(\|\mathbf{y}_k-\mathbf{H}_k\mathbf{x}_k\|^2_{\mathbf{R}_k^{-1}}+\|\mathbf{x}_k-\hat{\mathbf{x}}_{k|k-1}\|^2_{\mathbf{P}_{k|k-1}^{-1}}\right). \tag{8}$$

Since the cost function (8) is (strictly) convex on $\mathbf{x}_k$, it can be minimized w.r.t. $\mathbf{x}_k$. Then we get the unique MAP solution as

$$\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}}=\left(\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{H}_k+\mathbf{P}_{k|k-1}^{-1}\right)^{-1}\cdot\left(\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{y}_k+\mathbf{P}_{k|k-1}^{-1}\hat{\mathbf{x}}_{k|k-1}\right), \tag{9}$$

which is equal to the a-posteriori state estimate of the KF in (4), i.e., $\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}}=\hat{\mathbf{x}}_{k|k}$, with error covariance matrix $\mathbf{P}_{k|k}=(\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{H}_k+\mathbf{P}_{k|k-1}^{-1})^{-1}$ [10]. Hence, the optimization problem in (8) is equivalent to the MMSE problem which the KF solves under the Gaussian assumption.

4.2. DCKF

Here, we define $\mathcal{N}_j$ is the set of neighboring nodes directly connected to node $j\in\mathcal{J}$. To solve the problem (8) in a distributed way, we decompose (8) into J local estimation problems coupled with the consensus constraints $\mathbf{x}_{j,k}=\mathbf{x}_{i,k}$, $\forall j\in\mathcal{J}, i\in\mathcal{N}_j$ on the state estimate and further conditions $\hat{\mathbf{x}}_{j,k|k-1}=\hat{\mathbf{x}}_{i,k|k-1}$, $\mathbf{P}_{j,k|k-1}=\mathbf{P}_{i,k|k-1}$. We firstly focus on the consensus constraints and discuss how to fulfill further conditions later. Thus, (8) can be equivalently written into

$$\{\hat{\mathbf{x}}_{j,k|k}|j\in\mathcal{J}\}=\arg\min_{\{\mathbf{x}_{j,k}|j\in\mathcal{J}\}}\sum_{j=1}^{J}f(\mathbf{x}_{j,k})\quad\text{s.t.}\quad\mathbf{x}_{j,k}=\mathbf{x}_{i,k},\ \forall j\in\mathcal{J}, i\in\mathcal{N}_j, \tag{10}$$

[1]The Mahalanobis norm is defined as a weighted norm: $\|\mathbf{a}\|^2_{\mathbf{G}}=\mathbf{a}^{\mathrm{T}}\mathbf{G}\mathbf{a}$.

with $f(\mathbf{x}_{j,k}) = \|\mathbf{y}_{j,k} - \mathbf{H}_{j,k}\mathbf{x}_{j,k}\|^2_{\mathbf{R}^{-1}_{j,k}} + \frac{1}{J}\|\mathbf{x}_{j,k} - \hat{\mathbf{x}}_{j,k|k-1}\|^2_{\mathbf{P}^{-1}_{j,k|k-1}}$. This kind of distributed consensus-based problem in (10) can be solved by different in-network-processing (INP) algorithms proposed in [11]-[13]. Here, we refer to the idea in [13] and apply the augmented Lagrangian (AL) method [14] to solve (10). By introducing Lagrange multipliers $\boldsymbol{\lambda}_{ji} \in \mathbb{R}^m$ to associate the constraints, the AL cost function can be built over all the nodes:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{j=1}^{J} \left(\frac{1}{2} f(\mathbf{x}_{j,k}) - \sum_{i \in \mathcal{N}_j} \boldsymbol{\lambda}_{ji}^{\mathrm{T}} (\mathbf{x}_{j,k} - \mathbf{x}_{i,k}) + \frac{1}{2\mu} \sum_{i \in \mathcal{N}_j} \|\mathbf{x}_{j,k} - \mathbf{x}_{i,k}\|^2 \right) = \sum_{j=1}^{J} \mathcal{L}_j(\boldsymbol{x}, \boldsymbol{\lambda}), \tag{11}$$

where $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ denote the sets of variables $\mathbf{x}_{j,k}$ and $\boldsymbol{\lambda}_{ji}$, respectively, and μ is a penalty parameter. Then, the AL function of the whole network $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ is split into the sum of J AL sub-functions $\mathcal{L}_j(\boldsymbol{x}, \boldsymbol{\lambda})$ which can be minimized locally w.r.t. $\mathbf{x}_{j,k}$ at each node $j \in \mathcal{J}$ by performing an inner loop at each time instant k. To enable the parallel update of variable $\mathbf{x}_{j,k}$ among all nodes, decoupling is needed. Here we force the newest update $\mathbf{x}^l_{j,k}$ at inner iteration l equal to the last update $\mathbf{x}^{l-1}_{i,k}$ at inner iteration $l-1$ from the neighboring nodes, i.e., $\mathbf{x}^l_{j,k} = \mathbf{x}^{l-1}_{i,k}, \forall j \in \mathcal{J}, i \in \mathcal{N}_j$. When $l \to \infty$, the consensus constraints $\mathbf{x}_{j,k} = \mathbf{x}_{i,k}$ can be fulfilled through the whole network. Then we calculate $\mathbf{x}^l_{j,k}$ by minimizing the convex local cost function $\mathcal{L}_j(\mathbf{x}_{j,k}, \mathbf{x}^{l-1}_{i,k}, \boldsymbol{\lambda}^{l-1}_{ji})$ with respect to $\mathbf{x}_{j,k}$. Further the gradient descent method is applied to deal with $\mathcal{L}_j(\boldsymbol{\lambda}_{ji}, \mathbf{x}^{l-1}_{j,k}, \mathbf{x}^{l-1}_{i,k}, \boldsymbol{\lambda}^{l-1}_{ji})$ on $\boldsymbol{\lambda}_{ji}$ to obtain the update of multipliers $\boldsymbol{\lambda}^l_{ji}$. The update equations of $\mathbf{x}^l_{j,k}$ and $\boldsymbol{\lambda}^l_{ji}$ are shown as follows at inner iteration l:

$$\mathbf{x}^l_{j,k} = \left(\mathbf{H}^{\mathrm{T}}_{j,k} \mathbf{R}^{-1}_{j,k} \mathbf{H}_{j,k} + \frac{1}{J} \mathbf{P}^{-1}_{j,k|k-1} + \frac{|\mathcal{N}_j|}{\mu} \mathbf{I} \right)^{-1} \cdot \left[\mathbf{H}^{\mathrm{T}}_{j,k} \mathbf{R}^{-1}_{j,k} \mathbf{y}_{j,k} + \frac{1}{J} \mathbf{P}^{-1}_{j,k|k-1} \hat{\mathbf{x}}_{j,k|k-1} + \sum_{i \in \mathcal{N}_j} \left(\frac{\mathbf{x}^{l-1}_{i,k}}{\mu} + \boldsymbol{\lambda}^{l-1}_{ji} \right) \right], \tag{12}$$

$$\boldsymbol{\lambda}^l_{ji} = \boldsymbol{\lambda}^{l-1}_{ji} - \frac{1}{\mu} \left(\mathbf{x}^{l-1}_{j,k} - \mathbf{x}^{l-1}_{i,k} \right), \ \forall j \in \mathcal{J}, i \in \mathcal{N}_j. \tag{13}$$

At each time k, we set the initial value of $\mathbf{x}^0_{j,k}$ to the previous a-priori state estimate $\hat{\mathbf{x}}_{j,k|k-1}$ and multiplier $\boldsymbol{\lambda}^0_{ji} = \mathbf{0}$. After update steps (12) and (13), each node $j \in \mathcal{J}$ will share the local temporary state estimate $\mathbf{x}^l_{j,k}$ to its neighbors preparing for the next update. The update of $\boldsymbol{\lambda}_{ji}$ can be performed locally without information exchange. If we set the number of inner iteration to L, the a-posteriori state estimate of each node $j \in \mathcal{J}$ is obtained by $\hat{\mathbf{x}}_{j,k|k} \leftarrow \mathbf{x}^L_{j,k}$. We have proved that the equivalent problem of (8) is (10), when $l \to \infty$ the following relation will be fulfilled

$$\hat{\mathbf{x}}_{j,k|k} = \hat{\mathbf{x}}_{i,k|k} = \hat{\mathbf{x}}^{\mathrm{MAP}}_{k|k} = \hat{\mathbf{x}}_{k|k}. \tag{14}$$

Next, we take state dynamics (1) into account. Similar to the time update steps (5) and (6) in CKF, in our approach, each node $j \in \mathcal{J}$ will perform a local prediction:

$$\hat{\mathbf{x}}_{j,k+1|k} = \mathbf{A}_k \hat{\mathbf{x}}_{j,k|k}, \tag{15}$$

$$\mathbf{P}_{j,k+1|k} = \mathbf{A}_k \mathbf{P}_{j,k|k} \mathbf{A}^{\mathrm{T}}_k + \mathbf{Q}_k. \tag{16}$$

Then we will discuss how to meet the conditions $\hat{\mathbf{x}}_{j,k+1|k} = \hat{\mathbf{x}}_{i,k+1|k}$ and $\mathbf{P}_{j,k+1|k} = \mathbf{P}_{i,k+1|k}$ for the decomposition of the centralized objective function at next time instant $k+1$. We define $\mathbf{S}^0_{j,k} = \mathbf{H}^{\mathrm{T}}_{j,k} \mathbf{R}^{-1}_{j,k} \mathbf{H}_{j,k}$ which is locally available at each node $j \in \mathcal{J}$. By transmitting $\mathbf{S}^{l-1}_{j,k} \in \mathbb{R}^{m \times m}$ to the neighbors $i \in \mathcal{N}_j$ at every inner iteration l, each node $j \in \mathcal{J}$ performs an average consensus algorithm [15] in an inner loop as

$$\mathbf{S}^l_{j,k} \ \leftarrow \ \mathbf{S}^{l-1}_{j,k} + \epsilon \sum_{i \in \mathcal{N}_j} \left(\mathbf{S}^{l-1}_{i,k} - \mathbf{S}^{l-1}_{j,k} \right) \tag{17}$$

to asymptotically approach the global average

$$\lim_{l \to \infty} \mathbf{S}^l_{j,k} = \frac{1}{J} \sum_{j=1}^{J} \mathbf{H}^{\mathrm{T}}_{j,k} \mathbf{R}^{-1}_{j,k} \mathbf{H}_{j,k}. \tag{18}$$

Here, ϵ in (17) is a step size in the range $0 < \epsilon < \frac{1}{D_{\max}}$, where $D_{\max}$ is the maximum degree of the network graph $\mathcal{G}$. The larger ϵ is, the faster convergence will be achieved. In the centralized form for the error covariance matrix (3), $\mathbf{H}^{\mathrm{T}}_k \mathbf{R}^{-1}_k \mathbf{H}_k = \sum_{j=1}^{J} \mathbf{H}^{\mathrm{T}}_{j,k} \mathbf{R}^{-1}_{j,k} \mathbf{H}_{j,k}$ holds. When $l \to \infty$, we can reformulate (3) using (18) and each node $j \in \mathcal{J}$ has the same error covariance matrix, identical to the central one

$$\mathbf{P}_{j,k|k} = \lim_{l \to \infty} \left(J \mathbf{S}^l_{j,k} + \mathbf{P}^{-1}_{j,k|k-1} \right)^{-1} = \mathbf{P}_{k|k}. \tag{19}$$

Here, $\mathbf{P}_{j,k|k-1} = \mathbf{P}_{i,k|k-1}$ holds, based on the calculation from the previous time instant $k-1$. Then according to (16), $\mathbf{P}_{j,k+1|k} = \mathbf{P}_{i,k+1|k}$ holds. When $l \to \infty$, $\hat{\mathbf{x}}_{j,k|k} = \hat{\mathbf{x}}_{i,k|k}$ is obtained via inner iterations on the state estimate. According to (15), asymptotically $\hat{\mathbf{x}}_{j,k+1|k} = \hat{\mathbf{x}}_{i,k+1|k}$ will be achieved.

Definitely, we can not perform an infinite number of inner consensus iterations. When only L inner iterations are processed, an approximate form of (19) is

$$\tilde{\mathbf{P}}_{j,k|k} = \left(J \mathbf{S}^L_{j,k} + \mathbf{P}^{-1}_{j,k|k-1} \right)^{-1} \ \approx \ \mathbf{P}_{k|k}, \tag{20}$$

and the state estimate $\hat{\mathbf{x}}_{j,k|k} = \mathbf{x}^L_{j,k}$ will also be an approximation of the central solution. Hence, when considering a limited number of inner iterations, DCKF is suboptimal.

Based on the derivation above, the DCKF algorithm can be summarized in Algorithm 1. Note that two kinds of consensus operations (12), (13) as well as (17) can be performed in parallel. DCKF converges to the central solution with an infinite number of inner iterations.

Algorithm 1 Distributed consensus-based KF (DCKF)

Initialization: for all $j \in \{1, ..., J\}$, $\hat{\mathbf{x}}_{j,0|-1} = \mathbf{x}_0$, $\mathbf{P}_{j,0|-1} = \mathbf{P}_0$,
for $k = 1, \cdots K$, sensor j **do**
  set initial values: $\mathbf{S}_{j,k}^0 = \mathbf{H}_{j,k}^{\mathrm{T}} \mathbf{R}_{j,k}^{-1} \mathbf{H}_{j,k}$, $\mathbf{x}_{j,k}^0 = \hat{\mathbf{x}}_{j,k|k-1}$, $\boldsymbol{\lambda}_{ji}^0 = \mathbf{0}$
  for $l = 1, \cdots, L$ **do**
    transmit the previous matrix $\mathbf{S}_{j,k}^{l-1}$ and the estimate $\mathbf{x}_{j,k}^{l-1}$ to neighbors $i \in \mathcal{N}_j$, then perform (12), (13) as well as (17) in parallel
  end for
  obtain the a-posteriori state estimate $\hat{\mathbf{x}}_{j,k|k} \leftarrow \mathbf{x}_{j,k}^L$ and calculate the error covariance matrix according to (20)
  predict for the next time instant $k+1$ by calculating (15) and (16)
end for

5. PERFORMANCE EVALUATION

In this section, we perform numerical simulations to evaluate the performance of DCKF and compare the results to CKF [8], information-weighted consensus filter (ICF) [7] and distributed Kalman filter algorithm 2 (DKF2) with consensus step on the estimate [2]. For the sake of fairness, we also process L average consensus iterations on estimate in DKF2. It can be shown that the communication overhead, i.e., transmitted scalars, of DCKF and ICF are the same at each time k, which is slightly larger than that of DKF2. Consider a 2-dimensional dynamic system with matrices $\mathbf{A}_k = [0.992, -0.1247; 0.1247, 0.992]$, $\mathbf{Q}_k = \mathbf{I}_2$, $\forall k$. We randomly deploy 20 sensors connected in a network topology shown in Fig. 1 (a), to observe the state vector. The measurement matrix $\mathbf{H}_{j,k}$ is chosen to be either $[1, 0]$ or $[0, 1]$ randomly, but it is ensured that every element of the state vector can be measured by sensors. The measurement noise covariance matrices $\mathbf{R}_{j,k}$ collapse to scalars of value $\sqrt{j}, j = 1, \cdots, J$. We evaluate the performance by calculating an approximation of the average mean square error (aMSE) among J nodes

$$\text{aMSE} = \frac{1}{J} \sum_{j=1}^{J} \mathrm{E}\{\|\mathbf{x}_k - \hat{\mathbf{x}}_{j,k|k}\|^2\}, \tag{21}$$

averaging over 1000 Monte Carlo experiments with random realizations of $\mathbf{w}_k$, $\mathbf{v}_{j,k}$ and $\mathbf{H}_{j,k}$. We set the average step size $\epsilon = 1/(D_{\max} + 0.1)$ and the penalty parameter $\mu = 3$.

Fig.1 (b) presents an example of aMSE performance for a specific time instant k. We observe that DCKF asymptotically approaches CKF with increasing inner iteration l. With large l, DCKF is able to reach the optimal CKF performance. When a limited number of inner iterations, e.g., $L = 20$, is performed at each time instant k, we get Fig.1 (c) and (d). Fig.1 (c) depicts the aMSE for different algorithms over J nodes w.r.t. time instant k. Compared to the other algorithms, DCKF has a better aMSE performance which is close to the centralized performance. Because DKF2 is always suboptimal [2], increasing consensus iteration will not help it converge to the central solution. Considering the transmitted scalars of different algorithms at each k, a small L indicates low communication overhead. Hence, Fig.1 (c) can also reflect that to reach the same aMSE performance, DCKF can save communication overhead compared to ICF. Fig.1 (d) shows the steady-state MSE over different nodes. It can not only confirm the better MSE performance of DCKF but also illustrate that DCKF has consensus-based state estimates among nodes compared to the other distributed algorithms.

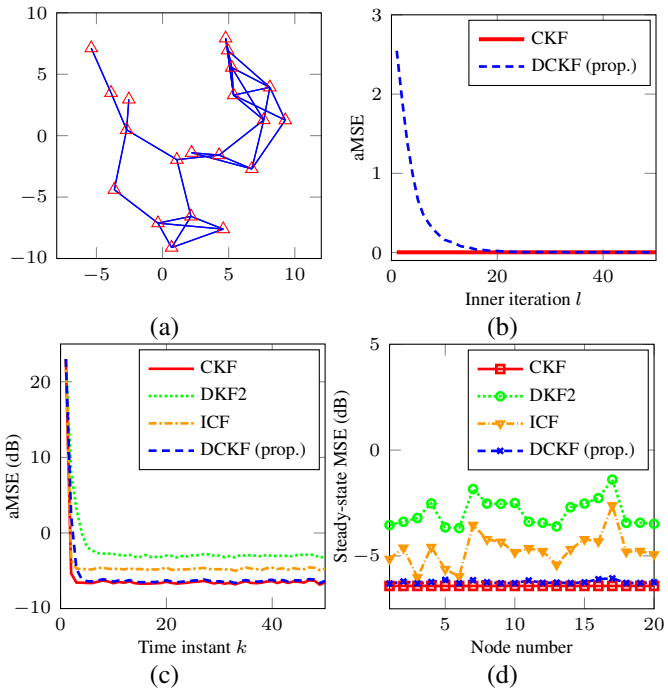

Fig. 1. (a) A random network topology, $D_{\max} = 6$; (b) An example of aMSE performance for our proposed DCKF by varying inner iteration l; (c) aMSE over the whole network as a function of time instant k when $L = 20$; (d) Steady-state MSE for different nodes after time instant 100 when $L = 20$

6. CONCLUSION

In this paper, we propose a distributed consensus-based Kalman filtering algorithm for the state estimation in linear dynamic systems, utilizing the equivalent relation between KF and MAP estimation. It can be shown analytically and experimentally that our proposed DCKF converges to the optimal central solution. With a limited number of inner iterations, DCKF has great potential to outperform the other state of the art algorithms, since it achieves a better MSE performance and consensus-based state estimates among nodes. The choice of penalty parameter and network topology will influence the convergence behavior of DCKF, so further investigations need to be performed to finalize the analysis.

7. REFERENCES

[1] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proc. of the 44th IEEE Conf. Decision Control and 8th European Control Conf.*, 2005, pp. 8179–8184.

[2] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proc. of the 46th IEEE Conf. on Decision and Control*, 2007, pp. 5492–5498.

[3] R. Olfati-Saber, "Kalman-consensus filter: optimality, stability, and performance," in *Proc. of the 48h IEEE Conf. on Decision and Control and 28th Chinese Control Conf.*, 2009, pp. 7036–7042.

[4] P. Alriksson and A. Rantzer, "Distributed Kalman filtering using weighted averaging," in *Proc. of the 17th International Symposium on Mathematical Theory of Networks and Systems*, 2006, pp. 2445–2450.

[5] F. S. Cattivelli and A. H. Sayed, "Diffusion strategies for distributed Kalman filtering and smoothing," *IEEE Trans. on Automatic Control*, vol. 55, no. 9, pp. 2069–2084, 2010.

[6] I. D. Schizas, G. B. Giannakis, S. I. Roumeliotis, and A. Ribeiro, "Anytime optimal distributed Kalman filtering and smoothing," in *IEEE/SP 14th Workshop on Statistical Signal Processing, SSP'07.*, 2007, pp. 368–372.

[7] A. T. Kamal, J. A. Farrell, and A. K. Roy-Chowdhury, "Information weighted consensus," in *Proc. of the 51st IEEE Conf. on Decision and Control*, 2012, pp. 2732–2737.

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[9] U. A. Khan and J. M. F. Moura, "Distributing the Kalman filter for large-scale systems," *IEEE Trans. on Signal Processing*, vol. 56, no. 10, pp. 4919–4935, 2008.

[10] Z. Chen, "Bayesian filtering: from Kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[11] H. Paul, J. Fliege, and A. Dekorsy, "In-network-processing: distributed consensus-based linear estimation," *IEEE Communications Letters*, vol. 17, no. 1, pp. 59–62, 2013.

[12] G. Xu, H. Paul, D. Wübben, and A. Dekorsy, "Fast distributed consensus-based estimation for cooperative wireless sensor networks," in *18th International ITG Workshop on Smart Antennas*, 2014.

[13] G. Xu, H. Paul, D. Wübben, and A. Dekorsy, "Distributed augmented Lagrangian method for cooperative estimation in small cell networks," in *Proc. of the 10th International ITG Conf. on Systems, Communications and Coding*, 2015.

[14] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Science+Business Media, 2nd edition, 2006.

[15] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.