# Distributed Consensus-Based Extended Kalman Filtering: A Bayesian Perspective

Shengdi Wang and Armin Dekorsy
*Department of Communications Engineering*
*University of Bremen*
Bremen, Germany
{wang, dekorsy}@ant.uni-bremen.de

***Abstract*—In this paper, we study the distributed state estimation problem where a set of nodes cooperatively estimate the hidden state of a nonlinear dynamic system based on sequential observations. As a common approach to solve this problem, the extended Kalman filter (EKF) is considered from a Bayesian perspective. After linearizing the state-space model using the first-order Taylor series, we construct an equivalent maximum-a-posteriori (MAP) estimation problem under linear Gaussian assumptions coupled with a consensus constraint. The consensus-based MAP problem is solved distributedly by the alternating direction method of multiplier (ADMM). The resulting distributed algorithm ensures robust consensus-based state estimates among nodes and is able to converge to the central solution.**

***Index Terms*—Nonlinear state estimation, distributed extended Kalman filter, maximum-a-posteriori estimation, consensus optimization**

## I. Introduction

Distributed state estimation of dynamic systems draws much attention in different domains such as electric power grid monitoring, environmental field estimation as well as tracking and localization in sensor networks [1], where no central processing unit is deployed. Distributed processing brings many benefits compared to the centralized one, e.g., the robustness and efficiency of the whole system are increased by overcoming the presence of single point failure and congestion of massive data.

For linear dynamic systems, the Kalman filter (KF) [2], which provides an optimal state estimation, is usually derived in the minimum-mean-square-error (MMSE) sense with the assumption of Gaussian uncertainties. However, most systems in real world applications are nonlinear. As a nonlinear version of KF, the idea of the extended Kalman filter (EKF) [3] is to use the first-order Taylor series to linearly approximate nonlinear functions in the state-space model. An alternative is the unscented Kalman filter (UKF), where the unscented transform is applied to approximate the mean and covariance of the Gaussian formed state density [4]. When it is not sufficient to use Gaussian distribution to approximate the state density, the sampling method-based particle filter (PF) [5] can be applied, but the computation complexity is increased.

In this work, we assume a dynamic system which is not highly nonlinear. Thus, the EKF is sufficient and widely applied. Here, we investigate it in a distributed scenario where each node in a network desires an accurate consensus-based estimate on the entire system state variables. This consensus property offers a flexibility to conduct further transmission or control. Most of the state of the art (SotA) distributed EKF approaches apply average consensus (AC) scheme [6] to deal with different terms in the update equations of the EKF, which are summarized and analyzed in [7]. Different from them, our previous work [8] considers the KF in another viewpoint by constructing an equivalent consensus optimization problem. The designed algorithm in [8] offers an accurate consensus-based solution with low communication overhead. Here, we extended the idea of our previous work to the nonlinear case by considering the EKF in a Bayesian perspective. Similar to the EKF, we first approximate the nonlinear model at specific working points using the first-order Taylor series. Then, by exploiting the equivalent relation between the KF and the maximum-a-posterior (MAP) estimation under linear Gaussian assumptions, we construct the general consensus optimization problem which is solved by the alternating direction method of multiplier (ADMM) in an efficient and distributed way. The resulting distributed algorithm is applied to the target tracking application in sensor networks. Analytical and simulated results illustrate that the proposed algorithm is able to converge to the central solution and offers robust consensus-based state estimates among nodes with low communication overhead.

## II. System Model

We consider a discrete-time nonlinear dynamic system with the hidden state vector $\mathbf{x}_k \in \mathbb{R}^m$. At each time instant $k$, $\mathbf{x}_k$ is observed by $J$ nodes in a network. This network is described by a time-invariant geometric undirected graph $\mathcal{G} = \{\mathcal{J}, \mathcal{E}\}$, in which $\mathcal{J} := \{1, \cdots, J\}$ and $\mathcal{E}$ denote the set of nodes and edges, respectively. We assume that the network is connected which means one node is able to reach any other node by multi-hop. All these nodes are homogeneous in terms of processing and communication capability. The inter-node communication links are assumed to be ideal. The dynamic process and the local observation of each node $j \in \mathcal{J}$ can be described using the following state-space model:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{G}_k\mathbf{w}_k, \tag{1}$$

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

where the state transition function $\mathbf{f}_k(\cdot) : \mathbb{R}^m \to \mathbb{R}^m$ and observation function $\mathbf{h}_{j,k}(\cdot) : \mathbb{R}^m \to \mathbb{R}^n$ are both

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

differentiable and possibly nonlinear. Here, $\mathbf{G}_k \in \mathbb{R}^{m\times p}$ is a matrix. The vectors $\mathbf{w}_k \in \mathbb{R}^p$ and $\mathbf{v}_{j,k} \in \mathbb{R}^n$ denote the process and observation noise, respectively. Both noise vectors are assumed to be zero mean Gaussian distributed with covariance

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

where $\delta_{cd}$ is Kronecker delta , i.e. $\delta_{cd} = 1$ only if $c = d$. The initial state is assumed to be $\mathbf{x}_0 = \mathcal{N}(\mathbf{m}_0, \mathbf{P}_0)$.

To make full use of all information in the network, we define a collection of the observations of the entire network as $\mathbf{y}_k = [\mathbf{y}_{1,k}^{\mathrm{T}}, \cdots, \mathbf{y}_{J,k}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nJ}$. Thus, the global nonlinear observation function w.r.t. the state $\mathbf{x}_k$ is defined by $\mathbf{h}_k(\mathbf{x}_k) = [\mathbf{h}_{1,k}(\mathbf{x}_k)^{\mathrm{T}}, \cdots, \mathbf{h}_{J,k}(\mathbf{x}_k)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nJ}$. Similarly, the global observation noise is $\mathbf{v}_k = [\mathbf{v}_{1,k}^{\mathrm{T}}, \cdots, \mathbf{v}_{J,k}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nJ}$ with the block diagonal covariance matrix $\mathbf{R}_k = \mathrm{blkdiag}[\mathbf{R}_{1,k}, \cdots, \mathbf{R}_{J,k}] \in \mathbb{R}^{nJ\times nJ}$. Thus, the global observation model is

$$\mathbf{y}_k \;=\; \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k \tag{3}$$

by stacking the $J$ nonlinear equations in (2).

The key objective is to infer the hidden state $\mathbf{x}_k$ at each time $k$ based on a set of available observations $\mathbf{y}_{1:k} := \{\mathbf{y}_1, \cdots, \mathbf{y}_k\}$. From a Bayesian viewpoint, at time $k$ we want to recursively estimate the predictive distribution (density) $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ given observations up to $k-1$ and the filtering distribution (density) $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ given observation up to $k$, as a description of general Bayesian filtering (BF) [9].

## III. EKF and Equivalent Problem Formulation

### A. Overview of EKF

With the global system model (1) and (3), in the EKF the filtering densities are assumed to be approximated by Gaussian distributions [10], i.e., $p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \mathcal{N}(\mathbf{x}_k|\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$ with filtered mean $\hat{\mathbf{x}}_{k|k}$ and covariance matrix $\mathbf{P}_{k|k}$. This evolution of Gaussian approximations is formed by linearizing the nonlinear function using the first-order Taylor series at current best state estimate, i.e., $\hat{\mathbf{x}}_{k|k-1} = \mathrm{E}\{\mathbf{x}_k|\mathbf{y}_{1:k-1}\}$ and $\hat{\mathbf{x}}_{k|k} = \mathrm{E}\{\mathbf{x}_k|\mathbf{y}_{1:k}\}$ in prediction and filtering step, respectively. Here, we use $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ to express the mean and covariance matrix of Gaussian approximated predictive distribution, i.e., $p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \approx \mathcal{N}(\mathbf{x}_k|\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$. Thus, the linearized state-space model at time $k$ is

$$\mathbf{x}_{k+1} \;\approx\; \mathbf{F}_k\mathbf{x}_k + \mathbf{G}_k\mathbf{w}_k + \mathbf{u}_k, \tag{4}$$

$$\bar{\mathbf{y}}_k \;\approx\; \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k \tag{5}$$

with Jacobian matrices $\mathbf{F}_k \in \mathbb{R}^{m\times m}$ and $\mathbf{H}_k \in \mathbb{R}^{nJ\times m}$:

$$\mathbf{F}_k = \left.\frac{\partial \mathbf{f}_k}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k}}, \quad \mathbf{H}_k = \left.\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k-1}} \tag{6}$$

as well as the deterministic input $\mathbf{u}_k$ and the reformulated observation vector $\bar{\mathbf{y}}_k$ defined by:

$$\mathbf{u}_k \;=\; \mathbf{f}_k(\hat{\mathbf{x}}_{k|k}) - \mathbf{F}_k\hat{\mathbf{x}}_{k|k}, \tag{7}$$

$$\bar{\mathbf{y}}_k \;=\; \mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}) + \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}. \tag{8}$$

Based on the linearized model (4) and (5), the KF is derived correspondingly [3]. With a proper initialization, the update equations of the EKF at each time $k$ are as follows [7]:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{u}_{k-1} = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}), \tag{9}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}_{k-1}^{\mathrm{T}} + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^{\mathrm{T}}, \tag{10}$$

$$\mathbf{P}_{k|k} = \left(\mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\mathbf{H}_k\right)^{-1}, \tag{11}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k}\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}(\bar{\mathbf{y}}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}) \tag{12}$$

with the Jacobian matrices (6) and modified observation (8). Note that (9)-(12) indicate the updates of the centralized EKF (CEKF) using complete observation data set of the network.

### B. Equivalent Optimization Problem

To infer the state based on the observation data set, besides the MMSE criterion, the maximum-a-posteriori estimation

$$\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}} = \arg\max_{\mathbf{x}_k}\; p(\mathbf{x}_k|\mathbf{y}_{1:k}) \tag{13}$$

can be conducted to obtain an filtering estimate in an optimal sense. Based on the Bayes rule, the maximization problem in (13) can be rewritten into

$$\max_{\mathbf{x}_k}\; p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \tag{14}$$

with the likelihood $p(\mathbf{y}_k|\mathbf{x}_k)$ and the prior $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ [9]. Thanks to the linearization and Gaussian approximations in the EKF, both densities in (14) are approximated by Gaussian distributions and the determination of their means and covariances is straightforward with linearized system model. Thus, (14) is further derived in Gaussian form resulting in a centralized minimization problem as

$$\min_{\mathbf{x}_k}\left(\|\bar{\mathbf{y}}_k - \mathbf{H}_k\mathbf{x}_k\|_{\mathbf{R}_k^{-1}}^2 + \|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}\|_{\mathbf{P}_{k|k-1}^{-1}}^2\right), \tag{15}$$

where the two norms are defined by $\|\mathbf{a}\|_{\mathbf{B}} = \sqrt{\mathbf{a}^{\mathrm{T}}\mathbf{B}\mathbf{a}}$. By minimizing the convex objective function in (15) w.r.t. $\mathbf{x}_k$, the MAP estimate is obtained as

$$\hat{\mathbf{x}}_{k|k}^{\mathrm{MAP}} = \mathbf{P}_{k|k}\left(\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\bar{\mathbf{y}}_k + \mathbf{P}_{k|k-1}^{-1}\hat{\mathbf{x}}_{k|k-1}\right). \tag{16}$$

Under the linear Gaussian assumptions, the MMSE and the MAP estimation are identical and share the same form of solution [11]. Hence, (16) is identical to the state estimate (12) of CEKF with the same covariance matrix (11) [9].

## IV. Distributed Consensus-Based Extended Kalman Filter (DCEKF)

Firstly, we consider the prediction step of DCEKF. At each node $j \in \mathcal{J}$, the nonlinear model is linearized locally w.r.t. the node specific state estimate at each time $k$. Thus, the predicted mean and covariance matrix of the state at node $j$ are

$$\hat{\mathbf{x}}_{j,k|k-1} = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{j,k-1|k-1}), \tag{17}$$

$$\mathbf{P}_{j,k|k-1} = \mathbf{F}_{j,k-1}\mathbf{P}_{j,k-1|k-1}\mathbf{F}_{j,k-1}^{\mathrm{T}} + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^{\mathrm{T}} \tag{18}$$

with Jacobian matrix $\mathbf{F}_{j,k-1}$ at $\hat{\mathbf{x}}_{j,k-1|k-1}$. By initializing $\hat{\mathbf{x}}_{j,0|0} = \mathbf{m}_0$, $\mathbf{P}_{j,0|0} = \mathbf{P}_0$ for all $j \in \mathcal{J}$, first predictions are identical over the network.

Next for the filtering step, the centralized problem (15) can be decomposed into the summation of local parallel minimization problems by introducing a consensus constraint to guarantee an agreement on state estimate at time $k$ as

$$\min_{\{\mathbf{x}_{j,k}|j\in\mathcal{J}\}} \sum_{j=1}^{J} g_j(\mathbf{x}_{j,k}) \quad \text{s.t.} \quad \mathbf{x}_{j,k} = \mathbf{x}_{i,k},\ \forall j \in \mathcal{J}, i \in \mathcal{N}_j \tag{19}$$

with $g_j(\mathbf{x}_{j,k}) = \|\bar{\mathbf{y}}_{j,k} - \mathbf{H}_{j,k}\mathbf{x}_{j,k}\|^2_{\mathbf{R}^{-1}_{j,k}} + \frac{1}{J}\|\mathbf{x}_{j,k} - \hat{\mathbf{x}}_{j,k|k-1}\|^2_{\mathbf{P}^{-1}_{j,k|k-1}}$. The $\mathcal{N}_j$ denotes a set of neighboring nodes (neighbors) of node $j$. Note that, this decomposition is under the condition that $\hat{\mathbf{x}}_{j,k|k-1}$ and $\mathbf{P}_{j,k|k-1}$ for all nodes $j \in \mathcal{J}$ are identical to the central ones, i.e., $\hat{\mathbf{x}}_{j,k|k-1} = \hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{j,k|k-1} = \mathbf{P}_{k|k-1}$. Later we will explain when this condition is fulfilled.

The constrained optimization problem (19) can be solved in a distributed fashion using the ADMM algorithm [12]. Here, we refer to the distributed consensus-based estimation (DiCE) algorithm [13] due to its good performance with a convergence guarantee towards a central solution [14]. In (19), $\mathbf{x}_{j,k}$ and $\mathbf{x}_{i,k}$ in the consensus constraint are directly coupled and cannot be updated in parallel. To enable a distributed processing, an auxiliary variable $\mathbf{z}_{j,k}$ at node $j$ is introduced. Then the constraint pairs $\mathbf{x}_{j,k} = \mathbf{z}_{j,k}, \mathbf{x}_{j,k} = \mathbf{z}_{i,k}$ are constructed and further merged into $\mathbf{x}_{j,k} = \mathbf{z}_{i,k}, i \in \mathcal{N}_j^+$ with $\mathcal{N}_j^+ := \mathcal{N}_j \cup \{j\}$. To associate this modified constraint with the objective function in (19), the augmented Lagrangian (AL) method [15] is applied and the centralized AL cost function can be further decomposed into a summation of local convex cost functions:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}) = \sum_{j=1}^{J} \Bigg[\frac{1}{2} g_j(\mathbf{x}_{j,k}) - \sum_{i\in\mathcal{N}_j^+} \boldsymbol{\lambda}_{ji}^{\mathrm{T}}(\mathbf{x}_{j,k} - \mathbf{z}_{i,k}) + \frac{1}{2\mu}\sum_{i\in\mathcal{N}_j^+} \|\mathbf{x}_{j,k} - \mathbf{z}_{i,k}\|^2\Bigg] = \sum_{j=1}^{J} \mathcal{L}_j(\mathbf{x}_{j,k}, \boldsymbol{z}, \boldsymbol{\lambda}). \tag{20}$$

Here $\boldsymbol{\lambda}_{ji} \in \mathbb{R}^m$ is a Lagrange multiplier at node $j$ and $\mu$ is a scalar penalty parameter. To obtain a state estimate $\hat{\mathbf{x}}_{j,k|k}$ at each time $k$, an inner-consensus iteration denoted by $l$ is proceed with initialization $\mathbf{z}^0_{j,k} = \hat{\mathbf{x}}_{j,k|k-1}, \boldsymbol{\lambda}^0_{ji} = \mathbf{0}, \forall j \in \mathcal{J}$. At each node $j$, in the $l$-th inner iteration, $\mathbf{x}^l_{j,k}$ is obtained by solving $\partial\mathcal{L}_j(\mathbf{x}_{j,k}, \mathbf{z}^{l-1}_{i,k}, \boldsymbol{\lambda}^{l-1}_{ji})/\partial\mathbf{x}_{j,k} = \mathbf{0}$. Similarly, to update $\mathbf{z}^l_{j,k}$, the cost function (20) can be rewritten w.r.t. $\mathbf{z}_{j,k}$ by reorganizing indices $j$ and $i$ [13] and then $\partial\mathcal{L}_j^{'}(\mathbf{z}_{j,k}, \mathbf{x}^l_{i,k}, \boldsymbol{\lambda}^{l-1}_{ij})/\partial\mathbf{z}_{j,k} = \mathbf{0}$ is solved. Further, to calculate $\boldsymbol{\lambda}^l_{ji}$, the gradient method is used to deal with $\mathcal{L}_j(\boldsymbol{\lambda}_{ji}, \mathbf{x}^l_{j,k}, \mathbf{z}^l_{i,k}, \boldsymbol{\lambda}^{l-1}_{ji})$ on $\boldsymbol{\lambda}_{ji}$ [12]. In this way, the updates of these variables in the $l$-th iteration become

$$\mathbf{x}^l_{j,k} = \Big(\mathbf{H}^{\mathrm{T}}_{j,k}\mathbf{R}^{-1}_{j,k}\mathbf{H}_{j,k} + \frac{1}{J}\mathbf{P}^{-1}_{j,k|k-1} + \frac{|\mathcal{N}_j+1|}{\mu}\mathbf{I}\Big)^{-1} \times \Bigg[\mathbf{H}^{\mathrm{T}}_{j,k}\mathbf{R}^{-1}_{j,k}\bar{\mathbf{y}}_{j,k} + \frac{1}{J}\mathbf{P}^{-1}_{j,k|k-1}\hat{\mathbf{x}}_{j,k|k-1} + \sum_{i\in\mathcal{N}_j^+}\Big(\frac{\mathbf{z}^{l-1}_{i,k}}{\mu} + \boldsymbol{\lambda}^{l-1}_{ji}\Big)\Bigg], \tag{21}$$

$$\mathbf{z}^l_{j,k} = \frac{\mu}{|\mathcal{N}_j+1|}\sum_{i\in\mathcal{N}_j^+}\Big(\frac{1}{\mu}\mathbf{x}^l_{i,k} - \boldsymbol{\lambda}^{l-1}_{ij}\Big), \tag{22}$$

$$\boldsymbol{\lambda}^l_{ji} = \boldsymbol{\lambda}^{l-1}_{ji} - \frac{1}{\mu}\left(\mathbf{x}^l_{j,k} - \mathbf{z}^l_{i,k}\right)\ \forall j \in \mathcal{J}, i \in \mathcal{N}_j^+. \tag{23}$$

During each inner-consensus update, each node transmits the estimate $\mathbf{x}_{j,k}$ and the auxiliary variable $\mathbf{z}_{j,k}$ to its neighbors. Then, all Lagrange multipliers can be calculated locally. When $l \to \infty$, the filtered state estimate $\hat{\mathbf{x}}_{j,k|k} \leftarrow \mathbf{x}^{\infty}_{j,k|k}$ for each node $j \in \mathcal{J}$ converges to the centralized MAP estimate which is identical to the CEKF solution, i.e., $\hat{\mathbf{x}}_{j,k|k} = \hat{\mathbf{x}}^{\mathrm{MAP}}_{k|k} = \hat{\mathbf{x}}_{k|k},\ \forall j \in \mathcal{J}$ is fulfilled.

In the following, we calculate the covariance matrix in the filtering step at node $j \in \mathcal{J}$ using

$$\mathbf{P}_{j,k|k} = \Big(\sum_{j=1}^{J}\mathbf{H}^{\mathrm{T}}_{j,k}\mathbf{R}^{-1}_{j,k}\mathbf{H}_{j,k} + \mathbf{P}^{-1}_{j,k|k-1}\Big)^{-1} \tag{24}$$

with Jacobian matrix $\mathbf{H}_{j,k}$ at $\hat{\mathbf{x}}_{j,k|k-1}$. When $\mathbf{P}_{j,k|k-1} = \mathbf{P}_{k|k-1}$, (24) is identical to (11). We observe that (24) requires the global information on $\mathbf{H}^{\mathrm{T}}_{j,k}\mathbf{R}^{-1}_{j,k}\mathbf{H}_{j,k}$ of all $j \in \mathcal{J}$. In a general case, this global information can be achieved by using an additional consensus step on $\mathbf{H}^{\mathrm{T}}_{j,k}\mathbf{R}^{-1}_{j,k}\mathbf{H}_{j,k}$ during the inner-consensus update [8]. However, for some specific applications such as target tracking discussed later, a reasonable assumption is that networked nodes are homogeneous with global information on observation model and level of observation noise. Thus, when $\hat{\mathbf{x}}_{j,k|k-1}$ is identical to the global one, $\mathbf{H}_{j,k}$ is the same for all $j \in \mathcal{J}$. Therefore, information exchange is not necessary for the local update of $\mathbf{P}_{j,k|k}$.

According to the analysis above, the filtered state estimate (mean) $\hat{\mathbf{x}}_{j,k|k}$ and covariance matrix $\mathbf{P}_{j,k|k}$ of the state for all $j \in \mathcal{J}$ are the same and identical to the central ones, when $l \to \infty$. Thus, at the next time instant, the predicted mean and covariance matrix at each node are identical to the central ones based on (17) and (18). This completes the condition $\hat{\mathbf{x}}_{j,k+1|k} = \hat{\mathbf{x}}_{k+1|k}$, $\mathbf{P}_{j,k+1|k} = \mathbf{P}_{k+1|k}$ mentioned before. Definitely, $l \to \infty$ is not feasible in practice. Hence, we only process limited $L$ inner-consensus iterations and the filtered state estimate is approximated by $\hat{\mathbf{x}}_{j,k|k} \leftarrow \mathbf{x}^L_{j,k}$. Note that in practice $L$ is determined by a proper stopping criterion during the inner-consensus iteration, but here for the analysis we consider $L$ as a parameter. Thus, the result of DCEKF is an approximation of the CEKF solution. The accuracy of this approximation depends on $L$. Along with increasing $L$, DCEKF has the ability to converge to the CEKF solution. The whole procedure of DCEKF is summarized in Algorithm 1.

## V. Performance Evaluation

In this section, the proposed DCEKF algorithm is tested in a scenario of distributed target tracking. We perform numerical simulations to evaluate the performance of DCEKF. As a comparison, we also evaluate the AC-based information-weighted consensus filter [16] under nonlinear setting analyzed in [7] and name it as E-ICF, which outperforms some other SotA algorithms. The good performance of E-ICF makes our

**Algorithm 1** Distributed consensus-based EKF (DCEKF)

**Initialization:** for all $j \in \mathcal{J}$, $\hat{\mathbf{x}}_{j,0|0} = \mathbf{m}_0$, $\mathbf{P}_{j,0|0} = \mathbf{P}_0$,
**for** $k = 1, \cdots K$, node $j$ **do**
  calculate the Jacobian matrix $\mathbf{F}_{j,k-1}$ at $\hat{\mathbf{x}}_{j,k-1|k-1}$
  predict the mean $\hat{\mathbf{x}}_{j,k|k-1}$ and the covariance matrix $\mathbf{P}_{j,k|k-1}$ by calculating (17) and (18)
  calculate the Jacobian matrix $\mathbf{H}_{j,k}$ and the modified observation $\bar{\mathbf{y}}_{j,k}$ at $\hat{\mathbf{x}}_{k|k-1}$
  set initial values: $\mathbf{z}^0_{j,k} = \hat{\mathbf{x}}_{j,k|k-1}$, $\boldsymbol{\lambda}^0_{ji} = \mathbf{0}$ and transmit $\mathbf{z}^0_{j,k}$ to neighbors $i \in \mathcal{N}_j$
  **for** $l = 1, \cdots, L$ **do**
    update (21) and transmit $\mathbf{x}^l_{j,k}$ to neighbors $i \in \mathcal{N}_j$
    update (22) and transmit $\mathbf{z}^l_{j,k}$ to neighbors $i \in \mathcal{N}_j$
    update (23)
  **end for**
  obtain the filtered state estimate (mean) $\hat{\mathbf{x}}_{j,k|k} \leftarrow \mathbf{x}^L_{j,k}$ and calculate the covariance matrix $\mathbf{P}_{j,k|k}$ using (24)
**end for**

comparison more challenging. Note that, we choose a relative large step size for AC in E-ICF to make it converge fast, which requires pre-knowledge on graph $\mathcal{G}$. However, DCEKF does not need to know $\mathcal{G}$ beforehand as one advantage.

### A. Simulation Setup

We consider a distributed target tracking scenario where one object moves in a two dimensional plane. The state of a target contains 2D coordinate $x$ and velocity $\dot{x}$, i.e., $\mathbf{x}_k = [x_{1,k}, x_{2,k}, \dot{x}_{1,k}, \dot{x}_{2,k}]^{\mathrm{T}}$. The dynamic model of state is

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{F}_k\mathbf{x}_k + \mathbf{G}_k\mathbf{w}_k \\ &= \begin{bmatrix}\mathbf{I}_2 & T_s\mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2\end{bmatrix}\mathbf{x}_k + \begin{bmatrix}0.5T_s^2\mathbf{I}_2 \\ T_s\mathbf{I}_2\end{bmatrix}\mathbf{w}_k, \end{aligned} \tag{25}$$

where $\mathbf{I}_2$ and $\mathbf{0}_2$ are $2 \times 2$ identical and zero matrix, respectively. This model is a Gaussian random walk commonly used for target tracking [5]. In this work, we set $T_s = 1$ denoting that the sampling interval is 1 second. Assume that the object is moving in a range of 50m $\times$ 50m. The positions of $J = 25$ sensor nodes are deployed semi-randomly[1] in this specific area with coordinates $(x_{1j}, x_{2j})$ for all $j \in \mathcal{J}$. These sensor positions are assumed to be global known and time-invariant. A simple nonlinear observation model is assumed at sensor $j \in \mathcal{J}$ w.r.t. the Euclidean distance to the target as

$$y_{j,k} = \sqrt{(x_{1j} - x_{1,k})^2 + (x_{2j} - x_{2,k})^2} + v_{j,k}. \tag{26}$$

In this paper, we simulate a case with two moving targets for comparisons of different algorithms. Thus, the process model (25) is extended with 8 dimensional state vector. We set the initial states of two objects to be $[20; 20; 0.1; 0.1]$ and $[30; 30; -0.1; 0.1]$. The process noise is zero mean Gaussian with covariance matrix $\mathbf{Q}_k = 0.01\mathbf{I}_4$. Accordingly, the observation in (26) is extended to an 2 dimensional vector and the local noise covariance matrix is set to be $\mathbf{R}_{j,k} = 0.1\mathbf{I}_2$. We randomly generate a network topology by setting the sensor communication radius $r_c = 15$m and fix this topology during the simulation. Note that, under this setting, the generated sensor network is relative sparse with connected ratio[2] 0.1367. As an example, we set the penalty parameter $\mu = 1$ for DCEKF. The step size of AC in E-ICF is set to be $\frac{0.9}{D_{\max}} = 0.1286$ in which $D_{\max}$ is the maximum degree of graph $\mathcal{G}$.

[1]We first generate a regular 2D grid positions and add a random jitter to each sensor position.

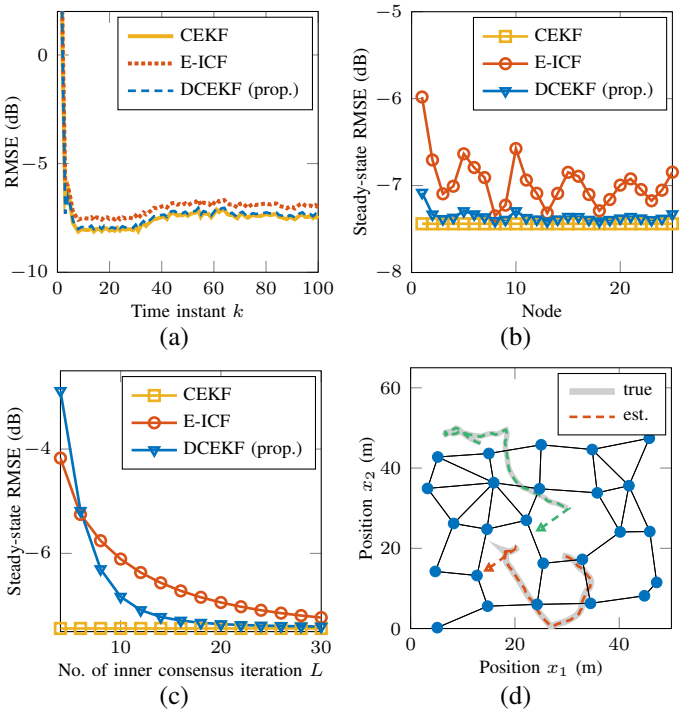


Fig. 1. (a) RMSE over $J = 25$ nodes as a function of time instant $k$ for $L = 20$; (b) Steady-state RMSE of different nodes for $L = 20$; (c) Steady-state RMSE over $J = 25$ w.r.t. the number of inner-consensus iteration $L$; (d) An example of two objects tracking over a sensor network for $L = 20$; Triangles are the initial positions in DCEKF.

For performance analysis, we perform $M = 500$ Monte Carlo runs with different realizations of $\mathbf{w}_k$, $\mathbf{v}_{j,k}$ and $J$ sensor positions. Next we define the root mean square error (RMSE) at each time instant $k$ as a key performance indicator:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{m=1}^{M}\sum_{j=1}^{J} e^2_{j,k,m}}{MJ}}, \tag{27}$$

where the position error $e_{j,k,m}$ is the Euclidean distance between the true and estimated position of the target at node $j$, at time instant $k$ in Monte Carlo run $m$.

### B. Simulation Results

The performance of DCEKF is evaluated compared to CEKF and E-ICF. The sensor sensing range is firstly assumed to be a fully coverage over the moving plane. The RMSE as a function of $k$ for $L = 20$ is shown in Fig.1 (a). We observe that compared to E-ICF, the RMSE performance of DCEKF is better and almost reach the CEKF result. Then we focus

[2]The connected ratio of the network is calculated by $\frac{2|\mathcal{E}|}{J\times(J-1)}$.

TABLE I
STEADY-STATE PERFORMANCE WITH LSR ($L = 20$, $r_s = 30$) AND COMMUNICATION OVERHEAD

| **Algorithms** | CEKF | E-ICF | DCEKF | DCEKF1 |
|---|---|---|---|---|
| steay-state RMSE (dB) | -6.14 | -5.77 | -5.61 | -6.14 |
| comm. overhead per $l$ | | $(m^2+m)J$ | $2mJ$ | $(m^2+2m)J$ |

on the steady-state RMSE by averaging the position errors $e_{j,k,m}$ over time instant 80 to 100 and simulation runs $M$. The node specific steady-state RMSE is depicted in Fig.1 (b). It reflects that DCEKF achieves relative consensus-based solutions among nodes. However, the curve of E-ICF is still fluctuating, which means consensus is not yet achieve among nodes. Fig.1 (c) illustrates the steady-state RMSE over the network w.r.t. the number of inner-consensus iteration $L$. It is obvious that along with the increase of $L$, both DCEKF and E-ICF approach to the central result. After about $L = 6$, DCEKF begins to outperform E-ICF with smaller steady-state RMSE. Around $L = 20$, DCEKF almost converges to CEKF, but E-ICF does not. This also indicates that DCEKF needs smaller $L$ to reach the same level of RMSE performance consuming less communication effort compared to E-ICF. In Fig.1 (d), one example with a sensor network, true trajectories and average estimates of DCEKF are demonstrated for $L = 20$. With randomly chosen initial state, DCEKF works well to track the true positions of multi-objects for this randomly generated sparse network. As discussed in [17], to solve the distributed consensus problem, the ADMM converges faster than the AC for sparse graphs. It provides an evidence to intuitively explain why the ADMM-based DCEKF converges faster compared to the AC-based E-ICF under the setup of sparse graph.

Next, we assume that each sensor has a limited sensing range (LSR), e.g., $r_\mathrm{s} = 30$m. Thus, at some time $k$ some sensors and even its neighbors may not get observations. In this case, we set both local $\mathbf{y}_{j,k}$ and $\mathbf{H}_{j,k}$ to be $\mathbf{0}$. Without exchanging information on current Jacobian matrix $\mathbf{H}_{j,k}$, the global optimization problem (19) of DCEKF is an approximation of the central one. Along with the increase of $r_\mathrm{s}$, this approximation is better. To make (19) identical to the central one, an additional consensus step on matrix $\mathbf{H}_{j,k}^{\mathrm{T}}\mathbf{R}_{j,k}^{-1}\mathbf{H}_{j,k}$ [8] is needed for DCEKF and we name it as DCEKF1. The steady-state RMSE considering LSR for $L = 20$ is shown in Table I. We find that DCEKF obtains comparable performance. With exchanging information on current $\mathbf{H}_{j,k}$, the performance of DCEKF1 is closer to CEKF among all.

For the communication overhead, we assume that each node broadcasts the local information to its neighbors and we count the transmitted scalar per inner iteration $l$ of each algorithm. The results are also shown in Table I. Note that here we do not discuss the communication overhead of CEKF which depends on specific routing protocol. Combined with the result in Fig.1, when the sensing range of sensors covers the entire field, DCEKF outperforms E-ICF with better RMSE performance and lower communication overhead. However, when sensors have LSR, our suggestion is to consider $r_\mathrm{s}$ and the trade-off between performance and communication overhead. When $r_\mathrm{s}$ is relative large, DCEKF can well approximate the central optimization problem and achieves a comparable solution with low communication overhead.

## VI. CONCLUSION

In this paper, we propose a distributed algorithm to solve the consensus-based state estimation problem for nonlinear dynamic systems. By considering the EKF from a Bayesian viewpoint, we construct an equivalent consensus MAP estimation problem under linear Gaussian assumptions. Analytical and simulated results illustrate that the proposed DCEKF algorithm converges to the central solution with robust consensus-based state estimates among nodes. Especially for sparse networks, DCEKF distinguishes itself in terms of fast convergence rate and low communication overhead without pre-knowledge on network graph. Even when the LSR of sensors is considered, DCEKF has the potential to achieve comparable performance meanwhile keeping low communication effort.

## REFERENCES


[1] M. S. Mahmoud and H. M. Khalid, "Distributed Kalman filtering: a bibliographic review," *IET Control Theory Applications*, vol. 7, no. 4, pp. 483–501, March 2013.

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

[3] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[4] E. A. Wan and R. V. D. Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Oct. 2000, pp. 153–158.

[5] O. Hlinka, F. Hlawatsch, and P. M. Djuric, "Distributed particle filtering in agent networks: a survey, classification, and comparison," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 61–81, Jan. 2013.

[6] 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.

[7] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano, "Consensus-based linear and nonlinear filtering," *IEEE Trans. on Automatic Control*, vol. 60, no. 5, pp. 1410–1415, May 2015.

[8] S. Wang, H. Paul, and A. Dekorsy, "Distributed optimal consensus-based Kalman filtering and its relation to MAP estimation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 3664–3668.

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

[10] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.

[11] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993.

[12] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[13] 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.

[14] Q. Ling, Y. Liu, W. Shi, and Z. Tian, "Weighted ADMM for fast decentralized network optimization," *IEEE Trans. on Signal Processing*, vol. 64, no. 22, pp. 5930–5942, 2016.

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

[16] 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.

[17] N. Bof, R. Carli, and L. Schenato, "Is ADMM always faster than average consensus?" *Automatica*, vol. 91, no. 1, pp. 311–315, 2018.