MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Distributed Kalman Filtering: When to Share Measurements

Greiff, Marcus; Berntorp, Karl

TR2022-158 December 09, 2022

Abstract

This paper considers the problem of designing distributed Kalman filters (DKFs) when the sensor measurement noise is correlated. To this end, we analyze several existing methods in terms of their Bayesian Cramér-Rao bounds (BCRB), and insights from the analysis motivates a departure from the conventional estimate-sharing frameworks in favor of measurement-sharing. We demonstrate that if the communication bandwidth and computational resources permit, the minimum mean-square error (MMSE) estimator is implementable under measurement-sharing protocols. Furthermore, such approaches may use less communication bandwidth than standard consensus methods for smaller estimation problems. The developments are verified in several numerical examples, including comparisons against previously reported methods.

IEEE Conference on Decision and Control (CDC) 2022

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Distributed Kalman Filtering: When to Share Measurements

Marcus Greiff[1] and Karl Berntorp[1]

***Abstract*— This paper considers the problem of designing distributed Kalman filters (DKFs) when the sensor measurement noise is correlated. To this end, we analyze several existing methods in terms of their Bayesian Cramér-Rao bounds (BCRB), and insights from the analysis motivates a departure from the conventional estimate-sharing frameworks in favor of measurement-sharing. We demonstrate that if the communication bandwidth and computational resources permit, the minimum mean-square error (MMSE) estimator is implementable under measurement-sharing protocols. Furthermore, such approaches may use less communication bandwidth than standard consensus methods for smaller estimation problems. The developments are verified in several numerical examples, including comparisons against previously reported methods.**

I. INTRODUCTION

In this paper, we consider the problem of designing distributed Kalman filters (DKFs) for moderately sized sensor networks where the measurement noise is correlated. In general, DKFs can be realized in two ways: either by having the sensors communicating their local measurements [1], or their local estimates [2]–[8]. The latter represents the more popular approach for large sensor networks, as the communication bandwidth can be made to scale with the dimensions of the estimates [3]. When the noise in the sensors are uncorrelated, the problem can be solved for various underlying assumptions. If nothing is known about the communication graph (apart from that it is connected), there exists a large number of consensus DKFs (CDKFs) that achieve an estimate consensus among the nodes based on iteratively weighting the estimates in a sensor's 1-hop neighborhood [2]. In such consensus filters, the weights can be designed such that convergence of the estimation error is ensured (e.g., [2], [9]). However, even in the cases where few assumptions are made on the communication graph, the measurement noise among the sensors is generally assumed to be uncorrelated. For correlated measurement noise, the estimation problem is significantly more difficult to solve, and established methods (e.g., [2], [5], [9]) may perform worse than approaches that rely solely on local information.

Example 1 *To illustrate this conceptually, consider two sensors that have no dynamics nor process noise, with states x that represent the same physical quantity in the two sensors. Assume that T measurements, $\{\boldsymbol{y}_k\}_{k=1}^T$, are sampled from*

$$\boldsymbol{y}_k = \begin{bmatrix} y_k^1 \\ y_k^2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} x, \begin{bmatrix} \sigma_1^2 & \sigma_2^2 \\ \sigma_2^2 & \sigma_1^2 \end{bmatrix}\right) \triangleq \mathcal{N}\left(\boldsymbol{C}x, \boldsymbol{R}\right),$$

with $(c_1, c_2) \in \mathbb{R}^2$, $\sigma_1 \geq \sigma_2 \geq 0$. Consider three cases:

(A) *x is estimated locally as $\hat{x}^A$, knowing $\{y_k^1\}_{k=1}^T$ and the parameters c_1 and σ_1 (and not $\{y_k^2\}_{k=1}^T$).*

(B) *x is estimated globally as $\hat{x}^B$ with knowledge of $\{\boldsymbol{y}_k\}_{k=1}^T$, the parameters c_1, c_2 and σ_1, but assuming that the noise is uncorrelated ($\sigma_2 = 0$) in the estimator.*

(C) *x is estimated globally as $\hat{x}^C$ with knowledge of $\{\boldsymbol{y}_k\}_{k=1}^T$ and the parameters c_1, c_2, σ_1, and σ_2.*

Assume that the estimates are computed with the best linear unbiased estimator (BLUE), provided by the Gauss-Markov theorem (see [10, Theorem 6.1]). It is simple to verify that

(A) $\mathrm{Var}(\hat{x}^A) = \sigma_1^2/(Tc_1^2)$,

(B) $\mathrm{Var}(\hat{x}^B) = (\sigma_1^2(c_1^2 + c_2^2) + 2\sigma_2^2 c_1 c_2)/(T(c_1^2 + c_2^2)^2)$,

(C) $\mathrm{Var}(\hat{x}^C) = (\sigma_1^4 - \sigma_2^4)/(T(\sigma_1^2(c_1^2 + c_2^2) - 2\sigma_2^2 c_1 c_2))$,

and $\mathrm{Var}(\hat{x}^A) \geq \mathrm{Var}(\hat{x}^C), \mathrm{Var}(\hat{x}^B) \geq \mathrm{Var}(\hat{x}^C)\ \forall c_i, \sigma_i, T$. *However, if the noise is correlated, it can sometimes be better to implement (A) over (B). For instance, if $c_1 = \sigma_1 = 1$, then*

$$0 < \frac{c_2 + c_2^3}{2} \leq \sigma_2^2 < 1 \Rightarrow \mathrm{Var}(\hat{x}^B) \geq \mathrm{Var}(\hat{x}^A), \quad \forall T. \quad (1)$$

The tightness of the inequalities depend nontrivially on the cross-correlation in $\boldsymbol{R}$ and the structure of $\boldsymbol{C}$. Note that $\mathrm{Var}(\hat{x}^B) \to \mathrm{Var}(\hat{x}^C)$ as $\sigma_2 \to 0$, that is, (B) and (C) are equivalent if the measurement noise is not cross-correlated.

1) Contributions: We discuss how and when measurement sharing should be considered in favor of estimate sharing for small to moderately sized sensor networks. In particular, we study the distributed filtering problem, and we show that estimate sharing cannot perform better than the minimum mean-square error (MMSE) estimator implemented using measurement sharing. This is done by analyzing the associated Bayesian Cramér-Rao bounds (BCRBs), and illustrated by numerical examples. We also demonstrate numerically that even in the filtering context, it may sometimes be advantageous to consider an estimator in (A) over (B) when the measurement noise is correlated.

2) Notation: We let $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ indicate that $\boldsymbol{x}$ is Gaussian distributed with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. The notation $\hat{\boldsymbol{x}}_{k|k}$ refers to the estimate of $\boldsymbol{x}$ at time step k given the set of measurements $\boldsymbol{y}_{0:k} \triangleq \{\boldsymbol{y}_0, \dots, \boldsymbol{y}_k\}$, and $\hat{\boldsymbol{x}}_{k|k-1}$ denotes the one-step prediction of $\hat{\boldsymbol{x}}_{k-1|k-1}$. In addition, let $\boldsymbol{x}_{\infty|\infty} \triangleq \lim_{k\to\infty} \boldsymbol{x}_{k|k}$ and let $\boldsymbol{y}_{a:b} = \emptyset$ for any $b < a$. With $p(\boldsymbol{x}_{0:k}|\boldsymbol{y}_{0:k})$, we mean the posterior density function of the state trajectory $\boldsymbol{x}_{0:k}$ from time step 0 to time step k given the measurement sequence $\boldsymbol{y}_{0:k}$, and $p(\boldsymbol{x}_k|\boldsymbol{y}_{0:k})$ denotes the corresponding marginal (filtering) posterior. The notation $\mathrm{diag}(\cdot)$ is a matrix composition where the arguments form blocks on the diagonal, and $\otimes$ is Kronecker product. Finally, for any symmetric and real matrix, $\boldsymbol{M}$, we let $\boldsymbol{M} \succeq \boldsymbol{0} \Leftrightarrow \boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x} \geq 0, \boldsymbol{M} \succ \boldsymbol{0} \Leftrightarrow \boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x} > 0\ \forall \boldsymbol{x} \neq \boldsymbol{0}$.

[1]Mitsubishi Electric Research Labs (MERL), 02139 Cambridge, MA, USA. Email: `greiff@merl.com`

II. Preliminaries

In this paper, we consider observable stable linear systems

$$\boldsymbol{x}_k = \boldsymbol{A}_k \boldsymbol{x}_{k-1} + \boldsymbol{q}_k, \qquad \boldsymbol{q}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_k), \tag{2a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{x}_k + \boldsymbol{r}_k \qquad \boldsymbol{r}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_k), \tag{2b}$$

where $\mathbb{E}[\boldsymbol{q}_k \boldsymbol{q}_s^\top] = \boldsymbol{0}$ and $\mathbb{E}[\boldsymbol{r}_k \boldsymbol{r}_s^\top] = \boldsymbol{0}$ for all $k \neq s$, and

$$\boldsymbol{y}_k = \begin{bmatrix} \boldsymbol{y}_{1,k} \\ \vdots \\ \boldsymbol{y}_{N,k} \end{bmatrix}, \quad \boldsymbol{C}_k = \begin{bmatrix} \boldsymbol{C}_{1,k} \\ \vdots \\ \boldsymbol{C}_{N,k} \end{bmatrix}, \quad \boldsymbol{r}_k = \begin{bmatrix} \boldsymbol{r}_{1,k} \\ \vdots \\ \boldsymbol{r}_{N,k} \end{bmatrix}, \tag{3a}$$

such that $\boldsymbol{y}_k^i = \boldsymbol{C}_{i,k}\boldsymbol{x}_k + \boldsymbol{r}_{i,k}$, with correlated noise

$$\boldsymbol{R}_k = \begin{bmatrix} \boldsymbol{R}_{11,k} & \cdots & \boldsymbol{R}_{1N,k} \\ \vdots & \ddots & \vdots \\ \boldsymbol{R}_{N1,k} & \cdots & \boldsymbol{R}_{NN,k} \end{bmatrix}. \tag{3b}$$

A connected graph, $\mathcal{G}$, is defined by a set of vertices $\mathcal{V} = \{\mathcal{V}_i\}_{i=1}^N$ connected by edges $\mathcal{E}_{ij}$. Let $d(i,j)$ denote the smallest number of edges connecting the nodes $\mathcal{V}_i$ and $\mathcal{V}_j$, let $K_i = \max_j d(i,j)$, and take $\bar{K} = \max_i K_i$. Furthermore, let $\mathcal{U}_i^n = \{j \in [1,...,N] | d(i,j) \leq n\}$ denote the n-hop neighborhood of a vertex $\mathcal{V}_i$ (including i). Assume that the transmission information along an edge takes exactly one time step and that the measurement $\boldsymbol{y}_k^i$ is sampled in the node $\mathcal{V}_i$ at a time step k. Then, if the nodes share their measurements over the network, the information available to a node $\mathcal{V}_i$ at a time step k can be defined as follows.

Definition 1 *Let $\mathcal{Y}_k = \{\boldsymbol{y}_{0:k}^j\}_{j=1}^N$ denote all measurements at a time step k, and let $\mathcal{Y}_k^i \subseteq \mathcal{Y}_k$ be the set of measurements known to the vertex $\mathcal{V}_i$ at a time k, as $\mathcal{Y}_k^i = \{\boldsymbol{y}_{0:k-d(i,j)}^j\}_{j=1}^N$. Furthermore, let $\mathcal{Y}_k^{N_c,i} = \{\boldsymbol{y}_{0:\max(k,k+N_c-1-K_i)}^j\}_{j=1}^N$ be the set of all measurements known to vertex $\mathcal{V}_i$ at a time step k if N_c delay-free transmissions are permitted on each time-step.*

Consider the conditional density function of the state $\boldsymbol{x}_a$ at a time step a given the measurements up until and including b. If the initial state distribution is Gaussian, $p(\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_0 | \hat{\boldsymbol{x}}_{0|0}, \boldsymbol{P}_{0|0})$, then it follows that $p(\boldsymbol{x}_a|\mathcal{Y}_b) = \mathcal{N}(\boldsymbol{x}_a | \hat{\boldsymbol{x}}_{a|b}, \boldsymbol{P}_{a|b})$ is Gaussian for all a and b. Furthermore, the moments $\hat{\boldsymbol{x}}_{a|b}, \boldsymbol{P}_{a|b}$ are given by the standard discrete-time KF recursions [11], consisting of a prediction step

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{A}_k \hat{\boldsymbol{x}}_{k-1|k-1}, \tag{4a}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{A}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{A}_k^\top + \boldsymbol{Q}_k, \tag{4b}$$

and an update step

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \boldsymbol{C}_k^\top (\boldsymbol{C}_k^\top \boldsymbol{P}_{k|k-1} \boldsymbol{C}_k + \boldsymbol{R}_k)^{-1}, \tag{5a}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k (\boldsymbol{y}_k - \boldsymbol{C}_k \hat{\boldsymbol{x}}_{k|k-1}), \tag{5b}$$

$$\boldsymbol{P}_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}_k) \boldsymbol{P}_{k|k-1}. \tag{5c}$$

As such, we can compute a Bayesian Cramér-Rao bound (BCRB) using the Bayesian information matrix in [12], [13],

$$\boldsymbol{J}_k = \mathbb{E}_{p(\boldsymbol{x}_k, \mathcal{Y}_k)}[-\nabla_{\boldsymbol{x}_k} \nabla_{\boldsymbol{x}_k}^\top \log(p(\boldsymbol{x}_k, \mathcal{Y}_k))], \tag{6}$$

which in the context of (2) can be computed recursively as

$$\boldsymbol{J}_k = (\boldsymbol{Q}_k + \boldsymbol{A}_k \boldsymbol{J}_{k-1}^{-1} \boldsymbol{A}_k^\top)^{-1} + \boldsymbol{C}_k^\top \boldsymbol{R}_k^{-1} \boldsymbol{C}_k, \tag{7}$$

and where for any unbiased estimate $\hat{\boldsymbol{x}}_k$ of $\boldsymbol{x}_k$,

$$\mathrm{Cov}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k) \succeq \mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k) = [\boldsymbol{J}_k]^{-1}. \tag{8}$$

The inequality in (8) can be replaced by an equality when $\hat{\boldsymbol{x}}_k$ is computed using (4) and (5). In the more general case, where the measurements are delayed with K time steps, we can compute $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_{k-K})$ using the same recursions, with $\boldsymbol{C}_i = \boldsymbol{0}$ for all $k - K < i \leq k$. By similar reasoning,

$$\mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k) \preceq \mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k^i) \tag{9a}$$

$$\preceq \mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_{k-K} \cup \boldsymbol{y}_{0:k}^i) \tag{9b}$$

$$\preceq \mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_{k-K}). \tag{9c}$$

III. Distributed Filtering Methods

In this section, we introduce four related but different strategies to solve the distributed filtering problem. This includes three common approaches based on estimate-sharing in Sections III-A–III-C, where only one is capable of incorporating knowledge of cross-correlated measurement noise. We also define the MMSE estimator in Section III-D and show that it is implementable with measurement sharing.

A. Consensus DKFs

The consensus filters locally perform a KF update and subsequently transmit the estimate to the 1-hop neighborhood [2], [14], [15]. The estimates are combined by one of many averaging rules, also known as consensus protocols, defined by a set of weights $\{w_{ij} \in \mathbb{R} | i = 1,...,N, j = 1,...,N, \sum_i w_{ij} = 1\}$. Three common examples include

$$\mathrm{P1}: \quad w_{ij} \triangleq N^{-1}, \qquad \forall i \neq j, \tag{10a}$$

$$\mathrm{P2}: \quad w_{ij} \triangleq (\max_{j \in \mathcal{U}_i^1} |\mathcal{U}_j^1|)^{-1}, \qquad \forall i \neq j, \tag{10b}$$

$$\mathrm{P3}: \quad w_{ij} \triangleq (1 + \max(|\mathcal{U}_i^1|, |\mathcal{U}_j^1|))^{-1}, \quad \forall i \neq j. \tag{10c}$$

Here, the protocol P3 being is a common Metropolis weight scheme [14]. We consider an implementation of these filters on the information form, with an information vector $\boldsymbol{\gamma}_{k|k}$ and information matrix $\boldsymbol{\Gamma}_{k|k}$ relating to the estimates of the CDKF as $\hat{\boldsymbol{x}}_{k|k}^i = (\boldsymbol{\Gamma}_{k|k}^{ii})^{-1} \boldsymbol{\gamma}_{k|k}^i$ and $\boldsymbol{P}_{k|k}^{ii} = (\boldsymbol{\Gamma}_{k|k}^{ii})^{-1}$. Generally, these methods perform a large number of iterations at each time step, and the consensus protocols are implemented by having the nodes iterate their information vectors (and information matrices) with their neighbors over N_c steps. Starting from $\hat{\gamma}_{k|k}^{i,(0)} \triangleq \hat{\gamma}_{k|k}^i$ and $\boldsymbol{\Gamma}_{k|k}^{ii,(0)} \triangleq \boldsymbol{\Gamma}_{k|k}^{ii}$, according to

$$\hat{\gamma}_{k|k}^{i,(n+1)} = \sum_{j \in \mathcal{U}_i^1} w_{ij} \hat{\gamma}_{k|k}^{j,(n)}, \quad \boldsymbol{\Gamma}_{k|k}^{ii,(n+1)} = \sum_{j \in \mathcal{U}_i^1} w_{ij} \boldsymbol{\Gamma}_{k|k}^{jj,(n)}, \tag{11}$$

with $n < N_c$ chosen large enough for the iterated estimates to approach a consensus in the estimates. For the communication protocols in (10), convergence to the consensus can be quantified explicitly based on the Laplacian of the graph [2]. In the following, we primarily consider the cases where one consensus iteration is performed with protocol P, denoted by P-CDKF-1, and when the consensus iterations are done until convergence, denoted by P-CDKF-∞.

B. Fused DKFs

An alternative but closely related approach is to implement a covariance intersection (CI) [16], [17] (or some similar fusion strategy) to fuse the local estimates in the DKF. One such example is the fused DKF (here FDKF) proposed in [5], where the weights w_{ij} in (11) are computed based on the relative uncertainty of the estimates in the 1-hop neighborhood of a vertex, as

$$w_{ij} = \frac{\mathrm{Trace}[(\mathbf{\Gamma}^{(n),i}_{k|k})^{-1}]}{\sum_{j\in\mathcal{U}_i^1}\mathrm{Trace}[(\mathbf{\Gamma}^{(n),j}_{k|k})^{-1}]}. \tag{12}$$

Eq. (12) provides an adaptive method for dynamically updating the weights of the consensus iterations, which are otherwise static in a CDKF. In [5], it is shown that (12) yields a conservative estimate of the posterior estimate covariance, that the resulting estimate is unbiased (when the measurement noise is uncorrelated), and that the estimates in the graph asymptotically achieve a consensus. In the following, we let FDKF-1 denote a filter that obeys a 1-delay communication constraint with the weights chosen according to (12) , and FDKF-∞ refers to a filter iterates the estimates are iterated using (11) until a consensus is reached.

C. Weighted DKFs

The weighted DKFs considered in this paper only propagate the first moments of the local estimates, transmitting these to the 1-hop neighborhood at each time step [3], [4]. After the update step, these estimates are fused using a set of weight matrices $\{\boldsymbol{W}_{ij}\}_{i=1,j=1}^{i=N,j=N}$, subject to the constraints

$$\boldsymbol{W}_{ij} = \mathbf{0} \quad \text{if} \quad j\notin\mathcal{U}_i^1, \qquad \sum_{j\in\mathcal{U}_i^1}\boldsymbol{W}_{ij} = \boldsymbol{I}. \tag{13}$$

An estimate can then be formed similar to CDKF,

$$\hat{\boldsymbol{x}}^{i,\mathrm{reg}}_{k|k} = \sum_{j\in\mathcal{U}_i^1}\boldsymbol{W}_{ij}\hat{\boldsymbol{x}}^j_{k|k}, \tag{14}$$

where this fused estimate is unbiased and the scheme implementable given $\mathcal{G}$ by the constraints imposed in (13). The appeal of this method lies in that the variance of the weighted estimate can be expressed in local estimate variance as

$$\boldsymbol{P}^{\mathrm{reg}}_{k|k} \triangleq \boldsymbol{W}\mathrm{diag}(\boldsymbol{P}^{11}_{k|k},, ..., \boldsymbol{P}^{NN}_{k|k})\boldsymbol{W}^\top. \tag{15}$$

As such, an optimization problem may be posed and solved offline over the weight matrices to minimize the weighted posterior covariance of the estimation error as outlined in [3],

$$\underset{\boldsymbol{W}}{\text{minimize}} \quad \mathrm{Trace}(\boldsymbol{P}^{\mathrm{reg}}_{\infty|\infty}) \tag{16a}$$
$$\text{subject to} \quad (13) \tag{16b}$$

Generally, $\boldsymbol{W}_{ij} \neq \boldsymbol{W}_{ji}$ if $i\neq j$ and $j\in\mathcal{U}_i^1$, and this approach also takes the cross-correlations in (3b) into account. The method can be adapted to reduce the number of parameters by constraining each weight matrix $\boldsymbol{W}_{ij} = w_{ij}\boldsymbol{I}$ for some scalar w_{ij}. However, this simplification is not done in this paper, and we refer to a DKF that uses *matrix-valued weights* computed by minimizing (16) as a weighted DKF (WDKF). We solve (16) using CVX in Matlab [18].

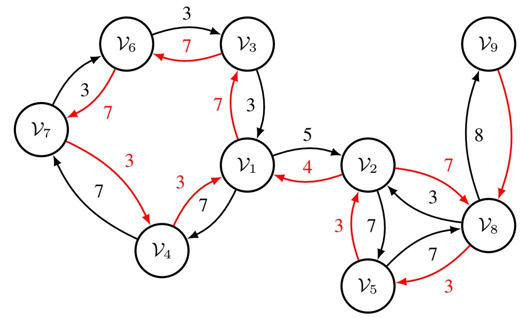

Fig. 1. A communication graph where each edge shows the cardinality $|\mathcal{S}_i^j|$ as depicted in red and black. Note that generally $|\mathcal{S}_i^j| \neq |\mathcal{S}_j^i|$.

D. Measurement-Sharing DKFs

From Section II, it is clear that the optimal MMSE estimator can be implemented if the measurements are shared among the vertices, and that the corresponding BCRB is attained. Now, consider an implementation where the nodes send and receive measurements with the following logic:

- At time step k, the vertex $\mathcal{V}_i$ samples a measurement $\boldsymbol{y}_k^i$ and receives a set of measurements $\mathcal{R}_i^j$ from the vertices in the 1-hop neighborhood $j\in\mathcal{U}_i^1$.
- Consider a set of all received measurements as $\mathcal{R}_i = \bigcup_{j\in\mathcal{U}_i^1}\mathcal{R}_i^j$, and take the most recently acquired measurement with respect to each measurement and form a set $\mathcal{S}_i = \{\max_a \boldsymbol{y}_a^b \in \mathcal{R}_i | b = [1, ..., N]\}$.
- Form a set of measurements to be sent to each node $j\in\mathcal{U}_i^1$ as $\mathcal{S}_i^j = \mathcal{S}_i\backslash(\mathcal{S}_i\cap\mathcal{R}_i^j)$, and transmit this set.

The measurements from $\mathcal{V}_i$ will reach the node $\mathcal{V}_j$ in $d(i,j)$ time steps. Furthermore, the scheme avoids superfluous sending of measurements, and a conservative upper bound on the cardinality of the set $\mathcal{S}_i^j$ is $|\mathcal{S}_j^i| < N$, and will generally be smaller, as clearly demonstrated in Fig. 1.

Note that a KF with an extended state vector $\boldsymbol{X}_k^i = ((\boldsymbol{x}_k^i)^\top, ..., (\boldsymbol{x}^i_{k-K_i})^\top)^\top$ and a measurement vector $\boldsymbol{Y}_k^i = ((\boldsymbol{y}^i_{k-d(i,1)})^\top, ..., (\boldsymbol{y}^i_{k-d(i,N)})^\top)^\top$ is not the MMSE estimator. Indeed, in node $\mathcal{V}_i$, such an implementation does not leverage the cross-correlation in the set $\mathcal{C}_i \triangleq \{\boldsymbol{R}_{lk} | l \neq k, d(l,i) \neq d(k,i)\}$. Such estimator may come close to, but will generally never achieve the BCRB of the estimate variance if any $\boldsymbol{R}_{ij}\in\mathcal{C}_i$ is nonzero (as will be shown in Proposition 1). Instead, the MMSE approach is to store a delayed estimate in each node, which at a time step k is computed based on $\mathcal{Y}_k^i$ as $p(\boldsymbol{x}_{k-K_i}|\mathcal{Y}_{k-K_i}) = \mathcal{N}(\boldsymbol{x}_k|\hat{\boldsymbol{x}}_{k|k-K_i}, \boldsymbol{P}_{k|k-K_i})$. Using the measurements in $\mathcal{Y}_k^i\backslash\mathcal{Y}_{k-K_i}$, the estimate $p(\boldsymbol{x}_k|\mathcal{Y}_k^i)$ can then be computed by (4) and (5), implementing (7) subject to the communication constraints. This filter yields a tight BCRB. We denote it as SDKF and summarize it in Algorithm 1. For later comparisons, we denote a global KF filter operating without communication delays by KF-∞.

IV. The Distributed Kalman Filter

The analysis becomes more complicated when introducing dynamics and process noise in the states. As previously, let $\bar{\boldsymbol{R}} = \mathrm{diag}(\boldsymbol{R}_{11}, ..., \boldsymbol{R}_{NN})$, but now consider two cases:

(B) a KF designed with respect to $\bar{\boldsymbol{R}}$,
(C) a KF designed with respect to $\boldsymbol{R}$,

Algorithm 1 Pseudocode of the SDKF

Initialize $\{\hat{\boldsymbol{x}}_{0|0}^i, \boldsymbol{P}_{0|0}^{ii}\}_{i=1}^N$
for $k = 1$ **to** T **do**
  for $i = 1$ **to** N **do**
    // Time update
    Evaluate $\{\hat{\boldsymbol{x}}_{k-K_i|k-K_i-1}^i, \boldsymbol{P}_{k-K_i|k-K_i-1}^{ii}\}$ by (4)
    // Local measurement update
    Assemble $\boldsymbol{y}_k$ from $\mathcal{R}_i$
    Evaluate $\{\hat{\boldsymbol{x}}_{k-K_i|k-K_i}^i, \boldsymbol{P}_{k-K_i|k-K_i}^{ii}\}$ by (5)
    // Externalization
    for $s = k - K_i + 1$ **to** k **do**
      Compute $\mathcal{I}_s = \{a_l\}_{l=1}^m = \{j | d(i,j) < k - s\}$
      Form the matrices

$$\boldsymbol{C}^{(s)} = \begin{bmatrix} \boldsymbol{C}_{a_1,s} \\ \vdots \\ \boldsymbol{C}_{a_m,s} \end{bmatrix}, \boldsymbol{Y}^{(s)} = \begin{bmatrix} \boldsymbol{Y}_{a_1,s} \\ \vdots \\ \boldsymbol{Y}_{a_m,s} \end{bmatrix}, \forall a_l \in \mathcal{I}_s$$

$$\boldsymbol{R}^{(s)} = \begin{bmatrix} \boldsymbol{R}_{a_1 a_1,s} & \cdots & \boldsymbol{R}_{a_1 a_m,s} \\ \vdots & \ddots & \vdots \\ \boldsymbol{R}_{a_m a_1,s} & \cdots & \boldsymbol{R}_{a_m a_m,s} \end{bmatrix}, \forall a_l \in \mathcal{I}_s.$$

      Evaluate $\{\hat{\boldsymbol{x}}_{s|s-1}, \boldsymbol{P}_{s|s-1}\}$ using (4)
      Evaluate $\{\hat{\boldsymbol{x}}_{s|s}, \boldsymbol{P}_{s|s}\}$ by (5) w.r.t. $\boldsymbol{R}^{(s)}, \boldsymbol{C}^{(s)}, \boldsymbol{Y}^{(s)}$
    end for
  end for
  // Transmission
  Send $\mathcal{S}_i^j$ to all $\{\mathcal{V}_j | j \in \mathcal{U}_i^1 \backslash \{i\}\}$
  Output $\{\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{P}_{k|k}\}$
  Store $\{\hat{\boldsymbol{x}}_{k-K|k-K}, \boldsymbol{P}_{k-K|k-K}\}$ and $\mathcal{Y}_k^i \backslash \mathcal{Y}_{k-K}$.
end for

Let $\{\hat{\boldsymbol{x}}_{k|k}^B, \boldsymbol{P}_{k|k}^B\}$ and $\{\hat{\boldsymbol{x}}_{k|k}^C, \boldsymbol{P}_{k|k}^C\}$ denote the estimates in both cases, respectively. Then the estimate error covariance can be expressed through the matrix inversion lemma as

$$\boldsymbol{P}_{k|k}^B = ((\boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{C})(\boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{R} \bar{\boldsymbol{R}}^{-1} \boldsymbol{C})^{-1} (\boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{C}) + (\boldsymbol{P}_{k|k-1}^B)^{-1})^{-1}, \tag{17a}$$

$$\boldsymbol{P}_{k|k}^C = (\boldsymbol{C}^\top \boldsymbol{R}^{-1} \boldsymbol{C} + (\boldsymbol{P}_{k|k-1}^C)^{-1})^{-1}. \tag{17b}$$

Proposition 1 *Given* (17)*, if the two filters are initialized with the same prior, then, for all time steps k,*

$$\boldsymbol{P}_{k-1|k-1}^C \preceq \boldsymbol{P}_{k-1|k-1}^B \Leftrightarrow \boldsymbol{P}_{k|k-1}^C \preceq \boldsymbol{P}_{k|k-1}^B \tag{18}$$

$$\Rightarrow \boldsymbol{P}_{k|k}^C \preceq \boldsymbol{P}_{k|k}^B. \tag{19}$$

Eq. (19) *is strict if* $\boldsymbol{C}^\top (\boldsymbol{R}^{-1} - \bar{\boldsymbol{R}}^{-1} \boldsymbol{R} \bar{\boldsymbol{R}}^{-1}) \boldsymbol{C} \succ \boldsymbol{0}$.

Proof: The equivalence holds by (4b), and the second implication can be shown by analyzing the difference $\boldsymbol{P}_{k|k}^B - \boldsymbol{P}_{k|k}^C$ using Schur complements: (19) holds if $\boldsymbol{M} \triangleq (\boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{C})^{-1} \boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{R} \bar{\boldsymbol{R}}^{-1} \boldsymbol{C} (\boldsymbol{C}^\top \bar{\boldsymbol{R}}^{-1} \boldsymbol{C})^{-1} - (\boldsymbol{C}^\top \boldsymbol{R}^{-1} \boldsymbol{C})^{-1} \succeq \boldsymbol{0}$, which can also be shown by Schur complements using the definition of $\boldsymbol{R}$ and $\bar{\boldsymbol{R}}$. ■

Here, the gap in the BCRB should inform the decision of whether a simpler DKF algorithm that ignores the measurement cross-correlation (e.g., a CDKF or FDKF) should be implemented in favor of a DKF algorithm that exploits this knowledge (e.g., a WDKF or SDKF). In all of these cases, the asymptotic variances are best computed and analyzed offline for the specific problem at hand.

TABLE I
STRUCTURE OF THE ASYMPTOTIC WEIGHT MATRICES.

Filter	Stationary Weight Matrix
CDKF-1,FDKF-1	$\boldsymbol{W} = \bar{\boldsymbol{W}} \otimes \boldsymbol{I}_{\dim(\boldsymbol{x})} \in \mathbb{R}^{N \dim(\boldsymbol{x}) \times N \dim(\boldsymbol{x})}$, subject to (13), with $[\bar{\boldsymbol{W}}]_{ij} > 0$
CDKF-∞,FDKF-∞	$\boldsymbol{W} = (\boldsymbol{1}_N \bar{\boldsymbol{w}}^\top) \otimes \boldsymbol{I}_{\dim(\boldsymbol{x})}$, subject to $\bar{\boldsymbol{w}}^\top \boldsymbol{1}_N = 1$, with $\bar{w}_i > 0$
WDKF	$\boldsymbol{W} \in \mathbb{R}^{N \dim(\boldsymbol{x}) \times N \dim(\boldsymbol{x})}$, subject to the constraint in (13)

A. Comparison of BCRBs

Consider the various DKFs, and view the estimate $\hat{\boldsymbol{x}}_k^i$ as a function of the measurements. If the estimates are computed locally in the node, $\hat{\boldsymbol{x}}_k^i := \hat{\boldsymbol{x}}_k^i(\boldsymbol{y}_{0:k})$. If the nodes communicate with its neighbors in a 1-hop neighborhood, then $\hat{\boldsymbol{x}}_k^i := \hat{\boldsymbol{x}}_k^i(\mathcal{Y}_k^i)$, and if communication is done in N_c iterations without delay, then $\hat{\boldsymbol{x}}_k^i := \hat{\boldsymbol{x}}_k^i(\mathcal{Y}_k^{N_c,i})$. Therefore, if $N_C = 1$, $\mathcal{Y}_k^{N_c,i} = \mathcal{Y}_k^i$, and if $N_c > K_i$, $\mathcal{Y}_k^{N_c,i} = \mathcal{Y}_k$. Thus, the filters for which $N_c > K_i$ should be evaluated against $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k)$, while the 1-delay DKFs (including the CDKF-1, FDKF-1, WDKF, and SDKF) should be evaluated against $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k | \mathcal{Y}_k^i)$. In general, the consensus schemes will not attain the BCRB (as they generally cannot be written as an SDKF), and even if the filters can be implemented as the MMSE estimator for $\bar{\boldsymbol{R}}$, there will exist a gap between their error covariance and the covariance of the corresponding MMSE estimator if $\boldsymbol{C}^\top (\boldsymbol{R}^{-1} - \bar{\boldsymbol{R}}^{-1} \boldsymbol{R} \bar{\boldsymbol{R}}^{-1}) \boldsymbol{C} \succ \boldsymbol{0}$.

To get a sense of these BCRB-gaps in a general setting, note that the estimate-sharing schemes all aspire to solve the optimization problem in (16) over a weighting protocol, but with various constraints. These protocols are derived heuristically and fixed in the CDKF and FDKF setting, but explicitly optimized in the WDKF, with the resulting matrices summarized in Table I. The asymptotic weight matrices of the CDKF-1 and FDKF-1 are of the same structure as that of the WDKF, but with the additional constraint that all $\boldsymbol{W}_{ij}$ are diagonal. Hence, even if the weighting protocol in the CDKF-1 and FDKF-1 are optimized with reference to (16), it will never do better than the WDKF if estimate sharing is done with respect to the same objects (i.e., the estimate and covariance, or information vector and information matrix).

Remark 1 *The gaps in MSE will generally vary among the nodes in the network when the noise is correlated, and are best computed for specific nodes of interest to inform decision-making as to which method to use. However, if* $\boldsymbol{R}$ *is known and there is a 1-delay communication constraint, no method can perform better than the SDKF.*

B. Comparison of Bandwidth

The communication bandwidth for the various methods is summarized in Table II, assuming that a double is represented in D bits. For simplification, assume that $\dim(\boldsymbol{x}) \approx \dim(\boldsymbol{y}^i)$ for all $i = 1, ..., N$. If the communication graph is large, the measurement sharing becomes infeasible, as it scales with the number of measurements in the system. Here, the lowest possible bandwidth is achieved by the WDKF, which is likely

TABLE II

COMMUNICATION BANDWIDTH FOR THE DISTRIBUTED FILTERS, WITH NUMBERS RELATED TO $\mathcal{G}$ IN FIG. 1 WHEN $(\dim(\boldsymbol{y}^i), \dim(\boldsymbol{x})) = (2, 3)$.

Method	Communication bandwidth	For $\mathcal{G}$ in Fig. 1
P-CDKF-1	$D(\dim(\boldsymbol{x})+1)\dim(\boldsymbol{x})$	$12D$
P-CDKF-∞	$DN_c(\dim(\boldsymbol{x})+1)\dim(\boldsymbol{x})$	$240D$
FDKF-1	$D(\dim(\boldsymbol{x})+1)\dim(\boldsymbol{x})$	$12D$
FDKF-∞	$DN_c(\dim(\boldsymbol{x})+1)\dim(\boldsymbol{x})$	$240D$
WDKF	$D\dim(\boldsymbol{x})$	$3D$
SDKF	$< D\dim(\boldsymbol{y})$	$< 18D$
KF-∞	$D\sum_{i=1}^{N}(N-1)\dim(\boldsymbol{y}^i)$	$< 144D$

to perform better in the context of cross-correlated noise as it selects the weights to minimize a posterior MSE. However, we stress that for small to moderately sized graphs, such as in Fig. 1, the optimal SDKF can typically be implemented at a slight increase in communication bandwidth (which differs among the edges) when compared to the CDKF, FDKF, and WDKF. Furthermore, note that for the example graph in Fig 1, the communication bandwidth of sharing all of the estimates is comparable to running the CDKF and FDKF at $N_c < \frac{144D}{12D} = 12$ iterations. However, the global KF (KF-∞) achieves an equality in the BCRB, unlike the CDKF-∞ and FKDF-∞, which assume that the noise is not correlated. The KF-∞ will outperform these methods in terms of MSE.

C. Numerical Example With Dynamics

To illustrate the potential of the SDKF, consider a graph with $N = 9$ vertices, $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{0}_{3\times 1}, \boldsymbol{I})$, $\boldsymbol{C} \in \mathbb{R}^{45\times 3}$ such that each element $[\boldsymbol{C}]_{ij} \sim \mathcal{N}(0, 1)$, and $\boldsymbol{R} \in \mathbb{R}^{45\times 45}$ is a random dense positive definite and symmetric matrix. The states evolve as a linear system characterized by

$$\boldsymbol{A} = 0.98\begin{bmatrix} 1 & 0.1 & 0.2 \\ 0 & 1 & 0.1 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{Q} = \begin{bmatrix} 0.12 & 0.02 & -0.03 \\ 0.02 & 0.15 & -0.01 \\ -0.03 & -0.01 & 0.19 \end{bmatrix}.$$

We consider all of the defined filters, including:

(i) P3-CDKF-1 (consensus, $N_c = 1$, Metropolis-Hastings);
(ii) P3-CDKF-∞ (consensus, Metropolis-Hastings);
(iii) FDKF-1 (consensus, $N_c = 1$, CI);
(iv) FDKF-∞ (consensus, iterated CI);
(iv) WDKF (offline optimization over 20 iterations);
(vi) SDKF as defined in Algorithm 1.

For the particular realization, $\lambda_m(\boldsymbol{M}) = 8.85 \cdot 10^{-3}$. Thus, the inequality in (19) is strict, and none of the methods that disregard the measurement noise correlations (CDKFs, FDKFs) can attain their theoretical BCRBs. To verify this, in addition to computing the estimates, we also compute the true estimation error covariance. As the methods in (i)–(iv) do not have a consistent covariance estimate, due to the introduced noise correlations, the MSE of these estimates is computed empirically from 10^3 Monte-Carlo runs. Finally, we compute the BCRBs when only using local information (removing all communication with the other nodes) as $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k^i|\boldsymbol{y}_{0:k})$, and when using the global information without any delays as $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k^i|\mathcal{Y}_k)$. When considering the communication graph in Fig. 1, for a specific realization of the problem, the resulting states and estimates are shown in Fig. 2. In addition, the MSE of the estimates in node $\mathcal{V}_5$ is shown as a function of time against the BCRBs. In Fig. 3, the asymptotic estimate covariances and BCRBs are depicted at the terminal time $T = 100$ for three graphs:

- $\mathcal{G}_1$ is the graph in Figure 1;
- $\mathcal{G}_2$ is bipartite, with all nodes communicate with $\mathcal{V}_1$;
- $\mathcal{G}_3$ is a path graph ($\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ are connected).

When studying the MSEs of the node in $\mathcal{V}_5$ for the simulation run with $\mathcal{G}_1$, the CDKF and FDKF filters yield very similar MSEs. However, when assuming a 1-delay communication (full) the CDKF-1 and FDKF-1 these filters are outperformed by both the WDKF and SDKF, and when con-

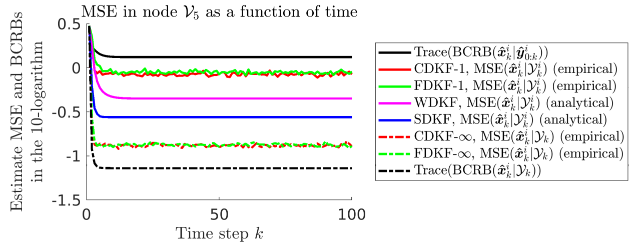

Fig. 2. MSE in time for the node $\mathcal{V}_5$ when considering the graph in Fig 1. The filters that operate with a delay associated with communication over the edges are drawn in full, and the filters which communicate without such constraints are dash-dotted. The figure also shows the traces of the associated BCRBs when using local (black full) and global (black dashed) information without delays. Being the MMSE estimator, SDKF (blue) achieves a tight BCRB under the communication delay constraint, thereby outperforming all other filters operating under the same communication constraints.

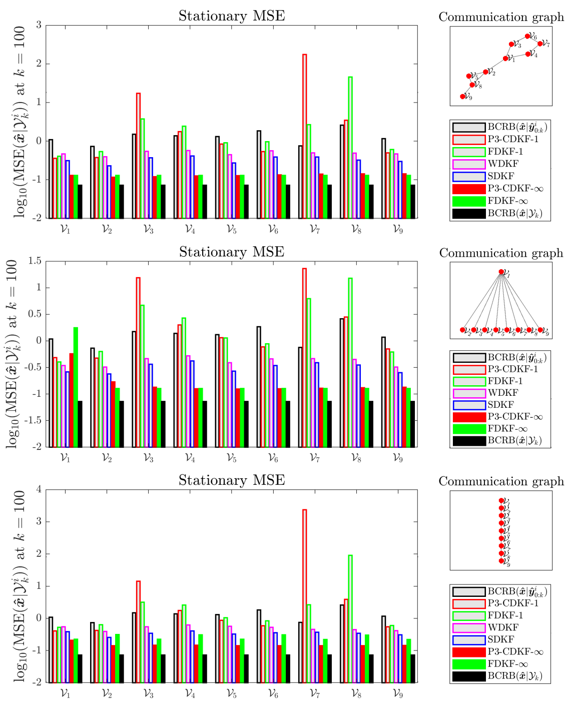

Fig. 3. Stationary MSE in the nodes for various graphs. *Top:* The communication graph in Fig 1 (as such, $\mathcal{V}_5$ in the top-most subplot depicts the MSEs in Fig. 2 at a time-step $k = 100$). *Center:* A bipartite graph where all of the nodes communicate through $\mathcal{V}_1$. *Bottom:* A graph with $N = 9$ nodes connected by 8 edges where the maximum delay is $K_1 = K_9 = 8$.

sidering delay-free communication (dash-dotted), the CDKF-∞ and FDKF-∞ do not come close to the corresponding BCRB (black). As the SDKF attains the BCRB for the 1-delay communication case, we see that it outperforms all other 1-delay approaches, and in this node the gaps are significant at all times. These conclusions do not just apply to $\mathcal{V}_5$ in $\mathcal{G}_1$, but to all vertices in all graphs (see Fig. 3).

When studying the stationary MSE in the 1-delay communication case (see Fig. 3), for all of the nodes, and for all graphs, the SDKF yields the lowest MSE. The MSE gap between the SDKF and WDKF is roughly the same in all cases, but the MSE of the CDKF-1 and FDKF-1 varies greatly. In particular, irrespective of the communication graph, their MSE (red/green) is significantly worse than just running a KF using the local information (black). This is the case in all of the simulations. Therefore, we conclude that this performance degradation is more related to the realization of the measurement model (here the same in all three cases) than the communication graph (varies in the three cases).

Also, for the communication graph $\mathcal{G}_2$, the delay-free estimates do not achieve a consensus in node $\mathcal{V}_1$ in the maximum of $N_c = 30$ iterations, which implies that more consensus iterations are required. However, already with $N_c = 30$, the communication bandwidth of running a global KF is significantly lower (see Table II), and achieves a far lower stationary MSE in all cases. As such, when considering cross-correlated measurement noise and the three graphs $\mathcal{G}_1$–$\mathcal{G}_3$, there is no reason to implement any filter other than the SDKF (or possibly the WDKF if the bandwidth is constrained) under the 1-delay constraint. Similarly, there is no reason to implement a CDKF-∞ or a FDKF-∞ over a global KF under a measurement sharing scheme.

V. Conclusions

In this paper, we have analyzed the distributed estimation and filtering problem in the context of correlated measurement noise in terms of of BCRBs. This analysis provided several insights, which can be summarized as follows:

- The distributed methods should always be compared under the same assumptions on the communication delays, as the corresponding BCRBs differ.
- For the case where communication over an edge takes one time-step, no estimator can perform better than the SDKF, as it attains $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k^i|\mathcal{Y}_k^i)$ in all nodes.
- For the case where communication over an edge is instantaneous, no estimator can perform better than a global KF, as it attains $\mathrm{BCRB}(\hat{\boldsymbol{x}}_k^i|\mathcal{Y}_k)$ in all nodes.
- For small to moderately sized graphs (in the sense that $\bar{K}$ is small), there is little distinction between the communication bandwidth of the 1-delay schemes.
- For small to moderately sized graphs (in a quantitative sense), the CDKF and FDKF often use more bandwidth than a global KF if communication is delay free.

In terms of estimation accuracy, for moderately sized networks and with correlated measurements, there is never any reason for using a method other than the SDKF unless the difference in communication bandwidth and computational power is deemed significant enough to warrant the implementation of a WDKF. If the network is larger, such that the communication bandwidth cannot pass the measurements and the network topology is known, then the WDKF is likely favorable, and if the network topology is not known then CDKF or FDKF should be considered. However, in the context of correlated measurement noise, the CDKF and FDKF should be used with caution, as it may sometimes be favorable to just run a local KF in each node (as shown in Fig 3, analogous to Example 1). It is also worth noting that all of the considered filters, including the WDKF, are easily extended to a linear time-varying setting. Regardless of the application, the performance of the considered methods should be evaluated in relation to the BCRB computed by the SDKF when deciding on which method to implement.

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, 2013.

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

[3] P. Alriksson and A. Rantzer, "Distributed Kalman filtering using weighted averaging," in *17th International Symposium on Mathematical Theory of Networks and Systems, 2006: MTNS 2006*, 2006.

[4] ——, "Model based information fusion in sensor networks," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 4150–4155, 2008.

[5] O. Hlinka, O. Sluvciak, F. Hlawatsch, and M. Rupp, "Distributed data fusion using iterative covariance intersection," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1861–1865.

[6] D. Marelli, M. Zamani, M. Fu, and B. Ninness, "Distributed kalman filter in a network of linear systems," *Systems & Control Letters*, vol. 116, pp. 71–77, 2018.

[7] K. Ryu and J. Back, "Distributed Kalman-filtering: Distributed optimization viewpoint," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2640–2645.

[8] Y. Zhang, B. Chen, L. Yu, and D. W. Ho, "Distributed Kalman filtering for interconnected dynamic systems," *IEEE Transactions on Cybernetics*, 2021.

[9] A. Khodabandeh, P. J. Teunissen, and S. Zaminpardaz, "Consensus-based distributed filtering for GNSS," in *Kalman Filters-Theory for Advanced Applications*. InTech, 2018, pp. 273–304.

[10] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Prentice-Hall, Inc., 1993.

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

[12] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on signal processing*, vol. 46, no. 5, pp. 1386–1396, 1998.

[13] C. Fritsche, E. Özkan, L. Svensson, and F. Gustafsson, "A fresh look at Bayesian Cramér-Rao bounds for discrete-time nonlinear filtering," in *17th International Conference on Information Fusion (FUSION)*. IEEE, 2014, pp. 1–8.

[14] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005*. IEEE, 2005, pp. 63–70.

[15] S. Bandyopadhyay and S.-J. Chung, "Distributed Bayesian filtering using logarithmic opinion pool for dynamic sensor networks," *Automatica*, vol. 97, pp. 7–17, 2018.

[16] L. Chen, P. O. Arambel, and R. K. Mehra, "Estimation under unknown correlation: Covariance intersection revisited," *IEEE Transactions on Automatic Control*, vol. 47, no. 11, pp. 1879–1882, 2002.

[17] B. Noack, J. Sijs, and U. D. Hanebeck, "Inverse covariance intersection: New insights and properties," in *2017 20th International Conference on Information Fusion (Fusion)*. IEEE, 2017, pp. 1–8.

[18] M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming, version 2.1," 2014.