Optimal Kalman Filter With Information-Weighted Consensus

Shiraz Khan, *Student Member, IEEE*, Raj Deshmukh, and Inseok Hwang, *Member, IEEE*

***Abstract*—The use of wireless sensor networks for distributed state estimation has been a popular research topic in the signal processing community. However, there is a distinct lack of emphasis on formal derivation and optimality of distributed state estimation algorithms in the literature. Furthermore, many existing algorithms utilize unweighted average consensus filtering, which has been shown to lead to poor estimation performance in the presence of sensor agents that cannot make measurements due to environmental obstructions or sensor limitations. In this article, a novel distributed minimum mean-squared error estimator is developed by generalizing the Kalman consensus filter to incorporate consensus on a weighted directed graph. By employing weighted consensus, the algorithm is able to achieve a directional flow of information in heterogeneous sensor networks, leading to improved performance in the presence of sensors that have low observability. Unlike several existing algorithms, the proposed algorithm does not rely on approximations or ad hoc parameter tuning and achieves optimal performance in a fully distributed setting. Through numerical simulations, it is demonstrated that the proposed algorithm has a smaller mean-squared estimation error and is robust in the aforementioned scenarios.**

***Index Terms*—Consensus, distributed state estimation, multiagent systems, wireless sensor networks.**

I. Introduction

The use of sensors to quantify physical activities is a critical component of several applications, such as robotics, in which machines are required to observe their surroundings in order to fulfill certain objectives [1]. However, measurement data are usually corrupted with noise, due to various factors, such as uncertainties in the sensor parameters. State estimation algorithms solve this problem by utilizing the redundancy of information in measurements made over a duration of time to filter out unwanted noise. The Kalman filter [2] is a popular state estimation algorithm, which uses a Bayesian approach to state estimation and achieves minimum mean-squared estimation error (MMSE) performance in the single-agent case, by assuming that the system under observation is linear and measurement noise is Gaussian.

In decentralized state estimation, multiple sensors are connected to each other via communication channels, which introduce further redundancy in measurement information. Unlike centralized state estimation, where a central computer is required to process measurement data, the computational load in decentralized state estimation is shared by all sensors in the network [3]. Distributed state estimation refers to the special case where each sensor computes its estimate using only locally available information, making it more scalable, more energy-efficient, and less susceptible to cyberattacks than centralized estimation [4]. The class of algorithms studied in this article is that of consensus-based distributed estimation, in which all sensors are required to arrive at a consensus on their estimate of the state of the system. Consensus-based distributed state estimation algorithms are critical components of applications, such as the flocking of autonomous drones, where the agents of the network are required to perform tasks cooperatively in a safety-critical manner [5], [6].

Manuscript received 19 January 2022; revised 20 July 2022; accepted 26 October 2022. Date of publication 8 November 2022; date of current version 30 August 2023. Recommended by Associate Editor P. G. Mehta. *(Corresponding author: Inseok Hwang.)*

Shiraz Khan and Inseok Hwang are with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47906 USA (e-mail: shiraz.yellow@gmail.com; ihwang@purdue.edu).

Raj Deshmukh is with the Aptiv, Troy, MI 48098 USA (e-mail: rdeshmu@purdue.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TAC.2022.3220528.

Digital Object Identifier 10.1109/TAC.2022.3220528

A. Prior Work

The problem of distributed state estimation is often posed as a target-tracking problem (without loss of generality) in the literature. One of the first consensus-based algorithms to be developed within this problem setup was the distributed Kalman filter (DKF), which utilizes a combination of a micro-Kalman filter and a consensus filter at each sensor in the network [7]. The Kalman consensus filter (KCF) is a suboptimal distributed MMSE estimator that improved upon the performance of DKF and serves as a seminal work on the topic [8], [9]. Unlike DKF, the KCF algorithm performs well even when the system is not fully observable at each sensor, as long as it is collectively observable by the sensor network. However, the performance of KCF has been shown to be poor and insufficient when some sensors are naive (i.e., have no measurement information about the state vector) [10]. This scenario occurs in many real-world applications, such as in vision-based sensors with a limited field of view (FoV). This lapse in performance of KCF arises because the KCF algorithm uses unweighted consensus, i.e., each sensor assigns equal weightage to the information received from other sensors in the network. It has been shown that this fact can cause the KCF algorithm to diverge [11]. The advantages of weighted consensus filtering over unweighted consensus filtering have been well documented in the literature on distributed consensus [12].

Subsequently distributed estimation algorithms that incorporate weighted consensus include generalized KCF (GKCF) and Information weighted Consensus Filter (ICF) [10], [13] among others [14], [15], [16], [17]. However, these algorithms have one or more of the following drawbacks:

1) the use of approximations or ad hoc steps in their derivation;
2) contingency of estimation performance on indeterminate design parameters;
3) the requirement of multiple subiterations after each measurement;
4) the requirement of global information about the sensor network at each sensor.

Another class of algorithms that were developed over the past decade is based on embedded average consensus and diffusion [18], [19], [20]. The main drawback of these algorithms is that their consensus weights must be chosen a priori (via ad hoc tuning or offline optimization), with there being no way of selecting the optimal consensus weights in a fully distributed setting when the network topology is unknown. The drawbacks of existing algorithms limit their viability in real-world applications. For example, the use of consensus subiterations in the aforementioned algorithms makes them inefficient for energy-constrained applications [21] and leads to poor state estimation performance in the presence of communication delays [22].

Furthermore, a vast majority of distributed estimation algorithms, including KCF and GKCF, do not take into account the correlation between the estimates of connected sensors, which leads to the suboptimal estimation performance of these algorithms [9], [23]. Recently, in the work on the optimal KCF (OKCF) [24], [25], the suboptimality of the KCF algorithm was addressed and remedied by tracking the cross-covariance matrices between all sensors in the sensor network. However, the OKCF algorithm still suffers from poor estimation performance in the presence of naive sensors, owing to its incorporation of unweighted consensus [11].

B. Contributions

In order to simultaneously address the limitations of existing algorithms, with regard to their suboptimality, poor performance in heterogeneous sensor networks, and reliance on subiterations, we propose a novel distributed state estimation algorithm that combines Kalman filtering with matrix-weighted consensus filtering. By incorporating consensus on a weighted directed graph (WDG), the proposed algorithm enables sensors to selectively assign higher weightage to accurate information and lower weightage to inaccurate or redundant information. The proposed algorithm uses optimal estimation gains that minimize the total MMSE of the sensor network. Through numerical simulations, it is demonstrated that the proposed algorithm achieves a smaller mean-squared error and is more robust than KCF and OKCF in the presence of naive sensors.

The rest of the article is organized as follows: In Section I-C, the distributed state estimation problem is formulated. The proposed algorithm is derived in Section II and compared against existing algorithms in Section III. Simulation results for the target-tracking problem are presented in Section IV, in order to illustrate the superior performance of the proposed algorithm to existing algorithms. Finally, concluding remarks are presented in Section V.

C. Preliminaries

The distributed estimation problem consists of a discrete-time linear time-varying dynamical system, which describes the evolution of the targets that need to be tracked. The dynamical system is observed using a network of N sensors. The dynamical system and sensor observation model are represented mathematically as

$$\begin{aligned} x(k+1) &= A(k)x(k) + B(k)w(k) \\ z_i(k) &= H_i(k)x(k) + v_i(k), \quad i = 1, 2, \dots, N \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the state vector of the system, $x(0) = x_0$ is the initial condition, $w(k) \in \mathbb{R}^m$ is the system noise. $z_i(k) \in \mathbb{R}^p$ is the measurement of sensor i at time step k and $v_i(k) \in \mathbb{R}^p$ is the corresponding measurement noise. The noise vectors $w(k)$ and $v_i(k)$ are modeled as mutually independent white Gaussian random variables, having the following covariance matrices:

$$\begin{aligned} E[w(r)w(s)^T] &= Q(r)\delta_{rs} \\ E[v_i(r)v_j(s)^T] &= R_i(r)\delta_{rs}\delta_{ij}. \end{aligned} \tag{2}$$

Here, $E[\cdot]$ denotes the expectation operator and δ_{rs} is the Kronecker delta, i.e., $\delta_{rs} = 1$ if $r = s$, and $\delta_{rs} = 0$, otherwise.

It is assumed that the state is collectively observable by all the sensors in the network; however, no constraint is imposed on the observability of individual sensors. The connectivity of the sensor network can be modeled as a dynamic weighted graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$ at time step k, such that the set of vertices $\mathcal{V} = \{1, 2, \dots, N\}$ represents the sensors, and the set of directed edges $\mathcal{E}(k) \subseteq \mathcal{V} \times \mathcal{V}$ represents the communication links between sensors.

The *prior estimate*, denoted by $\bar{x}_i(k)$, is defined as the estimate of $x(k)$ computed at sensor i using the information available at time step $k-1$. The *posterior estimate*, denoted by $\hat{x}_i(k)$, incorporates the new information received by sensor i at time step k. Accordingly, the prior and posterior estimation errors are denoted as $\bar{\eta}_i = \bar{x}_i - x$ and $\hat{\eta}_i = \hat{x}_i - x$, respectively. The prior and posterior cross-covariance matrices of the estimation errors are

$$\begin{aligned} P_{i,j} &= E[\bar{\eta}_i \bar{\eta}_j^T] \\ M_{i,j} &= E[\hat{\eta}_i \hat{\eta}_j^T] \end{aligned} \tag{3}$$

respectively. This completes the problem setup. The development of the proposed algorithm is presented in Section II.

II. Algorithm Development

In the Kalman filter algorithm, the posterior estimate is updated by fusing the prior estimate with measurement information at each time step. The KCF combines the Kalman filtering logic with that of unweighted average consensus, to yield a highly scalable algorithm. Omitting the time step index k, the KCF posterior estimate update equation at sensor i is

$$\hat{x}_i = \bar{x}_i + K_i(z_i - H_i\bar{x}_i) + C_i \sum_{j \in \mathcal{N}_i} (\bar{x}_j - \bar{x}_i) \tag{4}$$

where K_i is called the Kalman gain, C_i is called the consensus gain, $\mathcal{N}_i$ denotes the set of neighbors of sensor i, and all quantities are evaluated at time step k. The prior estimate for the next time step, denoted as $\bar{x}_i^+$, is computed by propagating the posterior estimate as $\bar{x}_i^+ = A\hat{x}_i(k)$.

The limitation of KCF arises from the fact that at a sensor following the KCF update equation (4), a single consensus gain C_i is used to weigh the estimates of neighboring sensors. This inhibits the sensor agents from being able to selectively incorporate information having low uncertainty and discard information having high uncertainty.

A. Kalman Filtering With Weighted Consensus

In this article, a variant of KCF is proposed, called KCF for WDGs (KCF-WDG), which employs weighted consensus instead of unweighted consensus. The KCF-WDG update equation is

$$\hat{x}_i = \bar{x}_i + K_i(z_i - H_i\bar{x}_i) + \sum_{j \in \mathcal{N}_i} [C_{j,i}(\bar{x}_j - \bar{x}_i)] \tag{5}$$

where $\{C_{j,i} | j \in \mathcal{N}_i\}$ is the set of consensus gains at sensor i. Using the update equation (5), lower weightage can be assigned to the estimates of sensors with low observability, in order to improve the collective performance of the sensor network. Consequently, a sensor network utilizing the proposed update equation (5) can be represented as a WDG, such that the matrix-valued weight of the directed edge from vertex j to vertex i is $C_{j,i}$. It may be noted that the KCF update equation (4) arises as a special case of (5) when $C_{m,i} = C_{n,i}\ \forall m, n \in \mathcal{N}_i$, whereas (5) is the more general update equation inspired by the optimal Bayes filter [26].

The set of estimation gains at sensor i, following the update equation (5), is $\{K_i, C_{j,i} | j \in \mathcal{N}_i\}$. It is desirable to choose the optimal estimation gains that maximize the estimation performance of the sensor network. The total mean-squared error of the sensor network is a commonly used metric of accuracy of a distributed state estimation algorithm [9] and is defined as

$$\text{Total Mean-Squared Error} = \sum_{i=1}^{N} E[\|\hat{x}_i - x\|^2]. \tag{6}$$

Theorem 1: The distributed estimation algorithm following the update equation (5) minimizes the total mean-squared error (6) at each time step if and only if the following conditions are satisfied at all sensors in the network:

$$K_i = \left(P_{i,i}H_i^T + \sum_{s\in\mathcal{N}_i} C_{s,i}(P_{s,i} - P_{i,i})H_i^T\right)(R_i + H_iP_{i,i}H_i^T)^{-1} \tag{7}$$

$$\left(I - K_iH_i - \sum_{s\in\mathcal{N}_i} C_{s,i}\right)(P_{i,j} - P_{i,i}) = \sum_{s\in\mathcal{N}_i} C_{s,i}(P_{s,i} - P_{s,j}) \quad \forall j \in \mathcal{N}_i. \tag{8}$$

Proof: The proof of this result can be found in [11]. ■

B. Optimization of Estimation Gains

In order to make the optimality conditions presented in Theorem 1 tractable, the KCF-WDG update equation needs to be expressed in the block matrix form. The proposed update equation (5) can be rewritten as

$$\hat{x}_i = K_iz_i + \left(I - K_iH_i - \sum_{j\in\mathcal{N}_i} C_{j,i}\right)\bar{x}_i + \sum_{j\in\mathcal{N}_i} C_{j,i}\bar{x}_j. \tag{9}$$

Consider the second term of (9); it is the weight assigned by sensor i to its prior estimate. This motivates us to define

$$C_{i,i} = I - K_iH_i - \sum_{j\in\mathcal{N}_i} C_{j,i}. \tag{10}$$

The definition of $C_{i,i}$ is an auxiliary step that enables us to conduct the rest of the analysis succinctly. The set of estimation gains to be determined at sensor i is $\{C_{j,i} | j \in \mathcal{N}_i \cup \{i\}\}$. This set of gains can be written in the form of a vector, which will be referred to as the *estimation gain vector*, as

$$\mathcal{C}_i = \begin{bmatrix} C_{\mathcal{N}_i(1),i} & C_{\mathcal{N}_i(2),i} & \dots & C_{\mathcal{N}_i(|\mathcal{N}_i|),i} & C_{i,i} \end{bmatrix} \tag{11}$$

where $\mathcal{C}_i \in \mathbb{R}^{n\times n(|\mathcal{N}_i|+1)}$ and $\mathcal{N}_i(k)$ refers to the kth element in $\mathcal{N}_i$. We define the following block matrix, consisting of prior covariance matrices of the sensors in $\mathcal{N}_i \cup \{i\}$:

$$\mathcal{P}_i = \begin{bmatrix} P_{\mathcal{N}_i(1),\mathcal{N}_i(1)} & \dots & P_{\mathcal{N}_i(1),\mathcal{N}_i(|\mathcal{N}_i|)} & P_{\mathcal{N}_i(1),i} \\ \vdots & \ddots & & \vdots \\ P_{\mathcal{N}_i(|\mathcal{N}_i|),\mathcal{N}_i(1)} & & & P_{\mathcal{N}_i(|\mathcal{N}_i|),i} \\ P_{i,\mathcal{N}_i(1)} & \dots & P_{i,\mathcal{N}_i(|\mathcal{N}_i|)} & P_{i,i} \end{bmatrix}. \tag{12}$$

The inverse of this matrix, $\mathcal{P}_i^{-1}$, will be referred to as the *distributed information matrix*. Theorem 2 shows the relevance of the distributed information matrix to the proposed algorithm.

Theorem 2 (Optimal Estimation Gains for KCF-WDG): The distributed estimation protocol following the update equation (5) minimizes the total mean-squared error (6) at every time step if and only if, at each sensor i in the network, the estimation gain vector $\mathcal{C}_i$ is given by

$$\mathcal{C}_i = \tilde{C}_i\mathbb{1}^T\mathcal{P}_i^{-1} \tag{13}$$

$$\tilde{C}_i = (\mathbb{1}^T\mathcal{P}_i^{-1}\mathbb{1} + H_i^TR_i^{-1}H_i)^{-1} \tag{14}$$

where

$$\mathbb{1} = \left(\begin{bmatrix}1 & 1 & \dots & 1\end{bmatrix} \otimes I\right)^T$$

I is the identity matrix of rank n and "$\otimes$" denotes the Kronecker product operator.

Proof: The system of equations (8), which are necessary conditions for optimality at sensor i, can be rewritten using the definition of $C_{i,i}$ (10) as

$$\sum_{s\in\mathcal{N}_i'} C_{s,i}(P_{s,i} - P_{s,j}) = 0I \qquad j \in \mathcal{N}_i \tag{15}$$

where $\mathcal{N}_i' = \mathcal{N}_i \cup \{i\}$. Expressing (15) in the block matrix form, we have

$$\mathcal{C}_i\left(\begin{bmatrix} P_{\mathcal{N}_i(1),i} \\ P_{\mathcal{N}_i(2),i} \\ \vdots \\ P_{\mathcal{N}_i(|\mathcal{N}_i|),i} \\ P_{i,i}\end{bmatrix}\mathbb{1}^T - \mathcal{P}_i\right) = \mathbb{0}^T \tag{16}$$

where

$$\mathbb{0} = \left(\begin{bmatrix}0 & 0 & \dots & 0\end{bmatrix} \otimes I\right)^T.$$

Using the definition of $\mathcal{P}_i$, we note that

$$\begin{bmatrix} P_{\mathcal{N}_i(1),i} \\ P_{\mathcal{N}_i(2),i} \\ \vdots \\ P_{\mathcal{N}_i(|\mathcal{N}_i|),i} \\ P_{i,i}\end{bmatrix} = \mathcal{P}_i\left(\begin{bmatrix}0\\ \vdots \\ 0 \\ 1\end{bmatrix} \otimes I\right). \tag{17}$$

Equation (17) can be substituted in (16) to give

$$\mathcal{C}_i(\mathcal{P}_i\left(\begin{bmatrix}0\\ \vdots \\ 0 \\ 1\end{bmatrix} \otimes I\right)\mathbb{1}^T - \mathcal{P}_i) = \mathbb{0}^T. \tag{18}$$

Recalling the mixed-product property of the Kronecker product [27], for matrices A_1, A_2, B_1, and B_2 of appropriate dimensions, the following expression holds:

$$(A_1 \otimes A_2)(B_1 \otimes B_2) = (A_1B_1) \otimes (A_2B_2).$$

Using the mixed-product and associative properties of the Kronecker product in (18), we get

$$\mathcal{C}_i\mathcal{P}_i\left(\begin{bmatrix} -1 & 0 & \dots & 0 & 0 \\ 0 & -1 & \dots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \dots & -1 & 0 \\ 1 & 1 & \dots & 1 & 0\end{bmatrix} \otimes I\right) = \mathbb{0}^T. \tag{19}$$

The square matrix in (19) has a one-dimensional left nullspace

$$\ker_L\left(\begin{bmatrix} -1 & 0 & \dots & 0 & 0 \\ 0 & -1 & \dots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \dots & -1 & 0 \\ 1 & 1 & \dots & 1 & 0\end{bmatrix}\right) = \operatorname{span}\left(\begin{bmatrix}1 & 1 & \dots & 1\end{bmatrix}\right) \tag{20}$$

where $\ker_L(A) = \{x | x^TA = 0, x \in \mathbb{R}^m\}$. So the solutions of (19) are of the form

$$\mathcal{C}_i\mathcal{P}_i = \tilde{C}_i\left(\begin{bmatrix}1 & 1 & \dots & 1\end{bmatrix} \otimes I\right) \tag{21}$$

where $\tilde{C}_i$ is a scaling factor. When the measurement noise is nondegenerate (i.e., $R_i \succ 0$), the estimates of the sensors are not perfectly correlated with each other. Consequently, $\mathcal{P}_i$ is positive definite and invertible, giving us (13). In the case where $\mathcal{P}_i$ is ill-conditioned or singular (for instance, in the limit $k \to \infty$ when consensus is achieved), $\mathcal{P}_i^{-1}$ can be replaced by the corresponding pseudoinverse to obtain the unique closed-form solution to (21).

We now use the optimality condition (7) to determine $\tilde{C}_i$. Substituting (10) into (7), we have

$$\sum_{s\in\mathcal{N}_i'} C_{s,i}[I + (P_{s,i} - P_{i,i})H_i^T\Delta_i^{-1}H_i] = I - P_{i,i}H_i^T\Delta_i^{-1}H_i. \tag{22}$$

Rewriting (22) in terms of the estimation gain vector

$$\mathcal{C}_i \begin{bmatrix} I - P_{i,i}H_i^T\Delta_i^{-1}H_i \\ I - P_{i,i}H_i^T\Delta_i^{-1}H_i \\ \vdots \\ I - P_{i,i}H_i^T\Delta_i^{-1}H_i \end{bmatrix} + \mathcal{C}_i \begin{bmatrix} P_{\mathcal{N}_i(1),i}H_i^T\Delta_i^{-1}H_i \\ P_{\mathcal{N}_i(2),i}H_i^T\Delta_i^{-1}H_i \\ \vdots \\ P_{\mathcal{N}_i(|\mathcal{N}_i|),i}H_i^T\Delta_i^{-1}H_i \end{bmatrix} = I - P_{i,i}H_i^T\Delta_i^{-1}H_i \tag{23}$$

where $\Delta_i = R_i + H_iP_{i,i}H_i^T$. Substituting (13) and (17) into (23), we have

$$\tilde{C}_i\mathbb{1}^T\mathcal{P}_i^{-1}\mathbb{1}(I - P_{i,i}H_i^T\Delta_i^{-1}H_i) + \tilde{C}_i\mathbb{1}^T\left(\begin{bmatrix}0\\ \vdots \\ 0 \\ 1\end{bmatrix} \otimes I\right)H_i^T\Delta_i^{-1}H_i = I - P_{i,i}H_i^T\Delta_i^{-1}H_i \tag{24}$$

which can be simplified to

$$\tilde{C}_i = (\mathbb{1}^T\mathcal{P}_i^{-1}\mathbb{1} + ((H_i^T\Delta_i^{-1}H_i)^{-1} - P_{i,i})^{-1})^{-1}. \tag{25}$$

Using the Woodbury Matrix Identity [28], (25) can be shown to be equivalent to (14), which concludes the proof of Theorem 2. ■

C. OKCF for WDGs

Let the distributed information matrix [defined below (12)] be denoted as a block matrix $\mathcal{F}_i$, such that $\mathcal{F}_i = [F_{r,s}^i] = \mathcal{P}_i^{-1}$, where $r, s \in \mathcal{N}_i'$ and $F_{r,s}^i$ are the submatrices of $\mathcal{F}_i$. The sum of columns of $\mathcal{F}_i$ can be expressed as

$$\mathbb{1}^T\mathcal{P}_i^{-1} = \mathbb{1}^T\mathcal{F}_i = \left[\sum_{r\in\mathcal{N}_i'} F_{r,\mathcal{N}_i(1)}^i \quad \cdots \quad \sum_{r\in\mathcal{N}_i'} F_{r,i}^i\right] \tag{26}$$

and the sum of all elements of $\mathcal{F}_i$ is

$$\mathbb{1}^T\mathcal{P}_i^{-1}\mathbb{1} = \mathbb{1}^T\mathcal{F}_i\mathbb{1} = \sum_{r\in\mathcal{N}_i'}\sum_{s\in\mathcal{N}_i'} F_{r,s}^i. \tag{27}$$

Using (26) and (27) in the result of Theorem 2, we arrive at the closed-form expression for the optimal consensus gains at sensor i, which is

$$C_{j,i} = \left(\sum_{r\in\mathcal{N}_i'}\sum_{s\in\mathcal{N}_i'} F_{r,s}^i + H_i^TR_i^{-1}H_i\right)^{-1} \sum_{r\in\mathcal{N}_i'} F_{r,j}^i \tag{28}$$

and the expression for the optimal Kalman gain at sensor i, which is

$$K_i = \left(\sum_{r\in\mathcal{N}_i'}\sum_{s\in\mathcal{N}_i'} F_{r,s}^i + H_i^TR_i^{-1}H_i\right)^{-1} H_i^TR_i^{-1}. \tag{29}$$

Algorithm 1: Optimal KCF-WDG.

Given: $P_{i,j}(0) = P_{i,j,0}\ \forall\, i,j;\ \bar{x}_i(0) = \bar{x}_{i,0}\ \forall\, i;\ x(0) = x_0$;
At sensor i, time step k,

1: Measurement z_i is made.
2: Prior estimate $\bar{x}_i$ and covariance matrices $P_{i,j}$ are transmitted to all sensors connected to sensor i, i.e., sensors $j \in \mathcal{N}_i$.
3: The distributed information matrix $\mathcal{F}_i = [F_{r,s}] = \mathcal{P}_i^{-1}$ is evaluated at sensor i.
4: The optimal estimation gains at sensor i are computed using (28) and (29).
5: The posterior estimate of sensor i, $\hat{x}_i$, is computed by fusing the information available at sensor i, as

$$\hat{x}_i = \bar{x}_i + K_i(z_i - H_i\bar{x}_i) + \sum_{j\in\mathcal{N}_i} C_{j,i}(\bar{x}_j - \bar{x}_i)$$

and the posterior error covariance matrices are determined using (30).
6: The values of $P_{i,j}$ and $\bar{x}_i$ for the next time step are obtained by propagating the posterior quantities as

$$P_{i,j} \leftarrow AM_{i,j}A^T + BQB^T$$
$$\bar{x}_i \leftarrow A\hat{x}_i$$

The estimation error covariance can be obtained by substituting (9), (28), and (29) into (3), as

$$M_{i,j} = \tilde{C}_i\sum_{r\in\mathcal{N}_i'}\sum_{t\in\mathcal{N}_j'}\left(\left(\sum_{s\in\mathcal{N}_i'} F_{s,r}^i\right)P_{r,t}\left(\sum_{s\in\mathcal{N}_j'} F_{t,s}^j\right)\right)\tilde{C}_j^T + K_iR_iK_i^T\delta_{ij}. \tag{30}$$

Thus, we arrive at the Optimal KCF-WDG algorithm, summarized in Algorithm 1.

III. Comparison With Existing Algorithms

In this section, we draw comparisons and distinctions between the proposed Optimal KCF-WDG algorithm and other comparable algorithms in the literature. Similar to the proposed algorithm, the GKCF [13] was developed by extending the KCF [9] to incorporate weighted consensus. However, the extension in [13] was done based on intuitive reasoning, without emphasis on optimality. Since $P_{i,i}$ is a measure of the uncertainty of the estimate of the sensor i, $P_{i,i}^{-1}$ can be considered as a measure of the accuracy of the estimation of the sensor. This observation was used to design the consensus weights of GKCF, in which the prior estimates of the sensors are weighted by the inverse of their corresponding covariance matrices. GKCF does not take into account the cross-correlation between sensor estimates, $P_{i,j}$. On the other hand, the OKCF [24], [25] tracks $P_{i,j}$ for all pairs of sensors in the network, but does not utilize weighted consensus.

To the authors' knowledge, the existing literature on distributed estimation does not discuss the distributed information matrix $\mathcal{P}_i^{-1}$, which was introduced in Section II-B. Instead, since most existing distributed estimation algorithms are derived from centralized algorithms, they often consider a centralized variant of $\mathcal{P}_i^{-1}$, which is defined as follows. Consider a block matrix $\mathcal{P}$ consisting of all pairwise estimation error covariances of the sensor network, constructed as

$$\mathcal{P} = \left[P_{i,j}\right], \quad i,j \in \{1,2,\ldots N\}. \tag{31}$$

The inverse of this matrix, denoted as $\mathcal{F} = \mathcal{P}^{-1}$, is referred to as the *information matrix* in the literature on state estimation [10].

The information matrix is utilized in centralized maximum *a posteriori* (MAP) estimation, in which all sensors can communicate with a central computer. The centralized MAP estimate $\hat{x}_{\text{MAP}}$ is

$$\hat{x}_{\text{MAP}} = \left(\sum_{r\in\mathcal{V}}\sum_{s\in\mathcal{V}} F_{r,s} + \sum_{r\in\mathcal{V}} H_r^T R_r^{-1} H_r\right)^{-1} \left(\sum_{r\in\mathcal{V}}\sum_{i\in\mathcal{V}} F_{r,i}\bar{x}_i + \sum_{i\in\mathcal{V}} H_i^T R_i^{-1} z_i\right) \tag{32}$$

where $F_{r,s}$ are submatrices of $\mathcal{F}$ and $r,s\in\mathcal{V}$. The term $(\sum_{r\in\mathcal{V}}\sum_{s\in\mathcal{V}} F_{r,s} + \sum_{r\in\mathcal{V}} H_r^T R_r^{-1} H_r)^{-1}$ in (32) can be considered as a normalization factor. The ICF [10] was derived from the centralized MAP estimator (32), by extending the algorithm to the distributed case. Similarly, the Information-driven Fully DKF (IFDKF) [14] was derived using the centralized weighted least squares estimator. In both algorithms, the prior information matrix $\mathcal{P}^{-1}$ is approximated at each sensor through the use of consensus subiterations.

Having introduced the defining characteristics of these distributed estimation algorithms, a few points of comparison between them and the proposed algorithm (Optimal KCF-WDG) can be made, as follows.

1) The weights assigned to prior information in Optimal KCF-WDG sum to I at each sensor. This is consistent with the result in Bayesian inference, which states that the Bayesian posterior is a convex combination of the prior information, when the priors have Gaussian distributions [26]. On the other hand, the estimation gains of the KCF and ICF algorithms depend on design parameters, and do not sum to I.
2) The normalization factor of the Optimal KCF-WDG gains (28) and (29) has an expression that closely resembles that of the centralized MAP algorithm (32) and ICF.
3) In the ICF and IFDKF algorithms, the full information matrix $\mathcal{P}^{-1}$ is approximated using subiterations. On the other hand, the analysis in Section II-B highlights the relevance of the distributed information matrix $\mathcal{P}_i^{-1}$ in optimal distributed MMSE estimation. Furthermore, the proposed algorithm is able to track the exact value of $\mathcal{P}_i^{-1}$ without using any approximations.

IV. Numerical Simulations

In this section, the Optimal KCF-WDG algorithm is validated and its estimation performance is compared against those of existing algorithms through an illustrative numerical simulation. We consider the problem of target tracking using a network of cameras, which is an example of a distributed estimation scenario where the performance of the KCF algorithm has been shown to be poor [10]. The dynamical system being observed over the camera network has a four-dimensional state vector, $x(k)$, such that the first two entries of $x(k)$ correspond to the coordinates (in meters) of the target on the horizontal plane, and the last two entries correspond to its velocity vector. The state vector is updated according to the linear time-varying system represented by (1), having the following system matrices:

$$A(k) = \begin{cases} A_1, & k\in\{0,1,\ldots,20\}\bigcup\{41,\ldots,60\} \\ A_2, & k\in\{21,\ldots,40\}\end{cases} \tag{33}$$

where

$$A_1 = \begin{bmatrix} 1 & 0 & 0.05 & 0 \\ 0 & 1 & 0 & 0.05 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} 1 & 0 & 0.05 & 0 \\ 0 & 1 & 0 & 0.05 \\ 0 & 0 & 0.9945 & -0.1045 \\ 0 & 0 & 0.1045 & 0.9945 \end{bmatrix}$$

and $B = I_4$. The process noise covariance Q is a diagonal matrix whose diagonal entries are $\begin{bmatrix} 0.05 & 0.05 & 0.75 & 0.75 \end{bmatrix}$. The initial value of the state vector is $x_0 = \begin{bmatrix} 20 & 20 & -25 & 0 \end{bmatrix}^T$. It can be noted that the system matrix A_1 corresponds to motion along a straight line, having constant velocity; A_2 corresponds to an anticlockwise turn of constant angular velocity.

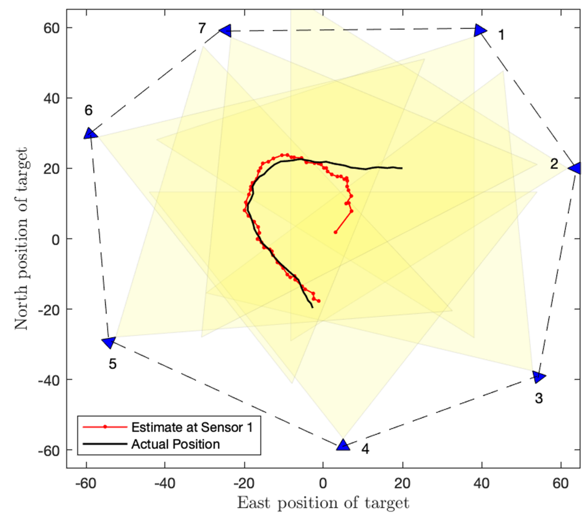

Fig. 1. Estimates at sensor 1 using the OKCF-WDG algorithm.

The target is observed using a sensor network consisting of seven camera sensors connected in a circle, as shown in Fig. 1. Cameras have a limited FoV; when the target moves out of the FoV of a camera, it is no longer observable by the camera, and the camera is said to be naive. The FoV of each camera (depicted by the yellow shaded region) is a triangle with an apex angle of 70° and height of 70 m, such that the apex of the triangle lies on the sensor.

Conventional cameras capture two-dimensional images and thereby cannot measure depth information directly [29]. Furthermore, sensors that are naive are unable to obtain a measurement of the target. To reflect these facts, the observation matrices of the cameras are chosen as follows:

$$H_i = \begin{cases} \begin{bmatrix} l_1^\perp & l_2^\perp & 0 & 0 \end{bmatrix}, & i\notin\mathcal{N}_{\text{Naive}} \\ \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}, & \text{otherwise} \end{cases}$$

where $\mathcal{N}_{\text{Naive}}$ indicates the set of sensors, which are naive at a given time step and $[l_1^\perp, l_2^\perp]$ is the unit vector perpendicular to the line of sight of the camera. The camera measurements have noise that can arise due to the rotation of the cameras in the wind [30]. The measurement noise covariance is chosen as $R_i = 20$ (which has the units m^2), corresponding to a camera rotation of about 4°.

In Fig. 1, the target trajectory is plotted along with the estimates of sensor 1 while using the proposed algorithm [OKCF for WDG (OKCF-WDG)]. It can be seen that the algorithm is able to estimate the position of the target with sufficient accuracy even when the target is outside the FoV of the sensor.

In Figs. 2 and 3, the root-mean-squared error (RMSE), defined as $E\|\hat{x}_i - x\|$, is plotted for sensors 1 and 4, respectively, using KCF, OKCF, and the proposed algorithm (OKCF-WDG). The results were averaged over 10 000 Monte Carlo simulations. Fig. 4 shows the total RMSE of the sensor network for each algorithm. It can be seen that the proposed algorithm has the lowest estimation error in each of the plots. In KCF, the estimation error of each sensor is higher during the time-steps when the corresponding sensor is naive. In contrast, in the proposed algorithm, there is relatively little difference in the performance of either sensor. The rate of convergence of the proposed algorithm is also seen to be higher than those of KCF and OKCF, which shows that the algorithm is able to effectively distribute the information across the network.

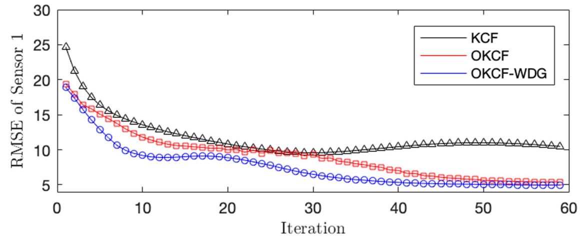

Fig. 2. RMSE of sensor 1, averaged over 10 000 Monte Carlo simulations.

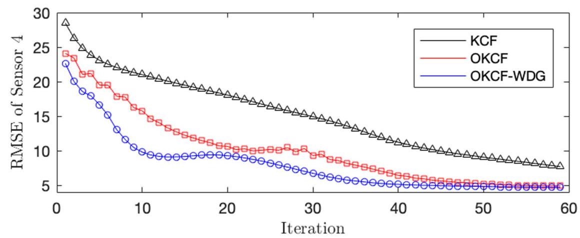

Fig. 3. RMSE of sensor 4, averaged over 10 000 Monte Carlo simulations.

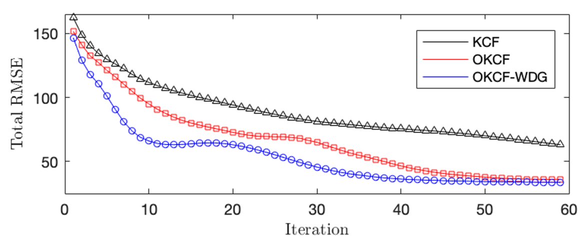

Fig. 4. Total RMSE of the sensor network, averaged over 10 000 Monte Carlo simulations.

V. Conclusion

In this article, a novel distributed estimation algorithm has been proposed, which extends the KCF to incorporate weighted consensus. The proposed algorithm, which we call OKCF-WDG, addresses certain limitations of existing distributed state estimation algorithms, such as their lack of robustness and optimality. Unlike the existing weighted consensus-based estimation algorithms, in which the estimation gains are proposed through intuitive arguments, the estimation gains of OKCF-WDG were optimized at each sensor to achieve the minimum mean-squared error in the distributed setting. The OKCF-WDG algorithm was compared with the KCF and OKCF algorithms through an illustrative numerical example of target tracking. Compared to the existing algorithms, OKCF-WDG was demonstrated to achieve a lower MMSE at all sensors.

Future work on OKCF-WDG would look into formalizing the estimation of the distributed information matrix such that it can be estimated efficiently in terms of computing and communication costs. A formal stability analysis of the algorithm will also be conducted.

References

[1] M. E. Stieber, M. McKay, G. Vukovich, and E. Petriu, "Vision-based sensing and control for space robotics applications," *IEEE Trans. Instrum. Meas.*, vol. 48, no. 4, pp. 807–812, Aug. 1999.

[2] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, pp. 35–45, Mar. 1960.

[3] B. Rao, H. F. Durrant-Whyte, and J. Sheen, "A fully decentralized multisensor system for tracking and surveillance," *Int. J. Robot. Res.*, vol. 12, no. 1, pp. 20–44, 1993.

[4] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *Int. J. High Perform. Comput. Appl.*, vol. 16, no. 3, pp. 293–313, 2002.

[5] R. Olfati-Saber and P. Jalalkamali, "Coupled distributed estimation and control for mobile sensor networks," *IEEE Trans. Autom. Control*, vol. 57, no. 10, pp. 2609–2614, Oct. 2012.

[6] Y. Wang, Q. Wu, Y. Wang, and D. Yu, "Consensus algorithm for multiple quadrotor systems under fixed and switching topologies," *J. Syst. Eng. Electron.*, vol. 24, no. 5, pp. 818–827, 2013.

[7] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proc. IEEE 44th Conf. Decis. Control*, 2005, pp. 8179–8184.

[8] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proc. IEEE 46th Conf. Decis. Control*, 2007, pp. 5492–5498.

[9] R. Olfati-Saber, "Kalman-consensus filter: Optimality, stability, and performance," in *Proc. IEEE 48th Conf. Decis. Control Held Jointly With 28th Chin. Control Conf.*, 2009, pp. 7036–7042.

[10] A. T. Kamal, J. A. Farrell, and A. K. Roy-Chowdhury, "Information weighted consensus filters and their application in distributed camera networks," *IEEE Trans. Autom. Control*, vol. 58, no. 12, pp. 3112–3125, Dec. 2013.

[11] S. Khan, R. Deshmukh, and I. Hwang, "Optimal Kalman consensus filter for weighted directed graphs," in *Proc. IEEE 58th Conf. Decis. Control*, 2019, pp. 7832–7837.

[12] M. H. Trinh, C. Van Nguyen, Y.-H. Lim, and H.-S. Ahn, "Matrix-weighted consensus and its applications," *Automatica*, vol. 89, pp. 415–419, 2018.

[13] A. T. Kamal, C. Ding, B. Song, J. A. Farrell, and A. K. Roy-Chowdhury, "A generalized Kalman consensus filter for wide-area video networks," in *Proc. IEEE 50th Conf. Decis. Control Eur. Control Conf.*, 2011, pp. 7863–7869.

[14] S. Wang, W. Ren, and Z. Li, "Information-driven fully distributed Kalman filter for sensor networks in presence of naive nodes," 2014, *arXiv:1410.0411*.

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

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

[17] S. Das and J. M. Moura, "Consensus innovations distributed Kalman filter with optimized gains," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 467–481, Jan. 2017.

[18] S. P. Talebi and S. Werner, "Distributed Kalman filtering and control through embedded average consensus information fusion," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4396–4403, Oct. 2019.

[19] C. Li, H. Dong, J. Li, and F. Wang, "Distributed Kalman filtering for sensor network with balanced topology," *Syst. Control Lett.*, vol. 131, 2019, Art. no. 104500.

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

[21] B. M. Sadler, "Fundamentals of energy-constrained sensor network systems," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 20, no. 8, pp. 17–35, Aug. 2005.

[22] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Math. Program.*, vol. 180, no. 1, pp. 237–284, 2020.

[23] J. Wu, A. Elser, S. Zeng, and F. Allgöwer, "Consensus-based distributed Kalman-Bucy filter for continuous-time systems," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 321–326, 2016.

[24] W. Li, Y. Jia, and J. Du, "Distributed Kalman consensus filter with intermittent observations," *J. Franklin Inst.*, vol. 352, no. 9, pp. 3764–3781, 2015.

[25] R. Deshmukh, C. Kwon, and I. Hwang, "Optimal discrete-time Kalman consensus filter," in *Proc. Amer. Control Conf.*, 2017, pp. 5801–5806.

[26] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.

[27] R. A. Horn, R. A. Horn, and C. R. Johnson, *Topics in Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1994.

[28] H. V. Henderson and S. R. Searle, "On deriving the inverse of a sum of matrices," *SIAM Rev.*, vol. 23, no. 1, pp. 53–60, 1981.

[29] Y. Wei, Z. Dong, and C. Wu, "Depth measurement using single camera with fixed camera parameters," *IET Comput. Vis.*, vol. 6, no. 1, pp. 29–39, 2012.

[30] Y. Liu et al., "Study of the error caused by camera movement for the stereo-vision system," *Appl. Sci.*, vol. 11, no. 20, 2021, Art. no. 9384.