2019 IEEE 58th Conference on Decision and Control (CDC)
Palais des Congrès et des Expositions Nice Acropolis
Nice, France, December 11-13, 2019

Optimal Kalman Consensus Filter for Weighted Directed Graphs

Shiraz Khan, Raj Deshmukh and Inseok Hwang

***Abstract*—The distributed estimation problem has proven to be a highly relevant topic today, due to its applicability in a wide variety of scenarios that do not accommodate a centralized supervisor. Decentralized algorithms can offer enhanced robustness and resilience to system failures and cyber-attacks. A seminal work on the topic was the development of the Kalman Consensus Filter (KCF), and recently the issue of the suboptimality of the KCF was addressed. However, in the KCF scheme, the sensor network is modeled as an unweighted undirected graph. This fact has been shown to severely degrade performance when certain assumptions on the network topology are not met, such as in the case of limited observability. Subsequent contributions in the field have implemented consensus based filters on directed graphs, but they either employ heuristic choices for the consensus gains or entail the sharing of information matrices between neighbors. In this paper, we address these issues by proposing an optimal distributed state estimation algorithm for weighted directed graphs. The proposed scheme is shown to be more versatile and offers critical performance improvements in scenarios where the KCF performs poorly. Specifically, we highlight the efficacy of the proposed algorithm in the presence of naïve sensors, through illustrative numerical examples.**

I. INTRODUCTION

Dynamical systems in the real world, as well as the sensors observing them, are seldom devoid of random disturbances. This necessitates the use of interconnected sensors that exchange information between them. Such an arrangement, called a sensor network, introduces a redundancy in the measurements which could be exploited to arrive at a much better estimate of the target state. The challenge then lies in devising an algorithm that fuses information from multiple sensors to arrive at the best possible estimate, while accounting for implementational constraints such as limited bandwidth and processing power.

The introduction of the Kalman Filtering logic provided a strong foundation for a metonymic class of estimation schemes, wherein the information fusion takes place at a centralized sensor [1][2]. Such centralized sensing systems have a wide array of limitations, including higher bandwidth requirements and high vulnerability to packet-losses and sensor failures [3][4]. Moreover, it is often infeasible to have a centralized computer in mobile applications such as drone-mounted sensors. In such scenarios where the sensors are mounted on autonomous agents, it becomes more practical to conduct the computation in a decentralized manner. This has been the motivation for the development of distributed estimation algorithms.

The Distributed Kalman Filter (DKF) uses localized Kalman filters to achieve concurrence between the estimates of all sensors [5][6]. Each sensor receives measurements broadcasted by the other sensors, then assimilates the information to improve its own estimate. The sensor network is said to then collectively observe the system's state. With the advent of such distributed estimation schemes, sensor networks became more scalable and deployment costs decreased considerably. The limitation of the standard DKF is in its requirement of a fully connected network, or all-to-all links. Consensus-based filters [6][7][8][9][10][11][12] relax that constraint by only sharing measurements between neighboring sensors, thereby reducing the communication overhead and further increasing scalability. In recent times, consensus-based techniques have been employed for a diverse variety of applications, such as target tracking [13], power distribution modelling [14][15], health monitoring [16], fault detection and estimation problems for safety-critical systems [17], and cooperative reconnaissance of unmanned aerial vehicles [18][19][20].

A notable contribution to the topic is that of the Kalman Consensus Filter (KCF), introduced in [9]. The KCF not only computes the system state estimate at each sensor, but also ensures that consensus is achieved between all the sensors, requiring only that the network be strongly connected.

While the KCF uses an approximated formulation of the update equations, which lowers bandwidth requirements, it is suboptimal. In our earlier work [12], an optimal formulation of the KCF was presented, where the error cross covariance matrices were used to compute globally optimal gains. However, in the KCF, the communication links are each given identical weights, notwithstanding the quality of estimation of each sensor. Consequently, the algorithm achieves consensus on an unweighted undirected graph, which does not allow directional flow of information with different weights. The restriction of the KCF to unweighted graphs enforces certain assumptions pertaining to the network topology as well, including the requirement of complete observability at each sensor. It performs poorly when these assumptions are not satisfied, such as in the presence of naïve sensors (a situation in which some sensors cannot observe the system), as was noted in [8].

Subsequently, there have been several contributions in distributed estimation which relax this limitation of the KCF. However, many of these algorithms assume the estimation errors of any two sensors to be uncorrelated [21] or require mul-

The authors are with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907, USA (email: khan243@purdue.edu, rdeshmu@purdue.edu, ihwang@purdue.edu)

978-1-7281-1398-2/19/$31.00 ©2019 IEEE

tiple consensus subiterations at each estimation step [8][22]. Moreover, there has been a significant lack of emphasis on optimality.

In this paper, we propose an optimal distributed state estimation algorithm which uses the Kalman consensus filtering logic in conjunction with weighted directed graphs. The proposed algorithm retains the communication complexity of the optimal KCF [12], but does not require the assumptions on the observability of the sensors. For instance, in the presence of sensor naïvety, the communication links directed outwards from a naïve sensor are given smaller weights, such that the collective estimation performance of the sensor network is not compromised. The mathematical framework presented in this paper also addresses the issue of suboptimality evident in the existing algorithms.

The organization of the rest of the paper is as follows: in Section II, we formulate the problem and Section III presents the development of the proposed distributed state estimation algorithm. We then demonstrate the performance of the proposed algorithm in Section IV, comparing the performance of the proposed algorithm to the existing algorithms. Finally, concluding remarks and some future work are discussed in Section V.

II. PROBLEM FORMULATION

The dynamical system to be estimated (referred to as target in this paper) is modeled as a linear time-varying discrete-time system:

$$x(k+1) = A(k)x(k) + B(k)w(k) \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the state, $w(k) \in \mathbb{R}^m$ is the system noise and $x(0) = x_0$, the initial condition.

The sensor agent connectivity at time k is given by a dynamic directed graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, and is assumed to be known. $\mathcal{V} = \{1, 2, \dots, N\}$ is the set of nodes (representing the N sensor agents) and $\mathcal{E}(k) \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges (representing the communication links). We assume that the communication link between any two sensors is two-way. As a consequence, the adjacency matrix of the graph is symmetric. The observation model of each sensor is

$$z_i(k) = H_i(k)x(k) + v_i(k) \qquad i = 1, 2 \dots N \tag{2}$$

where $z_i(k) \in \mathbb{R}^p$ is the measurement made by the i^{th} sensor and $v_i(k) \in \mathbb{R}^p$ is its measurement noise. $w(k)$ and $v_i(k)$ are each modeled as white Gaussian noise having the following statistical properties:

$$\begin{aligned} E[w(r)w(s)^T] &= Q(r)\delta_{rs} \\ E[v_i(r)v_j(s)^T] &= R_{i,j}(r)\delta_{rs} \end{aligned} \tag{3}$$

where $E[\cdot]$ is the expectation operator, and δ_{rs} is the Kronecker delta, i.e., $\delta_{rs} = 1$ if $r = s$, and $\delta_{rs} = 0$, otherwise. Given the history of measurements made by sensor i, $Z_i(k) \coloneqq \{z_i(0), z_i(1), \dots, z_i(k)\}$, we define:

$$\begin{aligned} \hat{x}_i(k) &= E[x(k) \mid Z_i(k)] \\ \bar{x}_i(k) &= E[x(k) \mid Z_i(k-1)] \end{aligned} \tag{4}$$

where $\hat{x}_i(k)$ and $\bar{x}_i(k)$ are referred to as the posterior and prior estimate of the target state, respectively. For brevity, we will omit the time index henceforth, referring to time step k unless otherwise specified.

$\hat{\eta}_i = \hat{x}_i - x$ and $\bar{\eta}_i = \bar{x}_i - x$ are the posterior and prior estimation errors at sensor i, respectively. The cross covariance matrices of the estimation errors are represented as:

$$\begin{aligned} M_{i,j} &= E[\hat{\eta}_i \hat{\eta}_j^T] \\ P_{i,j} &= E[\bar{\eta}_i \bar{\eta}_j^T] \end{aligned} \tag{5}$$

The Kalman Consensus Filter (KCF) [9] has the update equation:

$$\hat{x}_i = \bar{x}_i + K_i(z_i - H_i\bar{x}_i) + C_i \sum_{j \in \mathcal{N}_i} (\bar{x}_j - \bar{x}_i) \tag{6}$$

where C_i is the Kalman consensus gain for sensor i. $\mathcal{N}_i$ is the set of indices of the first-degree neighbors of sensor i, which includes all the sensors that can be reached from sensor i in one hop along the network. The second term of the expression is from the conventional Kalman filter. The third term in (6) aims to achieve a collaborative consensus between the estimates of connected sensors. The optimal expressions for K_i and C_i have been presented in our earlier work [12].

Considering that the KCF update equation in (6) has a single consensus gain C_i for sensor i, all the communication links of the sensor are equally weighted. Hence, the KCF achieves consensus on an unweighted graph. As noted in Section I, this could cause the performance of the KCF to be critically impacted during the presence of sensor failures or limited observability.

This motivates the development of an optimal distributed state estimation algorithm that does not suffer from such limitations. In the following section, we propose an optimal distributed estimation algorithm that achieves consensus on a weighted directed graph, by combining the Kalman Consensus Filtering logic with the concept of matrix-weighted consensus. Unlike the KCF, the proposed algorithm does not involve the assumptions regarding the observability of individual sensors, requiring only that the system be collectively observable by the sensor network.

III. ALGORITHM DEVELOPMENT

Consider a weighted directed graph, such as the one depicted in Fig. 1.

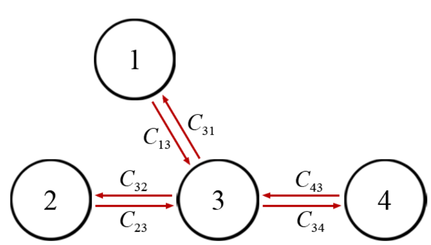

Fig. 1. An example of a weighted directed graph

In the graph, $C_{j,i} \in \mathbb{R}^{n \times n}$ is the matrix-valued weight of the directed edge from node j to node i. Since the nodes of this graph represent the sensor agents in the given network topology, we refer to this weight as the consensus gain for the communication link from sensor j to sensor i. The KCF update equation (6) can then be generalized to incorporate matrix-weighted consensus as:

$$\hat{x}_i = \bar{x}_i + K_i(z_i - H_i\bar{x}_i) + \sum_{j\in\mathcal{N}_i}[C_{j,i}(\bar{x}_j - \bar{x}_i)] \tag{7}$$

Note that (6) is the special case of (7) when $C_{m,i} = C_{n,i}$ for $m, n \in \mathcal{N}_i$. Matrix-weighted consensus has been studied in detail in [23].

Theorem 1. *The optimal gains (for sensor i) for the distributed state estimation algorithm that uses the update equation (7) and minimizes the mean squared estimation error $\sum_{i=1}^{N} E[\|\hat{x}_i - x\|^2]$ are given by:*

$$\begin{aligned} K_i &= (P_{i,i} + \sum_{r\in\mathcal{N}_i}[C_{r,i}(P_{r,i} - P_{i,i})])H_i^T\Delta_i^{-1} \\ \mathcal{C} &= \mathcal{B}\mathcal{A}^{-1} \end{aligned} \tag{8}$$

where the consensus gains are represented via a linear matrix equality. $\mathcal{C} = col(\mathcal{C}_1, \mathcal{C}_2 \dots \mathcal{C}_{|\mathcal{N}_i|})$ and $\mathcal{B} = col(\mathcal{B}_1, \mathcal{B}_2 \dots \mathcal{B}_{|\mathcal{N}_i|})$ are block column matrices, and $\mathcal{A}$ is a block matrix, of the form

$$\begin{aligned} \mathcal{C}_r &= C_{\mathcal{N}_i(r),i} \\ \mathcal{B}_r &= (I - P_{i,i}H_i^T\Delta_i^{-1}H_i)(P_{i,\mathcal{N}_i(r)} - P_{i,i}) \\ \mathcal{A}_{r,s} &= (P_{\mathcal{N}_i(r),i} - P_{i,i})H_i^T\Delta_i^{-1}H_i(P_{i,\mathcal{N}_i(s)} - P_{i,i}) \\ &\quad - [P_{\mathcal{N}_i(r),\mathcal{N}_i(s)} - P_{\mathcal{N}_i(r),i} - P_{i,\mathcal{N}_i(s)} + P_{i,i}] \end{aligned} \tag{9}$$

where $\Delta_i = R_i + H_iP_{i,i}H_i^T$, $\mathcal{N}_i(r)$ is the r^{th} element of the set $\mathcal{N}_i$, and $|\mathcal{N}_i|$ denotes the cardinality of $\mathcal{N}_i$.

Proof. For brevity, let us denote $M_{i,i}$, $P_{i,i}$ and $R_{i,i}$ as M_i, P_i and R_i, respectively. To find the expression for $M_{i,j}$, let us consider the following relation between the prior and posterior estimation errors of the sensors i and j. Subtracting x from both sides of (7) for sensors i and j,

$$\begin{aligned} \hat{\eta}_i &= F_i\bar{\eta}_i + \sum_{r\in\mathcal{N}_i}[C_{r,i}(\bar{\eta}_r - \bar{\eta}_i)] + K_iv_i \\ \hat{\eta}_j &= F_j\bar{\eta}_j + \sum_{s\in\mathcal{N}_j}[C_{s,j}(\bar{\eta}_s - \bar{\eta}_j)] + K_jv_j \end{aligned} \tag{10}$$

where $F_i = I - K_iH_i$. The cross covariance $M_{i,j}$ can be written as:

$$\begin{aligned} M_{i,j} &= F_iP_{i,j}F_j^T + F_i\sum_{s\in\mathcal{N}_j}E[\bar{\eta}_i(\bar{\eta}_s^T - \bar{\eta}_j^T)C_{s,j}^T] \\ &\quad + \sum_{r\in\mathcal{N}_i}E[C_{r,i}(\bar{\eta}_r - \bar{\eta}_i)\bar{\eta}_j^T]F_j^T + K_iR_{i,j}K_j^T \\ &\quad + \sum_{r\in\mathcal{N}_i}\sum_{s\in\mathcal{N}_j}E[C_{r,i}(\bar{\eta}_r - \bar{\eta}_i)(\bar{\eta}_s^T - \bar{\eta}_j^T)C_{s,j}^T] \end{aligned} \tag{11}$$

Substituting the outer product terms with the corresponding covariance matrices in (5), we have

$$\begin{aligned} M_i &= F_iP_iF_i^T + \sum_{r\in\mathcal{N}_i}[C_{r,i}(P_{r,i} - P_i)]F_i^T \\ &\quad + F_i\sum_{s\in\mathcal{N}_i}[(P_{i,s} - P_i)C_{s,i}^T] + K_iR_iK_i^T \\ &\quad + \sum_{r\in\mathcal{N}_i}\sum_{s\in\mathcal{N}_i}[C_{r,i}(P_{r,s} - P_{r,i} - P_{i,s} + P_i)C_{s,i}^T] \end{aligned} \tag{12}$$

The total mean squared estimation error equals $\sum_{i=1}^{N} E[\|\hat{x}_i - x\|^2]$, which is $\sum_{i=1}^{N} tr(M_i)$. Considering that the gains at sensor i only influence the value of M_i, the optimal K_i and $C_{j,i}$ are the solutions to

$$\frac{\partial tr(M_i)}{\partial K_i} = 0, \qquad \frac{\partial tr(M_i)}{\partial C_{j,i}} = 0 \tag{13}$$

Applying the matrix trace operator $tr(\cdot)$ to (12), we get

$$\begin{aligned} tr(M_i) &= tr(P_i) - 2tr(P_iH_i^TK_i^T) + tr(K_iH_iP_iH_i^TK_i^T) \\ &\quad + 2tr[(I - K_iH_i)\sum_{s\in\mathcal{N}_i}(P_{i,s} - P_i)C_{s,i}^T] \\ &\quad + tr(\sum_{r\in\mathcal{N}_i}\sum_{s\in\mathcal{N}_i}[C_{r,i}(P_{r,s} - P_{r,i} - P_{i,s} + P_{i,j})C_{s,i}^T]) \\ &\quad + tr(K_iR_iK_i^T) \end{aligned} \tag{14}$$

Taking the partial derivative of $tr(M_i)$ with respect to K_i,

$$\begin{aligned} \frac{\partial tr(M_i)}{\partial K_i} &= -2P_iH_i^T + 2K_i(H_iP_iH_i^T) \\ &\quad - 2\sum_{r\in\mathcal{N}_i}[C_{r,i}(P_{r,i} - P_i)]H_i^T + 2K_iR_i = 0 \end{aligned} \tag{15}$$

From (15), the optimal Kalman gain K_i in terms of $C_{j,i}$ is obtained as:

$$K_i = (P_i + \sum_{r\in\mathcal{N}_i}[C_{r,i}(P_{r,i} - P_i)])H_i^T\Delta_i^{-1} \tag{16}$$

where $\Delta_i = R_i + H_iP_iH_i^T$. Substituting the optimal Kalman gain determined in (16) into (14), and taking the derivative with respect to $C_{j,i}$, we have

$$\begin{aligned} \frac{\partial tr(M_i)}{\partial C_{j,i}} &= (I - P_iH_i^T\Delta_i^{-1}H_i)(P_{i,j} - P_i) \\ &\quad - \sum_{s\in\mathcal{N}_i}C_{s,i}(P_{s,i} - P_i)H_i^T\Delta_i^{-1}H_i(P_{i,j} - P_i) \\ &\quad + \sum_{s\in\mathcal{N}_i}C_{s,i}(P_{s,j} - P_{s,i} - P_{i,j} + P_i) = 0 \end{aligned} \tag{17}$$

Equation (17) is valid for any sensor j in $\mathcal{N}_i$. Thus, there are $|\mathcal{N}_i|$ independent linear equations for $|\mathcal{N}_i|$ variables. The system of equations describing the optimal consensus gains at each sensor can now be rewritten as a linear matrix equation, to arrive at the matrix equality presented in (8). This concludes the proof. □

The resulting optimal distributed state estimation algorithm is summarized in Algorithm 1.

Algorithm 1 Optimal KCF for Weighted Directed Graphs (at sensor i, time step k)

Given: $M_i(0) = M_0$, $x(0) = x_0$, $\mathcal{N}_i$ (the set of neighbors of sensor i)

1: Measurement z_i is obtained.

2: Prior estimate $\bar{x}_i$, as well as cross covariance matrices are transmitted to the first-degree neighbors, i.e., sensors $j \in \mathcal{N}_i$.

3: Messages transmitted by the neighboring sensors are received at sensor i.

4: Received information is assimilated to obtain the consensus gains via the matrix equality

$$\mathcal{C} = \mathcal{B}\mathcal{A}^{-1}$$

where $\mathcal{C}_m = C_{\mathcal{N}_i(m),i}$ is the block column matrix of consensus gains. $\mathcal{A}$ and $\mathcal{B}$ are defined as in (9). The Kalman gains K_i are given by

$$K_i = (P_i + \sum_{r \in \mathcal{N}_i} [C_{r,i}(P_{r,i} - P_i)])H_i^T \Delta_i^{-1}$$
$$\Delta_i = R_i + H_i P_i H_i^T$$

5: The optimal gains are used in the following update equations to determine the posterior estimate ($\hat{x}_i$) for sensor i, given by

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

as well as the cross covariance matrices, given by

$$\begin{aligned} M_{i,j} = {} & F_i P_{i,j} F_j^T + \sum_{r \in \mathcal{N}_i} C_{r,i}(P_{r,j} - P_{i,j})F_j^T \\ & + F_i \sum_{s \in \mathcal{N}_j} (P_{i,s} - P_{i,j})C_{s,i}^T + K_i R_{i,j} K_j^T \\ & + \sum_{r \in \mathcal{N}_i} \sum_{s \in \mathcal{N}_i} [C_{r,i}(P_{r,s} - P_{r,j} - P_{i,s} + P_{i,j})C_{s,j}^T] \end{aligned}$$

6: $P_{i,j}$ and $\bar{x}_i$ are updated as

$$\begin{aligned} P_{i,j} & \leftarrow A M_{i,j} A^T + BQB^T \\ \bar{x}_i & \leftarrow A\hat{x}_i \end{aligned}$$

In the next section, we implement the proposed algorithm to track the state of a dynamic target using a sensor network, and compare its performance to the existing distributed state estimation algorithms.

IV. SIMULATIONS

To demonstrate the efficacy of the proposed distributed state estimation algorithm, we consider an illustrative example where a target moves in a circular path over a sensor network. The target is subject to system noise that causes its trajectory to be perturbed. Such a system is described by (1), with the state-space matrices

$$A = \begin{bmatrix} \cos(\pi/200) & -\sin(\pi/200) \\ \sin(\pi/200) & \cos(\pi/200) \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{18}$$

and initial condition $x_0 = \begin{bmatrix} 20 & 0 \end{bmatrix}^T$. We consider two different scenarios for the sensor network, each highlighting a different aspect of the proposed algorithm.

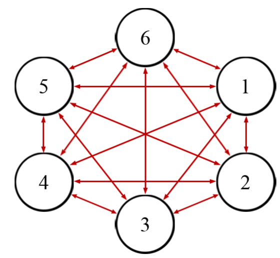

Fig. 2. Scenario I: Fully connected topology

Scenario I: Consider the fully connected network of 6 sensors shown in Fig. 2. In this scenario, there is a directed communication link between every pair of sensors. Each sensor follows the observation model presented in (2). The sensor noise and system noise have the specifications

$$R_{i,j} = \begin{bmatrix} \delta_{ij} & 0 \\ 0 & \delta_{ij} \end{bmatrix} \qquad Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{19}$$

respectively.

The dynamical system presented in (18) is simulated for 500 time steps, while the sensor network iteratively uses the proposed algorithm (Algorithm 1) to estimate the state of the target. Figure 3 shows the history of state estimates at sensor 3 within the fully connected network of 6 sensors shown in Fig. 2.

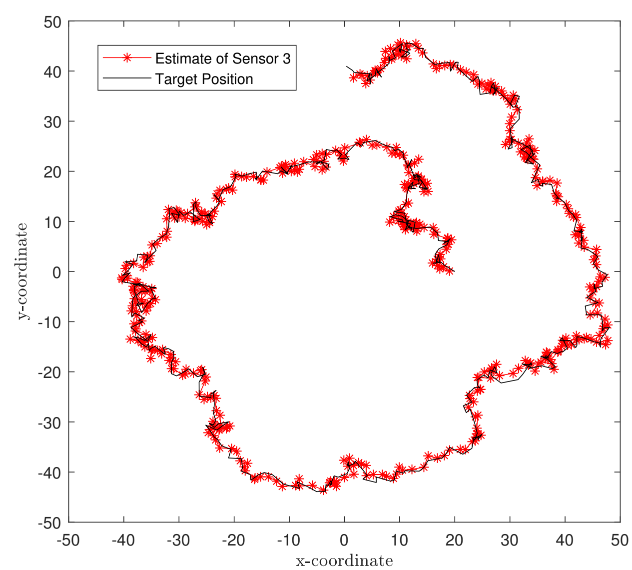

Fig. 3. History of target state estimates (at a sensor) juxtaposed with the actual target trajectory, for the fully connected sensor network shown in Fig. 2, using the proposed algorithm.

The steady state values of the Kalman gain and the consensus gains for the proposed algorithm are,

$$K_i = 0.565 I_2 \quad C_{j,i} = 0.0725 I_2 \quad \forall i, j \in \{1, 2 \ldots 6\}$$

where I_2 is the identity matrix of order 2. We also simulate scenario I using the optimal KCF algorithm, in which the sensor network in Fig. 2 is modeled as an unweighted undirected graph. The steady state values of the gains for the optimal KCF algorithm are

$$K_i = 0.565 I_2 \quad C_i = 0.0725 I_2 \quad \forall i \in \{1, 2 \ldots 6\}$$

Hence, when all the sensors have statistically identical observation models, the steady state performance of the proposed algorithm is consistent with that of the optimal KCF scheme.

While Fig. 3 shows that the proposed distributed state estimation algorithm succeeds in estimating the target state at sensor 3, the fully connected topology is an ideal scenario. To exemplify the superior performance of the proposed algorithm, we also consider a sensor configuration where the direction of flow of information plays an important role in the estimation process, and therefore the communication links have non-identical weights.

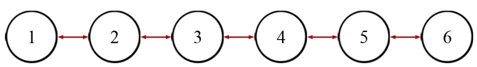

Fig. 4. Scenario II: Chain topology

Scenario II: In this simulation, the sensor network has a chain topology (as presented in Fig. 4), in which there are 6 sensors connected in series. The sensors at either end (1 and 6) have only one first-degree neighbor (2 and 5, respectively). Out of a total simulation time of 60 time steps, for time steps $0-20$ and $40-60$, all the sensors observe the target with statistically identical sensor noise, as specified in (19). For the period of $20-40$ steps, sensors 4, 5 and 6 have disproportionately high sensor noise (with $R_i = 10^6 I_2$, for $i \in \{4, 5, 6\}$). These sensors are essentially naïve, as they are oblivious of the target. In this scenario, it is to be expected that the information in the network must propagate from the functioning sensors to the naïve sensors (i.e., from left to right in Fig. 4), for optimal performance. A directional flow of information of this manner may be achieved using the proposed algorithm, but not with the KCF algorithm, as the latter uses an unweighted graph for the consensus logic.

We have performed 10,000 Monte Carlo simulations, and in Fig. 5, the mean squared estimation error is plotted for the KCF [9], optimal KCF [12] and the proposed algorithm (which is denoted as KCF-WDG in the plot), for the simulation specifications given in Scenario II. The data points within the shaded region correspond to the time period when sensors 4, 5 and 6 are naïve. It can be noted that within this region, the mean squared estimation errors of the suboptimal KCF and optimal KCF increase monotonously. The error of the proposed algorithm plateaus within a few time steps and remains bounded thereafter.

To analyze why the proposed algorithm improves performance in Scenario II, let us consider the consensus gains computed at sensor 4. For the proposed algorithm, these gains are $C_{3,4}$ and $C_{5,4}$, which are used to weigh the information

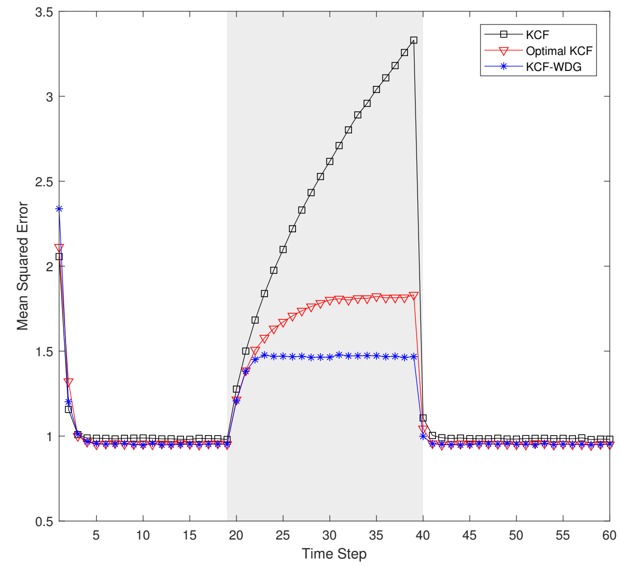

Fig. 5. Evolution of mean squared estimation errors (averaged over 10,000 Monte Carlo simulations) of the KCF, the optimal KCF and the proposed distributed state estimation algorithms, where the shaded region indicates naïvety of sensors 4, 5 and 6 in Scenario II.

received from sensors 3 and 5 respectively, at sensor 4. In the optimal KCF algorithm, a single consensus gain C_4 is computed at sensor 4. Since the consensus gains are matrices, we compare their Frobenius norms (denoted as $\|\cdot\|_{\mathcal{F}}$).

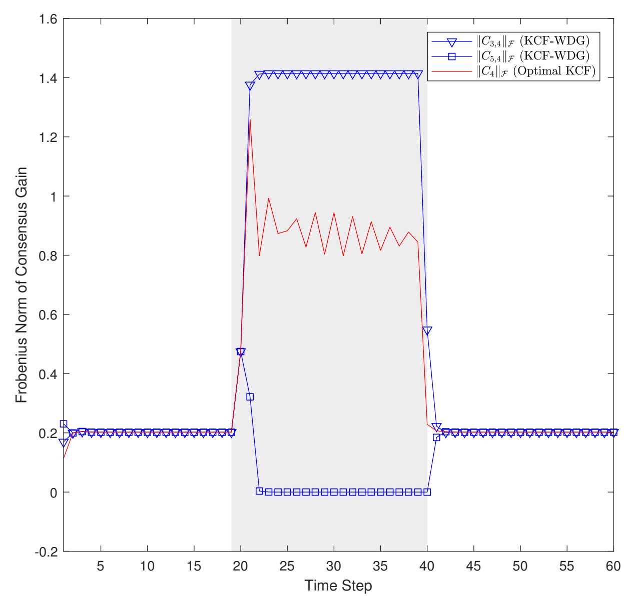

Fig. 6. Evolution of the consensus gains computed at sensor 4, for the optimal KCF and the proposed algorithms, where the shaded region indicates naïvety of sensors 4, 5 and 6 in Scenario II.

Figure 6 shows the evolution of $\|C_{3,4}\|_{\mathcal{F}}$ and $\|C_{5,4}\|_{\mathcal{F}}$ (for the proposed algorithm), as well as $\|C_4\|_{\mathcal{F}}$ (for the optimal KCF), for the simulation in Scenario II. Consider the period of $20 - 40$ time steps, during which sensors 4, 5 and 6 are naïve, in Fig. 6. At sensor 4, the proposed algorithm employs a higher consensus gain for the information being received from sensor 3 than for the information from sensor 5, i.e., $\|C_{3,4}\|_{\mathcal{F}}$ is greater than $\|C_{5,4}\|_{\mathcal{F}}$. This makes the entire sensor network achieve more accurate estimates by relying more on measurements made by sensors that are able to observe the target. On the other hand, in the optimal KCF scheme, the information going from sensor 3 to sensor 4, as well as the information going from sensor 5 to sensor 4, use the same consensus gain C_4. In the case of such algorithms which use unweighted graphs for the consensus logic, the information from the naïve sensors can be said to diffuse across the entire network and thus corrupt the estimates of the other sensors.

V. CONCLUSION

In this paper, a distributed estimation algorithm has been proposed, which estimates the state of a stochastic linear dynamical system observed over a sensor network. Unlike the Kalman Consensus Filter, the proposed algorithm does not require any assumptions on the observability of the individual sensors, and achieves consensus on a weighted directed graph. The issue of suboptimality in existing schemes has also been addressed. In this paper, a mathematically rigorous formulation of the algorithm has been presented, and the performance of the algorithm has been demonstrated using illustrative numerical examples. The simulation results have been found to validate the superior performance of the proposed scheme against the existing algorithms that are restricted to unweighted consensus. Lastly, the paper serves as a theoretical framework for future work on the subject, the possibilities for which include a formal stability analysis as well as extensions of the proposed scheme to scenarios involving packet losses and asynchronous communication.

REFERENCES

[1] L. Hong, "Centralized and distributed multisensor integration with uncertainties in communication networks," *IEEE transactions on aerospace and electronic systems*, vol. 27, no. 2, pp. 370–379, 1991.

[2] J. Xue-Bo and S. You-Xian, "Optimal centralized state fusion estimation for multi-sensor system with correlated measurement noise," in *Proceedings of 2003 IEEE Conference on Control Applications, 2003. CCA 2003.*, vol. 1. IEEE, 2003, pp. 770–772.

[3] P. Baran, "On distributed communications networks," *IEEE transactions on Communications Systems*, vol. 12, no. 1, pp. 1–9, 1964.

[4] A. M. Aziz, "A simple and efficient suboptimal multilevel quantization approach in geographically distributed sensor systems," *Signal Processing*, vol. 88, no. 7, pp. 1698–1714, 2008.

[5] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 8179–8184.

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

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

[8] 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 *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 7863–7869.

[9] R. Olfati-Saber, "Kalman-consensus filter: Optimality, stability, and performance," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, 2009, pp. 7036–7042.

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

[11] W. Li, Y. Jia, and J. Du, "Distributed kalman consensus filter with intermittent observations," *Journal of the Franklin Institute*, vol. 352, no. 9, pp. 3764–3781, 2015.

[12] R. Deshmukh, C. Kwon, and I. Hwang, "Optimal discrete-time kalman consensus filter," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 5801–5806.

[13] W. Li, Y. Jia, J. Du, and J. Zhang, "Distributed consensus filtering for jump markov linear systems," *IET Control Theory & Applications*, vol. 7, no. 12, pp. 1659–1664, 2013.

[14] I. Džafić, R. A. Jabr, I. Huseinagić, and B. C. Pal, "Multi-phase state estimation featuring industrial-grade distribution network models," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 609–618, 2017.

[15] N. da Silva, A. S. Costa, K. Clements, and E. Andreoli, "Simultaneous estimation of state variables and network topology for power system real-time modeling," *Electric Power Systems Research*, vol. 133, pp. 338–346, 2016.

[16] X. Liu, J. Cao, W.-Z. Song, P. Guo, and Z. He, "Distributed sensing for high-quality structural health monitoring using wsns," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 3, pp. 738–747, 2015.

[17] I. Shames, A. M. Teixeira, H. Sandberg, and K. H. Johansson, "Distributed fault detection for interconnected second-order systems," *Automatica*, vol. 47, no. 12, pp. 2757–2764, 2011.

[18] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. ACM, 1999, pp. 174–185.

[19] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference*, vol. 2. IEEE, 2002, pp. 1327–1332.

[20] S. Martinez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.

[21] J. Wu, A. Elser, S. Zeng, and F. Allgwer, "Consensus-based distributed kalman-bucy filter for continuous-time systems," vol. 49, no. 22, pp. 321 – 326, 2016, 6th IFAC Workshop on Distributed Estimation and Control in Networked Systems NECSYS 2016.

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

[23] 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.