A Framework for Distributed Estimation with Limited Information and Event-Based Communications

Journal Article

Author(s):
Yan, Jiaqi; Mo, Yilin; Ishii, Hideaki

Publication date:
2024-06

Permanent link:
https://doi.org/10.3929/ethz-b-000644721

Originally published in:
IEEE Transactions on Automatic Control 69(6), https://doi.org/10.1109/tac.2023.3309373

A Framework for Distributed Estimation with Limited Information and Event-Based Communications

Jiaqi Yan, Yilin Mo, and Hideaki Ishii

***Abstract*—In this paper, we consider the problem of distributed estimation in a sensor network, where multiple sensors are deployed to estimate the state of a linear time-invariant Gaussian system. By losslessly decomposing the Kalman filter, a framework of event-based distributed estimation is developed, where each sensor node runs a local filter using solely its own measurement, alongside with an event-based synchronization algorithm to fuse the neighboring information. One novelty of the proposed framework is that it decouples the local filters from the synchronization process. By doing so, we prove that a general class of triggering strategies can be applied in our framework, which yields stable distributed estimators under the requirements of collective system observability. Moreover, the developed results can be generalized to achieve a distributed implementation of any Luenberger observer. By solving a semi-definite programming (SDP), we further present a low-rank estimator design to obtain the (sub)optimal gains of a Luenberger observer such that the distributed estimation is realized under the constraint of message size. Therefore, as compared with existing works, the proposed algorithm is implemented with limited information since it enjoys lower data size at each transmission. Numerical examples are finally provided to demonstrate the efficacy of the proposed methods.**

***Index Terms*—Distributed estimation, Event-triggered control, Limited information, Low-rank estimator design.**

I. Introduction

Due to its wide applications in environment monitoring, target tracking, and robotics navigation, the problem of state estimation has attracted significant research attention in the past couple of decades ([1]–[4]). Within this field, a fundamental problem is to estimate the state of a linear time-invariant (LTI) Gaussian system, of which the optimal estimate is provided by the centralized Kalman filter ([5]). However, as both the number of sensors and data size increase in networks, the classical Kalman filter may not be suitable in many applications. As such, distributed estimation algorithms are needed, where each sensor aims to produce a stable local estimate only using its own measurements and limited information exchange.

In order to achieve a distributed implementation of Kalman filter, a number of consensus-based distributed estimators have been proposed in the literature including [6]–[14]. For example, by performing average consensus on local estimates, a Kalman-Consensus Filter (KCF) is proposed in [6]. This work has also motivated the development of a group of distributed estimators where local estimates are fused by consensus algorithms ([7], [8]). Different from them, Battistelli *et al.* [9] have suggested performing consensus algorithms on both measurements and information matrices. They proposed an estimator that can guarantee the stability of estimation error even when the system has nonlinear dynamics. Other approaches include [13] and [15], where distributed estimators are established by performing consensus on probability densities and coded estimates, respectively.

In the aforementioned works, the estimation algorithms require at least one transmission during each sampling period. In contrast, inspired by the fact that the sensors are often powered by energy-limited batteries, another group of works focuses on developing distributed estimators by using event-triggered transmission policies ([16]–[23]). For instance, [21] proposes a consensus-based distributed Kalman filter, where the transmission instants are triggered by both the state estimates and error covariance. Under the assumption that the system matrix is invertible and the communication topology is strongly connected, the authors have proven the mean-square boundedness of the estimation error. In [18], based on a stochastic triggering function determined by local estimates, a minimum mean-square error estimator is given. Under similar conditions, the distributed estimator is stable with a bounded mean-square estimation error. In a recent work [20], He *et al.* have suggested determining the transmission times by evaluating error covariance matrices. They have further quantified the communication rate which is necessary to guarantee the stability of estimation error.

Although the existing solutions differ in design and analysis, their information flow can be illustrated by a unified framework as shown in Fig. 1. In the figure, $\Delta_i(k_s^i)$ represents the message transmitted by sensor i, of which the transmission times could be determined by the triggering functions based on local estimates ([17], [18]), measurements ([19]), error covariance ([20], [21]), or innovations ([16], [22]). Towards achieving stable distributed estimators, consensus/synchronization algorithms are often performed on $\Delta_i(k_s^i)$.

From Fig. 1, it is also noticed that the local filters are usually coupled with synchronization algorithms in the existing solutions. As such, the performances of both processes are inevitably affected by the triggering mechanisms, bringing

J. Yan and H. Ishii are with the Department of Computer Science, Tokyo Institute of Technology, Japan. Emails: jyan@sc.dis.titech.ac.jp, ishii@c.titech.ac.jp. Yilin Mo is with the Department of Automation, BNRist, Tsinghua University, P. R. China. Email: ylmo@tsinghua.edu.cn.

This work was supported by National Natural Science Foundation of China under grant no. 62273196, and by JSPS under Grants-in-Aid for Scientific Research Grant No. 18H01460 and 21F40376.

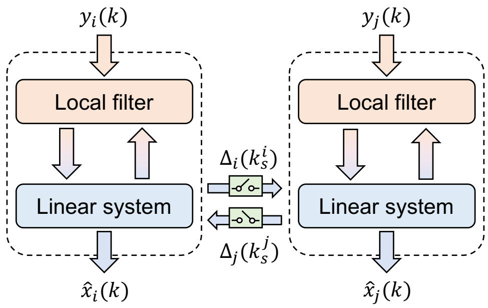

Fig. 1: The information flow of most existing event-based distributed estimation algorithms, where sensors i and j are immediate neighbors.

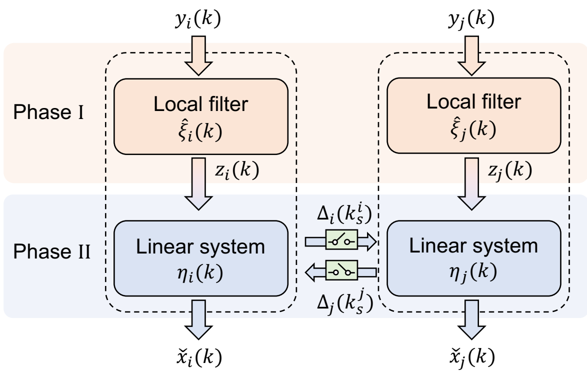

Fig. 2: The information flow of the proposed distributed estimation algorithm, where nodes i and j are immediate neighbors.

more challenges to algorithm design and analysis. Particularly, in order to guarantee the stability of local estimators, existing works usually have additional assumptions on the network topology or the system matrix. Driven by this concern, a question thus arises naturally: is it possible to construct a simpler implementation by decoupling the processes of local filters and synchronization?

This paper focuses on answering this question. To be specific, by decomposing the Kalman filter, we prove that the optimal Kalman estimate can be perfectly recovered as a weighted sum of a bank of local filters. Based on this decomposition, this paper presents a novel framework for the event-based distributed estimation. Here, each sensor performs local filtering solely with its own measurement, and global fusion is realized through the information exchange by running an event-based synchronization algorithm. With the proposed framework, we show that the local filter can be decoupled from the synchronization process as illustrated in Fig. 2, which is different from the existing solutions taking the form in Fig. 1.

With the particular aim of reducing communication burden of the developed algorithm, two efforts are made in this paper: 1) We apply event-based strategies to reduce the number of transmissions in the network. 2) By decomposing the estimation gain K of the Kalman filter according to its basis, the proposed algorithm is implemented with limited information, resulting in low message complexity[1] as $\mathrm{rank}(K) \leq \min\{m, n\}$, where m and n denote the dimensions of the output and the state of the system. As a result, the size of data exchanged during each transmission is also limited.

Finally, noticing that the message complexity is equal to the rank of the estimation gain, we are motivated to study the design of (sub)optimal gain under a rank constraint. Specifically, suppose that the message complexity that the network is willing to tolerate is $\tilde{r}$, where $0 < \tilde{r} \leq \min\{m, n\}$. We propose a semi-definite programming (SDP) to design a (sub)optimal estimation gain such that its rank is no more than $\tilde{r}$ while yielding the minimum estimation error.

The main contributions of this paper are summarized below:

1) By decomposing the Kalman filter, this paper presents a novel framework for the event-based distributed estimation. A merit of this framework is that it decouples the local filter from the synchronization process. As such, the performance of local filters will not be affected even when no sensors are triggered to transmit at certain times.

2) Instead of adopting any specific triggering function, we show that a general class of triggering strategies can be used in our framework. By proposing a c-martingale convergence lemma, the proposed estimator is shown as stable at each sensor side under the minimal requirements of collective system observability and network connectivity for synchronization, which depends on the instability of the system. To the best of our knowledge, this is the first time that the stability of distributed estimators is studied under these conditions and by using the martingale convergence theory.

3) Existing works such as [18]–[21], [23] usually require each sensor to send out messages of size n or even n^2+n when triggered. (For more comparison details, see also TABLE I in Section VII.) This is because in addition to the estimated state, the error covariance matrix must be transmitted. In contrast, our estimator requires only the reduced-order state estimate to be shared. Hence, it is implemented with limited information at lower message complexity as $\mathrm{rank}(K) \leq \min\{m, n\}$.

4) We also present a design of the low-rank estimator which yields the minimum performance loss. Specifically, this can be done for any $\tilde{r}$ such that $0 < \tilde{r} \leq \min\{m, n\}$, and the designed distributed estimator can be implemented with message complexity no more than $\tilde{r}$. To the best of our knowledge, this is the first work considering the optimal estimator design under the constraint of message complexity. By using it, computational complexity of our algorithm is also reduced.

The remainder of this paper is organized as follows. Section II introduces the problem of distributed estimation. A decomposition of the Kalman filter is discussed in Section III,

[1] In this paper, message complexity refers to the size of message in terms of the number of real values sent by each sensor to its neighbors at any transmission. A related notion is that of data rate, which is the number of bits per unit time required for message transmissions. Since in practice, it usually takes a fixed number of bits (8 or 16 bits) to transfer a real number, data rate increases linearly with the message complexity if the transmission times are the same. In this regard, these two notions are closely related. Note that under the event-based communication, it is difficult to know the frequency of transmissions in advance. Hence, in the theoretical analysis, we use the notion of message complexity. We will compute the frequency of transmissions in the numerical examples in Section VIII.

based on which Section IV presents our framework of event-based distributed estimation. The performance of this algorithm is next analyzed in Section V. We then discuss how to design the estimation gain under the constraint of message complexity in Section VI and compare the proposed framework with existing solutions in Section VII. The algorithm performance is then validated through numerical examples in Section VIII. Finally, we conclude this work in Section IX.

A preliminary version of this paper has been reported in [24]. The current version presents a different decomposition method of the Kalman filter and further proposes the low-rank estimator design to reduce the message complexity. Also, we present all the proofs as well as more extensive discussions and numerical examples here.

Notations: Throughout this paper, we denote by $\mathbf{0}_n$ and $\mathbf{1}_n$ the n-dimensional vectors of all zeros and all ones, respectively. We also denote by $\rho(A)$ the spectral radius of any matrix A. For a group of vectors $v_i \in \mathbb{R}^{m_i}$, the vector $\left[v_1^T, \ldots, v_N^T\right]^T$ is also written as $\operatorname{col}(v_1, \ldots, v_N)$. Moreover, given a positive semidefinite matrix U, let $U^{1/2}$ be the positive semidefinite matrix that satisfies $U^{1/2}U^{1/2} = U$. Finally, for any matrix $X \in \mathbb{R}^{n\times n}$, we denote by $p_X(s)$ its characteristic polynomial, i.e., $p_X(s) \triangleq \det(sI_n - X)$.

II. Problem Formulation

In this section, we shall discuss the problem of state estimation. The centralized Kalman filter and our framework for distributed estimation will be further introduced.

A. System setup for distributed estimator

Let us consider the LTI Gaussian system as given below:

$$x(k+1) = Ax(k) + w(k), \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the system state to be estimated, $w(k) \sim \mathcal{N}(0, Q)$ is an independent and identically distributed (i.i.d.) Gaussian noise with zero mean and covariance matrix $Q \geq 0$. Moreover, the initial state $x(0)$ also follows the Gaussian distribution which has zero mean.

A sensor network monitors the above system, where the measurement of each sensor $i \in \{1, 2, ..., m\}$ is given by[2]

$$y_i(k) = C_i x(k) + v_i(k), \tag{2}$$

where $y_i(k) \in \mathbb{R}$ is the measurement of sensor i and $C_i \in \mathbb{R}^{1\times n}$. By collecting the measurements from all sensors, we have

$$y(k) = Cx(k) + v(k), \tag{3}$$

where

$$y(k) \triangleq \begin{bmatrix} y_1(k) \\ \vdots \\ y_m(k) \end{bmatrix}, C \triangleq \begin{bmatrix} C_1 \\ \vdots \\ C_m \end{bmatrix}, v(k) \triangleq \begin{bmatrix} v_1(k) \\ \vdots \\ v_m(k) \end{bmatrix},$$

and $v(k)$ is a zero-mean i.i.d. Gaussian noise with covariance $R \geq 0$ and is independent of $w(k)$ and $x(0)$.

[2]Later in Remark 4, we will discuss an extension to the case where each sensor can output a vector measurement.

The system need not necessarily be Schur stable. That is, there may exist some eigenvalues of A with magnitudes no less than 1. Throughout this paper, we make the following assumption on system observability:

Assumption 1 (Collective observability)**.** *The system is collectively observable, i.e., the pair (A, C) is observable, while (A, C_i) is not necessarily observable for each sensor i.*

In this paper, we aim to design a distributed algorithm to estimate the system state. Communication over the sensor network is modeled by a connected undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Here, $\mathcal{V} = \{1, 2, ..., m\}$ and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ are the sets of sensors and edges, respectively. Moreover, the interaction among sensors is described by a weighted adjacency matrix $\mathcal{A} = [a_{ij}]$, where $a_{ij} \geq 0$ and $a_{ij} = a_{ji}, \forall i, j \in \mathcal{V}$. Notice that $a_{ij} > 0$ if and only if $(i, j) \in \mathcal{E}$. The degree matrix of $\mathcal{G}$ is defined as $\mathcal{D}_{\mathcal{G}} \triangleq \operatorname{diag}(d_1, \ldots, d_m)$, where $d_i = \sum_{j=1}^m a_{ij}$. The Laplacian matrix of $\mathcal{G}$ is calculated as $\mathcal{L}_{\mathcal{G}} \triangleq \mathcal{D}_{\mathcal{G}} - \mathcal{A}$. Since $\mathcal{G}$ is connected, let us arrange the eigenvalues of $\mathcal{L}_{\mathcal{G}}$ as

$$0 = \mu_1 < \mu_2 \leq \cdots \leq \mu_m. \tag{4}$$

B. Fundamental limit: Kalman filter

It is well known that if the measurements from all sensors can be collected by a single fusion center, then the centralized Kalman filter provides the optimal estimate. Therefore, the Kalman estimate acts as the fundamental limitation for all estimation schemes and will be briefly reviewed in this part.

Let $P(k)$ be the error covariance of Kalman estimate at time k. Under Assumption 1, the error covariance will converge to the steady state exponentially fast ([5], [25]), and thus let

$$P \triangleq \lim_{k\to\infty} P(k). \tag{5}$$

Since a sensor network typically operates for a long period of time, we consider the steady-state Kalman filter, which has the fixed gain

$$K = PC^T \left(CPC^T + R\right)^{-1}. \tag{6}$$

By using K, the optimal Kalman estimate $\hat{x}(k) \in \mathbb{R}^n$ is calculated recursively as

$$\hat{x}(k+1) = (A - KCA)\hat{x}(k) + Ky(k+1). \tag{7}$$

C. Framework of the proposed distributed estimator

The Kalman filter is a centralized solution since the optimal estimate (7) fuses the measurements of all sensors. To make the algorithm feasible in distributed networks, this paper aims to propose a distributed implementation of the Kalman filter such that each sensor can obtain a stable local estimate by communicating with only immediate neighbors.

Specifically, our distributed estimation algorithm is developed based on a lossless decomposition of the centralized Kalman filter (see Fig. 3). We show that the performance of Kalman filter is equivalent to a bank of local filters fused by a weighted sum. In our approach, the distributed estimator is designed as illustrated in Fig. 2. It has two phases, where Phase I implements the local filters solely based on the

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3309373

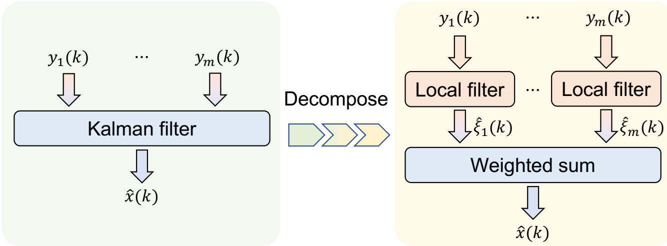

Fig. 3: The information flows of centralized Kalman filter (left) and decomposition of Kalman filter (15) (right).

local measurement of each sensor and Phase II fuses the neighboring states by replacing the weighted sum in Fig. 3 with a synchronization algorithm. In what follows, we shall detail the framework by respectively introducing the phases of decomposing the Kalman filter and synchronizing the local states.

III. A Lossless Decomposition of Kalman Filter

This section shows how to decompose the Kalman filter. To this end, we first prove that the Kalman estimate (7) can be losslessly recovered as a weighted sum of a bank of local filters. After that, a low-order decomposition of the Kalman filter will be developed by performing model reduction. We highlight that the results in this section are essential for us to design a framework of distributed estimation later in this paper.

To begin with, without loss of generality, suppose that the system matrix A takes a decomposed form as

$$A = \begin{bmatrix} A^u & \\ & A^s \end{bmatrix}, \tag{8}$$

where $A^u \in \mathbb{R}^{n^u \times n^u}$ and $A^s \in \mathbb{R}^{n^s \times n^s}$; any eigenvalue of A^u lies on or outside the unit circle while all the eigenvalues of A^s are strictly inside.

A. Preliminaries: Local filter design

We would first present the design of local filters. It was proposed in our previous work [15] and is an important preliminary of the approach of this work.

The results are based on the following lemmas:

Lemma 1 ([26])**.** *For any $\Lambda \in \mathbb{R}^n$, if Λ is non-derogatory*[3] *and in the Jordan form, then $(\Lambda, \mathbf{1}_n)$ is controllable.*

Lemma 2 ([15])**.** *Suppose that (X, p) is controllable, where $X \in \mathbb{R}^{n\times n}$ and $p \in \mathbb{R}^n$. For any $q \in \mathbb{R}^n$, if $X + pq^T$ and X do not share any eigenvalues, then $(X^T + qp^T, q)$ is controllable.*

Lemma 3 ([15])**.** *Suppose that (X, p) is controllable, where $X \in \mathbb{R}^{n\times n}$ and $p \in \mathbb{R}^n$. Denote the characteristic polynomial of X by $p_X(s) = \det(sI - X)$. Let $Y \in \mathbb{R}^{m\times m}$ and $q \in \mathbb{R}^m$ be such that $p_X(Y)q = 0$ holds. Then there exists $T \in \mathbb{R}^{m\times n}$ such that the following equations are satisfied*[4]*:*

$$TX = YT,\ Tp = q. \tag{9}$$

Let us decompose the Kalman gain (6) as

$$K = [K_1, \cdots, K_m],$$

namely, $K_i \in \mathbb{R}^n$ is the i-th column of K, which will be used as the gain of the i-th local filter. Accordingly, we can rewrite the Kalman estimate (7) as

$$\hat{x}(k+1) = (A - KCA)\hat{x}(k) + \sum_{i=1}^{m} K_i y_i(k+1). \tag{10}$$

Since (A, C) is observable, it is not difficult to conclude that the matrix $A - KCA$ is strictly stable. As such, one can always construct a Jordan matrix $\Lambda \in \mathbb{R}^{n\times n}$ satisfying that

1) Λ is strictly stable and non-derogatory.
2) The characteristic polynomials of Λ and $A - KCA$ are the same.

By virtue of Lemma 1, one knows that $(\Lambda, \mathbf{1}_n)$ is controllable. Then, as guaranteed by Lemma 3, for each $i = 1, \cdots, m$, a matrix $F_i \in \mathbb{R}^{n\times n}$ exists such that

$$F_i\Lambda = (A - KCA)F_i,\ F_i\mathbf{1}_n = K_i. \tag{11}$$

These matrices F_i will help us to reconstruct the optimal state estimate $\hat{x}(k)$ from the outputs of the local filters. The relation $F_i\mathbf{1}_n = K_i$ in (11) is the key to see that the gain in the local filter to be presented (in (14)) is $\mathbf{1}_n$.

Next, we would also design $\beta \in \mathbb{R}^n$ and $S \in \mathbb{R}^{n\times n}$ such that

$$S = \Lambda + \mathbf{1}_n\beta^T, \tag{12}$$

and the following statements hold:

1) Let $p_S(s)$ and $p_{A^u}(s)$ be the characteristic polynomials of S and A^u, respectively. Then it should hold that $p_{A^u}(s) | p_S(s)$. Namely, there exists a polynomial $\sigma(s)$ such that
$$p_S(s) = p_{A^u}(s)\sigma(s). \tag{13}$$
2) Any root of $\sigma(s)$ is strictly within the unit circle but not an eigenvalue of Λ.

Therefore, the unstable and stable eigenvalues of S are the roots of $p_{A^u}(s)$ and $\sigma(s)$, respectively. Moreover, the unstable ones should coincide with the eigenvalues of A^u, while the stable ones can be freely designed (but should not be the eigenvalues of Λ). Since Λ is strictly stable, S does not share any eigenvalues with Λ. As a result of Lemma 2, we conclude that (S^T, β) is controllable.

Remark 1. *In practice, one can find β and S by following the procedure below:*

i) Pre-determine the eigenvalues of S such that the unstable eigenvalues are identical to the ones of A^u, while the stable ones are not the same as those of Λ.

[3] A matrix is said to be non-derogatory if every eigenvalue of it has geometric multiplicity 1.

[4] The solution of (9) can be obtained by following the construction proof of [15, Lemma 2].

ii) *Through pole placement, find β such that the eigenvalues of $\Lambda + \mathbf{1}_n \beta^T$ are placed at the desired locations. Notice that β always exists since $(\Lambda, \mathbf{1}_n)$ is controllable.*

iii) *Calculate S by* (12)*.*

Now we propose the local filters, which are performed by each sensor i solely based on its own measurement $y_i(k)$:

$$\begin{aligned} z_i(k) &= y_i(k+1) - \beta^T \hat{\xi}_i(k), \\ \hat{\xi}_i(k+1) &= S\hat{\xi}_i(k) + \mathbf{1}_n\, z_i(k), \end{aligned} \tag{14}$$

where $\hat{\xi}_i(k) \in \mathbb{R}^n$ is the local response of the i-th local filter with $\hat{\xi}_i(0) = 0$, and $z_i(k) \in \mathbb{R}$ is the local innovation.

The following lemma shows that the optimal Kalman filter can be losslessly recovered by a linear combination of the local responses $\hat{\xi}_i(k), i \in \{1, \cdots, m\}$. Moreover, for any sensor i, the signal $z_i(k)$ is stable[5]. For the sake of readability, we provide the proof in Appendix A.

Lemma 4. *Suppose that each sensor implements the local filter* (14)*. Then the following statements hold at any k:*

1) *For any sensor i, the covariance of $z_i(k)$ is bounded.*
2) *The optimal Kalman estimate $\hat{x}(k)$ in* (7) *can be losslessly recovered from the local estimates $\hat{\xi}_i(k), i = 1, 2, \cdots, m$, by*

$$\hat{x}(k) = \sum_{i=1}^{m} F_i \hat{\xi}_i(k), \tag{15}$$

where F_i is the solution of (11)*.*

For illustration, the information flows of the Kalman filter (7) and its decomposition (15) are presented in Fig. 3.

B. A new decomposition of the Kalman filter with low order

In order to construct a distributed implementation of the Kalman filter, our next step is to present (15) with a matrix realization. Here, we provide a new approach by decomposing the Kalman gain K according to its basis. The intention is to deal with gains of lower rank.

To this end, let us denote by r the rank of $K \in \mathbb{R}^{n\times m}$:

$$r \triangleq \mathrm{rank}(K) \le \min\{m, n\}. \tag{16}$$

As a result, there exists a matrix $V \in \mathbb{R}^{r\times m}$ of rank r such that K can be decomposed as

$$K = \begin{bmatrix} K_1 & \cdots & K_m \end{bmatrix} = \begin{bmatrix} \tilde{K}_1 & \cdots & \tilde{K}_r \end{bmatrix} V = \tilde{K} V, \tag{17}$$

where $\{\tilde{K}_i\}_{i\in\{1,\cdots,r\}}$ are linearly independent. Then by (15), let us rewrite the Kalman estimate as below:

$$\begin{aligned} \hat{x}(k+1) &= \sum_{i=1}^{m} F_i \hat{\xi}_i(k+1) \\ &= \sum_{i=1}^{m} F_i(\Lambda + \mathbf{1}\beta^T)\hat{\xi}_i(k) + \sum_{i=1}^{m} F_i\, \mathbf{1}_n\, z_i(k) \\ &= (A - KCA) \sum_{i=1}^{m} F_i \hat{\xi}_i(k) + \sum_{i=1}^{m} K_i \beta^T \hat{\xi}_i(k) + \sum_{i=1}^{m} K_i z_i(k), \end{aligned} \tag{18}$$

[5]We say a signal is *stable* if the covariance of it is bounded at all time.

where the second and third equalities hold respectively by (14) and (11). We next consider the second term of the far right-hand side, i.e., $\sum_{i=1}^{m} K_i \beta^T \hat{\xi}_i(k)$. By (17), it follows that

$$\sum_{i=1}^{m} K_i \beta^T \hat{\xi}_i(k) = \sum_{i=1}^{r} \tilde{K}_i \beta^T \sum_{j=1}^{m} v_{ij} \hat{\xi}_j(k), \tag{19}$$

where v_{ij} is the (i, j)th entry of V. Moreover, it follows from (14) that

$$\sum_{j=1}^{m} v_{ij} \hat{\xi}_j(k+1) = S \sum_{j=1}^{m} v_{ij} \hat{\xi}_j(k) + \sum_{j=1}^{m} v_{ij}\, \mathbf{1}_n\, z_i(k). \tag{20}$$

To simplify notations, let us denote

$$\vartheta(k) \triangleq \begin{bmatrix} \sum_{i=1}^{m} F_i \hat{\xi}_i(k) \\ \sum_{j=1}^{m} v_{1j} \hat{\xi}_j(k) \\ \vdots \\ \sum_{j=1}^{m} v_{rj} \hat{\xi}_j(k) \end{bmatrix} \in \mathbb{R}^{n(r+1)}. \tag{21}$$

It thus follows from (18)–(20) that

$$\vartheta(k+1) = H\vartheta(k) + Lz(k), \tag{22}$$

where

$$\begin{aligned} H &\triangleq \begin{bmatrix} A - KCA & \tilde{K}_1\beta^T & \cdots & \tilde{K}_r\beta^T \\ & S & & \\ & & \ddots & \\ & & & S \end{bmatrix} \in \mathbb{R}^{n(r+1)\times n(r+1)}, \\ L &\triangleq \begin{bmatrix} K \\ V \otimes \mathbf{1}_n \end{bmatrix} \in \mathbb{R}^{n(r+1)\times m}, \\ z(k) &\triangleq \begin{bmatrix} z_1(k) & \cdots & z_m(k) \end{bmatrix}^T \in \mathbb{R}^m. \end{aligned} \tag{23}$$

For convenience, we denote by L_i the i-th column of L, namely,

$$L = \begin{bmatrix} L_1 & \cdots & L_m \end{bmatrix}. \tag{24}$$

In view of (18), the optimal Kalman estimate $\hat{x}(k)$ is indeed the vector consisting of the first n entries of $\vartheta(k)$. Therefore, the optimal estimate can be losslessly recovered by (14) and (22). Note that a center is however required to fuse $\hat{\xi}_i(k)$ and $z_i(k)$ from all sensors. In the next section, we will show how to use (14) and (22) to design a distributed implementation of the Kalman filter.

Remark 2. *Our decomposition approach here is an extension of [15] where the Kalman gain K is used directly. It is noted that, to achieve a distributed implementation, the message complexity should be $\min\{m, n\}$ in [15]. In contrast, as will be shown in Section IV,* (22) *can be implemented with lower message complexity, namely, $r \le \min\{m, n\}$. In general, however, the rank of estimation gain K may not have a lower rank. Later in Section VI of the paper, we will also provide a design method to find an estimation gain matrix for a given rank while minimizing the estimation error. This will allow us to further reduce the message complexity with some tradeoff in its estimation performance. Moreover, from the simulation results in Section VIII, we observe that the performance loss caused by the designed low-rank estimators can be very minor.*

IV. An Event-Based Distributed Implementation of Kalman Filter

In this section, we shall show how to perform the distributed estimation by implementing (14) and (22) in a distributed manner. In particular, we present event-based communication strategies to reduce the transmission frequencies for each sensor node. Notice that the conventional distributed estimators including [6], [8]–[12], [27]–[33] require that each sensor broadcasts its local information to neighbors at least one time during the sampling interval. Our approach will solve such issues with limited loss in estimation performance.

A. Framework of the event-based distributed estimation

We shall leverage the results established in Section III to design a distributed estimator. Specifically, based on the decomposition of Kalman filter, for any sensor i, its update during each sampling period is divided into two phases:

- Phase I: Sensor i performs the local filter (14) solely using its own measurement without communicating with others.
- Phase II: For synchronization, sensor i fuses the neighboring information based on (22).

In view of (22), it is clear that the Kalman estimate fuses $\hat{\xi}_i(k)$ and $z_i(k)$ from all sensors. However, since each local sensor is only capable of accessing the information in its neighborhood, we aim to implement Kalman filter by running a distributed synchronization algorithm. For the particular purpose of decreasing the transmission frequency, event-based communication strategies will be adopted.

To be concrete, let each sensor i keep a local state as below:

$$\eta_i(k) \triangleq \begin{bmatrix} \eta_{0,i}(k) \\ \eta_{1,i}(k) \\ \vdots \\ \eta_{r,i}(k) \end{bmatrix} \in \mathbb{R}^{n(r+1)}, \tag{25}$$

where $\eta_{j,i}(k) \in \mathbb{R}^n, j = 0, 1, \cdots, r$. In order to approach the performance of Kalman filter, the local state will be updated through the following synchronization algorithm:

$$\eta_i(k+1) = H\eta_i(k) + L_i z_i(k) + B\sum_{j=1}^{m} a_{ij}(\widehat{\Delta}_j(k) - \widehat{\Delta}_i(k)), \tag{26}$$

where $\eta_i(0) = 0$, H and L_i are respectively defined in (23) and (24), and

$$B = \begin{bmatrix} \mathbf{0}_{n\times r} \\ I_r \otimes \mathbf{1}_n \end{bmatrix} \in \mathbb{R}^{n(r+1)\times r}. \tag{27}$$

The sensors exchange their local knowledge on $\eta_i(k)$ but with limited information in the sense that their dimensions as well as the transmission frequencies are reduced. First, sensor i broadcasts the r-dimensional vector $\widehat{\Delta}_i(k) \in \mathbb{R}^r$ given by

$$\widehat{\Delta}_i(k) = T\widehat{\eta}_i(k), \tag{28}$$

where

$$T = \begin{bmatrix} \mathbf{0}_{r\times n} & I_r \otimes \Gamma \end{bmatrix} \in \mathbb{R}^{r\times n(r+1)} \tag{29}$$

such that $\mathbf{0}_{r\times n}$ is the r-by-n matrix of all zeros and $\Gamma \in \mathbb{R}^{1\times n}$ is the synchronization gain matrix to be designed. Moreover, the intermediate state $\widehat{\eta}_i(k)$ is the local state from the recent past updated based on the triggering function. Specifically, letting k_s^i be the triggering instants, it is given by

$$\widehat{\eta}_i(k) = H^{k-k_s^i}\eta_i(k_s^i), \quad k \in [k_s^i, k_{s+1}^i), \tag{30}$$

To determine the triggering instants k_s^i, each sensor i considers a triggering function $f_i(k)$ in the following form:

$$f_i(k) = ||\epsilon_i(k)||^2 - h_i(k), \tag{31}$$

where

$$\epsilon_i(k) = \widehat{\eta}_i(k) - \eta_i(k), \tag{32}$$

and $h_i(k)$ is a threshold function as will be discussed later in Section IV-B. Once the triggering function satisfies $f_i(k) \geq 0$, sensor i will be triggered. It then broadcasts $\Delta_i(k)$ to neighbors, resetting $\epsilon_i(k)$ to zero. Hence, the sequence of triggering instants is determined recursively as

$$k_{s+1}^i \triangleq \min\left\{k > k_s^i \mid f_i(k) \geq 0\right\}, \; k_0^i = 0. \tag{33}$$

That is, the local state is transmitted only when the difference between the current local state $\eta_i(k)$ and its processed version $\widehat{\eta}_i(k)$ is sufficiently large.

By collecting Phases I and II together, the update of sensor i is summarized in Algorithm 1. Fig. 2 presents the information flow of Algorithm 1, which requires no fusion center and is achieved in a distributed manner. As compared with Fig. 1, the novelty of the proposed algorithm lies in the decoupling of the local filter from the fusion process. Therefore, the communication occurs only in Phase II, and the performance of local filters will not be affected even when no sensors are triggered to transmit at certain times. As we will see later, this structure also enables us to simplify the analysis by using arguments based on martingales.

Algorithm 1 An event-based distributed estimation algorithm for sensor i at time $k > 0$

1: (Phase I) Solely using its own measurement, sensor i computes $z_i(k)$ and updates the output of the local filter by (14).

2: (Phase II) By fusing the information most recently received from its neighbors, sensor i updates $\eta_i(k+1)$ according to the synchronization algorithm (26), (28), and (29).

3: Sensor i obtains the local estimate as

$$\breve{x}_i(k+1) = m\eta_{0,i}(k+1). \tag{34}$$

4: Sensor i checks the triggering function (31). If $f_i(k) \geq 0$, it broadcasts $\Delta_i(k+1)$ to neighbors.

Remark 3. *By* (28)*, instead of directly transmitting the local state* $\widehat{\eta}_i(k) \in \mathbb{R}^{n(r+1)}$*, each sensor node broadcasts a "coded" vector* $\Delta_i(k) \in \mathbb{R}^r$*. Therefore, the data size for each transmission is* $r = \mathrm{rank}(K) \leq \min\{m, n\}$*. As compared with existing works, e.g., [18]–[21], [23], which usually require information exchange on the local covariance matrix of size* $n^2 + n$*, the proposed algorithm can perform* *with lower message complexity.*

Considering the computational complexity of Algorithm 1, although matrix H has a size of $n(r+1)\times n(r+1)$, it is sparse. Thus, it can be verified that the computational overhead for performing (26) is $O(rn^2)$, which is lower than many existing solutions with a complexity of $O(n^3)$. These differences will be further discussed later in Section VII.

Remark 4. *Now, we extend the current results on scalar measurements to vector scenarios. In this generalized setting, each sensor $i \in 1,\cdots,s$ outputs a vector measurement as follows:*

$$y_i(k) = C_i x(k) + v_i(k), \tag{35}$$

where $y_i(k) = [y_i^1(k),\cdots,y_i^{m_i}(k)]^T \in \mathbb{R}^{m_i}$, $C_i \in \mathbb{R}^{m_i\times n}$, and $v_i(k) \in \mathbb{R}^{m_i}$ represents the m_i-dimensional Gaussian white noise.

It is important to note that for any sensor i, it has direct access to all components of $y_i(k)$. In other words, it can access $y_i^\ell(k)$ for $\ell \in 1,\cdots,m_i$. Consequently, after performing the local filter (14) and calculating $z_i^\ell(k)$ from each $y_i^\ell(k)$, the sensor fuses information within itself before communicating with other sensors. To accommodate this vector scenario, we extend the synchronization algorithm in (26) as follows:

$$\eta_i(k+1) = H\eta_i(k) + L_i z_i(k) + B\sum_{j=1}^{s} a_{ij}(\widehat{\Delta}_j(k) - \widehat{\Delta}_i(k)),$$

where $\eta_i(k) \in \mathbb{R}^{n(r+1)}$ and

$$L_i \triangleq [L_i^1,\cdots,L_i^{m_i}] \in \mathbb{R}^{n(r+1)\times m_i},$$
$$z_i(k) \triangleq [z_i^1(k),\cdots,z_i^{m_i}(k)]^T \in \mathbb{R}^{m_i}.$$

With the same design in (28)–(33), it follows that $\widehat{\Delta}_i(k) \in \mathbb{R}^r$. Hence, we can verify that all the results in this paper remain valid, including the convergence results and discussions on communication and computation costs.

B. A general class of triggering functions

Instead of adopting any specific triggering function, we show that a general class of triggering strategies can be applied in our framework to yield stable distributed estimates. Specifically, we intend to design the triggering function (31) such that $||\epsilon_i(k)||^2$ is upper bounded by some $\hbar < \infty$, namely,

$$||\epsilon_i(k)||^2 \le \hbar,\ \forall k \ge 0. \tag{36}$$

Clearly, this requires the threshold $h_i(k)$ to be carefully chosen. We now present several designs of $h_i(k)$ that are commonly used in the literature:

1) Static time-dependent triggering function ([34]–[36]):

$$h_i(k) = c_0 + c_1\alpha^k, \tag{37}$$

where $c_0 > 0$, $c_1 \ge 0$, and $\alpha \in (0,1)$.

2) Static state-dependent triggering function ([37], [38]):

$$\begin{aligned}\widehat{q}_i(k) &= \min\left\{\frac{1}{2}\sum_{j=1}^{m} a_{ij}\left|\left|\widehat{\Delta}_j(k_s^j) - \widehat{\Delta}_i(k_s^i)\right|\right|^2, \ell\right\},\\ h_i(k) &= \alpha_i(k)\widehat{q}_i(k),\end{aligned} \tag{38}$$

where $\ell > 0$ and $\alpha_i(k)$ takes nonnegative values and exponentially decreases to zero.

3) Dynamic triggering function ([37], [38]):

$$\begin{aligned}\chi_i(k+1) &= \beta_i\chi_i(k) + \alpha_i(k)\widehat{q}_i(k) - ||\epsilon_i(k)||^2,\\ h_i(k) &= \frac{1}{\theta_i}\chi_i(k) + \alpha_i(k)\widehat{q}_i(k),\end{aligned} \tag{39}$$

where $\chi_i(0) > 0$, $\beta_i \in (0,1)$ and $\theta_i > 1/\beta_i$. Moreover, $\widehat{q}_i(k)$ and $\alpha_i(k)$ are defined in (38).

It is not difficult to verify that (36) can be guaranteed by each of these designs.

As will be shown later in Section IV, one merit of our framework is that, by decoupling the local filters from the communication process, we can reformulate the problem of distributed estimation to that of stochastic linear systems synchronization. We will further prove that any event-based algorithm guaranteeing (36) can facilitate the synchronization of stochastic linear systems. Therefore, any of them including (37)–(39) can be used in our framework to produce stable distributed estimators. However, as one might imagine, different triggering functions result in different triggering frequencies and estimation accuracy.

V. Estimation Performance Analysis

This section will theoretically analyze the performance of Algorithm 1. To this end, we will first resort to probability theory and propose a c-martingale convergence lemma. By using this lemma, we are able to establish the mean-squared synchronization of local states, namely $\eta_i(k)$'s. This result will then be leveraged to prove stability of the distributed estimators.

A. Synchronization of local states

In order to establish the synchronization among local states, let us introduce the following lemma. Here, we provide a method for the design of the feedback gain Γ used in the update law (26) through the transformation (28) and (29).

Lemma 5. *Suppose that the Mahler measure*[6] *of matrix S meets the following condition:*

$$\prod_j |\lambda_j^u(S)| < \frac{1+\mu_2/\mu_m}{1-\mu_2/\mu_m}, \tag{40}$$

where $\lambda_j^u(S)$ represent the unstable eigenvalues of S and μ_j are the eigenvalues of the Laplacian matrix $\mathcal{L}_\mathcal{G}$ from (4). Let

$$\Gamma = \frac{2}{\mu_2+\mu_m}\frac{\boldsymbol{I}_n^T\mathcal{P}S}{\boldsymbol{I}_n^T\mathcal{P}\boldsymbol{I}_n} \in \mathbb{R}^{1\times n}, \tag{41}$$

[6]The Mahler measure of a matrix is defined as the absolute product of its unstable eigenvalues.

where $\mathcal{P} > 0$ solves the following modified algebraic Riccati inequality:

$$\mathcal{P} - S^T \mathcal{P} S + \left(1 - \zeta^2\right) \frac{S^T \mathcal{P} \boldsymbol{1}_n \boldsymbol{1}_n^T \mathcal{P} S}{\boldsymbol{1}_n^T \mathcal{P} \boldsymbol{1}_n} > 0, \tag{42}$$

and ζ satisfies that

$$\prod_j \left|\lambda_j^u(S)\right| < \zeta^{-1} \le \frac{1 + \mu_2/\mu_m}{1 - \mu_2/\mu_m}. \tag{43}$$

Then for any $j \in \{2, ..., n\}$, it holds that

$$\rho(H - \mu_j BT) < 1. \tag{44}$$

Proof. The proof is provided in Appendix B. □

In this lemma, the condition (40) implies that the more unstable the system whose state is to be estimated is, the more connected the network topology of the local estimators should be. Such conditions appear in consensus problems of agents with general linear dynamics (see, e.g., [39]).

Let us denote the average of local states of all sensors as

$$\bar{\eta}(k) \triangleq \frac{1}{m} \sum_{i=1}^{m} \eta_i(k). \tag{45}$$

We next show the synchronization among local states:

Theorem 1. *Suppose that the condition* (40) *holds, and Γ is designed based on* (41) *and* (42)*. By applying the synchronization algorithm* (26) *with an event-based communication strategy that guarantees* (36)*, synchronization among local states is reached in the mean square sense. That is, the following statements hold at any time k:*

1) *Consistency condition:*

$$\bar{\eta}(k+1) = H\bar{\eta}(k) + \bar{L}_z(k), \tag{46}$$

where $\bar{L}_z(k) \triangleq \frac{1}{m}\sum_{i=1}^{m} L_i z_i(k)$.

2) *Consensus condition: There exists $\Xi > 0$ such that*

$$\mathrm{cov}[\eta_i(k) - \bar{\eta}(k)] \le \Xi, \ \forall k. \tag{47}$$

Before proving Theorem 1, we shall introduce some useful lemmas. Because of the presence of stochastic signals $z_i(k)$, the approach of Lyapunov stability for deterministic systems cannot be directly used. We therefore resort to a stochastic analogue of it. Notice that in our previous work [15], we have provided a framework for showing the stability of local estimation error in full transmission scenario. Specifically, it is established by using Cauchy-Schwarz inequality assuming that the communication among sensors is independent of the system states and sensor measurements. However, in Algorithm 1, the communication inevitably relies on these states since it is triggered by certain events depending on them. This prevents the methodologies in [15] from being used. Therefore, this paper instead views the local estimation errors as c-martingales.

To see this, let $\{\mathcal{F}(t)\}_{t\ge 0}$ be a filtration in the probability space $(\Omega, \mathcal{F}, \mathcal{P})$ and $\{V(k)\}$ is a nonnegative stochastic process. Let us define

$$\Delta V(k) \triangleq V(k+1) - V(k).$$

We can write

$$\mathbb{E}[\Delta V(k)|\mathcal{F}(k)] = \mathbb{E}[V(k+1)|\mathcal{F}(k)] - V(k). \tag{48}$$

In order to analyze the systems which have non-zero noises at the origin, we shall extend the classical results on stability of supermartingales. Specifically, we will consider processes that are almost supermartingales, in the sense that

$$\mathbb{E}[\Delta V(k)|\mathcal{F}(k)] \le -\rho V(k) + c(k), \tag{49}$$

for some $\mathcal{F}(k)$-measurable random process $c(k)$. Such processes are termed as c-martingale in the literature [40]–[42]. Based on their definitions, we propose a c-martingale convergence lemma:

Lemma 6 (c-martingale convergence lemma)**.** *Suppose that there exist $\rho > 0$ and $\bar{c} > 0$ such that* (49) *holds and $\mathbb{E}[c(k)] \le \bar{c} < \infty$. Then it follows for any $k \ge 0$ that $\mathbb{E}\left[V(k)\right]$ is bounded.*

Proof. It follows from (49) that

$$0 \le \mathbb{E}[V(k+1)|\mathcal{F}(k)] \le (1-\rho)V(k) + c(k). \tag{50}$$

By taking expectation on both sides of (50), it yields that

$$\begin{aligned} 0 &\le \mathbb{E}\left[V(k+1)\right] \le (1-\rho)\,\mathbb{E}\left[V(k)\right] + \bar{c} \\ &\le (1-\rho)^{k+1}\,\mathbb{E}\left[V(0)\right] + \bar{c}\sum_{t=0}^{k}(1-\rho)^t. \end{aligned} \tag{51}$$

The proof is thus completed. □

Proof of Theorem 1. In order to prove Theorem 1, we shall respectively establish the consistency and consensus conditions.

Consistency: By (28) and (32), we rewrite the dynamics of the local state of sensor i in (26) as

$$\begin{aligned} \eta_i(k+1) = H\eta_i(k) + L_i z_i(k) + BT\sum_{j=1}^{m} a_{ij}(\eta_j(k) - \eta_i(k)) \\ + BT\sum_{j=1}^{m} a_{ij}(\epsilon_j(k) - \epsilon_i(k)). \end{aligned} \tag{52}$$

The consistency condition is verified by summing (52) over $i \in \{1, \cdots, m\}$.

Consensus: To simplify the notation, let us define the aggregated vector of the states $\eta_i(k)$ and the matrix of local gains L_i (given in (24)) as below:

$$\eta(k) \triangleq \begin{bmatrix} \eta_1(k) \\ \vdots \\ \eta_m(k) \end{bmatrix}, \ L_\eta \triangleq \begin{bmatrix} L_1 & & \\ & \ddots & \\ & & L_m \end{bmatrix}.$$

Collecting (52) from each sensor yields:

$$\begin{aligned} &\eta(k+1) \\ &= (I_m \otimes H)\eta(k) - [I_m \otimes (BT)](\mathcal{L}_\mathcal{G} \otimes I_{n(r+1)})\eta(k) \\ &\quad - [I_m \otimes (BT)](\mathcal{L}_\mathcal{G} \otimes I_{n(r+1)})\epsilon(k) + L_\eta z(k) \\ &= [I_m \otimes H - \mathcal{L}_\mathcal{G} \otimes (BT)]\eta(k) - [\mathcal{L}_\mathcal{G} \otimes (BT)]\epsilon(k) \\ &\quad + L_\eta z(k), \end{aligned} \tag{53}$$

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3309373

where $z(k)$ is defined in (23). Let us rewrite $\bar{\eta}(k)$ in (45) as

$$\bar{\eta}(k) = \frac{1}{m}\sum_{i=1}^{m}\eta_i(k) = \frac{1}{m}(\mathbf{1}_m^T \otimes I_{n(r+1)})\eta(k). \tag{54}$$

Since $\mathbf{1}_m^T\mathcal{L}_\mathcal{G} = 0$, it follows that

$$\begin{aligned}\bar{\eta}(k+1) &= \frac{1}{m}(\mathbf{1}_m^T \otimes I_{n(r+1)})\Big([I_m \otimes H - \mathcal{L}_\mathcal{G} \otimes (BT)]\eta(k) \\ &\quad - [\mathcal{L}_\mathcal{G} \otimes (BT)]\epsilon(k) + L_\eta z(k)\Big) \\ &= H\bar{\eta}(k) + \frac{1}{m}(\mathbf{1}_m^T \otimes I_{n(r+1)})L_\eta z(k).\end{aligned}$$

Furthermore, we define for each sensor i that

$$\delta_i(k) \triangleq \eta_i(k) - \bar{\eta}(k).$$

By stacking $\delta_i(k)$ together, let us denote $\delta(k) \triangleq \mathrm{col}(\delta_1(k), \cdots, \delta_m(k))$. We therefore have

$$\begin{aligned}\delta(k+1) &= [I_m \otimes H - \mathcal{L}_\mathcal{G} \otimes (BT)]\delta(k) \\ &+ [(I_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T) \otimes I_{n(r+1)}]L_\eta z(k) - [\mathcal{L}_\mathcal{G} \otimes (BT)]\epsilon(k).\end{aligned} \tag{55}$$

By [39], there always exists a unitary matrix $\Phi \triangleq [\frac{1}{\sqrt{m}}\mathbf{1}_m, \phi_2, \cdots, \phi_m]$, with which the Laplacian matrix can be diagonalized as

$$\Phi^T\mathcal{L}_\mathcal{G}\Phi = \mathrm{diag}(0, \mu_2, \cdots, \mu_m).$$

One hence concludes

$$\begin{aligned}&(\Phi \otimes I_{n(r+1)})^T[\mathcal{L}_\mathcal{G} \otimes (BT)](\Phi \otimes I_{n(r+1)}) \\ &= \mathrm{diag}(0, \mu_2 BT, ..., \mu_m BT), \\ &(\Phi \otimes I_{n(r+1)})^T[I_m \otimes H - \mathcal{L}_\mathcal{G} \otimes (B\Gamma)](\Phi \otimes I_{n(r+1)}) \\ &= \mathrm{diag}(H, H - \mu_2 BT, ..., H - \mu_m BT),\end{aligned} \tag{56}$$

which holds by the property of Kronecker product. Denote

$$\tilde{\delta}(k) \triangleq (\Phi \otimes I_{n(r+1)})^T\delta(k),\ \tilde{\epsilon}(k) \triangleq (\Phi \otimes I_{n(r+1)})^T\epsilon(k). \tag{57}$$

Let us further partition $\tilde{\delta}(k)$ into two parts, i.e., $\tilde{\delta}(k) = [\tilde{\delta}_1^T(k), \tilde{\delta}_2^T(k)]^T$, where $\tilde{\delta}_1(k) \in \mathbb{R}^{n(r+1)}$ consists of the first $n(r+1)$ entries of $\tilde{\delta}(k)$. One thus obtains from (55) that

$$\begin{aligned}\tilde{\delta}_1(k+1) &= \frac{1}{\sqrt{m}}\sum_{i=1}^{m}\delta_i(k+1) = 0, \\ \tilde{\delta}_2(k+1) &= A_\delta\tilde{\delta}_2(k) + L_z z(k) + B_\epsilon\tilde{\epsilon}_2(k),\end{aligned} \tag{58}$$

where $A_\delta \triangleq \mathrm{diag}(H - \mu_2 BT, \cdots, H - \mu_m BT)$, $B_\epsilon \triangleq \mathrm{diag}(-\mu_2 BT, \cdots, -\mu_m BT)$, and L_z is formed by the last $mn(r+1) - n(r+1)$ rows of $[(\Phi^T - \frac{1}{m}\Phi^T\mathbf{1}_m\mathbf{1}_m^T) \otimes I_{n(r+1)}]L_\eta$.

Clearly, $\tilde{\delta}_1(k+1)$ is stable. We thus focus on the stability of $\tilde{\delta}_2(k+1)$. In view of Lemma 5, A_δ is stable. Hence, there exist $\widehat{\mathcal{P}} > 0, \mathcal{Q} > 0$ and $\sigma_1, \sigma_2 > 0$ such that

$$(1 + \sigma_1 + \sigma_2)A_\delta^T\widehat{\mathcal{P}}A_\delta - \widehat{\mathcal{P}} + \mathcal{Q} = 0. \tag{59}$$

Then let us consider the following Lyapunov candidate:

$$V(k) = \tilde{\delta}_2^T(k)\widehat{\mathcal{P}}\tilde{\delta}_2(k). \tag{60}$$

It thus follows that

$$\begin{aligned}V(k) &= \mathrm{tr}(\tilde{\delta}_2^T(k)\widehat{\mathcal{P}}\tilde{\delta}_2(k)) = \mathrm{tr}(\widehat{\mathcal{P}}\tilde{\delta}_2(k)\tilde{\delta}_2^T(k)) \\ &\le \mathrm{tr}(\widehat{\mathcal{P}})\,\mathrm{tr}(\tilde{\delta}_2(k)\tilde{\delta}_2^T(k)) = \mathrm{tr}(\widehat{\mathcal{P}})||\tilde{\delta}_2(k)||^2.\end{aligned} \tag{61}$$

The difference of $V(k)$ along (58) is given by

$$\begin{aligned}&\mathbb{E}[\Delta V(k)|\mathcal{F}(k)] \triangleq \mathbb{E}[V(k+1) - V(k)|\mathcal{F}(k)] \\ &= \tilde{\delta}_2^T(k)(A_\delta^T\widehat{\mathcal{P}}A_\delta - \widehat{\mathcal{P}})\tilde{\delta}_2(k) + 2\tilde{\delta}_2^T(k)A_\delta^T\widehat{\mathcal{P}}L_z\mathbb{E}[z(k)|\mathcal{F}(k)] \\ &\quad + 2\tilde{\delta}_2^T(k)A_\delta^T\widehat{\mathcal{P}}B_\epsilon\tilde{\epsilon}(k) + 2\mathbb{E}[z^T(k)|\mathcal{F}(k)]L_z^T\widehat{\mathcal{P}}B_\epsilon\tilde{\epsilon}(k) \\ &\quad + L_z^T\widehat{\mathcal{P}}L_z\mathbb{E}[z^T(k)z(k)|\mathcal{F}(k)] + \tilde{\epsilon}^T(k)B_\epsilon^T\widehat{\mathcal{P}}B_\epsilon\tilde{\epsilon}(k).\end{aligned}$$

Now using Young's inequality, one concludes that

$$\begin{aligned}&\mathbb{E}[\Delta V(k)|\mathcal{F}(k)] \\ &\le \tilde{\delta}_2^T(k)[(1 + \sigma_1 + \sigma_2)A_\delta^T\widehat{\mathcal{P}}A_\delta - \widehat{\mathcal{P}}]\tilde{\delta}_2(k) \\ &\quad + (1 + \sigma_2^{-1} + \sigma_3)L_z^T\widehat{\mathcal{P}}L_z\mathbb{E}[z^T(k)z(k)|\mathcal{F}(k)] \\ &\quad + (1 + \sigma_1^{-1} + \sigma_3^{-1})\tilde{\epsilon}^T(k)B_\epsilon^T\widehat{\mathcal{P}}B_\epsilon\tilde{\epsilon}(k) \\ &\le -\lambda_{\min}(\mathcal{Q})||\tilde{\delta}_2(k)||^2 + c(k) \\ &\le -\frac{\lambda_{\min}(\mathcal{Q})}{\mathrm{tr}(\widehat{\mathcal{P}})}V(k) + c(k),\end{aligned} \tag{62}$$

where σ_1, σ_2 are given in (59), and the last inequality holds by (61). As proved in Lemma 4, $\mathrm{cov}(z_i(k))$ is bounded at any time. Moreover, $||\epsilon(k)||^2$ is also bounded by (36). It thus follows that $\mathbb{E}[c(k)] < \infty$. In view of Lemma 6, we conclude that $\mathbb{E}[V(k)]$ is bounded. As a result of (61), $\mathrm{cov}(\tilde{\delta}_2(k)) < \infty$. Combining it with (57), we have $\mathrm{cov}[\eta_i(k) - \bar{\eta}(k)]$ is bounded for any i, which completes the proof. □

The consistency condition (46) claims that the dynamics of $\bar{\eta}(k)$ is governed by $z(k)$ only. Therefore, the interaction among sensors only affects the evolution of each local state but not their average value $\bar{\eta}(k)$. On the other hand, (47) states that, despite the signal $z(k)$, each local state can track $\bar{\eta}(k)$ with bounded error covariance. These conditions will next help us establish the stability of local estimators.

Remark 5. *As implied by Theorem 1, a general class of triggering functions that guarantee* (36) *can be used to synchronize the local states governed by stochastic linear dynamics* (26)*. This problem is known as stochastic linear systems synchronization in the literature ([43]–[45]).*

B. Stability analysis of local estimators

In Theorem 1, we have proven that the synchronization algorithm (26) facilitates both the consistency and consensus conditions among local states. We shall, in this subsection, show how these conditions will help to achieve a stable local estimate at each sensor side.

First, we show that the average of the local estimates of all sensors is indeed the optimal Kalman estimate (7). This is particularly guaranteed by the consistency condition (46):

Lemma 7. *Suppose that the condition* (40) *holds, and* Γ *is designed based on* (41) *and* (42)*. By performing Algorithm 1, it holds at any* $k \ge 0$ *that*

$$\frac{1}{m}\sum_{i=1}^{m}\breve{x}_i(k) = \hat{x}(k). \tag{63}$$

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3309373

Proof. By (46), it follows for any $j \in \{1, \cdots, r\}$ that

$$\sum_{i=1}^{m} \eta_{j,i}(k+1) = S \sum_{i=1}^{m} \eta_{j,i}(k) + \sum_{i=1}^{m} v_{ji}\, \mathbf{1}_n\, z_i(k). \quad (64)$$

Comparing it with (14), we can obtain for any time k that $\sum_{i=1}^{m} v_{ji}\hat{\xi}_i(k) = \sum_{i=1}^{m} \eta_{j,i}(k)$. Therefore, it follows at any $k \geq 0$ that:

$$\begin{aligned}\sum_{i=1}^{m} \eta_{0,i}(k+1) &= (A - KCA)\sum_{i=1}^{m} \eta_{0,i}(k) \\ &\quad + \sum_{j=1}^{r} \tilde{K}_j \beta^T \sum_{i=1}^{m} \eta_{j,i}(k) + \sum_{i=1}^{m} K_i z_i(k) \\ &= (A - KCA)\sum_{i=1}^{m} \eta_{0,i}(k) + \sum_{j=1}^{r} \tilde{K}_j \beta^T \sum_{i=1}^{m} v_{ji}\hat{\xi}_i(k) \\ &\quad + \sum_{i=1}^{m} K_i z_i(k).\end{aligned}$$

Comparing it with (18) and (19), one concludes that

$$\hat{x}(k) = \sum_{i=1}^{m} \eta_{0,i}(k) = \frac{1}{m}\sum_{i=1}^{m} \breve{x}_i(k). \quad (65)$$

□

On the other hand, it is also desired to analyze the stability of local estimators. We shall prove it by using the consensus condition (47), as stated in the following theorem:

Theorem 2. *Suppose that the condition* (40) *holds, and* Γ *is designed based on* (41) *and* (42)*. By performing Algorithm 1, it holds at any* $k \geq 0$ *that*

$$\mathrm{cov}(\breve{x}_i(k) - x(k)) < \infty,\ \forall i. \quad (66)$$

Namely, the error covariance of each local estimate is bounded.

Proof. Let us consider the local estimate of any sensor i, i.e., $\breve{x}_i(k)$. By virtue of (47), we conclude that $\mathrm{cov}(\eta_{0,i}(k) - \bar{\eta}_0(k))$ is bounded at any time k, where $\bar{\eta}_0(k) = \frac{1}{m}\sum_{i=1}^{m} \eta_{0,i}(k)$. Then in order to prove the boundedness of $\mathrm{cov}(\breve{x}_i(k) - x(k))$, let us denote

$$\bar{e}_i(k) \triangleq \breve{x}_i(k) - \hat{x}(k), \quad (67)$$

which is the distance between $\breve{x}_i(k)$ and the optimal Kalman estimate. Combining it with (65) yields

$$\bar{e}_i(k) = m(\eta_{0,i}(k) - \bar{\eta}_0(k)). \quad (68)$$

Thus, the local estimation error of sensor i is calculated as

$$\begin{aligned}\breve{e}_i(k) &\triangleq \breve{x}_i(k) - x(k) = (\breve{x}_i(k) - \hat{x}(k)) + (\hat{x}(k) - x(k)) \\ &= \bar{e}_i(k) + \hat{e}(k),\end{aligned} \quad (69)$$

where $\hat{e}(k)$ is the estimation error of Kalman filter. According to the orthogonality principle [46], $\bar{e}_i(k)$ is orthogonal to $\hat{e}(k)$. Therefore, it follows that

$$\begin{aligned}\mathrm{cov}(\breve{e}_i(k)) &= \mathrm{cov}(\bar{e}_i(k)) + \mathrm{cov}(\hat{e}(k)) \\ &= m^2\, \mathrm{cov}(\eta_{0,i}(k) - \bar{\eta}_0(k)) + P,\end{aligned} \quad (70)$$

where P is the steady-state error convariance of the Kalman filter as defined in (5). Since $\mathrm{cov}(\eta_{0,i}(k) - \bar{\eta}_0(k))$ is bounded, we complete the proof. □

Based on Theorem 2, we establish that each sensor provides a stable local estimate under the minimum requirement of collective system observability and condition (40) for synchronization. Furthermore, analyzing (70), we conclude that the performance gap between our estimator and the optimal Kalman filter results purely from the consensus error $\mathrm{cov}[\eta_{0,i}(k) - \bar{\eta}_0(k)]$. This insight allows us to leverage existing results on synchronization algorithms for stochastic linear systems within our proposed framework to address this consensus error. Consequently, our distributed estimation problem can be effectively solved by employing any algorithm designed for achieving synchronization in stochastic linear systems, as supported by Theorems 1 and 2. This finding bridges the gap between these two fields.

Additionally, in the proof of Theorem 1, we observe that the consensus error is introduced by the local innovation signals $\{z_i(t)\}_{t \leq k}$ and the event-triggering function. Notably, if we allow for infinite consensus iterations between two consecutive sampling times to generate $\{z_i(t)\}$, the consensus error diminishes, and the performance of our local estimator aligns with that of the Kalman filter.

Remark 6. *As shown in Figs. 1 and 2, this paper provides a novel framework which decouples the local filter from the synchronization process. Notice that this decoupling structure also simplifies our convergence analysis. To be concrete, in this work, the stability of local estimates is studied through the supermartingale convergence theory. Moreover, for the synchronization of the local states, we assume* (40)*, which relates the network topology and the instability of the system. To the best of our knowledge, this is the first time that the algorithm convergence is analyzed under these conditions in the context of distributed estimation. This is because the coupling structure in the existing solutions makes it difficult to properly define a supermartingale or c-martingale. Instead, these works establish the stability under stronger assumptions which make the proof possible. For example, the authors in [21] and [18] proved the stability of estimation errors by assuming that the system matrix is invertible and the network is strongly connected. Moreover, in [17], Meng et al. conducted the proof by ignoring noises in the system.*

Remark 7. *In Theorem 2, we have established the boundedness of the estimation error. Nevertheless, it is crucial to acknowledge that in this study, the communication is triggered by noisy states. The interplay between communication and random noises introduces complexity, making it challenging to exactly calculate the convergence rate and estimation error. This challenge is aligned with investigations conducted in other event-based distributed estimation algorithms, such as [17]–[21], [47], [48].*

VI. Low Message Complexity Estimator Design

In practice, a communication channel is usually limited by a finite bandwidth. Inspired by it, this section further investigates the design of distributed estimators under the constraint of message complexity. Specifically, suppose that the message complexity that the network is willing to tolerate is $\tilde{r} > 0$. We shall show how to design the distributed estimate algorithm such that each sensor only needs to send out a message of length no greater than $\tilde{r}$ at each transmission.

A. Extension to Luenberger observers

Our design is achieved by implementing a centralized Luenberger observer with the proposed framework. To see this, we start with a Luenberger observer for estimating (1) given by

$$\hat{x}(k+1) = (A - K_{\tilde{r}}CA)\hat{x}(k) + K_{\tilde{r}}y(k+1), \tag{71}$$

where $K_{\tilde{r}}$ is the estimation gain of the Luenberger observer. Clearly, the steady-state Kalman filter (7) also belongs to the class of Luenberger observers.

By replacing the Kalman gain K with $K_{\tilde{r}}$, it is not difficult to verify that all results in Sections III–V still hold. That is, we can generalize the results in Section III to obtain a lossless decomposition of the Luenberger observer (71). Based on this decomposition, Algorithm 1 can be applied to achieve a distributed implementation of (71), where all parameters (e.g., $\tilde{K}$, V, F, Γ, *etc.*) are calculated by using $K_{\tilde{r}}$ instead of K. Notice that, as stated in Remark 3, the message complexity of this implementation is $\mathrm{rank}(K_{\tilde{r}})$. As a result, one can reduce the message complexity by using a Luenberger observer with a gain $K_{\tilde{r}}$ such that $\mathrm{rank}(K_{\tilde{r}}) \leq \tilde{r}$.

In what follows, we show how to design the (sub)optimal estimation gain $K_{\tilde{r}}$ such that its rank is upper bounded by $\tilde{r}$ while yielding the minimum estimation error. It is obvious that for any $K_{\tilde{r}} \in \mathbb{R}^{n\times n}$, it can be factorized as

$$K_{\tilde{r}} = \bar{K}W, \tag{72}$$

where $\bar{K} \in \mathbb{R}^{n\times\tilde{r}}$ and $W \in \mathbb{R}^{\tilde{r}\times m}$. Therefore, the (sub)optimal gain $K_{\tilde{r}}$ can be obtained as follows:

1) Suppose that W is given. We first show that the optimal $\bar{K}$ can be directly calculated through a function of W. As such, $K_{\tilde{r}}$ purely depends on W.
2) The second step finds the (sub)optimal W under the constraint that $\mathrm{rank}(W) \leq \tilde{r}$.
3) Finally, one can obtain the (sub)optimal $K_{\tilde{r}}$ via (72). It is easy to verify that $\mathrm{rank}(K_{\tilde{r}}) \leq \mathrm{rank}(W) \leq \tilde{r}$.

Particularly, if $\tilde{r} \geq \min\{m, n\}$, the (sub)optimal $K_{\tilde{r}}$ obtained from the above procedure is indeed the Kalman gain. In the rest of this section, we shall detail these steps.

B. Optimal $\bar{K}$ when W is given

First, we shall show how to design the optimal $\bar{K}$ with a given W. Let us consider the following measurements given by a "virtual" sensor network:

$$\tilde{y}(k) = \tilde{C}x(k) + \tilde{v}(k), \tag{73}$$

where

$$\tilde{y}(k) = Wy(k),\ \tilde{C} = WC,\ \tilde{v}(k) = Wv(k). \tag{74}$$

Suppose that this "virtual" sensor network is monitoring the system (1) and a Luenberger observer is performed with estimation gain $\bar{K}$, where $\bar{K}$ is defined in (72). Let us respectively denote by $\tilde{x}(k)$ and $\tilde{P}$ the corresponding estimate and error covariance. Specifically, they can be expressed as

$$\tilde{x}(k+1) = (A - \bar{K}\tilde{C}A)\tilde{x}(k) + \bar{K}\tilde{y}(k+1), \tag{75}$$

and

$$\tilde{P}(k) = \mathrm{cov}(\tilde{x}(k) - x(k)),\ \tilde{P} = \lim_{k\to\infty}\tilde{P}(k). \tag{76}$$

The following result is immediate:

Lemma 8. *Let $P_{\tilde{r}}$ be the steady-state estimation error covariance of the Luenberger observer* (71)*. Then it follows that*

$$P_{\tilde{r}} = \tilde{P}. \tag{77}$$

As a result of Lemma 8, we now focus on finding the optimal $\bar{K}$ which minimizes $\mathrm{tr}(\tilde{P})$. Since W is given, clearly this optimal solution is provided by the Kalman filter, where the steady-state error covariance can be calculated as

$$\tilde{P} = [(A\tilde{P}A^T + Q)^{-1} + \tilde{C}^T(\tilde{R})^{-1}\tilde{C}]^{-1}, \tag{78}$$

where

$$\tilde{R} = WRW^T. \tag{79}$$

Moreover, the optimal $\bar{K}$ is given by

$$\bar{K} = (A\tilde{P}A^T + Q)\tilde{C}^T[\tilde{C}(A\tilde{P}A^T + Q)\tilde{C}^T + \tilde{R}]^{-1}. \tag{80}$$

C. Towards finding the (sub)optimal W

As seen from (78) and (79), the error covariance $\tilde{P}$ is a function of W. Therefore, we next aim to find the optimal W in the sense that $\mathrm{tr}(\tilde{P})$ is minimized under the constraint that $\mathrm{rank}(W) = \tilde{r}$. Notice that W appears only in the term $\tilde{C}^T(\tilde{R})^{-1}\tilde{C}$ of (78). We thus rewrite it as

$$\begin{aligned}\tilde{C}^T(\tilde{R})^{-1}\tilde{C} &= C^TW^T(WRW^T)^{-1}WC \\ &= C^TR^{-1/2}[R^{1/2}W^T(WRW^T)^{-1}WR^{1/2}]R^{-1/2}C.\end{aligned} \tag{81}$$

Let us denote

$$X \triangleq R^{1/2}W^T(WRW^T)^{-1}WR^{1/2} \in \mathbb{R}^{m\times m},\ \bar{C} \triangleq R^{-1/2}C. \tag{82}$$

It is easy to verify that X is a symmetric projection matrix, namely, $X^2 = X$ and $X = X^T$. Moreover, $\mathrm{rank}(X) = \mathrm{rank}(W) = \tilde{r}$. On the other hand, given any symmetric projection matrix X which is of rank $\tilde{r}$, one can also find a matrix W as

$$W = \left(R^{-1/2}\begin{bmatrix}v_1 & \cdots & v_m\end{bmatrix}\right)^T, \tag{83}$$

where $\{v_1, \cdots, v_m\}$ is an orthonormal basis of the column space of X. We can verify that this W satisfies (82). Therefore, instead of minimizing $\mathrm{tr}(\tilde{P})$ over W, we can minimize it over X. That is,

$$\begin{aligned}\underset{X,\,\tilde{P}}{\text{minimize}}\quad & \mathrm{tr}(\tilde{P}) \\ \text{subject to}\quad & \tilde{P} = [(A\tilde{P}A^T + Q)^{-1} + \bar{C}^TX\bar{C}]^{-1}, \\ & X^T = X,\ X^2 = X,\ \mathrm{rank}(X) = \tilde{r}.\end{aligned} \tag{84}$$

We shall follow [49] and manipulate the first constraint into Linear Matrix Inequalities (LMIs). Moreover, since the last constraint on rank is not convex, we further use the convex relaxation proposed in [49] and compute X by solving the following SDP:

$$\begin{aligned} \underset{X,\tilde{P},\Theta}{\text{minimize}} \quad & \operatorname{tr}(\tilde{P}) \\ \text{subject to} \quad & \begin{bmatrix} \tilde{P} & I \\ I & \Theta \end{bmatrix} \geq 0, \\ & \begin{bmatrix} Q^{-1} - \Theta + \bar{C}^T X \bar{C} & Q^{-1}A \\ A^T Q^{-1} & \Theta + A^T Q^{-1} A \end{bmatrix} \geq 0, \\ & X^T = X,\, 0 \leq X \leq I_m,\, \operatorname{tr}(X) = \tilde{r}. \end{aligned} \tag{85}$$

Remark 8. *The SDP* (85) *is always solvable for $\tilde{r} > 0$, since one can verify that $X = \frac{\tilde{r}}{m} I_m$ is a feasible solution of it.*

Remark 9. *Since we have relaxed the non-convex constraint on rank, the SDP* (85) *is no longer equivalent to the original problem* (84)*. However, as proved in [49, Lemma 4], the feasible region of this SDP is the convex hull formed by all feasible solutions of the original problem. Moreover, the optimal value of* (85) *provides a lower bound on the optimal value of* (84)*. Importantly, if the optimal solution of this SDP is feasible for* (84)*, it coincides with the optimal solution of the original problem.*

For (85), we can obtain the optimal solution X_* and $\tilde{P}_*$. However, since the constraint on the rank of X has been relaxed, the matrix X_* may not be a projection with rank $\tilde{r}$. In such a case, one can obtain an approximation based on X_*. Specifically, we apply an eigendecomposition to X_* as

$$X_* = U_* \operatorname{diag}(\lambda_1, \ldots, \lambda_m) U_*^T,$$

where U_* is orthonormal and $\lambda_1 \geq \cdots \geq \lambda_m$ are the eigenvalues of X_*. We thus can obtain a projection matrix X_0 as

$$X_0 = U_* \operatorname{diag}(\underbrace{1,\ldots,1}_{\tilde{r}}, \underbrace{0,\ldots,0}_{m-\tilde{r}}) U_*^T.$$

It is easy to verify $\operatorname{rank}(X_0) = \tilde{r}$. Therefore, from X_0, the (sub)optimal W can be obtained via (83). Then combining (72), (78), and (80), we finally obtain a (sub)optimal estimation gain $K_{\tilde{r}}$, the rank of which is no more than $\tilde{r}$. As discussed previously, one can implement the Luenberger observer with $K_{\tilde{r}}$ in a distributed manner by performing Algorithm 1, where the message complexity is at most $\tilde{r}$.

Remark 10. *As observed from Remark 3, using a low-rank estimator also reduces the computational overhead of performing* (26)*. This point will be discussed further in Section VII.*

Our motivation for proposing the low-rank estimator design is driven by the objective to reduce message complexity in the developed distributed framework. The use of a low-rank gain offers a promising solution to achieve this goal. To the best of our knowledge, this work represents the first attempt to explore the design of the (sub)optimal Luenberger observer with a low-rank estimation gain.

In the literature, one relevant problem is the design of reduced-order Luenberger observers (see [26]). Under the assumption that the number of states is no less than that of measurements, i.e., $n \geq m$, and the measurement matrix $C \in \mathbb{R}^{m\times n}$ is of low rank, i.e., $\operatorname{rank}(C) = p < m$, the results therein effectively reduce the order of estimators from n to $n - p$, and yields an estimation gain $\widehat{K} \in \mathbb{R}^{(n-p)\times p}$ in the *noise-free* environment. Since $\operatorname{rank}(\widehat{K}) \leq \min(n-p, p)$, the reduced-order Luenberger observer also leads to an estimation gain with low rank.

However, this work takes stochastic noises into account, presenting a departure from the above-mentioned approach. In our design, we directly optimize estimation performance over the gain K, under the constraint that $\operatorname{rank}(K) \leq \tilde{r}$. Notice that, given any $1 \leq \tilde{r} \leq \min(m, n)$, the proposed optimization problem is always feasible. This means that our design is more general than the ones using reduced-order observers, since $\tilde{r}$ need not be determined by rank of the measurement matrix C. Consequently, our method remains effective even when $n < m$ or when C is of full rank. Furthermore, our approach finds the optimal gain among all feasible ones, resulting in the minimum performance loss among all solutions, including the gains of the reduced-order observers. This highlights the superiority of our method in achieving the reduced message complexity and the optimal estimation performance.

VII. Comparison with Existing Works

Before closing the main sections of this paper, we finally compare our algorithm with some existing works.

Recall that, the framework proposed in this paper is built based on our previous work [15]. However, in order to save the communication efforts, we non-trivially extend the results therein from three aspects: 1) We apply event-based strategies to reduce the number of transmissions in the network. 2) A new method for decomposing the Kalman filter is proposed to reduce the message complexity. By doing so, the size of message exchanged at each transmission is also limited. 3) We propose the design of low-rank estimators, which allows us to further reduce the message complexity with some tradeoff in its estimation performance.

In TABLE I, we also present a comparison between our work and several existing event-based solutions. In particular, from the table, it is observed that both message complexity and computational overhead are fixed in the existing works. In contrast, our framework, as indicated in Remark 3, allows these complexities to be determined by $\operatorname{rank}(K)$. This characteristic introduces a new level of flexibility, enabling us to reduce these complexities effectively through the design of an estimation gain with a lower rank, as proposed in Section VI. Moreover, since $\operatorname{rank}(K) \leq \min(m, n)$, the proposed algorithm always outperforms existing works in terms of message complexity and computational overhead, regardless of the network size. The table also reveals that our algorithm achieves superior applicability and performance by not requiring the invertibility of the system matrix A and by accommodating a wide range of triggering functions.

Algorithm	Consensus on	Message transmitted to neighbors	Message complexity	Computational overhead	Triggering function	Stability
[21]	Information	Information matrices and vectors	n^2+n	$O(n^3)$	Specific	Yes (requires that A is invertible)
[17]	Estimates	Estimates	n	$O(n^2)$	Specific	Yes (requires that A is invertible)
[18]	Information	Information matrices and vectors	n^2+n	$O(n^3)$	Specific	Yes (requires that A is invertible)
[20]	Estimates	Covariance matrices and estimates	n^2+n	$O(n^3)$	Specific	Yes (requires that A is invertible)
This work	Estimates	"Encoded" estimates	$1 \leq \tilde{r} \leq \min(m,n)$	$O(\tilde{r}n^2)$	A large class of triggering functions can be used	Yes (no requirement on A)

TABLE I: Comparison with different event-based distributed estimation algorithms.

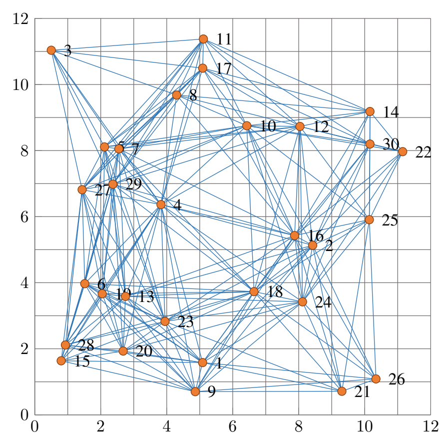

Fig. 4: The topology of 30 sensors in the 12×12 grid.

VIII. Numerical Examples

In this section, we aim to verify the established results through some numerical examples.

As shown in Fig. 4, let us consider a network of $m = 30$ sensors, which is deployed to monitor the temperature within a region represented by a 12×12 grid. Each sensor has the communication radius of 6. By discretizing the partial differential equation which describes heat transfer process, this problem can be modeled as a linear Gaussian system as in (1) and (2). Due to space limitation, we omit the detailed constructions but refer the readers to [50].

The temperature of each grid is taken as a state. Therefore, $n = 144$. The covariances of system and measurement noises are respectively chosen as $Q = 0.8I_n$ and $R = I_m$.

A. Performance of Algorithm 1 at different communicate rates

In this example, (37) is selected as the triggering function. Adjusting parameters in the function yields different communication rate, which is defined by

$$\sigma \triangleq \frac{1}{m}\sum_{i=1}^{m} \frac{\#\text{ of triggering instants of sensor } i}{\#\text{ of total instants}}. \tag{86}$$

In Fig. 5, we show the performance of Algorithm 1 at different communicate rates. Specifically, we characterize the estimation performance by averaging the norm of estimation error among all sensors.

From the figure, it is observed that similar performances are achieved at the beginning stage. This is because at this stage, estimation error is far from the steady state. Therefore, the sensors are always triggered in all situations. However, it is clear that the steady-state error increases when the communication becomes less frequent at the later stage.

Fig. 6 further demonstrates the steady-state estimation error at different communication rates. The results are obtained through 1000-run Monte Carlo trials, where the initial states and noises are randomly set for each trial. They are given in box and whisker diagrams where the bottom and top of the box represent the first and third quartiles, the (red) band inside the box represents the median of the data, and the ends of the whiskers represent the minimum and maximum of the data. From this figure, it is easy to conclude the trade-off between the communication rate and estimation performance.

B. Performance comparison of different algorithms

Next, we compare the estimation performance of our algorithm with the centralized Kalman filter and other event-based distributed estimators listed in TABLE I, i.e., those from [21], [17], [18], and [20]. We repeat the simulation for 100 times. From Fig. 7, it is observed that with the similar communication rate at around 80%, our algorithm enjoys the largest convergence rate as compared to the others.

Notice that, although the estimator in [21] yields less steady-state error than ours, from TABLE I, we know that

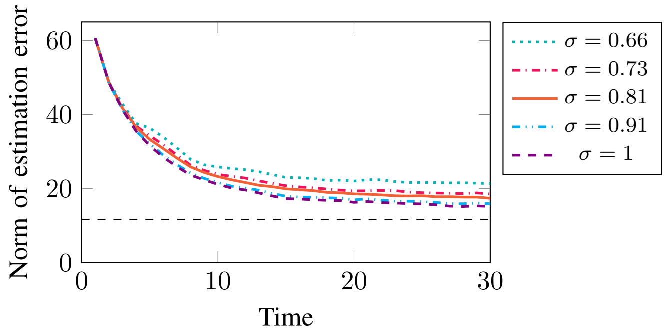

Fig. 5: Time responses of Algorithm 1 at different communicate rates, where the black dashed line denotes the steady-state error of the Kalman filter.

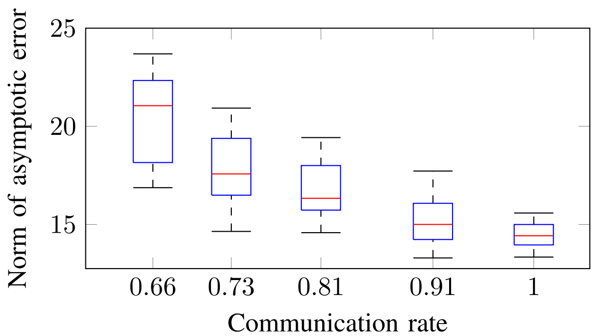

Fig. 6: Asymptotic performance of Algorithm 1 at different communicate rates.

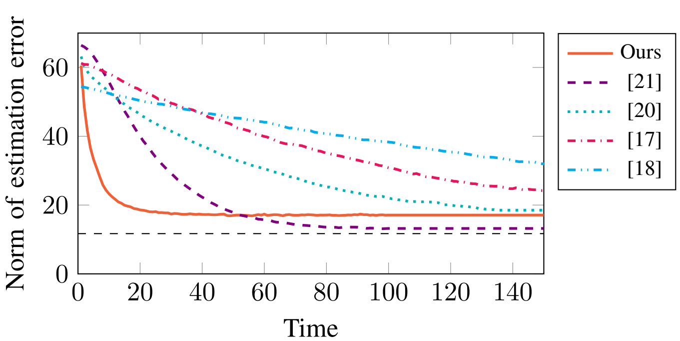

Fig. 7: Time responses of different event-triggered distributed algorithms, where the black dashed line denotes the steady-state error of Kalman filter. Communication rates in different algorithms are 80.60% (our algorithm), 81.13% ([21]), 80.05% ([17]), 81.22% ([18]), and 79.80% ([20]).

this is achieved at the expense of high message complexity. Specifically, at each transmission, the sizes of data to be transmitted in different algorithms are $\mathrm{rank}(K) = 30$ (the proposed algorithm), $n^2+n = 20880$ ([21]), $n = 144$ ([17]), $n^2+n = 20880$ ([18]), and $n^2+n = 20880$ ([20]). Therefore, although with the almost same number of transmissions, we require the least size of data to be transmitted. Clearly, our advantage in reducing the data size will be more apparent in a larger network with increasing number of states and sensors.

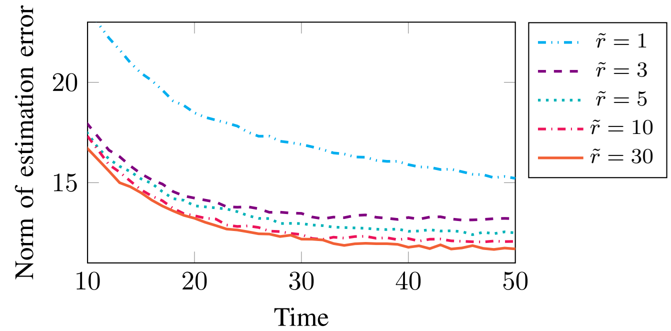

Fig. 8: Time responses of the designed low-rank estimators.

C. Performance of low-rank estimators

Using the system above, we present the performance of the low-rank estimators designed in Section VI. In Fig. 8, we depict the estimation errors of these designed estimators for different ranks: $\tilde{r} = 1, 3, 5, 10, 30$. Notably, the estimator with $\tilde{r} = 30$ corresponds to the optimal Kalman filter. It is evident from the figure that even with $\tilde{r} = 10$, meaning that only one-third of the degrees of freedom are utilized for the estimator design, the performance loss is minor, approximately at the level of 10%. Therefore, we can significantly reduce the message complexity by implementing this rank-10 estimator in a distributed manner using our algorithm.

IX. Conclusion

This paper has addressed the problem of distributed estimation with event-based communication protocols. By decomposing the centralized estimator, we have reformulated the problem of distributed estimation to that of stochastic linear systems synchronization, in which a large class of triggering functions are effective in yielding a stable local estimate at every sensor side. Given any $\tilde{r}$, an SDP has been presented, which gives the (sub)optimal gain of centralized estimator. By implementing it with our distributed algorithm, we have shown that the proposed estimator requires lower message complexity of no more than $\tilde{r}$.

Appendix A
Proof of Lemma 4

The local filters of the form (14) were proposed in [15]. Lemma 4 is included in the results there, but not explicitly. Here we present the proof for the sake of completeness.

1) It follows from (1) and (8) that

$$x^s(k+1) = A^s x^s(k) + Jw(k), \tag{87}$$

where $J = \begin{bmatrix} 0 & \mathbf{1}_{n^s} \end{bmatrix} \in \mathbb{R}^{n^s \times n}$ and $x(k) = \mathrm{col}(x^u(k), x^s(k))$ with $x^u(k) \in \mathbb{R}^{n^u}$ and $x^s(k) \in \mathbb{R}^{n^s}$. Moreover, let us partition C_i in accordance with (8) as $C_i = \begin{bmatrix} C_i^u & C_i^s \end{bmatrix}$, with $C_i^u \in \mathbb{R}^{1\times n^u}$ and $C_i^s \in \mathbb{R}^{1\times n^s}$.

It is not difficult to verify from (12) and (14) that $\hat{\xi}_i(k)$ can be rewritten as

$$\hat{\xi}_i(k+1) = \Lambda \hat{\xi}_i(k) + \mathbf{1}_n\, y_i(k+1). \tag{88}$$

By Lemma 2, (S^T, β) is controllable. From (13), we also conclude $p_{S^T}(A^u) = 0$. Then, by Lemma 3, for any $i \in \mathcal{V}$, we can find $G_i^u \in \mathbb{R}^{n \times n^u}$ such that

$$(G_i^u)^T S^T = (A^u)^T (G_i^u)^T,\ (G_i^u)^T \beta = (C_i^u A^u)^T,$$

which implies that

$$\begin{aligned} G_i^u A^u - \mathbf{1}_n C_i^u A^u &= S G_i^u - \mathbf{1}_n \beta^T G_i^u \\ &= (\Lambda + \mathbf{1}_n \beta^T) G_i^u - \mathbf{1}_n \beta^T G_i^u = \Lambda G_i^u, \\ \beta^T G_i^u &= C_i^u A^u. \end{aligned} \tag{89}$$

Therefore, we conclude

$$\begin{aligned} \begin{bmatrix} G_i^u & 0 \end{bmatrix} A - \mathbf{1}_n C_i A &= \begin{bmatrix} G_i^u A^u & 0 \end{bmatrix} - \mathbf{1}_n \begin{bmatrix} C_i^u A^u & C_i^s A^s \end{bmatrix} \\ &= \Lambda \begin{bmatrix} G_i^u & 0 \end{bmatrix} - \mathbf{1}_n \begin{bmatrix} 0 & C_i^s A^s \end{bmatrix}, \\ \beta^T \begin{bmatrix} G_i^u & 0 \end{bmatrix} &= \begin{bmatrix} C_i^u A^u & 0 \end{bmatrix} = C_i A - \begin{bmatrix} 0 & C_i^s A^s \end{bmatrix}. \end{aligned} \tag{90}$$

For simplicity, let us denote $G_i \triangleq \begin{bmatrix} G_i^u & 0 \end{bmatrix} \in \mathbb{R}^{n \times n}$. Moreover, define

$$\epsilon_i(k) \triangleq G_i x(k) - \hat{\xi}_i(k). \tag{91}$$

Following [15], it is verified that $\mathrm{cov}(\epsilon_i(k))$ is bounded.

Next, by (14) and (91), it follows $z_i(k) = y_i(k+1) - \beta^T (G_i x(k) - \epsilon_i(k))$. Then, by (1), (2), and (90),

$$\begin{aligned} z_i(k) &= C_i(Ax(k) + w(k)) + v_i(k+1) + \beta^T \epsilon_i(k) \\ &\quad - (C_i A - \begin{bmatrix} 0 & C_i^s A^s \end{bmatrix}) x(k) \\ &= \beta^T \epsilon_i(k) + C_i^s A^s x^s(k) + C_i w(k) + v_i(k+1). \end{aligned} \tag{92}$$

From (8), $x^s(k)$ has bounded covariance. Since $\epsilon_i(k)$ is stable, we conclude that $\mathrm{cov}(z_i(k))$ is also bounded.

2) To prove (15), let us multiply both sides of (88) by F_i from the left, which gives

$$F_i \hat{\xi}_i(k+1) = F_i \Lambda \hat{\xi}_i(k) + F_i \mathbf{1}_n y_i(k+1). \tag{93}$$

Since F_i solves (11), one obtains

$$F_i \hat{\xi}_i(k+1) = (A - KCA) F_i \hat{\xi}_i(k) + K_i y_i(k+1). \tag{94}$$

Summing up (94) for all $i \in \{1, \cdots, m\}$ yields that

$$\sum_{i=1}^m F_i \hat{\xi}_i(k+1) = (A - KCA) \sum_{i=1}^m F_i \hat{\xi}_i(k) + \sum_{i=1}^m K_i y_i(k+1).$$

By comparing this with (10), we complete the proof.

Appendix B
Proof of Lemma 5

Consider any $j \in \{2, \ldots, n\}$. It follows that

$$H - \mu_j BT = \begin{bmatrix} A - KCA & \tilde{K}_1 \beta^T & \cdots & \tilde{K}_r \beta^T \\ & S - \mu_j \mathbf{1}_n \Gamma & & \\ & & \ddots & \\ & & & S - \mu_j \mathbf{1}_n \Gamma \end{bmatrix}.$$

As $(\Lambda, \mathbf{1}_n)$ is controllable, $(S, \mathbf{1}_n)$ is also controllable by (23). Hence, by the choice of ζ, there exists $\mathcal{P} > 0$ that solves (42) ([39]). Then, following similar arguments as presented in [15, Lemma 6], one concludes that $\rho(S - \mu_j \mathbf{1}_n \Gamma) < 1$. Since $A - KCA$ is stable, our proof is completed.

References

[1] M. V. Subbotin and R. S. Smith, "Design of distributed decentralized estimators for formations with fixed and stochastic communication topologies," *Automatica*, vol. 45, no. 11, pp. 2491–2501, 2009.

[2] Z.-Q. Luo, "Universal decentralized estimation in a bandwidth constrained sensor network," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 2210–2219, 2005.

[3] B. Jia, K. D. Pham, E. Blasch, D. Shen, Z. Wang, and G. Chen, "Cooperative space object tracking using space-based optical sensors via consensus-based filters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1908–1936, 2016.

[4] J. Yan, C. Deng, and C. Wen, "Resilient output regulation in heterogeneous networked systems under Byzantine agents," *Automatica*, vol. 133, p. 109872, 2021.

[5] B. D. Anderson and J. B. Moore, *Optimal Filtering*. Courier Corporation, 2012.

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

[7] B. Chen, G. Hu, D. W. Ho, and L. Yu, "Distributed Kalman filtering for time-varying discrete sequential systems," *Automatica*, vol. 99, pp. 228–236, 2019.

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

[9] 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, 2014.

[10] G. Battistelli and L. Chisci, "Stability of consensus extended Kalman filter for distributed state estimation," *Automatica*, vol. 68, pp. 169–178, 2016.

[11] S. Del Favero and S. Zampieri, "Distributed estimation through randomized gossip Kalman filter," in *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with the 28th Chinese Control Conference*. IEEE, 2009, pp. 7049–7054.

[12] S. Kar and J. M. Moura, "Gossip and distributed Kalman filtering: Weak consensus under weak detectability," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1766–1784, 2010.

[13] G. Battistelli and L. Chisci, "Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability," *Automatica*, vol. 50, no. 3, pp. 707–718, 2014.

[14] M. Farina, G. Ferrari-Trecate, and R. Scattolini, "Distributed moving horizon estimation for linear constrained systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2462–2475, 2010.

[15] J. Yan, X. Yang, Y. Mo, and K. You, "A distributed implementation of steady-state Kalman filter," *IEEE Transactions on Automatic Control*, to appear, 2022. DOI: 10.1109/TAC.2022.3175925. Full version: arXiv preprint arXiv:2101.10689.

[16] Q. Liu, Z. Wang, X. He, and D.-H. Zhou, "Event-based recursive distributed filtering over wireless sensor networks," *IEEE Transactions on Automatic Control*, vol. 60, no. 9, pp. 2470–2475, 2015.

[17] X. Meng and T. Chen, "Optimality and stability of event triggered consensus state estimation for wireless sensor networks," in *Proceedings of 2014 American Control Conference*. IEEE, 2014, pp. 3565–3570.

[18] D. Yu, Y. Xia, L. Li, and D.-H. Zhai, "Event-triggered distributed state estimation over wireless sensor networks," *Automatica*, vol. 118, p. 109039, 2020.

[19] W. Song, J. Wang, S. Zhao, and J. Shan, "Event-triggered cooperative unscented Kalman filtering and its application in multi-UAV systems," *Automatica*, vol. 105, pp. 264–273, 2019.

[20] X. He, C. Hu, Y. Hong, L. Shi, and H.-T. Fang, "Distributed Kalman filters with state equality constraints: Time-based and event-triggered communications," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 28–43, 2020.

[21] G. Battistelli, L. Chisci, and D. Selvi, "A distributed Kalman filter with event-triggered communication and guaranteed stability," *Automatica*, vol. 93, pp. 75–82, 2018.

[22] L. Yan, X. Zhang, Z. Zhang, and Y. Yang, "Distributed state estimation in sensor networks with event-triggered communication," *Nonlinear Dynamics*, vol. 76, no. 1, pp. 169–181, 2014.

[23] X. He, Y. Xing, J. Wu, and K. H. Johansson, "Event-triggered distributed estimation with decaying communication rate," *SIAM Journal on Control and Optimization*, vol. 60, no. 2, pp. 992–1017, 2022.

[24] J. Yan, Y. Mo, and H. Ishii, "Event-based control for synchronization of stochastic linear systems with application to distributed estimation," *IFAC-PapersOnLine*, vol. 55, no. 13, pp. 115–120, 2022.

[25] L. Shi, P. Cheng, and J. Chen, "Sensor data scheduling for optimal state estimation with communication energy constraint," *Automatica*, vol. 47, no. 8, pp. 1693–1698, 2011.

[26] P. J. Antsaklis and A. N. Michel, *Linear Systems*. Birkhäuser, 1997.

[27] 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.

[28] W. Li and Y. Jia, "Consensus-based distributed multiple model UKF for jump Markov nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 227–233, 2011.

[29] K. Ma, S. Wu, Y. Wei, and W. Zhang, "Gossip-based distributed tracking in networks of heterogeneous agents," *IEEE Communications Letters*, vol. 21, no. 4, pp. 801–804, 2016.

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

[31] J. Hu, L. Xie, and C. Zhang, "Diffusion Kalman filtering based on covariance intersection," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 891–902, 2011.

[32] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion strategies for distributed Kalman filtering: Formulation and performance analysis," *Proc. Cognitive Information Processing*, pp. 36–41, 2008.

[33] A. Haber and M. Verhaegen, "Moving horizon estimation for large-scale interconnected systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2834–2847, 2013.

[34] Y. Kadowaki and H. Ishii, "Event-based distributed clock synchronization for wireless sensor networks," *IEEE Transactions on Automatic Control*, vol. 60, no. 8, pp. 2266–2271, 2014.

[35] C. Nowzari, E. Garcia, and J. Cortés, "Event-triggered communication and control of networked systems for multi-agent consensus," *Automatica*, vol. 105, pp. 1–27, 2019.

[36] L. Xing, H. Lin, C. Wang, Z. Liu, and M. Liu, "Event-triggered controller design from the control input perspective," *International Journal of Robust and Nonlinear Control*, vol. 32, no. 10, pp. 5866–5880, 2022.

[37] X. Yi, K. Liu, D. V. Dimarogonas, and K. H. Johansson, "Dynamic event-triggered and self-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3300–3307, 2018.

[38] R. K. Mishra and H. Ishii, "Dynamic event-triggered consensus control of discrete-time linear multi-agent systems," *International Journal of Robust and Nonlinear Control*, vol. 33, no. 1, pp. 159–176, 2023.

[39] K. You and L. Xie, "Network topology and communication data rate for consensusability of discrete-time multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2262–2275, 2011.

[40] Q.-C. Pham, N. Tabareau, and J.-J. Slotine, "A contraction theory approach to stochastic incremental stability," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 816–820, 2009.

[41] J. Steinhardt and R. Tedrake, "Finite-time regional verification of stochastic non-linear systems," *International Journal of Robotics Research*, vol. 31, no. 7, pp. 901–923, 2012.

[42] M. Wang and D. P. Bertsekas, "Stochastic first-order methods with random constraint projection," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 681–717, 2016.

[43] L. Ma, Z. Wang, and H.-K. Lam, "Event-triggered mean-square consensus control for time-varying stochastic multi-agent system with sensor saturations," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3524–3531, 2016.

[44] D. Ding, Z. Wang, B. Shen, and G. Wei, "Event-triggered consensus control for discrete-time stochastic multi-agent systems: The input-to-state stability in probability," *Automatica*, vol. 62, pp. 284–291, 2015.

[45] L. Cheng, Z.-G. Hou, and M. Tan, "A mean square consensus protocol for linear multi-agent systems with communication noises and fixed topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 261–267, 2013.

[46] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, 1993.

[47] Q. Liu, Z. Wang, X. He, and D. Zhou, "Event-based distributed filtering over Markovian switching topologies," *IEEE Transactions on Automatic Control*, vol. 64, no. 4, pp. 1595–1602, 2019.

[48] D. Shi, T. Chen, and M. Darouach, "Event-based state estimation of linear dynamic systems with unknown exogenous inputs," *Automatica*, vol. 69, pp. 275–288, 2016.

[49] Y. Yuan and Y. Mo, "Security in cyber-physical systems: Controller design against known-plaintext attack," in *Proceedings of the 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 5814–5819.

[50] Y. Mo, R. Ambrosino, and B. Sinopoli, "Sensor selection strategies for state estimation in energy constrained wireless sensor networks," *Automatica*, vol. 47, no. 7, pp. 1330–1338, 2011.

Jiaqi Yan received the bachelor's degree in Automation from Xi'an Jiaotong University, China, in 2016 and the Ph.D. degree from Nanyang Technological University, Singapore, in 2021. She is currently working as a postdoc research fellow at Tokyo Institute of Technology, Japan. Prior to this, she was a Research Assistant with the Tsinghua University, China, from August 2020 to July 2021, and a Visiting Scholar with the California Institute of Technology, USA, from February 2019 to August 2019. Her research interests include distributed estimation and security and resilience of cyber-physical systems. She received JSPS Postdoctoral Fellowship for Research in Japan (Standard) in 2021.

Yilin Mo is an Associate Professor in the Department of Automation, Tsinghua University. He received his Ph.D. in Electrical and Computer Engineering from Carnegie Mellon University in 2012 and his Bachelor of Engineering degree from Department of Automation, Tsinghua University in 2007. Prior to his current position, he was a postdoctoral scholar at Carnegie Mellon University in 2013 and California Institute of Technology from 2013 to 2015. He held an assistant professor position in the School of Electrical and Electronic Engineering at Nanyang Technological University from 2015 to 2018. His research interests include secure control systems and networked control systems, with applications in sensor networks and power grids.

Hideaki Ishii (M'02-SM'12-F'21) received the M.Eng. degree in applied systems science from Kyoto University, Kyoto, Japan, in 1998, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2002. He was a Postdoctoral Research Associate with the Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign, Urbana, IL, USA, from 2001 to 2004, and a Research Associate with the Department of Information Physics and Computing, The University of Tokyo, Tokyo, Japan, from 2004 to 2007. Currently, he is a Professor at the Department of Computer Science, Tokyo Institute of Technology, Yokohama, Japan. He was a Humboldt Research Fellow at the University of Stuttgart in 2014–2015. He has also held visiting positions at CNR-IEIIT at the Politecnico di Torino, the Technical University of Berlin, and the City University of Hong Kong. His research interests include networked control systems, multiagent systems, distributed algorithms, and cyber-security of control systems.

Dr. Ishii has served as an Associate Editor for the IEEE Control Systems Letters and the Mathematics of Control, Signals, and Systems and previously for Automatica, the IEEE Transactions on Automatic Control, and the IEEE Transactions on Control of Network Systems. He is the Vice President for the IEEE CSS since 2022 and has served on the IEEE CSS Board of Governors as an elected member in 2014-2016. He is the Chair of the IFAC Coordinating Committee on Systems and Signals since 2017 and was the Chair of the IFAC Technical Committee on Networked Systems for 2011–2017. He was the IPC Chair for the IFAC World Congress 2023 held in Yokohama, Japan. He received the IEEE Control Systems Magazine Outstanding Paper Award in 2015. Dr. Ishii is an IEEE Fellow.