Distributed Dynamic State Estimation and LQG Control in Resource-Constrained Networks

Yasin Yılmaz, *Member, IEEE,* Mehmet Necip Kurt, and Xiaodong Wang, *Fellow, IEEE*

***Abstract*—In this paper, the discrete-time distributed dynamic state estimation and Linear Quadratic Gaussian (LQG) control problems are analyzed for resource-constrained networked systems. Following a holistic approach we provide a complete system design for the signal processing, communication, and control tasks involved in the problems; and evaluate their performance. In the presence of a controller node and a number of sensor nodes, the sensor nodes, in a resource-efficient way, report their information entities to the controller node using an event-triggered sampling technique called level-crossing sampling. We demonstrate the performance gains due to level-crossing sampling over conventional time-triggered uniform sampling, as well as the advantages of processing data locally before transmitting to the controller. In particular, it is shown that the proposed decentralized schemes with local processing and level-crossing sampling ensure a very close approximation, with a bounded error, to the optimum (centralized) estimation and control schemes, and as a result yield order-2 asymptotic optimality. Moreover, non-ideal communication between sensors and the controller is considered, and optimal modulation techniques are provided for different channel models. Simulation results are provided to support the presented discussions.**

***Index Terms*– Networked control systems, level-crossing sampling, distributed Kalman filter, LQG control, asymptotic optimality, unreliable communications.**

I. Introduction

With the interplay between signal processing, communications, and control, and the recent advancements in these fields, it is now possible to control dynamic systems remotely via networked control systems (NCS) [1], [2]. In NCS, sensors and controllers are geographically separated and there is a communication system between them to exchange information. They are widely used due to ease of installation, ease of maintenance, system flexibility, low complexity, and low costs [1], [2]. However, traditional signal processing and control techniques do not directly apply to NCS due to strict communication constraints.

In resource-constrained systems, the communication rate should be reduced as much as possible while maintaining a satisfactory system performance such that resources are used in a smart way. Although conventional time-triggered systems have the advantage of ease of design and analysis based on well-established theories, they typically suffer from inefficient use of resources, such as energy and communication bandwidth. As a result, there is an increasing interest in the event-triggered signal processing and control techniques over the past years [3]. For instance, in a networked control system, when the system is at its steady state, periodic sampling and transmission between sensors and the controller wastes bandwidth, energy, and computation resources. On the other hand, in event-triggered systems, actions depend on a set of predefined events, and information transmission takes place only when such events occur, see e.g., [4], [5]. With properly defined events, event-triggered signal processing and control is preferred for networked systems [6], [7].

A main task of a networked system is estimating the system state based on collected measurements over the network. For a linear dynamic system, if the noise terms are Gaussian, the centralized Kalman filter is the optimal state estimator in minimizing the mean squared error [8]. However, if the system is located in a geographically large area, then collecting and processing measurements at a single node is infeasible due to communication constraints. Hence, distributed Kalman filtering has received significant attention in the literature, see e.g., [9] for a review. For a network in which only neighboring nodes are allowed to communicate with each other, the Kalman Consensus Filter (KCF) is proposed in [10] that combines local Kalman filters and a dynamic consensus algorithm for reducing disagreements on state estimates of the local Kalman filters. With the aim of reducing the communication rate in the KCF, event-triggered communication schemes are proposed in [11] and [12]. Furthermore, in [13], an event-based distributed Kalman filter is proposed, where each node decides on sharing its local measurements with the other nodes depending on the level of deviation of the local measurements from the predicted ones. It is also proposed in [14], [15] that at each time a sensor sends a quantized measurement innovation signal, which is the difference between the actual measurement and the predicted measurement by the Kalman filter, to all other sensors, which then update their state estimates based on the broadcasted message.

The presence of a communication system in networked systems brings some challenges such as limited data rate, quantization, and unreliable channels with data losses and communication delays, whose effects on the system performance are extensively studied, see e.g., [16]–[23]. For instance, [16] examines the linear quadratic Gaussian control problem in NCS in the presence of medium access constraints and delays. Delay compensator is used together with a medium access policy. In [17], communication channels have limited data rates. A quantization, coding and control scheme is designed

This work was supported in part by the U.S. National Science Foundation (NSF) under grants CNS-1737598, ECCS-1405327, in part by the U.S. Office of Naval Research (ONR) under grant N000141410667, and in part by the SCEEE-17-03 grant.

Y. Yılmaz is with the Electrical Engineering Department, University of South Florida, Tampa, FL 33620 (yasiny@usf.edu). M.N. Kurt and X. Wang are with the Electrical Engineering Department, Columbia University, New York, NY 10027 (m.n.kurt@columbia.edu, wangx@ee.columbia.edu).

to have the smallest possible data rate to achieve stability and desired control performance. Moreover, the tradeoff between linear quadratic cost and the data rate is shown. Authors of [18] considered distributed event-triggered control for NCS in the presence of both data losses and communication delays. Maximum allowable number of successive data dropouts and transmission delay deadlines are predicted locally. In [19], joint effect of delay and quantization on the stability of NCS is examined. The relationship between choosing the quantization parameters and communication delays is shown.

In [20], discrete-time LQG control problem is examined over lossy networks for both observation and control packets. It is argued that if packets are acknowledged at the receivers in a TCP-like network, then separation principle in LQG control holds and the optimal control input is a linear function of the estimated state provided that the arrival rates of packets are above some critical levels. However, if the packets are not acknowledged in a UDP-like network, then the separation principle does not hold and optimal control law is nonlinear in general. In [21], in a TCP-like network, acknowledgment messages are also subject to random data dropouts. In this case, the optimal control law is again nonlinear. It is also shown that with quantization, the separation principle does not hold. In [22], an unreliable channel that randomly erases transmitted data is considered and a threshold-based rule is used as the event-triggering mechanism. Control performance is measured by a linear-quadratic cost for a given event threshold level and the tradeoff between control performance and communication cost is illustrated. Cyber-security is also a concern in NCS. For instance, in [23], random packet losses due to interrupting jamming attacks are considered and conditions for stability of the system are provided.

The event-triggered paradigm is mainly used for nonuniform sampling in the signal processing applications [24]. In conventional uniform sampling, sampling times are periodic and predetermined, in general, according to the highest expected spectral frequency in the signal. Such high-frequency sampling is a waste of energy when the lower-frequency components are dominant in the signal. On the other hand, in event-triggered sampling, sampling times are dynamically determined based on the signal. Specifically, a sample is taken when a predefined event occurs in the signal (e.g., signal amplitude crosses a predetermined level). Consequently, event-triggered sampling encodes the signal in the sampling times, whereas in uniform sampling the signal is encoded in the sample amplitude. In real-time applications, the time-encoding feature of event-triggered sampling results in a significant advantage as the sampling times can be tracked using simple one-bit signaling [25].

In level-crossing sampling, sampling is triggered when the signal amplitude crosses one of the predetermined levels, which are usually uniformly spaced. When the successive crossing of the same level is ignored, the sampling mechanism is called level-crossing sampling with hysteresis [3]. In this paper, we use level-crossing sampling with hysteresis to transmit the local information entities at the sensor nodes to the controller. For simplicity, we drop the term hysteresis throughout the paper.

In this paper, we consider the discrete-time dynamic state estimation and LQG control problems in a resource-constrained networked system. Following a holistic approach we provide a complete decentralized system design for the signal processing, communication, and control tasks in a network of multiple sensor nodes and a controller. The sensor nodes take several noisy observations of the time-varying system states, and after processing observations they transmit their local information to the controller in a resource-efficient way using level-crossing sampling. We show that the proposed decentralized schemes closely approximate the optimum centralized schemes, achieving strong (i.e., order-2) asymptotic optimality under reliable communication channels. We then consider noisy communication channels between sensors and the controller. For different channel models, we derive the optimal modulation techniques. Through simulations, we illustrate the advantages of transmitting a finalized information compared to transmitting raw measurements to the controller, and also the advantages of using level-crossing sampling over conventional time-triggered sampling.

The remainder of the paper is organized as follows. The system model and dynamic state estimation are explained in Section II. Distributed state estimation for resource-constrained networked systems is examined in Section III. Non-ideal communications is considered and corresponding optimal modulation schemes are presented in Section IV. Distributed LQG control problem for resource-constrained networked systems is studied in Section V, which is followed by the numerical results in Section VI. Finally, Section VII concludes the paper. In this paper, we represent vectors and matrices with boldface small and capital letters, respectively, and all vectors are column vectors.

II. System Description

A. System Model

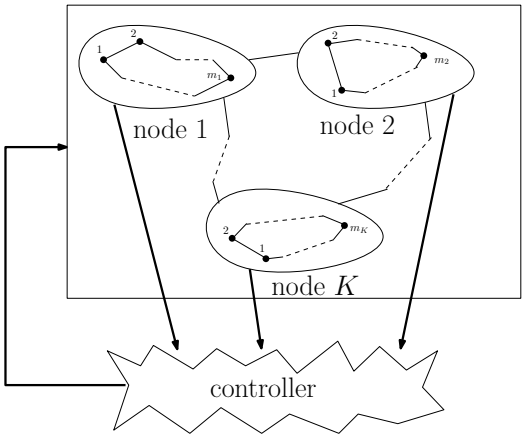

Fig. 1. The considered networked control system.

Consider a networked control system with K nodes and a controller, as shown in Fig. 1. At each time $t \in \mathbb{N}$ each node k takes m_k noisy measurements of n dynamic states, resulting in a total of $m = \sum_{k=1}^{K} m_k$ measurements systemwide. The controller, gathering some kind of local information from

nodes, applies a control input to the system. Specifically, we assume the following linear state space model

$$\begin{aligned} \boldsymbol{x}_{t+1} &= \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{v}_t \\ \boldsymbol{z}_t^k &= \boldsymbol{C}_k\boldsymbol{x}_t + \boldsymbol{w}_t^k, \end{aligned} \tag{1}$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$ is the state vector; $\boldsymbol{A}$ is the state matrix; $\boldsymbol{u}_t \in \mathbb{R}^p$ is the input, i.e., control vector; $\boldsymbol{B}$ is the input matrix; $\boldsymbol{v}_t$ is the additive white Gaussian system noise with the covariance matrix $\boldsymbol{R}_1$; and $\boldsymbol{z}_t^k \in \mathbb{R}^{m_k}$, $\boldsymbol{C}_k$, $\boldsymbol{w}_t^k$ are the observation vector, observation matrix, and the additive white Gaussian noise with the covariance matrix $\boldsymbol{R}_{2,k}$, respectively, at node k. We further assume that the noise vectors $\boldsymbol{v}_t$ and $\boldsymbol{w}_t^k$ are independent, hence we have $\mathsf{E}\left[\boldsymbol{v}_t\boldsymbol{v}_{t'}^T\right] = \delta(t-t')\boldsymbol{R}_1$, $\mathsf{E}\left[\boldsymbol{w}_t^k(\boldsymbol{w}_{t'}^k)^T\right] = \delta(t-t')\boldsymbol{R}_{2,k}$, and $\mathsf{E}\left[\boldsymbol{w}_t^k\boldsymbol{v}_{t'}^T\right] = \boldsymbol{0},\ \forall k,t,t'$, where $\mathsf{E}[\cdot]$ is the expectation operator, and $\delta(t)$ is the Dirac delta function at $t=0$.

We assume $\boldsymbol{A}$, $\boldsymbol{B}$ and $\{\boldsymbol{C}_k\}$ are given and satisfy the observability and controllability conditions for our discrete-time linear time-invariant system. In Section III, the input $\boldsymbol{u}_t$ is assumed to be given, and in Section V, it is designed for LQG control. Instead of transmitting the raw observations $\boldsymbol{z}_t^k$ from nodes to the controller, leveraging the information form of Kalman filter we propose to transmit local contributions to the optimum estimator $\hat{\boldsymbol{x}}_t$ (Section III) and the optimum control input $\boldsymbol{u}_t$ (Section V). Since $\boldsymbol{x}_t$ and $\boldsymbol{u}_t$ typically have much smaller dynamic ranges and number of dimensions than $\boldsymbol{z}_t^k$ (see Fig. 8), the proposed method turns out to be quite advantageous over the traditional methods (see Table I). We also show through simulations that the control input produced by the proposed method is closer to the optimum value than that produced by transmitting $\boldsymbol{z}_t^k$ (see Fig. 7).

B. Information Filter

We have a discrete-time linear dynamic system as described in (1). Since the noise terms are Gaussian, Kalman filter is the optimal state estimator [8]. The information filter, or inverse covariance filter, is algebraically equivalent to the Kalman filter. Instead of the estimates of the state covariance matrix and the state vector, it uses the following information matrix and information vector, respectively,

$$\boldsymbol{Y}_{t|t'} \triangleq \boldsymbol{P}_{t|t'}^{-1} \tag{2}$$

$$\boldsymbol{y}_{t|t'} \triangleq \boldsymbol{P}_{t|t'}^{-1}\hat{\boldsymbol{x}}_{t|t'} = \boldsymbol{Y}_{t|t'}\hat{\boldsymbol{x}}_{t|t'}, \tag{3}$$

where $\boldsymbol{P}_{t|t'}$ and $\hat{\boldsymbol{x}}_{t|t'}$ are the predicted (for $t' = t-1$) and updated (for $t' = t$) covariance matrix and state estimate in the Kalman filter, respectively. The information matrix and vector are predicted as

$$\boldsymbol{Y}_{t|t-1} = (\boldsymbol{I} - \boldsymbol{F}_{t-1})\boldsymbol{G}_{t-1}, \tag{4}$$

$$\boldsymbol{y}_{t|t-1} = (\boldsymbol{I} - \boldsymbol{F}_{t-1})\boldsymbol{A}^{-T}\boldsymbol{y}_{t-1|t-1} + \boldsymbol{\xi}_{t-1}, \tag{5}$$

and updated as

$$\boldsymbol{Y}_{t|t} = \boldsymbol{Y}_{t|t-1} + \sum_{k=1}^{K} \boldsymbol{C}_k^T\boldsymbol{R}_{2,k}^{-1}\boldsymbol{C}_k, \tag{6}$$

$$\boldsymbol{y}_{t|t} = \boldsymbol{y}_{t|t-1} + \sum_{k=1}^{K} \boldsymbol{C}_k^T\boldsymbol{R}_{2,k}^{-1}\boldsymbol{z}_t^k, \tag{7}$$

$$\text{where}\quad \boldsymbol{G}_t \triangleq \boldsymbol{A}^{-T}\boldsymbol{Y}_{t|t}\boldsymbol{A}^{-1}, \tag{8}$$

$$\boldsymbol{F}_t \triangleq \boldsymbol{G}_t\left(\boldsymbol{G}_t + \boldsymbol{R}_1^{-1}\right)^{-1}, \tag{9}$$

$$\boldsymbol{\xi}_{t-1} \triangleq \boldsymbol{Y}_{t|t-1}\boldsymbol{B}\boldsymbol{u}_{t-1}, \tag{10}$$

and $\boldsymbol{I}$ is the identity matrix [26]. Denote the information contributions from node k with

$$\boldsymbol{\Phi}_k \triangleq \boldsymbol{C}_k^T\boldsymbol{R}_{2,k}^{-1}\boldsymbol{C}_k, \tag{11}$$

$$\text{and}\quad \boldsymbol{\phi}_t^k \triangleq \boldsymbol{C}_k^T\boldsymbol{R}_{2,k}^{-1}\boldsymbol{z}_t^k. \tag{12}$$

Then, from (6) and (7) we can write $\boldsymbol{Y}_{t|t} = \boldsymbol{Y}_{t|t-1} + \boldsymbol{\Phi}$ and $\boldsymbol{y}_{t|t} = \boldsymbol{y}_{t|t-1} + \boldsymbol{\phi}_t$, where $\boldsymbol{\Phi} \triangleq \sum_{k=1}^{K}\boldsymbol{\Phi}_k$ and $\boldsymbol{\phi}_t \triangleq \sum_{k=1}^{K}\boldsymbol{\phi}_t^k$ are the systemwide information contributions.

In this study, for state estimation, we prefer to use the information filter over the Kalman filter for several reasons. Firstly, the information filter is more convenient for a distributed setting due to its simple update rules, given in (6) and (7). In particular, in the classical distributed implementation of the information filter, each node k reports its new information contribution $\boldsymbol{\phi}_t^k$ at each time t to the controller, which sums them to update the information vector $\boldsymbol{y}_{t|t-1}$. On the other hand, in the Kalman filter each node k needs to report its raw observation $\boldsymbol{z}_t^k$ at each time t, which is processed at the controller before the update, hence causes propagation of quantization/communication errors. Secondly, at each time t, the inverse of an $n \times n$ matrix is computed to obtain $\boldsymbol{F}_t$ in the information filter, whereas in the Kalman filter an $m \times m$ matrix is inverted, where usually $m \gg n$. However, we note that the structure of the information filter requires the matrices $\boldsymbol{A}$, $\boldsymbol{R}_1$, and $\{\boldsymbol{R}_{2,k}\}$ to be invertible.

III. Distributed State Estimation For Resource-Constrained Systems

In resource-constrained systems with strict energy and bandwidth limitations, e.g., wireless sensor networks, nodes cannot exactly report their information contributions. Specifically, each node k needs to sample and quantize its information entity before transmitting it to the controller. Since such a sampling and quantization process induces information loss, which grows with further processing at the controller, we intend to perform the necessary information processing as locally as possible before transmitting the information. To this end, defining the shorthand subscript $t \triangleq t|t$ we present the following lemma.

Lemma 1. *The systemwide optimum state estimate $\hat{\boldsymbol{x}}_t$ can be written as the sum of local contributions and a term related to the previous-time control inputs, i.e.,*

$$\hat{\boldsymbol{x}}_t = \boldsymbol{\varphi}_t + \sum_{k=1}^{K}\hat{\boldsymbol{x}}_t^k, \tag{13}$$

where $\boldsymbol{\varphi}_t$ is the input related term, defined in (23) *and* (18)*, and $\hat{\boldsymbol{x}}_t^k$ is the local contribution from node k to $\hat{\boldsymbol{x}}_t$.*

Proof: From (5) and (7), we write

$$\boldsymbol{y}_t = \boldsymbol{\xi}_{t-1} + \boldsymbol{\Omega}_{t-1}\boldsymbol{y}_{t-1} + \sum_{k=1}^{K} \boldsymbol{\phi}_t^k, \tag{14}$$

$$\text{where} \quad \boldsymbol{\Omega}_{t-1} \triangleq (\boldsymbol{I} - \boldsymbol{F}_{t-1})\boldsymbol{A}^{-T}. \tag{15}$$

Note that due to (10), $\boldsymbol{\xi}_{t-1}$ is related to $\boldsymbol{u}_{t-1}$. We split $\boldsymbol{y}_t$ into two parts as

$$\boldsymbol{y}_t = \boldsymbol{\lambda}_{t-1} + \boldsymbol{\zeta}_t \tag{16}$$

where $\boldsymbol{\lambda}_{t-1}$ is equal to the sum of all terms related to the previous-time control inputs and $\boldsymbol{\zeta}_t$ is the contribution from information entities. We split $\boldsymbol{y}_{t-1}$ as in (16), and rewrite (14) as follows:

$$\begin{aligned}\boldsymbol{y}_t &= \boldsymbol{\xi}_{t-1} + \boldsymbol{\Omega}_{t-1}(\boldsymbol{\lambda}_{t-2} + \boldsymbol{\zeta}_{t-1}) + \sum_{k=1}^{K} \boldsymbol{\phi}_t^k \\ &= \underbrace{\boldsymbol{\xi}_{t-1} + \boldsymbol{\Omega}_{t-1}\boldsymbol{\lambda}_{t-2}}_{\boldsymbol{\lambda}_{t-1}} + \underbrace{\boldsymbol{\Omega}_{t-1}\boldsymbol{\zeta}_{t-1} + \sum_{k=1}^{K} \boldsymbol{\phi}_t^k}_{\boldsymbol{\zeta}_t}. \end{aligned} \tag{17}$$

Based on (17), we write the evolutions of the terms $\boldsymbol{\lambda}_t$ and $\boldsymbol{\zeta}_t$ in time as follows:

$$\boldsymbol{\lambda}_t = \boldsymbol{\Omega}_t\boldsymbol{\lambda}_{t-1} + \boldsymbol{\xi}_t, \tag{18}$$

$$\boldsymbol{\zeta}_t = \boldsymbol{\Omega}_{t-1}\boldsymbol{\zeta}_{t-1} + \sum_{k=1}^{K} \boldsymbol{\phi}_t^k, \tag{19}$$

where $\boldsymbol{\lambda}_{-1} = \boldsymbol{0}$ and $\boldsymbol{\zeta}_0 = \boldsymbol{0}$ are the initial values of these terms, respectively. Furthermore, from (19) we have $\boldsymbol{\zeta}_1 = \sum_{k=1}^{K} \boldsymbol{\phi}_1^k$, $\boldsymbol{\zeta}_2 = \boldsymbol{\Omega}_1 \sum_{k=1}^{K} \boldsymbol{\phi}_1^k + \sum_{k=1}^{K} \boldsymbol{\phi}_2^k$, and so on. Defining $\boldsymbol{\Psi}_s^t \triangleq \boldsymbol{\Omega}_t\boldsymbol{\Omega}_{t-1}\cdots\boldsymbol{\Omega}_s$ we write $\boldsymbol{\zeta}_t$ as

$$\begin{aligned}\boldsymbol{\zeta}_t &= \sum_{s=1}^{t} \boldsymbol{\Psi}_s^{t-1} \sum_{k=1}^{K} \boldsymbol{\phi}_s^k, \\ &= \sum_{k=1}^{K} \sum_{s=1}^{t} \boldsymbol{\Psi}_s^{t-1} \boldsymbol{\phi}_s^k, \\ &= \sum_{k=1}^{K} \boldsymbol{y}_t^k, \end{aligned} \tag{20}$$

where $\boldsymbol{\Psi}_t^{t-1} = \boldsymbol{I}$ and $\boldsymbol{y}_t^k \triangleq \sum_{s=1}^{t} \boldsymbol{\Psi}_s^{t-1}\boldsymbol{\phi}_s^k$. Note that $\boldsymbol{\phi}_s^k = \boldsymbol{C}_k^T \boldsymbol{R}_{2,k}^{-1} \boldsymbol{z}_s^k$ is the local information of node k at time s, and $\boldsymbol{\Psi}_s^{t-1} = \boldsymbol{\Omega}_{t-1}\boldsymbol{\Omega}_{t-2}\cdots\boldsymbol{\Omega}_s$ is a function of $\boldsymbol{A}$ and $\{\boldsymbol{F}_r\}_{r=s}^{t-1}$, where $\boldsymbol{F}_r$, given by (9), is a function of $\boldsymbol{A}$, $\boldsymbol{R}_1$, and $\boldsymbol{Y}_r$. From (6), $\boldsymbol{Y}_r$ is computed using $\{\boldsymbol{C}_k\}$ and $\boldsymbol{R}_{2,k}$. Since $\boldsymbol{A}$, $\boldsymbol{R}_1$, $\boldsymbol{R}_{2,k}$, and $\{\boldsymbol{C}_k\}$ are known and time-independent, each node k can compute its local information entity $\boldsymbol{y}_t^k$ at each time t. Moreover, similar to (19), it can recursively update $\boldsymbol{y}_t^k$ as

$$\boldsymbol{y}_t^k = \boldsymbol{\Omega}_{t-1}\boldsymbol{y}_{t-1}^k + \boldsymbol{\phi}_t^k. \tag{21}$$

The terms related to the previous inputs, i.e., $\{\boldsymbol{\lambda}_t\}$ can be computed and updated by the controller node using (18). Hence, there is no need to transmit these information entities from the nodes. Then, each node k can locally process $\boldsymbol{\phi}_t^k$, given by (12), and transmit $\boldsymbol{y}_t^k$ instead of transmitting $\boldsymbol{\phi}_t^k$ as in the standard distributed implementation of the information filter. In fact, to minimize the information loss at the controller, we can further process $\boldsymbol{y}_t^k$ locally and transmit a finalized information entity. Specifically, from (3), (16), and (20), the global state estimate at time t is given by

$$\begin{aligned}\hat{\boldsymbol{x}}_t &= \boldsymbol{Y}_t^{-1}\left(\boldsymbol{\lambda}_{t-1} + \sum_{k=1}^{K} \boldsymbol{y}_t^k\right) \\ &= \boldsymbol{\varphi}_t + \sum_{k=1}^{K} \hat{\boldsymbol{x}}_t^k, \end{aligned} \tag{22}$$

$$\text{where} \quad \boldsymbol{\varphi}_t \triangleq \boldsymbol{Y}_t^{-1}\boldsymbol{\lambda}_{t-1}, \tag{23}$$

and $\hat{\boldsymbol{x}}_t^k \triangleq \boldsymbol{Y}_t^{-1}\boldsymbol{y}_t^k$. Note that the matrix $\boldsymbol{Y}_t$ is available to all nodes and from (4) and (6), it can be recursively computed as

$$\boldsymbol{Y}_t = \boldsymbol{\Omega}_{t-1}\boldsymbol{Y}_{t-1}\boldsymbol{A}^{-1} + \sum_{k=1}^{K} \boldsymbol{\Phi}_k. \tag{24}$$

■

A. Information Transmission using Level-Crossing Sampling

As a result of Lemma 2, we propose that the nodes transmit $\{\hat{\boldsymbol{x}}_t^k\}_{k=1}^K$ to the controller, which sums them and the term related to the previous-time inputs to obtain the optimal state estimate $\hat{\boldsymbol{x}}_t$. An *event-triggered* sampling technique, called the *level-crossing sampling*, can be used to accurately report $\hat{\boldsymbol{x}}_t^k$ in an energy- and bandwidth-efficient way. The level-crossing sampling procedure is quite simple: Firstly, a set of signal levels is selected to trigger sampling. Here we use a set with uniform spacing Δ, as shown in Fig. 2. As the signal is observed sequentially, a new sample is taken when the signal crosses a sampling level that is different than the most recently crossed one.

This method is, in fact, known as level-crossing sampling with hysteresis in the literature. In the original level-crossing sampling procedure, a sample is taken every time a sampling level is crossed, even when the same level is crossed consecutively, which makes the procedure even simpler. However, this method does not serve our purpose of reporting the changes in the local information entity (see Fig. 2). For simplicity, here we drop the term hysteresis.

In this *adaptive* sampling scheme, the sampling times are dynamically determined by the signal, hence random, as opposed to the traditional time-triggered uniform sampling, in which samples are periodically taken in time. In the traditional sampling, the time axis is partitioned uniformly, whereas in the level-crossing sampling, we partition the magnitude axis uniformly, time being the dependent variable.

In particular, node k samples the ith entry $\hat{x}_t^{k,i}$ of $\hat{\boldsymbol{x}}_t^k$ at the random sampling times $\{\tau_q^{k,i}\}_q$ given by

$$\tau_q^{k,i} \triangleq \min\left\{t \in \mathbb{N} : |\hat{x}_t^{k,i} - \gamma_{q-1}^{k,i}\Delta| \geq \Delta\right\}, \tag{25}$$

where $\gamma_{q-1}^{k,i} \in \mathbb{N}$ is the sampling level in terms of Δ that was most recently crossed. In other words, each node k runs n

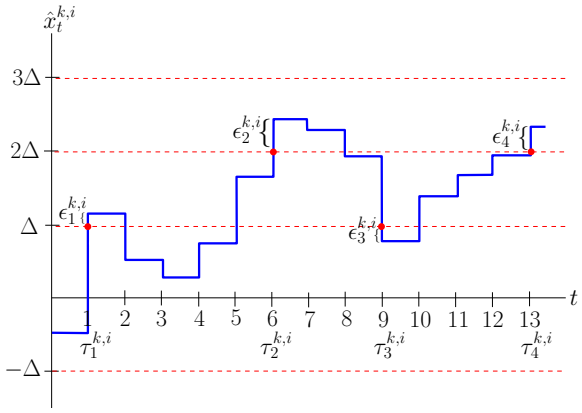

Fig. 2. The level-crossing sampling procedure. A new sample is taken when a sampling level that is different than the one at the last sampling time is crossed. At each sampling time $\tau_q^{k,i}$, a discrete-time signal over/under shoots the sampling level by $\epsilon_q^{k,i}$.

parallel level-crossing samplers for $\{\hat{x}_t^{k,i}\}_{i=1}^n$. Note that the threshold Δ in (25) determines the sampling rate. The smaller it is, the more frequently samples are taken in a nonuniform fashion. A small Δ may also cause $\hat{x}_t^{k,i}$ to cross multiple sampling levels at once, i.e., at time $\tau_q^{k,i}$ the number of crossed sampling levels is given by

$$\theta_q^{k,i} \triangleq \left\lfloor \frac{|\hat{x}_q^{k,i} - \gamma_{q-1}^{k,i}\Delta|}{\Delta} \right\rfloor \geq 1. \tag{26}$$

Then, at each sampling time $\tau_q^{k,i}$ node k transmits $\theta_q^{k,i}$ repetitions of the sign bit

$$\delta_q^{k,i} \triangleq \text{sign}(\hat{x}_q^{k,i} - \gamma_{q-1}^{k,i}\Delta) = \text{sign}(\gamma_q^{k,i} - \gamma_{q-1}^{k,i}), \tag{27}$$

indicating the current sampling level

$$\gamma_q^{k,i} = \gamma_{q-1}^{k,i} + \delta_q^{k,i}\theta_q^{k,i}. \tag{28}$$

Here we use the shorthand notation $\hat{x}_q^{k,i}$ for the signal level, and $\gamma_q^{k,i}$ for the sampling level (in terms of Δ) at time $\tau_q^{k,i}$.

The controller, upon receiving $\delta_q^{k,i}$ (and possible repetitions) at time $\tau_q^{k,i}$, updates its estimate for $\hat{x}_t^{k,i}$ by $\delta_q^{k,i}\theta_q^{k,i}\Delta$, i.e.,

$$\tilde{\hat{x}}_q^{k,i} = \tilde{\hat{x}}_{q-1}^{k,i} + \delta_q^{k,i}\theta_q^{k,i}\Delta, \quad \tilde{\hat{x}}_0^{k,i} = 0, \tag{29}$$

where $\tilde{\hat{x}}_{q-1}^{k,i}$ is the estimate of $\hat{x}_t^{k,i}$ at the controller node during the time interval $\tau_{q-1}^{k,i} \leq t < \tau_q^{k,i}$. Since $\hat{\boldsymbol{x}}_t = \boldsymbol{\varphi}_t + \sum_{k=1}^K \hat{\boldsymbol{x}}_t^k$, as stated in (13), in fact, the controller updates its estimates for each entry of $\hat{\boldsymbol{x}}_t$ regardless of which of the nodes transmitting the information bits. We define $\tilde{\hat{\boldsymbol{x}}}_q = [\tilde{\hat{x}}_q^1, \tilde{\hat{x}}_q^2, \ldots, \tilde{\hat{x}}_q^n] \triangleq \sum_{k=1}^K \tilde{\hat{\boldsymbol{x}}}_q^k$, where $\tilde{\hat{x}}_q^{k,i}$ is the ith entry of $\tilde{\hat{\boldsymbol{x}}}_q^k$. The controller node, upon receiving the qth (in the global order) bit δ_q^i from node k_q, it performs the following update

$$\tilde{\hat{x}}_q^i = \tilde{\hat{x}}_{q-1}^i + \delta_q^i\Delta, \quad \tilde{\hat{x}}_0^i = 0, \tag{30}$$

and uses $\tilde{\hat{\boldsymbol{x}}}_q$ as the estimate of $\sum_{k=1}^K \hat{\boldsymbol{x}}_t^k$ until the next received bit. If multiple bits arrive at the same time, then it processes them in a random order. The proposed procedures at node k and the controller are summarized in Algorithms 1 and 2, respectively.

Algorithm 1 The proposed procedure at node k

1: Compute $\{\boldsymbol{Y}_t\}_{t\geq 0}$ and $\{\boldsymbol{\Omega}_t\}_{t\geq 0}$ as in Fig. 3
2: Initialization: $t \leftarrow 0$, $\boldsymbol{y} \leftarrow 0$, $\gamma_i \leftarrow 0$, $i = 1, \ldots, n$
3: **while** $t \geq 0$ **do**
4: $\quad t \leftarrow t + 1$
5: $\quad \boldsymbol{y} \leftarrow \boldsymbol{\Omega}_{t-1}\boldsymbol{y} + \boldsymbol{\phi}_t$
6: $\quad \hat{\boldsymbol{x}} \leftarrow \boldsymbol{Y}_t^{-1}\boldsymbol{y}$
7: $\quad$ **if** $|\hat{x}_i - \gamma_i\Delta| \geq \Delta$, $i = 1, \ldots, n$ **then**
8: $\quad\quad \theta_i \leftarrow \left\lfloor \frac{|\hat{x}_i - \gamma_i\Delta|}{\Delta} \right\rfloor$
9: $\quad\quad$ Transmit $\delta_i = \text{sign}(\hat{x}_i - \gamma_i\Delta)$ to the controller (θ_i times)
10: $\quad\quad \gamma_i \leftarrow \gamma_i + \delta_i\theta_i$
11: $\quad$ **end if**
12: **end while**

Algorithm 2 The proposed procedure at the controller

1: Compute $\{\boldsymbol{Y}_t\}_{t\geq 0}$ and $\{\boldsymbol{\Omega}_t\}_{t\geq 0}$ as in Fig. 3
2: Compute $\{\boldsymbol{Y}_{t|t-1}\}_{t\geq 1}$ as in (4)
3: Initialization: $t \leftarrow 0$, $\tilde{\hat{x}}_q^i \leftarrow 0$, $i = 1, \ldots, n$, $\boldsymbol{\lambda}_{-1} \leftarrow \boldsymbol{0}$
4: **while** $t \geq 0$ **do**
5: $\quad t \leftarrow t + 1$
6: $\quad \boldsymbol{\xi}_{t-1} \leftarrow \boldsymbol{Y}_{t|t-1}\boldsymbol{B}\boldsymbol{u}_{t-1}$
7: $\quad \boldsymbol{\lambda}_{t-1} \leftarrow \boldsymbol{\xi}_{t-1} + \boldsymbol{\Omega}_{t-1}\boldsymbol{\lambda}_{t-2}$
8: $\quad \boldsymbol{\varphi}_t \leftarrow \boldsymbol{Y}_t^{-1}\boldsymbol{\lambda}_{t-1}$
9: $\quad$ **if** δ_q^i arrives, $i = 1, \ldots, n$ **then**
10: $\quad\quad \tilde{\hat{x}}_q^i \leftarrow \tilde{\hat{x}}_{q-1}^i + \delta_q^i\Delta$
11: $\quad$ **end if**
12: $\quad \tilde{\hat{\boldsymbol{x}}}_t \leftarrow \boldsymbol{\varphi}_t + \tilde{\hat{\boldsymbol{x}}}_q$
13: **end while**

Note that lines 7-11 in Algorithm 1, and 9-11 in Algorithm 2 are processed for each entry i in parallel. In Algorithm 1 and Algorithm 2, the matrices $\boldsymbol{Y}_t$ and $\boldsymbol{\Omega}_t$ are iteratively computed offline following the flow chart in Fig. 3.

B. Performance Analysis

The optimum state estimate $\hat{\boldsymbol{x}}_t$, is achievable only under a centralized setup, in which the controller has access to all observations systemwide. In a distributed system with resource constraints, a decentralized state estimate $\tilde{\hat{\boldsymbol{x}}}_t$ inevitably incurs a nonzero performance gap. Given a specific decentralized scheme, the lower and upper bounds on the difference $\hat{x}_t^i - \tilde{\hat{x}}_t^i$ define the 100% confidence interval for each entry $\hat{x}_t^i$. We here analyze both the non-asymptotic and asymptotic behaviors of this maximum-level, i.e., deterministic confidence interval, as a measure of performance/reliability for the proposed decentralized scheme based on level-crossing sampling. Specifically, we show that this interval is easily controllable through parameter selection, and remains bounded for all $\hat{x}_t^i$, i.e., $\hat{x}_t^i - \tilde{\hat{x}}_t^i = O(1)$[1] even if $|\hat{x}_t^i| \to \infty$, yielding a strong type of asymptotic optimality called *order-2*.

[1] $O(\cdot)$ is the big-O notation and $O(1)$ denotes a constant.

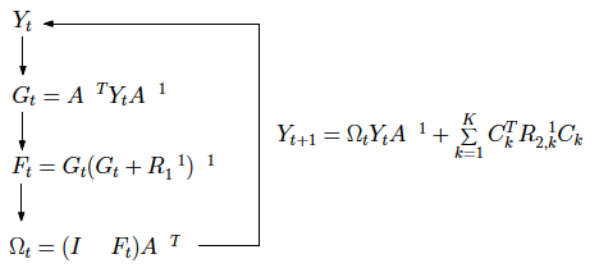

Fig. 3. Flow chart for iteratively computing $\boldsymbol{Y}_t$, $\boldsymbol{G}_t$, $\boldsymbol{F}_t$, and $\boldsymbol{\Omega}_t$. See (4)–(8) and (15) for the mathematical relations.

This strong type of asymptotic optimality provides a higher performance standard than the conventional *order-1* asymptotic optimality, which necessitates the convergence of $\tilde{\hat{x}}_t^i$ to $\hat{x}_t^i$, i.e., $\frac{\tilde{\hat{x}}_t^i}{\hat{x}_t^i} = 1 + o(1)$[2] as $|\hat{x}_t^i| \to \infty$. Note that in order-1 asymptotic optimality, the deterministic confidence interval may become unbounded at a lower rate than $\hat{x}_t^i$, that is, $\hat{x}_t^i - \tilde{\hat{x}}_t^i = o(\hat{x}_t^i)$ as $|\hat{x}_t^i| \to \infty$. Order-2 asymptotic optimality is used as a benchmark for high performance in resource-constrained distributed systems, e.g., [25], [27]. In the following theorem, we show that the proposed decentralized scheme achieves this high-performance benchmark.

Theorem 1. *In the decentralized state estimation scheme based on level-crossing sampling, given in Algorithms 1 and 2, each state estimate $\tilde{\hat{x}}_t^i$ is guaranteed to lie within $K\Delta$ bound of the optimum state estimate $\hat{x}_t^i$, i.e.,*

$$|\hat{x}_t^i - \tilde{\hat{x}}_t^i| < K\Delta, \ \forall i, t, \tag{31}$$

where K is the number of nodes, Δ is the level-crossing sampling threshold, $\hat{x}_t^i$ and $\tilde{\hat{x}}_t^i$ are the ith entries of state estimate vectors at time t in the optimum scheme and the proposed scheme, respectively. As a result, it is order-2 asymptotically optimum, i.e., $\hat{x}_t^i - \tilde{\hat{x}}_t^i = O(1)$ even if $|\hat{x}_t^i| \to \infty$.

Proof: Due to the sampling rule in (25) the current signal level $\hat{x}_t^{k,i}$ at each node k is within Δ bound of the current sampling level $\gamma_q^{k,i}\Delta$, that was most recently crossed. The controller keeps track of the current sampling level $\gamma_q^{k,i}$, given by (28), for each local signal $\hat{x}_t^{k,i}$ as it is informed of each sampling level crossing through $\delta_q^{k,i}$, given by (27). It estimates $\hat{x}_t^{k,i}$ with $\tilde{\hat{x}}_t^{k,i} = \gamma_q^{k,i}\Delta$, as shown in (29), hence $|\hat{x}_t^{k,i} - \tilde{\hat{x}}_t^{k,i}| < \Delta, \forall k, i, t$. Let the ith entry of the vector $\boldsymbol{\varphi}_t$ be denoted with φ_t^i. Due to (13), $\hat{x}_t^i = \varphi_t^i + \sum_{k=1}^K \hat{x}_t^{k,i}$ and $\tilde{\hat{x}}_t^i = \varphi_t^i + \sum_{k=1}^K \tilde{\hat{x}}_t^{k,i}$. Hence, we have

$$|\hat{x}_t^i - \tilde{\hat{x}}_t^i| < \sum_{k=1}^{K} |\hat{x}_t^{k,i} - \tilde{\hat{x}}_t^{k,i}| < K\Delta, \quad \forall i, t,$$

proving (31). Then, the asymptotic optimality result follows for fixed K and Δ. ∎

Theorem 1 presents a strong result (i.e., the confidence bound $K\Delta$) that holds for any $\hat{x}_t^i$ value (small/large, non-asymptotic/asymptotic), constituting an important advantage over the conventional scheme based on conventional time-triggered sampling. In particular, in the scheme which uniformly samples and quantizes $\hat{x}_t^{k,i}$, the confidence interval increases with range of $\hat{x}_t^{k,i}$, that is, the larger values $\hat{x}_t^{k,i}$ gets, the larger the confidence interval is. Furthermore, in such a scheme with a small number of quantization bits, the confidence interval is, in general, much larger than $K\Delta$.

[2] $o(\cdot)$ is the little-o notation and $o(1)$ denotes a diminishing value.

IV. Non-Ideal Communications

Existing works assume there exists a communication system (a pair of transmitter and receiver for some channel model), and analyze the effects of noisy communications in discrete-time. Similarly, we start our discussion with a discrete-time channel model with error probability p, e.g., binary erasure channel (BEC) and binary symmetric channel (BSC). In such a model, a transmitted bit $\delta_q^{k,i}$ is received as $\tilde{\delta}_q^{k,i}$ at the controller, where $\tilde{\delta}_q^{k,i} = \delta_q^{k,i}$ with probability $1-p$ and $\tilde{\delta}_q^{k,i} \neq \delta_q^{k,i}$ (i.e., $\tilde{\delta}_q^{k,i} = 0$ for BEC and $\tilde{\delta}_q^{k,i} = -\delta_q^{k,i}$ for BSC) with probability p. When a bit error occurs (i.e., a bit is lost or flipped), the controller performs an erroneous update $\tilde{\delta}_q^{k,i}\Delta$ with $\tilde{\delta}_q^{k,i} \neq \delta_q^{k,i}$.

If such errors happen infinitely often (i.o.), the discrepancy $|\hat{x}_t^i - \tilde{\hat{x}}_t^i|$ may grow unboundedly if the system is unstable, i.e., $|\hat{x}_t^i| \to \infty$. If the system tends to be unstable in time for some reason, it can be stopped and restarted to let the system states return to initial values, e.g., [29]. However, for a stable system, as long as bit error happens at the same rate p for both $\delta_q^{k,i} = 1$ and $\delta_q^{k,i} = -1$, the stability of the system is preserved as $t \to \infty$, as shown next.

Theorem 2. *In a practical system with resource constraints and a nonzero channel error probability p, the deterministic confidence interval η for the proposed LCS-based state estimate $\tilde{\hat{x}}_t^i$ (i.e., $|\hat{x}_t^i - \tilde{\hat{x}}_t^i| < \eta$) is unbounded, i.e., $\eta \to \infty$, as $|\hat{x}_t^i| \to \infty$. Recall from Theorem 1 that for $p = 0$, $\eta = K\Delta, \forall t$. Hence, as opposed to Theorem 1, for $p > 0$, order-2 asymptotic optimality does not hold. However, for a stable system where $|\hat{x}_t^i| < \infty$ as $t \to \infty$, our estimate is also stable, i.e., $|\tilde{\hat{x}}_t^i| < \infty$, for any $p > 0$.*

Proof: From the Borel-Cantelli lemma,

$$\mathrm{P}(\text{bit error occurs i.o.}) = 0 \quad \text{if} \quad \sum_{t=1}^{\infty} p < \infty,$$

which does not hold unless $p \to 0$ at a rate at least as fast as $1/t$, that is, $p = O(1/t)$. In the channel coding theorem, $p \to 0$ only if the block length goes to infinity, i.e., the number of bits $\theta_q^{k,i} \to \infty$. Obviously, this is not the case here as the local signal $\hat{x}_t^{k,i}$ has only finite jumps. Moreover, due to the resource constraints, we cannot transmit a large number of bits per sample. As a result, $\mathrm{P}(\text{bit error occurs i.o.}) > 0$. If $|\hat{x}_t^i| \to \infty$, that means either the number of Δ or $-\Delta$ changes (i.e., the number of $\delta_q^{k,i} = 1$ or $\delta_q^{k,i} = -1$) goes to infinity faster than the other. In this case one of the bit types is lost infinitely more often than the other, and we cannot deterministically bound the error $|\hat{x}_t^i - \tilde{\hat{x}}_t^i|$.

For a stable system where $|\hat{x}_t^i| < \infty \ \forall t, i$, the number of Δ and $-\Delta$ changes will go to infinity as $t \to \infty$ at the same rate

such that the state estimate remains bounded. Hence, even if bit error occurs infinitely often, errors for $\delta_q^{k,i}=1$ and $\delta_q^{k,i}=-1$ will occur randomly at the same rate p, canceling each other, so the error will not accumulate to infinity. ■

In theory, the order-1 asymptotic optimality is possible, as shown next.

Theorem 3. *If $\hat{x}_t^i \to \infty$ at a rate faster than t, that is, $\hat{x}_t^i = \omega(t)$, the proposed LCS-based scheme achieves order-1 asymptotic optimality, i.e., $\frac{\tilde{x}_t^i}{\hat{x}_t^i} = 1 + o(1)$ as $|\hat{x}_t^i| \to \infty$.*

Proof: This is because $\frac{\tilde{x}_t^i}{\hat{x}_t^i} = 1 + \frac{\tilde{x}_t^i - \hat{x}_t^i}{\hat{x}_t^i}$, and from Theorem 2, $\tilde{x}_t^i - \hat{x}_t^i$ may become unbounded as fast as t, i.e., $\tilde{x}_t^i - \hat{x}_t^i = O(t)$, since there may occur an error of $2\Delta \sum_{k=1}^K \theta_q^{k,i} < \infty$ at each time t. Hence, $\frac{\tilde{x}_t^i}{\hat{x}_t^i} = 1 + o(1)$ if $\hat{x}_t^i = \omega(t)$. ■

However, such an extreme case, where $\hat{x}_t^i \to \infty$ at a rate faster than t, is not of practical interest. In this section we consider improving the non-asymptotic performance by designing practical (continuous-time) communication systems. We first consider the simplest case in which each node k reports each ith entry $\hat{x}_t^{k,i}$ of its local vector $\hat{\boldsymbol{x}}_t^k$ to the controller through a separate channel that is orthogonal to the other channels in the network, resulting in Kn parallel channels in total. In this model, we identify the optimum modulation techniques under additive white Gaussian noise (AWGN) and fading channels. Then, we propose more bandwidth-efficient models via multiple-access channels.

A. Parallel Channels for Nodes

1) AWGN Channels: Suppose each node k sends the waveforms $a\ s(t)$ and $b\ s(t)$ for $\delta_q^{k,i}=1$ and $\delta_q^{k,i}=-1$, respectively, through an AWGN channel. The controller, applying a matched filter and sampling uniformly, receives the following discrete-time signal

$$r_t^{k,i} = c_t^{k,i} + n_t^{k,i}, \tag{32}$$

where $c_t^{k,i}$ is either a or b, and $n_t^{k,i}$ is the (zero-mean) white Gaussian noise with variance σ^2. Hence, $r_t^{k,i} \sim \mathcal{N}(c_t^{k,i}, \sigma^2)$, that is, the mean is a, b, and 0 for $\delta_q^{k,i}=1$, $\delta_q^{k,i}=-1$, and no transmission, respectively, and the variance is the same in all cases. To minimize the probability of demodulation error, we should separate a, b, and 0 as much as we can. Thus, under the peak transmission power constraint $\max(a^2, b^2) \le P^2$, the antipodal signaling $a=-b=P$ is optimum, as shown in Fig. 4.

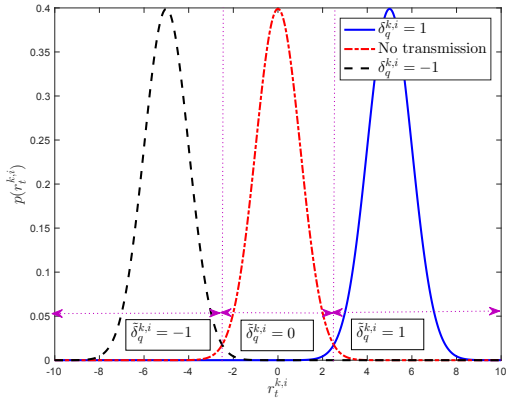

Fig. 4. Probability density function (pdf) of the signal $r_t^{k,i}$, given in (32), received by the controller through an AWGN channel when the optimum antipodal signaling $a=-b=5$ is used. In this example, the peak power constraint is $P^2=25$ and the variance is $\sigma^2=1$. For maximum likelihood (ML) demodulation, the decision boundaries and the corresponding decisions are shown with the vertical dotted lines and the boxed texts, respectively. The controller indeed makes such a decision at each time t. To minimize the modulation error, represented by the intersection of pdfs, signaling levels a and b are set apart as much as the peak power constraint allows.

2) Fading Channels: Under a fading channel model, the controller receives

$$r_t^{k,i} = h_t^{k,i} c_t^{k,i} + n_t^{k,i}, \tag{33}$$

where the channel coefficients $\{h_t^{k,i}\}, \forall t,i,k$ are i.i.d. with the distribution $\mathcal{N}(\mu, \rho^2)$ for Rician fading ($\mu \neq 0$) and Rayleigh fading ($\mu = 0$). Let us first analyze the Rayleigh fading case. The received signal $r_t^{k,i}$ is zero-mean Gaussian with variance $a^2\rho^2+\sigma^2$, $b^2\rho^2+\sigma^2$, and σ^2 for $\delta_q^{k,i}=1$, $\delta_q^{k,i}=-1$, and no transmission, respectively (see Fig. 5). This time we should separate a^2, b^2, and 0 to minimize the error probability. Hence, an asymmetric constellation is optimum under Rayleigh fading, as shown in Fig. 5. Under the same peak transmission power constraint, either a^2 or b^2 should be P^2, say $a^2=P^2$. Then, the choice for $b^2 \in [0, P^2]$ is given by the following theorem.

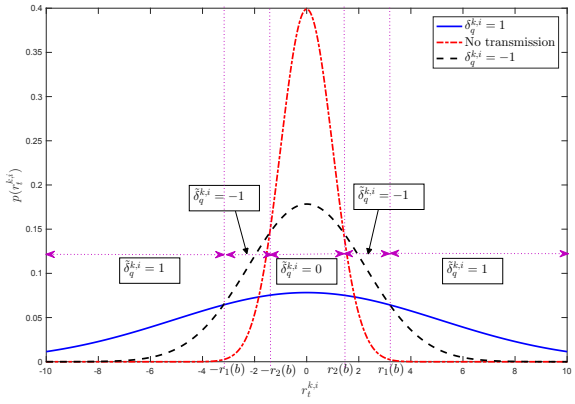

Fig. 5. Probability density function (pdf) of the signal $r_t^{k,i}$, given in (33), received by the controller through a Rayleigh fading channel when the optimum asymmetric signaling, $a^2=25$ and $b^2=4$, is used. In this example, the peak power constraint is $P^2=25$; the variances are $\sigma^2=\rho^2=1$; and the transmission probabilities are $\mathsf{P}(\delta_q^{k,i}=1)=\mathsf{P}(\delta_q^{k,i}=-1)=0.23$ and $\mathsf{P}(\text{No transmission})=0.54$. For maximum likelihood (ML) demodulation, the decision boundaries $\pm r_1(b), \pm r_2(b)$, and the corresponding decisions are shown with the vertical dotted lines and the boxed texts, respectively. The controller indeed makes such a decision at each time t. To minimize the expected modulation error, represented by the intersection of pdfs in the wrong decision regions, the signaling level b is determined based on the transmission probabilities, as shown in (34).

Theorem 4. *Consider Rayleigh fading channels between the nodes and the controller [see* (33)*], a peak power constraint P^2, and maximum likelihood (ML) demodulation. Then, the optimum signaling levels, in terms of minimizing the MSE per bit, for transmitting the LCS bits [see* (27)*] are given by $a_{ray} =$*

$\pm P$ *and*

$$b_{ray} = \arg\min_b \mathsf{P}(\delta=1)\left[4\Phi\left(\frac{r_1(b)}{\alpha}\right) - 3\Phi\left(\frac{r_2(b)}{\alpha}\right)\right] - \mathsf{P}(\delta=0)\Phi\left(\frac{r_2(b)}{\sigma}\right) + \mathsf{P}(\delta=-1)\left[-4\Phi\left(\frac{r_1(b)}{\beta(b)}\right) + \Phi\left(\frac{r_2(b)}{\beta(b)}\right)\right], \quad (34)$$

where a and b are interchangeable; $\Phi(\cdot)$ is the cumulative distribution function (cdf) of the standard Gaussian distribution; $\mathsf{P}(\delta=1)$, $\mathsf{P}(\delta=-1)$, $\mathsf{P}(\delta=0)$ are a node's probabilities of transmitting $+1$, -1, and nothing, respectively; $\alpha=\sqrt{a^2\rho^2+\sigma^2}$ and $\beta(b)=\sqrt{b^2\rho^2+\sigma^2}$ are the standard deviations of the signal received by the controller [see (33)] when $+1$ and -1 are transmitted, respectively; and

$$r_1(b) = \alpha\beta(b)\sqrt{\frac{2\log(\alpha/\beta(b))}{\alpha^2-\beta(b)^2}}$$
$$r_2(b) = \beta(b)\sigma\sqrt{\frac{2\log(\beta(b)/\sigma)}{\beta(b)^2-\sigma^2}} \quad (35)$$

are the decision boundary values for the received signal for ML demodulation (see Fig. 5).

Proof: From (30), the MSE per bit is given by

$$\begin{aligned}\mathsf{E}[(\delta\Delta-\tilde{\delta}\Delta)^2] = &\\ &(2\Delta)^2\mathsf{P}(\tilde{\delta}=-1,\delta=1)+\Delta^2\mathsf{P}(\tilde{\delta}=0,\delta=1)\\ &+\Delta^2\mathsf{P}(\tilde{\delta}=-1,\delta=0)+\Delta^2\mathsf{P}(\tilde{\delta}=1,\delta=0)\\ &+\Delta^2\mathsf{P}(\tilde{\delta}=0,\delta=-1)+(2\Delta)^2\mathsf{P}(\tilde{\delta}=1,\delta=-1) \quad (36)\end{aligned}$$

where $\delta\in\{-1,0,1\}$ is the transmitted bit, and $\tilde{\delta}\in\{-1,0,1\}$ is the received bit through the Rayleigh fading channel. Above we don't show the indices for node, dimension, and time since we assume i.i.d. channels. As shown in Fig. 5, the MSE per bit can be written as

$$\begin{aligned}\mathsf{E}[(\delta\Delta-\tilde{\delta}\Delta)^2] = &\\ &(2\Delta)^2\ \mathsf{P}\left(r_2(b)\le|r|\le r_1(b)\mid\delta=1\right)\ \mathsf{P}(\delta=1)\\ &+\Delta^2\ \mathsf{P}(-r_2(b)\le r\le r_2(b)\mid\delta=1)\ \mathsf{P}(\delta=1)\\ &+\Delta^2\ \mathsf{P}(r_2(b)\le|r|\le r_1(b)\mid\delta=0)\ \mathsf{P}(\delta=0)\\ &+\Delta^2\ \mathsf{P}(|r|\ge r_1(b)\mid\delta=0)\ \mathsf{P}(\delta=0)\\ &+\Delta^2\ \mathsf{P}(-r_2(b)\le r\le r_2(b)\mid\delta=-1)\ \mathsf{P}(\delta=-1)\\ &+(2\Delta)^2\ \mathsf{P}(|r|\ge r_1(b)\mid\delta=-1)\ \mathsf{P}(\delta=-1). \quad (37)\end{aligned}$$

After some manipulations, it is straightforward to show that

$$\begin{aligned}\mathsf{E}[(\delta\Delta-\tilde{\delta}\Delta)^2] = &\\ 2\Delta^2\Bigg\{&\ \mathsf{P}(\delta=1)\ \left[4\Phi\left(\frac{r_1(b)}{\alpha}\right)-3\Phi\left(\frac{r_2(b)}{\alpha}\right)-\frac{1}{2}\right]\\ &+\mathsf{P}(\delta=0)\ \left[1-\Phi\left(\frac{r_2(b)}{\sigma}\right)\right]\\ &+\mathsf{P}(\delta=-1)\ \left[\frac{7}{2}-4\Phi\left(\frac{r_1(b)}{\beta(b)}\right)+\Phi\left(\frac{r_2(b)}{\beta(b)}\right)\right]\Bigg\}, \quad (38)\end{aligned}$$

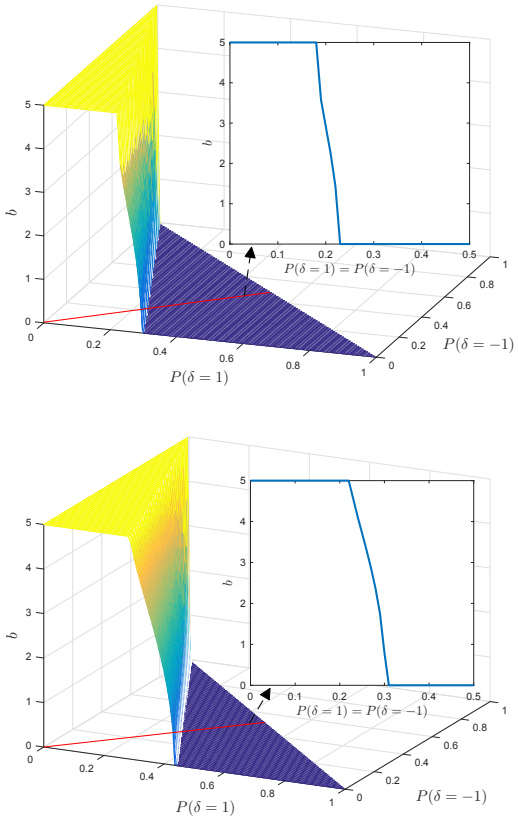

Fig. 6. The optimum signaling level b under Rayleigh fading channel vs. the transmission probabilities for the MSE per bit criterion [left figure - see (34)] and the MAE per bit criterion [right figure - see (39)]. $\mathsf{P}(\delta=1)$ and $\mathsf{P}(\delta=-1)$ denote the probability of transmitting at the level a and b, respectively. The probability of no transmission is given by $1-\mathsf{P}(\delta=1)-\mathsf{P}(\delta=-1)$. In practice, the case in which $\mathsf{P}(\delta=1)=\mathsf{P}(\delta=-1)$ is of special interest since $\hat{x}_t^{k,i}$ is neither increasing or decreasing. This case, represented by the red line, is shown specifically in the upper right corner in both figures.

minimizing which over b is equivalent to minimizing (34). The decision boundaries $r_1(b)$ and $r_2(b)$ are the intersection points at which the zero-mean Gaussian pdf with variance $\beta(b)^2$ coincides with the zero-mean Gaussian pdfs with variances α^2 and σ^2, respectively. Equating the pdf expressions at the same point, for instance for $r_1(b)$,

$$\frac{e^{-r_1(b)^2/2\alpha^2}}{\sqrt{2\pi}\alpha} = \frac{e^{-r_1(b)^2/2\beta(b)^2}}{\sqrt{2\pi}\beta(b)},$$

we get the closed-form solutions given in (35). ■

Fig. 6 shows the optimum signaling level b for different transmission probabilitites according to MSE per bit as in Theorem 4, and also mean absolute error (MAE) per bit $\mathsf{E}[|\delta\Delta-\tilde{\delta}\Delta|]$. For the MAE criterion, following the proof of Theorem 4 we can show that the optimum signaling levels

are given by $a_{\text{ray}} = \pm P$ and

$$b_{\text{ray}} = \arg\min_b \mathsf{P}(\delta=1)\left[2\Phi\left(\frac{r_1(b)}{\alpha}\right) - \Phi\left(\frac{r_2(b)}{\alpha}\right)\right] - \mathsf{P}(\delta=0)\Phi\left(\frac{r_2(b)}{\sigma}\right) + \mathsf{P}(\delta=-1)\left[-2\Phi\left(\frac{r_1(b)}{\beta(b)}\right) + \Phi\left(\frac{r_2(b)}{\beta(b)}\right)\right]. \quad (39)$$

In Fig. 6, it is seen that as the probability $\mathsf{P}(\delta = 1)$ of transmitting the level a increases from 0 to 1, the optimum level for b, as expected, transitions from 5, which is the same as a, to 0, which correpsonds to no transmission. In the MAE case, the transition takes place later, and the transition region is wider because the penalty for flipping a bit in this case is milder (2 times the penalty for losing a bit) than that in the MSE case (4 times the penalty for losing a bit).

Note that the transmission probabilities are determined by the sampling threshold Δ, as well as the signal to be sampled $\hat{x}_t^{k,i}$. In particular, larger Δ causes smaller $\mathsf{P}(\delta = 1)$ and $\mathsf{P}(\delta = -1)$, and thus larger no transmission probability. In practice, by setting Δ accordingly, we tend to attain moderate transmission rates (e.g., 1 bit per dimension per unit time), which correspond to the transition regions in Fig. 6, hence no information loss due to the identical signaling levels.

It might make more sense to minimize the cumulative MSE $= \mathsf{E}[\|\hat{\boldsymbol{x}}_t - \tilde{\hat{\boldsymbol{x}}}_t\|_2^2]$ instead of the MSE per bit. In this case, the optimum b value that minimizes the cumulative MSE can be numerically found through offline simulations.

Since Rician fading includes AWGN and Rayleigh fading as extreme cases with $\rho^2 = 0$ and $\mu^2 = 0$, respectively, the optimum signaling scheme under Rician fading is a function of μ^2 and ρ^2.

Recall from (26) that node k transmits $\theta_q^{k,i}$ repetitions of the sign bit $\delta_q^{k,i}$, where $\theta_q^{k,i}$ denotes the number of level crossings. Instead of transmitting $\theta_q^{k,i}$ times, one may choose to encode $\theta_q^{k,i}$ in time or frequency or amplitude based on available resources. Specifically, assuming $\theta_q^{k,i} < M$, for all q, i, k and for some $M > 0$, which easily holds in practice, one can uniformly partition the available time/frequency/amplitude interval into M subintervals, and transmit $\delta_q^{k,i}$ in the $\theta_q^{k,i}$th subinterval. Such an encoding enjoys a single transmission, hence higher resource efficiency, whereas multiple transmissions (i.e., $\theta_q^{k,i}$ repetitions of $\delta_q^{k,i}$) provide robustness against the demodulation errors. Note that encoding $\theta_q^{k,i}$ in energy levels (i.e., pulse-amplitude modulation) works well only under AWGN channels, but not fading channels due to the multiplicative noise. We next discuss using multiple-access channels.

B. Multiple-Access Channels

Since a received bit at the controller, regardless of its source node, causes the same update (Δ or $-\Delta$), nodes can synchronously transmit in multiple-access channels in a more bandwidth-efficient way than the previous setup with Kn parallel channels.

Under fading channels with asymmetric signaling, same polarity bits that correspond to the same state estimate, i.e., $\{\delta_q^{k,i} = j\}_{k,q},\ i = 1, \ldots, n,\ j = \pm 1$, can be transmitted in the same multiple-access channel, resulting in $2n$ parallel channels. Then, the controller receives

$$r_t^{ij} = c_j \sum_{k=1}^{K_t^{ij}} h_t^{k,ij} + n_t^{ij}, \quad (40)$$

where K_t^{ij} is the number of nodes transmitting bits at time t regarding the state estimate i with polarity j, and the asymmetric transmission levels $c_{+1} = a$ and $c_{-1} = b$. The update information carried by r_t^{ij} for $\hat{x}_t^i$ is $K_t^{ij} j\Delta$, where K_t^{ij} is unknown. Hence, the controller needs to estimate K_t^{ij} from r_t^{ij}. Defining the effective channel coefficient $h_t^{ij} \triangleq \sum_{k=1}^{K_t^{ij}} h_t^{k,ij}$ we have $h_t^{ij} \sim \mathcal{N}(K_t^{ij}\mu, K_t^{ij}\rho^2)$ and thus $r_t^{ij} \sim \mathcal{N}(c_j K_t^{ij}\mu, c_j^2 K_t^{ij}\rho^2 + \sigma^2)$. As a result, using the maximum likelihood (ML) criterion the controller can estimate K_t^{ij} as

$$\tilde{K}_t^{ij} = \arg\min_{K_t^{ij} \in \{0,1,\ldots,K\}} \frac{(r_t^{ij} - c_j K_t^{ij}\mu)^2}{c_j^2 K_t^{ij}\rho^2 + \sigma^2} + \log(c_j^2 K_t^{ij}\rho^2 + \sigma^2), \quad (41)$$

and update $\tilde{\hat{x}}_q^i$ with $\tilde{K}_t^{ij} j\Delta$ [cf. (30)].

The optimum total update for $\tilde{\hat{x}}_t^i$ at time t is $(K_t^{i,+1} - K_t^{i,-1})\Delta$ and estimated at the controller as $(\tilde{K}_t^{i,+1} - \tilde{K}_t^{i,-1})\Delta$. Note that in symmetric signaling (i.e., $b = -a$), we can transmit all bits regarding the same input, that is, $\{\delta_q^{k,i}\}_{k,q}$, in a single multiple-access channel without losing the necessary and sufficient information $K_t^i \triangleq K_t^{i,+1} - K_t^{i,-1}$. In other words, we can combine different polarity channels regarding the same input, resulting in n parallel channels, decreasing the number of required parallel channels from $2n$ (in the asymmetric case) to n. Specifically, the signal received by the controller at time t for $\hat{x}_t^i$ is given by

$$r_t^i = a\left(\sum_{k=1}^{K_t^{i,+1}} h_t^{k,i,+1} - \sum_{k=1}^{K_t^{i,-1}} h_t^{k,i,-1}\right) + n_t^{ij}. \quad (42)$$

Defining the effective channel coefficient $h_t^i \triangleq \sum_{k=1}^{K_t^{i,+1}} h_t^{k,i,+1} - \sum_{k=1}^{K_t^{i,-1}} h_t^{k,i,-1}$ we see that $h_t^i \sim \mathcal{N}(K_t^i\mu, K_t^i\rho^2)$ and thus $r_t^i \sim \mathcal{N}(aK_t^i\mu, a^2K_t^i\rho^2 + \sigma^2)$. Similar to (41), the controller finds the ML estimator as

$$\tilde{K}_t^i = \arg\min_{K_t^i \in \{-K,\ldots,K\}} \frac{(r_t^i - aK_t^i\mu)^2}{a^2K_t^i\rho^2 + \sigma^2} + \log(a^2K_t^i\rho^2 + \sigma^2), \quad (43)$$

and updates $\tilde{\hat{x}}_q^i$ with $\tilde{K}_t^i\Delta$. Note that in (43), $K_t^i \in \{-K, \ldots, K\}$, whereas in (41) $K_t^{ij} \in \{0, 1, \ldots, K\}$.

Similarly, under AWGN channels with symmetric signaling, we can transmit all the bits regarding the same input in a single multiple-access channel. In this simpler case, the controller receives

$$r_t^i = aK_t^i + n_t^i, \quad (44)$$

which is clearly distributed according to $\mathcal{N}(aK_t^i, \sigma^2)$. Hence, the ML estimator is given by

$$\tilde{K}_t^i = \arg\min_{K_t^i \in \{-K,\ldots,K\}} (r_t^i - aK_t^i)^2 = \arg\min_{K_t^i \in \{-K,\ldots,K\}} \left|K_t^i - \frac{r_t^i}{a}\right|. \quad (45)$$

That is, $\tilde{K}_t^i$ is the closest $K_t^i \in \{-K, \ldots, K\}$ to $\frac{r_t^i}{a}$, which we can write as

$$\tilde{K}_t^i = \min\left(\max\left(\text{round}\left(\frac{r_t^i}{a}\right), -K\right), K\right). \quad (46)$$

It is clearly seen that synchronous communication over multiple-access channels provides significant bandwidth savings (n or $2n$ vs. Kn parallel channels). The number of level crossings $\theta_q^{k,i}$ can be reported to the controller by again either multiple transmissions or encoding in time/frequency/energy, as discussed in Section IV-A.

V. Distributed LQG Control For Resource-Constrained Systems

As a practical application of the proposed distributed state estimation method in networked control systems, we consider the finite-horizon linear quadratic Gaussian (LQG) control problem in which the following quadratic cost function J_N is minimized,

$$J_N \triangleq \mathsf{E}\left[\boldsymbol{x}_N^T \boldsymbol{Q}_0 \boldsymbol{x}_N + \sum_{t=0}^{N-1}\left(\boldsymbol{x}_t^T \boldsymbol{Q}_1 \boldsymbol{x}_t + \boldsymbol{u}_t^T \boldsymbol{Q}_2 \boldsymbol{u}_t\right)\right], \quad (47)$$

where $\boldsymbol{Q}_0$, $\boldsymbol{Q}_1$, and $\boldsymbol{Q}_2$ are symmetric and positive semidefinite matrices of appropriate dimensions. Our objective is to find the optimal control strategy $\{\boldsymbol{u}_t\}$ that minimizes the cost function J_N above.

Due to the separation principle [30, page 157] in the optimal LQG control (cf. Remark 1 at the end of this section), we first obtain the optimum state estimate $\hat{\boldsymbol{x}}_t$, and then the optimum control vector

$$\boldsymbol{u}_t = -\boldsymbol{L}_t \hat{\boldsymbol{x}}_t, \quad (48)$$

where $\boldsymbol{L}_t$ is the optimum feedback gain matrix

$$\boldsymbol{L}_t = (\boldsymbol{B}^T \boldsymbol{S}_{t+1} \boldsymbol{B} + \boldsymbol{Q}_2)^{-1} \boldsymbol{B}^T \boldsymbol{S}_{t+1} \boldsymbol{A}, \quad (49)$$

and $\boldsymbol{S}_t$ is given by the recursive equation

$$\boldsymbol{S}_t = \boldsymbol{A}^T \boldsymbol{S}_{t+1} \boldsymbol{A} - \boldsymbol{A}^T \boldsymbol{S}_{t+1} \boldsymbol{B} \boldsymbol{L}_t + \boldsymbol{Q}_1 \quad (50)$$

for $t = N-1, \ldots, 1$ with the initial value $\boldsymbol{S}_N = \boldsymbol{Q}_0$. Note that given the system matrices $\boldsymbol{A}, \boldsymbol{B}$, and the cost matrices $\boldsymbol{Q}_0, \boldsymbol{Q}_1, \boldsymbol{Q}_2$, using (49) and (50) the optimum feedback gain $\boldsymbol{L}_t$ can be computed offline. Hence, in the optimal LQG control only the state estimate $\hat{\boldsymbol{x}}_t$ is computed online.

Lemma 2. *The systemwide optimum control input $\boldsymbol{u}_t$ can be written as the sum of local contributions as follows*

$$\boldsymbol{u}_t = \sum_{k=1}^{K} \boldsymbol{u}_t^k,$$

where $\boldsymbol{u}_t^k$ is the local contribution from node k, given by

$$\boldsymbol{u}_t^k = -\boldsymbol{L}_t \left(\boldsymbol{Y}_t^{-1} \sum_{s=1}^{t-1} \boldsymbol{\Psi}_s^{t-1} \boldsymbol{Y}_s \boldsymbol{A}^{-1} \boldsymbol{B} \boldsymbol{u}_s^k + \hat{\boldsymbol{x}}_t^k\right). \quad (51)$$

Proof: Using (13), (23) and (48), we can write

$$\begin{aligned}\boldsymbol{u}_t &= -\boldsymbol{L}_t \left(\boldsymbol{\varphi}_t + \sum_{k=1}^{K} \hat{\boldsymbol{x}}_t^k\right) \\ &= -\boldsymbol{L}_t \left(\boldsymbol{Y}_t^{-1} \boldsymbol{\lambda}_{t-1} + \sum_{k=1}^{K} \hat{\boldsymbol{x}}_t^k\right). \end{aligned} \quad (52)$$

The update rules for $\boldsymbol{\lambda}_t$ and $\boldsymbol{\zeta}_t$, given by (18) and (19), respectively, share the same structure. Thus, similar to (20), we can show that

$$\boldsymbol{\lambda}_t = \sum_{s=1}^{t} \boldsymbol{\Psi}_{s+1}^t \boldsymbol{\xi}_s \quad (53)$$

where from (10), (4), (8) and the definition of $\boldsymbol{\Omega}_t$ in (15), we have

$$\boldsymbol{\xi}_t = \boldsymbol{\Omega}_t \boldsymbol{Y}_t \boldsymbol{A}^{-1} \boldsymbol{B} \boldsymbol{u}_t. \quad (54)$$

Combining (52), (53) and (54) we get

$$\boldsymbol{u}_t = -\boldsymbol{L}_t \left(\boldsymbol{Y}_t^{-1} \sum_{s=1}^{t-1} \boldsymbol{\Psi}_s^{t-1} \boldsymbol{Y}_s \boldsymbol{A}^{-1} \boldsymbol{B} \boldsymbol{u}_s + \sum_{k=1}^{K} \hat{\boldsymbol{x}}_t^k\right), \quad (55)$$

Note that $\boldsymbol{u}_1 = \sum_{k=1}^{K} -\boldsymbol{L}_t \hat{\boldsymbol{x}}_1^k$, and as a result, for all t, $\boldsymbol{u}_t$ is a function of $\left\{\sum_{k=1}^{K} \hat{\boldsymbol{x}}_t^k\right\}_{t \geq 1}$ and globally known matrices. Hence, (55) can be rewritten as

$$\begin{aligned}\boldsymbol{u}_t &= \sum_{k=1}^{K} -\boldsymbol{L}_t \left(\boldsymbol{Y}_t^{-1} \sum_{s=1}^{t-1} \boldsymbol{\Psi}_s^{t-1} \boldsymbol{Y}_s \boldsymbol{A}^{-1} \boldsymbol{B} \boldsymbol{u}_s^k + \hat{\boldsymbol{x}}_t^k\right) \\ &= \sum_{k=1}^{K} \boldsymbol{u}_t^k. \end{aligned}$$

■

Note that each node can compute the matrix $\boldsymbol{L}_t$ using (49) and (50), as well as $\boldsymbol{Y}_t, \boldsymbol{\Psi}_s^t, \boldsymbol{A}, \boldsymbol{B}$. Hence, we propose that each node k computes $\boldsymbol{u}_t^k$ as in (51), and reports it to the controller node using level-crossing sampling, as described in Section III-A for $\hat{\boldsymbol{x}}_t^k$. In particular, each node k runs p parallel level-crossing samplers for $\{u_t^{k,i}\}_{i=1}^p$, where $u_t^{k,i}$ is the ith entry of $\boldsymbol{u}_t^k$. Let $\tilde{u}_q^{k,i}$ denote the approximation of $u_t^{k,i}$ at the controller node during the time interval $\tau_q^{k,i} \leq t < \tau_{q+1}^{k,i}$. Then, the controller computes the input vector $\tilde{\boldsymbol{u}}_t = [\tilde{u}_t^1, \ldots, \tilde{u}_t^p]^T$, where $\tilde{u}_t^i = \sum_{k=1}^{K} \tilde{u}_q^{k,i}$, and applies it to the system.

Corollary 1. *For each input u_t^i, under ideal communications, the performance gap between the proposed decentralized scheme and optimum centralized scheme is deterministically bounded, i.e.,*

$$|u_t^i - \tilde{u}_t^i| < K\Delta, \ \forall i, t, \quad (56)$$

and yields order-2 asymptotic optimality, i.e., $u_t^i - \tilde{u}_t^i = O(1)$ even if $|u_t^i| \to \infty$.

Proof of (56) is similar to the proof of Theorem 1. Moreover, since in a large system, in general, $p \ll n$, this marks another advantage of reporting $\boldsymbol{u}_t^k$ instead of $\boldsymbol{\phi}_t^k$ (or $\boldsymbol{y}_t^k$ or $\hat{\boldsymbol{x}}_t^k$) in the distributed LQG control.

Following a similar discussion to Theorem 2, it can be shown that as $|u_t^i| \to \infty$ the confidence interval $|u_t^i - \tilde{u}_t^i|$ is

unbounded under non-ideal communication channels, hence order-2 asymptotic optimality is not satisfied. However, for a practical scenario where $|u_t^i| < \infty$ as $t \to \infty$, we also have $|\tilde{u}_t^i| < \infty$ for any $p > 0$. Similar to the arguments given in Theorem 3, order-1 asymptotic optimality is possible, i.e., $\frac{\tilde{u}_t^i}{u_t^i} = 1 + o(1)$ as $|u_t^i| \to \infty$, if $u_t^i \to \infty$ at a rate faster than t, that is, $u_t^i = \omega(t)$. Since this condition is practically infeasible, the modulation techniques presented in Section IV can be used to improve the non-asymptotic performance.

Remark 1: Our discussion and results in this section are based on the separation principle and system stability, which are not guaranteed to hold in general, especially under noisy communication channels. In particular, they require some special conditions, e.g., a minimum level of data rate and a maximum level of communication delay, as discussed in Introduction in more detail. Hence, the performance shown in Corollary 1 is valid only if such conditions are satisfied.

Remark 2: Together with Theorem 2, the results presented in Corollary 1 and the following discussions show that the proposed entire solution (i.e., decentralized state estimator and LQG control scheme) is bounded-input bounded-output (BIBO) stable even under noisy channels as long as the optimum centralized solution is BIBO stable.

VI. Numerical Results

In this section, we illustrate the advantages of the proposed schemes based on level-crossing sampling (LCS) through numerical results considering a practical electrical power grid. In particular, we examine the performance of the distributed LQG control scheme presented in Section V since it is a practical application that includes both state estimation and control. Throughout this section we use the IEEE 57-bus data in MATPOWER [31] with $n = 57$ state variables, $m = 80$ measurements, $p = 7$ control inputs, and $\boldsymbol{Q}_0$, $\boldsymbol{Q}_1$, $\boldsymbol{Q}_2$, $\boldsymbol{R}_1$, $\boldsymbol{R}_{2,k}$ equal to the identity matrices of appropriate sizes. In Fig. 7, in a system consisting of four nodes (i.e., $K = 4$), in terms of mean squared error (MSE), given by $\mathsf{E}[\|\boldsymbol{u}_t - \tilde{\boldsymbol{u}}_t\|_2^2] = \sum_{i=1}^{p} \mathsf{E}[(u_t^i - \tilde{u}_t^i)^2]$, we compare the proposed scheme with the conventional schemes that report $\boldsymbol{z}_t^k$, $\boldsymbol{\phi}_t^k$, $\boldsymbol{u}_t^k$ via uniform sampling (US) and quantization. In the literature, it is conventional that each node k transmits the raw measurement $\boldsymbol{z}_t^k$ or the information vector $\boldsymbol{\phi}_t^k$, and the control center computes the Kalman filter or the information filter, respectively. In that sense, a scheme that transmits the local control vector $\boldsymbol{u}_t^k$ based on (51) is not conventional. It is conventional in terms of transmission method if it uses uniform sampling and quantization. For fair comparison, in the proposed scheme, on average a single bit is transmitted per dimension per unit time, and in the other schemes based on uniform sampling at each time a single quantization bit is transmitted per dimension.

A. MSE vs. Time

As shown in the top figure in Fig. 7, the "fully" conventional scheme that transmits $\boldsymbol{\phi}_t^k$ via uniform sampling suffers a huge performance gap due to the insufficient one-bit representations of the wide range of $\boldsymbol{\phi}_t^k$ values (see Fig. 8). We see in Fig. 7 that the scheme that transmits $\boldsymbol{z}_t^k$ using uniform sampling, on the contrary, attains an acceptable performance. Since the range of $\boldsymbol{z}_t^k$ values is much smaller than that of $\boldsymbol{\phi}_t^k$ (see Fig. 8), the approximate $\tilde{\boldsymbol{\phi}}_t^k$ that is obtained from the recovered $\tilde{\boldsymbol{z}}_t^k$ yields a better result than the quantized and recovered $\tilde{\boldsymbol{\phi}}_t^k$. Note that the number of dimensions in $\boldsymbol{\phi}_t^k$ and $\boldsymbol{z}_t^k$ are $n = 57$ and $m = 80$, respectively. In other words, in the schemes US-ϕ and US-z, at each time 57 and 80 bits are transmitted systemwide, respectively, whereas only $p = 7$ bits are transmitted at each time in the scheme US-u and the proposed scheme LCS-u (on average). We allow this unfairness in Fig. 7 because LCS-u still perform better than US-ϕ and US-z.

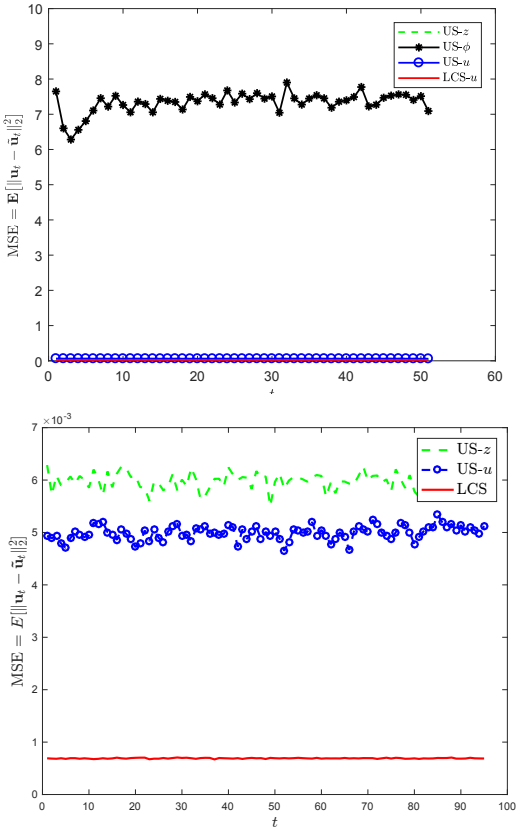

Fig. 7. Mean squared error vs. time for the proposed scheme based on level-crossing sampling (LCS) and the conventional schemes that transmit $\boldsymbol{z}_t^k$, $\boldsymbol{\phi}_t^k$, $\boldsymbol{u}_t^k$ via uniform sampling (US) and quantization.

Among the schemes that transmit the local control vector $\boldsymbol{u}_t^k$, which is the final information entity in the system, we observe in the bottom figure in Fig. 7 that the proposed scheme based on level-crossing sampling outperforms the "half" conventional scheme based on uniform sampling. It is "half" conventional because it unconventionally transmits $\boldsymbol{u}_t^k$, as we propose in (52). Quantization threshold of each $u_t^{k,i}$ is set to zero, and the two quantization levels are set as the means of the positive ($u_t^{k,i} > 0$) and negative ($u_t^{k,i} < 0$) values, that are estimated offline. The fluctuations in the conventional schemes are due to the coarse (one-bit) quantization. It is seen in the bottom figure in Fig. 7 that the performance improves

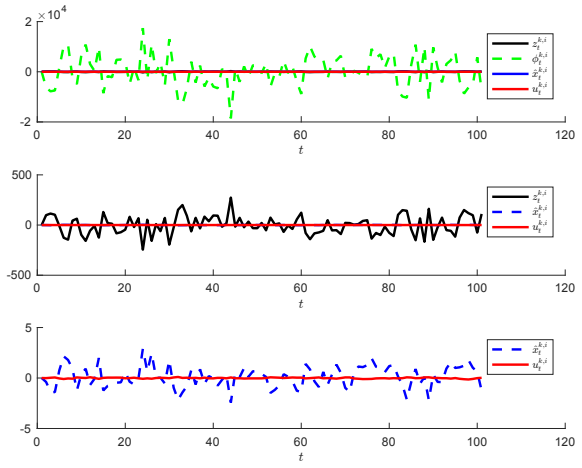

Fig. 8. Sample paths of $u_t^{k,i}$, $\hat{x}_t^{k,i}$, $\phi_t^{k,i}$, and $z_t^{k,i}$. It is seen that $u_t^{k,i}$ is much smoother than $\phi_t^{k,i}$, $z_t^{k,i}$ (top and middle figures), and $\hat{x}_t^{k,i}$ (bottom figure).

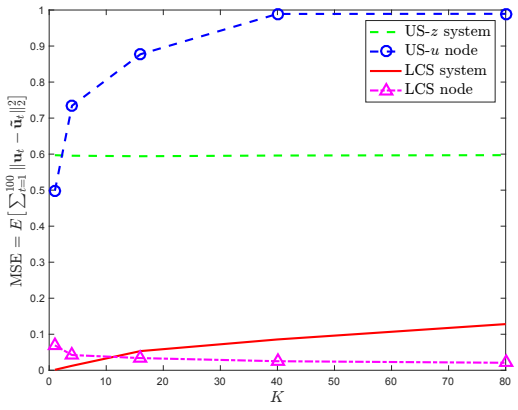

Fig. 9. Mean squared error vs. number of nodes for the resource constraints on the system (i.e., 80 bits per unit time systemwide) and on the nodes (i.e., 1 bit per dimension per unit time per node, which corresponds to $7K$ bits per unit time systemwide). With the increasing node diversity, the proposed LCS-based scheme, under both system and node resource constraints, significantly outperforms its counterparts.

by the transmission of a more finalized information entity, as anticipated in Section V.

B. MSE vs. Number of Nodes

We next discuss the effect of node diversity on the cumulative MSE performance $\sum_{t=1}^{100} \mathsf{E}[\|\boldsymbol{u}_t - \tilde{\boldsymbol{u}}_t\|_2^2]$ for the proposed scheme and its counterparts based on uniform sampling. As shown in Fig. 9, the performance of the conventional scheme that transmits $\boldsymbol{z}_t^k$ via uniform sampling is not affected by the increasing number of nodes since the number of transmitted measurements is constant ($m = 80$ in this case) regardless of the number of nodes. For the controller, it does not matter who sends the measurements as long as the measurements are the same regardless of the number and identity of the senders. Hence, in this case, the number of bits transmitted systemwide per unit time, which is 80, does not change with the number of nodes. Considering this case as a resource (communication and energy) constraint on the system we use the name "US-z system" to differentiate between the resource usages of the control schemes.

In the scheme that transmits $\boldsymbol{u}_t^k$ using uniform sampling, more resources are used as the number of nodes increases. Specifically, each node k transmits 1 bit per dimension i of $\{u_t^{k,i} : i = 1, \ldots, p\}$ per unit time, resulting in Kp bits per unit time systemwide. Since this corresponds to a resource constraint on the nodes, we use the name "US-u node" to express its resource usage. Despite the increase in resource consumption, its MSE performance stays nearly constant with the increasing node diversity, as shown in Fig. 9. At first glance this may look counterintuitive as K represents the number of information sources. However, in this case, larger K does not mean more information, but means more distributed (or decentralized) information. Specifically, larger K means more local signals $\boldsymbol{u}_t^k$ to report to the controller, hence more quantization losses unless the range of $\boldsymbol{u}_t^k$ shrinks as fast as $1/K$. Apparently, for smaller K values, the range of $\boldsymbol{u}_t^k$ shrinks slower than $1/K$.

We next compare the proposed scheme that transmits $\boldsymbol{u}_t^k$ using level-crossing sampling with the conventional schemes "US-z system" and "US-u node" under the same system and node resource constraints, respectively. Under the system constraints, in which 80 bits are transmitted systemwide per unit time, the performance of "LCS-u system" deteriorates as the number of nodes increases due to the decentralization of the systemwide optimum control vector $\boldsymbol{u}_t$. However, as seen in Fig. 9, it still significantly outperforms "US-z system" even in the most decentralized case with 80 nodes (i.e., one measurement per node). Under the resource constraints on each node, in which each node on average transmits 1 bit per dimension per unit time, the performance of the proposed scheme "LCS-u node" improves with the increasing node diversity, as opposed its counterpart "US-u node" (see Fig. 9). This is because as K increases each local control input $u_t^{k,i}$ gets smaller values, and as a result we decrease the sampling threshold Δ to ensure 1 bit is transmitted on average for each $u_t^{k,i}$ per unit time. Note that the performances of "LCS-u system" and "LCS-u node" coincide at around $K = 11$ as expected since the resource consumption becomes the same (i.e., 80 bits systemwide per unit time) for both.

C. Discussions

According to (56), under ideal communication channels, it is certain (not probabilistic) that the optimum control input u_t^i lies in the confidence interval $(\tilde{u}_t^i - K\Delta, \tilde{u}_t^i + K\Delta)$, which shrinks with small Δ. Accordingly, in selecting the Δ value, there is a trade-off between performance and resource-efficiency. Specifically, small Δ improves the performance, as stated in (56), but at the same time results in more frequent transmissions [see e.g., (25) and (26) in the case of state estimation], consuming more energy and bandwidth. Hence, Δ should be selected to strike a desired balance between the performance and resource efficiency.

Considering the LQG cost, given in (47), as the performance measure for the control system, the expected tradeoff between the control performance and the resource efficiency, based on

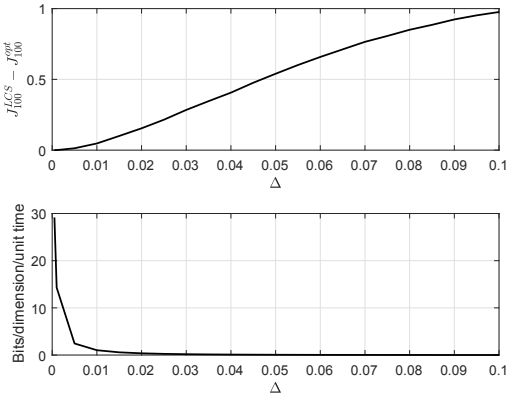

Fig. 10. Tradeoff between the control performance and resource usage. Control performance of the proposed scheme is represented by the additional cost $J_{100}^{LCS} - J_{100}^{opt}$ incurred due to decentralization at time 100 [see (47)]. Resource usage is denoted by the average number of bits transmitted by each node per dimension per unit time.

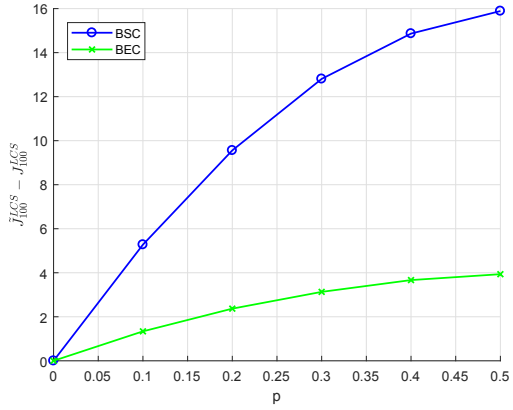

Fig. 11. Additional LQG cost vs. probability of error for BEC and BSC. The sampling threshold Δ is taken as 0.01.

the selection of the sampling threshold Δ, is illustrated in Fig. 10. In the figure, the average number of transmitted bits is given per unit time per node per dimension. For different Δ levels, J_{100}, given by (47), is calculated for the proposed decentralized scheme, and denoted by J_{100}^{LCS}. The additional cost in J_{100}^{LCS} with respect to the optimum centralized scheme, denoted by J_{100}^{opt}, in which all information is fully available to the controller (i.e., $\Delta \to 0$), is shown by $J_{100}^{LCS} - J_{100}^{opt}$ in the figure.

Performance	• minimizes the processing of lossy information at the controller (see (56) and Fig. 7)
Resource-efficiency (Energy & Bandwidth)	• smaller number of parallel samplers, and thus transmissions per node (e.g., $p = 7 \ll n = 57$ and $m = 80$ in IEEE 57-bus) • smoother signal with smaller jumps, hence smaller number of transmissions per sample (see Fig. 8 and (26))

TABLE I
ADVANTAGES OF REPORTING $\boldsymbol{u}_t^k$ INSTEAD OF $\boldsymbol{z}_t^k$, $\boldsymbol{\phi}_t^k$, $\hat{\boldsymbol{x}}_t^k$, GIVEN BY (1), (12), (22), RESPECTIVELY.

In addition to Δ, the smoothness of $u_t^{k,i}$ determines the transmission frequency. In particular, big jumps in $u_t^{k,i}$ may lead to frequent sampling and/or large number of transmissions per sample, hence more resource consumption. Fortunately, $u_t^{k,i}$, being a control input applied to the system, is much smoother than the observation $z_t^{k,i}$, the information entity $\phi_t^{k,i}$, and the state estimate $\hat{x}_t^{k,i}$, as shown in Fig. 8. The advantages of reporting $\boldsymbol{u}_t^k$ in the distributed LQG control instead of some earlier products in local processing, such as $\boldsymbol{z}_t^k$, $\boldsymbol{\phi}_t^k$, and $\hat{\boldsymbol{x}}_t^k$, are summarized in Table I.

Finally, the effect of nonzero error probability, p, in terms of the additional LQG cost $\tilde{J}_{100}^{LCS} - J_{100}^{LCS}$, where J_{100}^{LCS} and $\tilde{J}_{100}^{LCS}$ are the costs at time 100 [see (47)] for the proposed decentralized scheme under reliable and unreliable channels, respectively, is illustrated in Fig. 11. As expected, as the error probability p increases, the additional cost increases as well, i.e., the control performance degrades. Moreover, under BSC, the control performance degrades more compared to the BEC case since an incorrect update on the information entities is more destructive on the system performance compared to the no-update case (i.e., $\pm 2\Delta$ error vs. $\pm\Delta$ error).

VII. CONCLUSIONS

In this paper, the distributed dynamic state estimation and LQG control problems have been analyzed for a networked control system using an event-triggered sampling scheme called *level-crossing sampling*. The advantages of processing the information locally and sampling it with level-crossing sampling have been shown through theoretical analysis and simulations. Particularly, it has been shown that the information loss in the proposed decentralized schemes has a deterministic upper bound, and thus yields an order-2 asymptotic optimality under reliable communication channels. Furthermore, in transmitting information from sensors to the controller, noisy channels have been considered and corresponding optimal modulation techniques have been proposed to improve non-asymptotic performance of the proposed schemes.

REFERENCES

[1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, Jan 2007.
[2] X. Zhang, Q. L. Han, and X. Yu, "Survey on recent advances in networked control systems," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2015.
[3] Y. Yilmaz, G. Moustakides, X. Wang, and A. Hero, "Event-based statistical signal processing," in *Event-Based Control and Signal Processing*, M. Miskowicz, Ed. CRC Press.
[4] Y. Yılmaz, G. Moustakides, and X. Wang, "Channel-aware decentralized detection via level-triggered sampling," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 300–315, Jan 2013.
[5] Y. Yılmaz and X. Wang, "Sequential decentralized parameter estimation under randomly observed Fisher information," *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1281–1300, Feb 2014.
[6] Z. Heng, P. Chen, Z. Jin, and Z. Chu, "Event-triggered control in networked control systems: A survey," in *The 27th Chinese Control and Decision Conference (2015 CCDC)*, May 2015, pp. 3092–3097.

[7] X. M. Zhang, Q. L. Han, and B. L. Zhang, "An overview and deep investigation on sampled-data-based event-triggered control and filtering for networked systems," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2016.

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

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

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

[11] L. Wenshuang, Z. Shanying, C. Cailian, and G. Xinping, "Distributed consensus filtering based on event-driven transmission for wireless sensor networks," in *Proceedings of the 31st Chinese Control Conference*, July 2012, pp. 6588–6593.

[12] W. Li, Y. Jia, and J. Du, "Event-triggered Kalman consensus filter over sensor networks," *IET Control Theory Applications*, vol. 10, no. 1, pp. 103–110, 2016.

[13] S. Trimpe and R. D'Andrea, "Event-based state estimation with variance-based triggering," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3266–3281, Dec 2014.

[14] E. J. Msechu, S. I. Roumeliotis, A. Ribeiro, and G. B. Giannakis, "Decentralized quantized Kalman filtering with scalable communication cost," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3727–3741, 2008.

[15] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, "Soi-kf: Distributed Kalman filtering with low-cost communications using the sign of innovations," *IEEE Transactions on signal processing*, vol. 54, no. 12, pp. 4782–4795, 2006.

[16] D. H. Varsakelis and L. Zhang, "LQG control of networked control systems with access constraints and delays," *International Journal of Control*, vol. 81, no. 8, pp. 1266–1280, 2008.

[17] Q. Liu and F. Jin, "LQG control of networked control systems with limited information," *Mathematical Problems in Engineering*, 2014.

[18] X. Wang and M. D. Lemmon, "Event-triggering in distributed networked control systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 586–601, March 2011.

[19] E. Garcia and P. J. Antsaklis, "Model-based event-triggered control for systems with quantization and time-varying network delays," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 422–434, Feb 2013.

[20] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, Jan 2007.

[21] M. Moayedi, Y. K. Foo, and Y. C. Soh, "Networked LQG control over unreliable channels," in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 5851–5856.

[22] B. Demirel, V. Gupta, D. E. Quevedo, and M. Johansson, "On the trade-off between communication and control cost in event-triggered dead-beat control," *IEEE Transactions on Automatic Control*, vol. PP, no. 99, pp. 1–1, 2016.

[23] A. Cetinkaya, H. Ishii, and T. Hayakawa, "Event-triggered control over unreliable networks subject to jamming attacks," in *54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 4818–4823.

[24] J. W. Mark and T. Todd, "A nonuniform sampling approach to data compression," *Communications, IEEE Transactions on*, vol. 29, no. 1, pp. 24–32, Jan 1981.

[25] Y. Yılmaz, G. Moustakides, and X. Wang, "Cooperative sequential spectrum sensing based on level-triggered sampling," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4509–4524, Sept 2012.

[26] A. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. Boca Raton, FL: CRC, 1998.

[27] G. Fellouris and G. Moustakides, "Decentralized sequential hypothesis testing using asynchronous communication," *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 534–548, Jan 2011.

[28] X. Liu and A. Goldsmith, "Kalman filtering with partial observation losses," in *2004 43rd IEEE Conference on Decision and Control (CDC)*, vol. 4, Dec 2004, pp. 4180–4186 Vol.4.

[29] V. Lippiello, F. Ruggiero, and D. Serra, "Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 4782–4788.

[30] F. Fairman, *Linear Control Theory: The State Space Approach*. Chichester, England: Wiley, 1998.

[31] R. Zimmerman, C. Murillo-Snchez, and R. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, Feb 2011.

Yasin Yılmaz (S'11-M'14) received the B.Sc., M.Sc., and Ph.D. degrees in Electrical Engineering from Middle East Technical University, Ankara, Turkey in 2008, Koc University, Istanbul, Turkey in 2010, and Columbia University, New York, NY, in 2014, respectively. He is currently an Assistant Professor of Electrical Engineering at the University of South Florida, Tampa. He received the Collaborative Research Award from Columbia University in 2015. His research interests include statistical signal processing, machine learning, and their applications to cybersecurity, IoT networks, social networks, communication systems, and cyber-physical systems.

Mehmet Necip Kurt received the B.S. and the M.S. degrees in 2014 and 2016, respectively from the Department of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey. Currently, he is working towards the Ph.D. degree in electrical engineering at Columbia University, New York. His current research interests include statistical signal processing with applications in cyber-physical systems.

Xiaodong Wang (S'98-M'98-SM'04-F'08) received the Ph.D degree in Electrical Engineering from Princeton University. He is a Professor of Electrical Engineering at Columbia University in New York. Dr. Wang's research interests fall in the general areas of computing, signal processing and communications, and has published extensively in these areas. Among his publications is a book entitled "Wireless Communication Systems: Advanced Techniques for Signal Reception", published by Prentice Hall in 2003. His current research interests include wireless communications, statistical signal processing, and genomic signal processing. Dr. Wang received the 1999 NSF CAREER Award, the 2001 IEEE Communications Society and Information Theory Society Joint Paper Award, and the 2011 IEEE Communication Society Award for Outstanding Paper on New Communication Topics. He has served as an Associate Editor for the *IEEE Transactions on Communications*, the *IEEE Transactions on Wireless Communications*, the *IEEE Transactions on Signal Processing*, and the *IEEE Transactions on Information Theory*. He is a Fellow of the IEEE and listed as an ISI Highly-cited Author.