Stability Analysis of Distributed Event-Based State Estimation

Sebastian Trimpe

***Abstract*— An approach for distributed and event-based state estimation that was proposed in previous work [1] is analyzed and extended to practical networked systems in this paper. Multiple sensor-actuator-agents observe a dynamic process, sporadically exchange their measurements over a broadcast network according to an event-based protocol, and estimate the process state from the received data. The event-based approach was shown in [1] to mimic a centralized Luenberger observer up to guaranteed bounds, under the assumption of identical estimates on all agents. This assumption, however, is unrealistic (it is violated by a single packet drop or slight numerical inaccuracy) and removed herein. By means of a simulation example, it is shown that non-identical estimates can actually destabilize the overall system. To achieve stability, the event-based communication scheme is supplemented by periodic (but infrequent) exchange of the agents' estimates and reset to their joint average. When the local estimates are used for feedback control, the stability guarantee for the estimation problem extends to the event-based control system.**

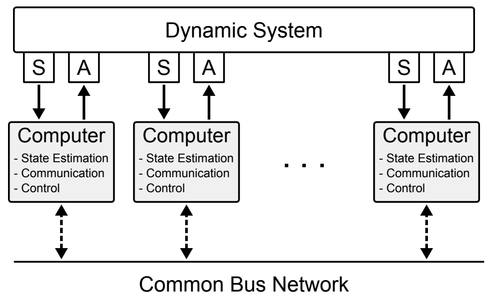

Fig. 1. Networked control system. Multiple control agents (each with an embedded computer, sensors (S) and actuators (A)) are distributed along a dynamic system. Each agent estimates the state of the dynamic system, computes local control inputs, and decides when to communicate with the other units over a common bus network.

I. INTRODUCTION

Event-based algorithms have recently received a lot of attention in the controls community (see recent overview articles [2]–[5]). With event-based methods, data is transmitted between the components of a control systems only when certain *events* indicate that new data is required, for example, to meet some control specification. This is in contrast to traditional control systems, where communication between sensors, actuators, and controllers usually occurs at predetermined, periodic time instants. Event-based strategies are especially attractive when many components are connected over a shared network, such as in networked control systems (NCSs) [6] or cyber-physical systems [7].

In this paper, we analyze and extend the distributed and event-based state estimation method that was proposed in [1] for NCSs such as in Fig. 1, where multiple sensor-actuator-agents exchange data over a common bus. In that approach, the event-based estimator consists of a switching Luenberger-type observer implemented on each agent of the NCS, in combination with a threshold-based event generation mechanism, which triggers when local sensor measurements are sent over the bus, see Fig. 2. The estimators are updated with data received from the bus, and since all agents have access to this data and run the same estimation algorithm, the agents' state estimates are basically the same. They capture the common information in the network, and their predictions are used by the event generator for the triggering decision: a measurement is broadcast only if the prediction of this measurement is not accurate enough. Accordingly, new measurement data is communicated *only when needed*.

Stability of this event-based estimation method in the sense of bounded estimation errors is proven in [1] under the assumption of (exactly) identical estimates on all agents. Clearly, this is an unrealistic assumption since even a single packet drop, different initial conditions, or slight differences in the numerical computations may cause some estimates to differ. Therefore, removing this assumption and establishing the stability of the event-based estimation scheme under non-ideal circumstances is essential for any practical implementation.

Herein, we study the case of non-identical estimates by introducing a (bounded) disturbance signal on each agent's estimate. The disturbances are independent of each other and may hence cause the agents' estimates to differ. Firstly, we show by means of a simulation example that inter-agent differences in the estimates can actually destabilize the system. Secondly, we propose a simple synchronous averaging mechanism to circumvent this, and establish stability of the inter-agent error and thus the overall estimation system (in the sense of bounded errors for bounded disturbances).

When the event-based state estimate on each agent is used for feedback control (as indicated in Fig. 2 in gray), one has a distributed event-based control system. This architecture was successfully used in [1] to balance a cube on one of its edges. We formally establish stability of the closed-loop control system in this paper by straightforward extension of the analysis for the estimation problem.

Related Work: The method proposed herein mainly falls into the category of *event-based state estimation* methods since triggering decisions are based on the estimation perfor-

S. Trimpe is with the Autonomous Motion Department at the Max Planck Institute for Intelligent Systems, 72076 Tübingen, Germany strimpe@tuebingen.mpg.de

This work was supported in part by the Max Planck Society and the Swiss National Science Foundation.

Accepted final version. To appear in *Proc. of the 53rd IEEE Conference on Decision and Control, 2014*.

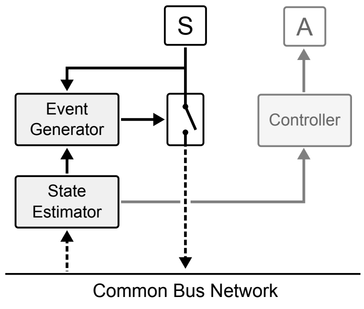

Fig. 2. Components of the event-based control system implemented on each agent of the NCS in Fig. 1. Event-triggered communication is indicated by dashed arrows, and periodic communication by solid ones. The event-based estimator (shown in black) consists of the *state estimator* and the *event generator*, which triggers the communication of local sensor measurements. The common bus ensures that all agents receive the same measurement data as inputs to the estimators and hence establishes consistency in the network. An event-based controller results when the local estimate is used for feedback control (gray). In this case, the control inputs are assumed to be shared between all agents (not shown) to be known to all estimators.

mance. Event-based estimation problems with a single sensor and estimator node have been considered in [8]–[14], for example. Distributed event-based estimation problems with an underlying broadcast communication architecture, such as the one considered herein, are also discussed in [15]–[17]. Related problems, where network communication is according to a graph topology, are treated in [18], [19].

When the state estimators are connected to state-feedback controllers as shown in Fig. 2, this structure represents an *event-based output-feedback control* system. Related problems on event-based control with output measurements (i.e. without full state measurements) have been considered, for example, in [20]–[24] for a single event-based control loop, and in [25]–[27] for a distributed setting.

Notation: A matrix is called stable if all of its eigenvalues have magnitude less than one. For an estimate of $x(k)$ computed from measurement data until time $\ell \leq k$, we write $\hat{x}(k|\ell)$. To simplify notation, we also write $\hat{x}(k)$ for $\hat{x}(k|k)$.

II. Preliminaries

In this section, we introduce the networked system and summarize a standard (centralized) observer-based control design, which serves as the basis for the event-based estimation and control methods of this paper.

A. Networked System

Consider the discrete-time, linear, time-invariant system

$$x(k) = A\,x(k-1) + B\,u(k-1) + v(k-1) \tag{1}$$
$$y(k) = C\,x(k) + w(k) \tag{2}$$

with time index k (corresponding to a sampling time T_s), state $x(k) \in \mathbb{R}^n$, control input $u(k) \in \mathbb{R}^q$, measurement $y(k) \in \mathbb{R}^p$, disturbances $v(k) \in \mathbb{R}^n$, $w(k) \in \mathbb{R}^p$, and all matrices of corresponding dimensions. We assume that (A, B) is stabilizable and (A, C) is detectable.

Let there be N agents, each of which measures a portion of $y(k)$ by its local sensors; that is,

$$\underbrace{\begin{bmatrix} y_1(k) \\ \vdots \\ y_N(k) \end{bmatrix}}_{y(k)} = \underbrace{\begin{bmatrix} C_1 \\ \vdots \\ C_N \end{bmatrix}}_{C} x(k) + \underbrace{\begin{bmatrix} w_1(k) \\ \vdots \\ w_N(k) \end{bmatrix}}_{w(k)} \tag{3}$$

with $y_j(k), w_j(k) \in \mathbb{R}^{p_j}$ and $p = \sum_{j=1}^N p_j$. Similarly, we consider the decomposition of the input vector

$$u^{\mathrm{T}}(k) = \begin{bmatrix} u_1^{\mathrm{T}}(k) & u_2^{\mathrm{T}}(k) & \cdots & u_N^{\mathrm{T}}(k) \end{bmatrix} \tag{4}$$

where $u_j(k) \in \mathbb{R}^{q_j}$ is the input associated with agent j's actuators, and $q = \sum_{j=1}^N q_j$. Without loss of generality, each agent j is assumed to have both sensors ($y_j(k)$) and actuators ($u_j(k)$) (if it has not, the corresponding dimension p_j or q_j may be set to zero). We do not make any assumption on stabilizability and detectability for the individual agents; that is, (A, B_j) may be not stabilizable, and (A, C_j) not detectable.

The agents are connected over a broadcast network such as in Fig. 1 and can exchange their measurements and inputs with each other. They are assumed to be synchronized in time (all have the same time index k). Network communication is assumed instantaneous without delay.

In this paper, we focus on the estimation problem and the reduction of measurement communications. We assume that the input $u(k)$ is known to all agents:

Assumption 1: All agents have access to the input u.

The assumption is satisfied, for example, when u is an external reference signal. When the components u_j are computed locally on different agents (such as in Sec. V where the local estimates are used for control), Assumption 1 requires that these inputs are communicated between all agents.

B. Centralized Observer-Based Control

A centralized control system (i.e. one that has periodic access to all measurements $y(k)$ and computes all inputs $u(k)$) can be designed as the combination of a linear state estimator and a state-feedback controller. Let

$$\hat{x}(k|k-1) = A\,\hat{x}(k-1|k-1) + B\,u(k-1) \tag{5}$$
$$\hat{x}(k|k) = \hat{x}(k|k-1) + L\left(y(k) - C\,\hat{x}(k|k-1)\right), \tag{6}$$

be the state estimator whose gain L is chosen such that $(I - LC)A$ is stable; and let

$$u(k) = F\hat{x}(k) \tag{7}$$

be the state-feedback controller with gain F such that $A + BF$ is stable (recall that $\hat{x}(k) = \hat{x}(k|k)$). The observer and controller gains L and F can be designed using standard state-space design methods (see e.g. [28]). Notice that (6) can be rewritten as

$$\hat{x}(k) = \hat{x}(k|k-1) + \sum_{i \in \{1,\ldots,N\}} L_i\big(y_i(k) - C_i\hat{x}(k|k-1)\big) \tag{8}$$

where

$$L = [L_1, L_2, \ldots, L_N], \quad L_i \in \mathbb{R}^{n \times p_i} \tag{9}$$

is the decomposition of the estimator gain according to the dimensions of the individual measurements $y_i(k)$.

It is straightforward to establish the stability of the centralized closed-loop control system given by (1), (2), (5), (6), and (7) from the stability of $(I-LC)A$ and $A+BF$.

III. Distributed & Event-Based Estimation

The distributed and event-based state estimation method from [1] is summarized in this section. It consists of an event generator and a state estimator implemented on each agent as shown in Fig. 2. To study stability for the case of non-identical estimates in later sections, we augment the framework from [1] by introducing disturbance signals on the individual estimates.

A. Event Generator

The event generator on agent j decides at every step k, whether or not the local measurement $y_j(k)$ is sent to all other agents. The following rule is used for making this transmit decision:

$$\text{transmit } y_j(k) \quad \Leftrightarrow \quad \|y_j(k) - C_j\hat{x}_j(k|k-1)\| \geq \delta_j \tag{10}$$

where $\delta_j \geq 0$ is a design parameter, $\hat{x}_j(k|k-1)$ is agent j's prediction of the state $x(k)$ based on measurements until time $k-1$ (to be defined in the next subsection), and $C_j\hat{x}_j(k|k-1)$ is agent j's prediction of its measurement $y_j(k)$. Hence, measurement $y_j(k)$ is transmitted if, and only if, its prediction from the previous estimate deviates by more than the tolerable threshold δ_j.

We denote the indices of those agents that transmit their measurement at time k by $I(k)$; that is,

$$I(k) := \big(i \mid 1 \leq i \leq N, \, \|y_i(k) - C_i\hat{x}_i(k|k-1)\| \geq \delta_i\big).$$

B. State Estimator

Agent j's state estimator recursively computes an estimate $\hat{x}_j(k)$ of the system state $x(k)$ from the measurements $I(k)$ transmitted at time k:

$$\hat{x}_j(k|k-1) = A\,\hat{x}_j(k-1|k-1) + B\,u(k-1) \tag{11}$$

$$\hat{x}_j(k|k) = \hat{x}_j(k|k-1) + \sum_{i\in I(k)} L_i\big(y_i(k) - C_i\hat{x}_j(k|k-1)\big) \tag{12}$$

where L_i is as defined in (9). That is, the estimator equations and the gains L_i are the same as for the centralized estimator (5), (8). The event-based estimator, however, updates its estimate with a subset of all measurements (compare summation in (12) and (8)).

In [1], it is assumed that all agents' estimates are identical ($\hat{x}^j(k) = \hat{x}^i(k)$ for all i, j, and k). Since this requires perfect communication and computation, it is an unrealistic assumption. To account for differences in the estimates from imperfect conditions, we introduce a disturbance signal d_j in each estimate and replace (12) with

$$\hat{x}_j(k|k) = \hat{x}_j(k|k-1) + \sum_{i\in I(k)} L_i\big(y_i(k) - C_i\hat{x}_j(k|k-1)\big) + d_j(k) \tag{13}$$

for the following analysis. The disturbances d_j are assumed bounded.

IV. Analysis

In contrast to the original approach in [1], the disturbance signals d_j cause the individual estimates to differ. Thus, the difference between any two agents' estimates must be taken into account in the stability analysis. The error dynamics for the modified scheme are derived in Sec. IV-A. In Sec. IV-B, we propose a synchronous averaging mechanism as an extension to [1], by means of which boundedness of the individual estimation errors is established in Sec. IV-C.

A. Estimation Error

Let $e_j(k) := x(k) - \hat{x}_j(k)$ be the estimation error of agent j, and let $\epsilon_{ji}(k) := \hat{x}_j(k) - \hat{x}_i(k)$ be the inter-agent error of agents j and i. For the estimation error, we obtain from (1), (3), (11), (13) and straightforward manipulation,

$$\begin{aligned}
e_j(k) &= Ae_j(k-1) + v(k-1) \\
&\quad - \sum_{i\in\{1,\ldots,N\}} L_i\big(y_i(k) - C_i\hat{x}_j(k|k-1)\big) \\
&\quad + \sum_{i\in\bar{I}(k)} L_i\big(y_i(k) - C_i\hat{x}_j(k|k-1)\big) - d_j(k) \quad (14)\\
&= Ae_j(k-1) + v(k-1) - L\big(y(k) - C\hat{x}_j(k|k-1)\big) \\
&\quad + \sum_{i\in\bar{I}(k)} L_i\big(y_i(k) - C_i\hat{x}_j(k|k-1)\big) - d_j(k) \quad (15)\\
&= (I-LC)Ae_j(k-1) + (I-LC)v(k-1) - Lw(k) \\
&\quad + \sum_{i\in\bar{I}(k)} L_i\big(y_i(k) - C_i\hat{x}_i(k|k-1)\big) \\
&\quad - \sum_{i\in\bar{I}(k)} L_iC_iA\epsilon_{ji}(k-1) - d_j(k) \quad (16)
\end{aligned}$$

where

$$\begin{aligned}
\bar{I}(k) &:= (1,\ldots,N) \setminus I(k) \\
&= \big(i \mid 1 \leq i \leq N, \, \|y_i(k) - C_i\hat{x}_i(k|k-1)\| < \delta_i\big) \quad (17)
\end{aligned}$$

is the set of measurements that are not transmitted at time k. Notice that the term $\sum_{i\in\bar{I}(k)} L_i(y_i(k) - C_i\hat{x}_i(k|k-1))$ in (16) is bounded because of (17). This observation is the key step in the stability proof in [1]. Indeed, under the assumption that all estimates are exactly identical, we have $\epsilon_{ji}(k) = 0$ and $d_j(k) = 0$ for all k, and stability of the estimation error follows from (16) with (17) and $(I-LC)A$ being stable. Here, we need to establish in addition that the inter-agent errors $\epsilon_{ji}(k)$ are bounded.

From (11) and (13), we obtain

$$\begin{aligned}
\epsilon_{ji}(k) &= A\epsilon_{ji}(k-1) - \sum_{\ell\in I(k)} L_\ell C_\ell A\epsilon_{ji}(k-1) + d_j(k) - d_i(k) \\
&= (I - L_{I(k)}C_{I(k)})A\epsilon_{ji}(k-1) + d_j(k) - d_i(k) \quad (18)
\end{aligned}$$

where $L_{I(k)}$ and $C_{I(k)}$ denote the matrices constructed from the corresponding submatrices L_ℓ and C_ℓ, $\ell \in I(k)$. Obviously, the inter-agent error $\epsilon_{ji}(k)$ is governed by the time-varying dynamics $(I - L_{I(k)}C_{I(k)})A$, and we cannot simply infer stability of the event-based estimation from stability of $(I-LC)A$ as in [1]. In Sec. VI, an example is presented where the inter-agent errors are unstable despite stability of $(I-LC)A$. Next, we present a straightforward extension to actually ensure boundedness of $\epsilon_{ji}(k)$.

B. Synchronous Averaging Mechanism

The inter-agent error $\epsilon_{ji}(k)$ is the difference between the state estimates by agent i and j. We therefore have full control over it: we can make it zero at any time by resetting the two agents' state estimates to the same value, for example, their average. Therefore, a straightforward strategy to guarantee bounded inter-agent errors is to periodically reset all agents' estimates to their joint average. Clearly, this strategy increases the communication load on the network. If, however, the disturbances d_j are small or only occur rarely, the required resetting period may be very large in comparison to the underlying sampling time T_s.

Let $\hat{x}_j(k-)$ and $\hat{x}_j(k+)$ denote agent j's estimate before and after resetting, and let $K \in \mathbb{N}$ be the resetting period. Each agent j implements the following synchronous averaging mechanism: for k a multiple of K,

$$\text{transmit } \hat{x}_j(k-); \text{ receive } \hat{x}_i(k-), i \in \{1,\dots,N\} \setminus \{j\};$$
$$\text{set } \hat{x}_j(k+) = \text{avg}(\hat{x}_i(k-)) = \frac{1}{N}\sum_{i=1}^{N} \hat{x}_i(k-) \tag{19}$$

where avg denotes the average over all N agents as shown.

We assume that the network capacity is such that the mutual exchange of the estimates can happen in one time step (as is the case for the system in [1]), and that no data is lost in the transfer (e.g. through appropriate low-level protocols using acknowledgments).

C. Stability of the Estimation Error

With the synchronous averaging mechanism, we can now establish boundedness of the estimation error:

Theorem 1: Assume that the disturbances v, w, and d_j are bounded and that $(I-LC)A$ is stable. Then, all estimation errors e_j resulting from the distributed event-based estimator (10), (11), (13) with synchronous averaging (19) are bounded for any initial conditions $\hat{x}_j(0)$ and $x(0)$.

For the proof and the later development, we define the following signals: the average estimate $\bar{x}(k) := \text{avg}(\hat{x}_j(k)) = \frac{1}{N}\sum_{j=1}^{N} \hat{x}_j(k)$, the average estimation error $\bar{e}(k) := x(k) - \bar{x}(k)$, and agent j's deviation from the average $\epsilon_j(k) := \bar{x}(k) - \hat{x}_j(k)$.

Proof: From the previous definitions, $e_j(k) = \bar{e}(k) + \epsilon_j(k)$, and we establish the claim by showing boundedness of $\epsilon_j(k)$ and $\bar{e}(k)$.

For the average estimate $\bar{x}(k)$, we have from (11), (13),

$$\bar{x}(k|k-1) = A\bar{x}(k-1|k-1) + Bu(k-1)$$
$$\bar{x}(k|k) = \bar{x}(k|k-1) + \sum_{i\in I(k)} L_i\big(y_i(k) - C_i\bar{x}(k|k-1)\big) + \bar{d}(k)$$

where $\bar{x}(k|k) = \bar{x}(k)$, $\bar{x}(k|k-1) := \text{avg}(\hat{x}_j(k|k-1))$, and $\bar{d}(k) := \text{avg}(d_j(k))$. The dynamics of the error $\epsilon_j(k) = \bar{x}(k) - \hat{x}_j(k)$ are therefore described by

$$\epsilon_j(k) = (I - L_{I(k)}C_{I(k)})A\epsilon_j(k-1) + \bar{d}(k) - d_j(k) \tag{20}$$
$$\epsilon_j(k+) = 0, \quad \text{for } k = \kappa K \text{ with some } \kappa \in \mathbb{N} \tag{21}$$

where (20) is obtained by direct calculation analogous to (18), and (21) follows from (19). That is, ϵ_j is periodically reset to 0 and evolves according to (20) in-between resetting instants. Since $\bar{d}$, d_j, and $(I-L_{I(k)}C_{I(k)})A$ are bounded, boundedness of ϵ_j for all $j \in \{1,\dots,N\}$ follows.

Since $\bar{e}(k) = \text{avg}(e_j(k))$, we obtain from (16)

$$\begin{aligned}\bar{e}(k) = (I-LC)A\bar{e}(k-1) &+ (I-LC)v(k-1) - Lw(k) \\ &+ \sum_{i\in \bar{I}(k)} L_i\big(y_i(k) - C_i\hat{x}_i(k|k-1)\big) \\ &- \sum_{i\in \bar{I}(k)} L_iC_iA\epsilon_i(k-1) - \bar{d}(k)\end{aligned} \tag{22}$$

where we used $\text{avg}(\epsilon_{ji}(k-1)) = \text{avg}_j(\hat{x}_j(k-1) - \hat{x}_i(k-1)) = \bar{x}(k-1) - \hat{x}_i(k-1) = \epsilon_i(k-1)$. Note that (22) fully describes the evolution of $\bar{e}$. In particular, the resetting (19) does not affect $\bar{e}$ since, at resetting instant $k = \kappa K$, $\bar{e}(k+) = x(k) - \frac{1}{N}\sum_{j=1}^{N}\hat{x}_j(k+) = x(k) - \frac{1}{N}\sum_{j=1}^{N}(\frac{1}{N}\sum_{\ell=1}^{N}\hat{x}_\ell(k-)) = x(k) - \frac{1}{N}\sum_{\ell=1}^{N}\hat{x}_\ell(k-) = \bar{e}(k-)$.

All input terms in (22) are bounded: v, w, and $\bar{d}$ by assumption; $\sum_{i\in\bar{I}(k)} L_i(y_i(k) - C_i\hat{x}_i(k|k-1))$ by the event-triggering mechanism (see (17)); and ϵ_i for all i by the previous argument. Since $\tilde{e}(k) = (I-LC)A\tilde{e}(k-1)$ is exponentially stable, it follows that $\bar{e}(k)$ is bounded for all k [29, Thm. 75, p. 218], which completes the proof. ■

V. Distributed & Event-Based Control

In this section, we analyze the stability of the distributed event-based control system that is obtained when the components $u_j(k)$ of the control vector (4) are computed locally by the agents from their state estimates $\hat{x}_j(k)$ and the state-feedback law (7).

Let $F^{\mathrm{T}} = [F_1^{\mathrm{T}} \dots F_N^{\mathrm{T}}]$, $F_j \in \mathbb{R}^{q_j\times n}$, be the decomposition of the state-feedback gain in (7) according to the input decomposition (4). Agent j implements

$$u_j(k) = F_j\,\hat{x}_j(k) \tag{23}$$

which can be rewritten as $u_j(k) = F_j\,(x(k) - \bar{e}(k) - \epsilon_j(k))$. From this and (1), it follows

$$\begin{aligned}x(k) &= (A+BF)x(k-1) + v(k-1) \\ &\quad - BF\bar{e}(k-1) - \sum_{j=1}^{N} B_jF_j\epsilon_j(k-1)\end{aligned} \tag{24}$$

where $B = [B_1 \dots B_N]$ with $B_j \in \mathbb{R}^{n\times q_j}$. Update equations for $\bar{e}(k)$ and $\epsilon_j(k)$ were derived in the proof of Theorem 1 in (22) and (20). The closed-loop system consisting of the plant (1), (3), the event-based estimators (10), (11), (13), (19), and the state-feedback controllers (23) is fully described by the state-space equations (24), (22), and (20). We thus have the following result:

Theorem 2: Assume that v, w, and d_j are bounded and that $(I-LC)A$ and $A+BF$ are stable. Then, the state $(x(k), \bar{e}(k), \epsilon_1(k), \dots, \epsilon_N(k))$ of the distributed event-based control system given by (1), (3), (10), (11), (13), (19), and (23) is bounded for all initial conditions $\hat{x}_j(0)$ and $x(0)$.

Proof: Theorem 1 establishes the boundedness of $\bar{e}(k)$, $\epsilon_1(k), \dots, \epsilon_N(k)$. The statement then follows directly from (24) and $A+BF$ being stable. ■

VI. Simulation Example

In this section, we present simulation results to highlight certain aspects of the analysis in previous sections. In particular, we simulate random packet drops causing the agents' estimates to differ. As simulation example, we consider an inverted pendulum being balanced by two sensor-actuator-agents. Matlab files to reproduce the simulations results of this section are available as supplementary material to this paper (contact the author or download from his web page).

A. Simulation Model

We consider the inverted pendulum system depicted in Fig. 3. The pendulum is to be stabilized about its upright position ($\phi = 0$) by appropriate motion of its two rotating "arms." The system can be regarded as an abstraction of the Balancing Cube [30], which was used as the experimental platform in [1]. The Balancing Cube uses six rotating arms to balance its cubic structure on any of its edges or corners. The arms represent the control units, which carry sensors and actuators, and communicate with each other over a common bus as in Fig. 1.

A state-space model of the system linearized about $\phi = \varphi_1 = \varphi_2 = 0$, which is used for state estimation, is given by

$$x(k) = A\,x(k-1) + B_1\,u(k-1) + B_2\,u(k-2) + v(k-1)$$
$$y(k) = C\,x(k) + w(k)$$

with (see supplementary Matlab files for other matrices)

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -0.0001 & -0.0001 & 1.0007 & 0.01 \\ -0.0151 & -0.0151 & 0.1492 & 1.0007 \end{bmatrix} \tag{25}$$

$$C = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{26}$$

state $x = (\varphi_1, \varphi_2, \phi, \dot{\phi})$, and sampling $T_s = 1/100\,\mathrm{s}$. The model includes the effect of local feedback on the arm velocities $\dot{\varphi}_1$ and $\dot{\varphi}_2$. The inputs $u \in \mathbb{R}^2$ are the reference velocities for these inner loops. The particular structure of the state equation, which also includes the input at time $k-2$, follows from a time-scale separation algorithm used to compute an approximate system model assuming sufficiently fast tracking for the velocity feedback loops (see [30], [31] for details). The state estimator equation (11) is modified accordingly to include the additional input.

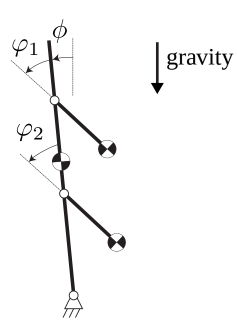

Fig. 3. Inverted pendulum system used for the simulations. The system (confined to the vertical plane) consists of three links: the pendulum, whose angle against the vertical is denoted by ϕ, and two "arms" rotating relative to the pendulum with angles φ_1 and φ_2.

From (25) and (26), it follows that (A, C_i) is *not* detectable for any i; that is, communication between the agents is *required* for stable state estimation.

The noise variables $v(k)$ and $w(k)$ are modeled as uniform random variables. The sensor noise intensity is chosen comparable to the experiment [1] (noise on angle sensors y_1, y_2 is negligible, noise on angular rate sensor y_3 is significant). To account for non-ideal actuation, we simulate input noise uniform in $[-0.05, 0.05]$ rad/s.

In order to study non-identical estimates, we simulate random packet drops: any measurement $y_1(k)$, $y_2(k)$, $y_3(k)$ that is transmitted between agent 1 and 2 is lost with a probability of 5%. Packet drops are represented by the disturbance $d_j(k)$ in (13) as follows: if $y_\ell(k)$, $\ell \in I(k)$ is a measurement not received at agent j, then $d_j(k) = -L_\ell\big(y_\ell(k) - C_\ell \hat{x}_j(k|k{-}1)\big)$ accounts for the lost packet (however, we do not establish boundedness of d_j in this case, see Sec. VII for further discussion). We assume that the communication required to perform the periodic averaging of estimates (19) is not affected by data loss.

B. Event-Based Estimation & Control Design

The event-based control system presented in Sec. V is applied to stabilize the pendulum about $x = 0$. The two control agents compute individual state estimates $\hat{x}_1(k)$ and $\hat{x}_2(k)$ according to (11), (13), and apply the control (23). The event trigger (10) is applied to each sensor measurement $y_1(k)$ to $y_3(k)$ individually. Agent 1 is responsible for measurement $y_1(k)$, and agent 2 decides for both $y_2(k)$ and $y_3(k)$ (i.e. $\hat{x}_3(k|k-1) = \hat{x}_2(k|k-1)$ in (10)). We chose $\delta_i = 0.005$ for all triggering thresholds, and $K = 200$ as period for the synchronous averaging (19) (reset every 2 s).

The centralized observer (5), (6) and the state-feedback controller (7), which form the basis of the event-based implementation, are designed from the linearized dynamics using standard techniques, such that $(I - LC)A$ and $A + BF$ are stable (see supplementary material for details).

C. Simulation Results: Effect of Synchronous Averaging

Figures 4 and 5 show system trajectories and communication rates for a typical simulation of the event-based control system. The communication rates are comparable to those observed in the experiments in [1]. At $t = 10\,\mathrm{s}$, an impulsive disturbance on u_1 is applied. As expected, the communication rate for the corresponding angle measurement y_1 goes up temporarily.

Figure 6 shows the inter-agent error ϵ_{21} for a different segment of the same simulation (in blue), as well as for a separate simulation, where synchronous resetting (19) is disabled (in green). Without resetting, the inter-agent errors diverge causing the control system to destabilize (the pendulum falls). This demonstrates that stability may be lost due to small deviations in the agents' estimates, which is neglected in the analysis in [1] by assuming identical estimates.

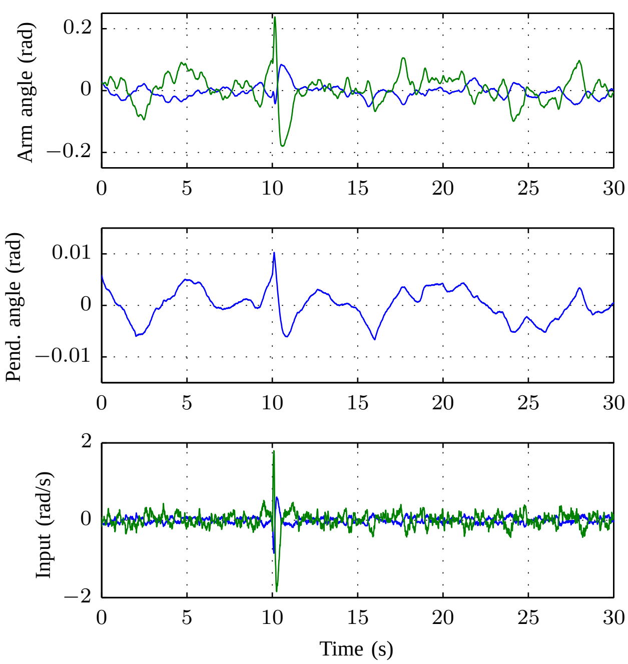

Fig. 4. Typical state and input trajectories for the simulation example. TOP: arm angles x_1 (blue) and x_2 (green). MIDDLE: pendulum angle x_3. BOTTOM: control inputs u_1 (blue) and u_2 (green). At $t = 10$s, an impulsive disturbance is applied on input u_1.

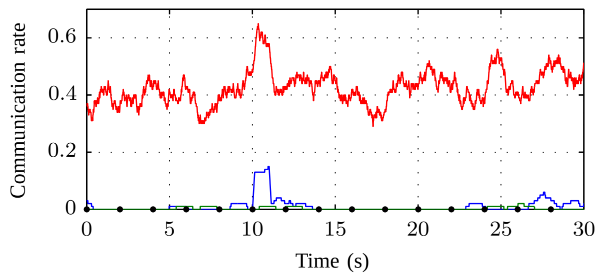

Fig. 5. Communication rates (moving average over 100 steps) for the same simulation experiment as in Fig. 4. The rates are shown in blue and green for the arm angle measurements y_1 and y_2, and in red for the pendulum angular rate measurement y_3. The black dots on the time axis indicate the instants when the estimates are reset according to (19).

D. Performance Comparison to Ideal Case

Table I shows root mean square (RMS) values of the individual estimation errors e_1 and e_2, as well as the inter-agent error ϵ_{21}. The values represent the average over 1000 simulation runs, with one run representing 300 seconds of balancing. In contrast to the simulation in Fig. 4, no impulsive disturbance was applied (but process and sensor noise were still active). The table shows the results for the simulation scenario discussed so far (with 5% packet drop probability), as well as the case of no packet drops. For the latter, $\hat{x}_1$ and $\hat{x}_2$ are identical and no synchronous averaging (19) is required, which corresponds to the analysis in [1].

Apparently, the RMS values of e_1 and e_2 are only slightly larger for the case with packet drops compared to the idealized case without data loss. Furthermore, the inter-agent error ϵ_{21} is relatively small. This indicates that analyzing the simpler case of identical estimates is helpful to approximate the estimation performance, while it is not sufficient to guarantee stability as discussed before.

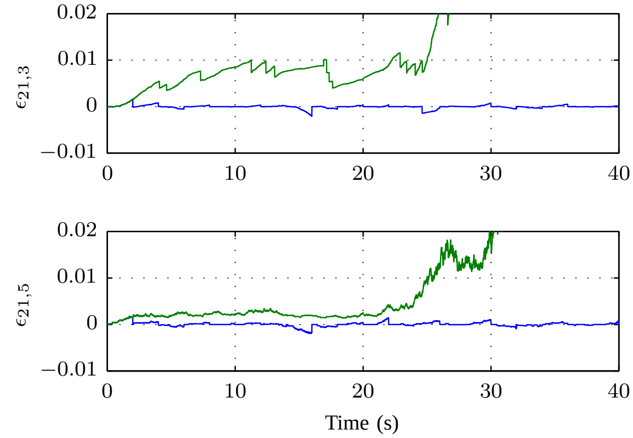

Fig. 6. Two components of the inter-agent error ϵ_{21} for simulation with synchronous averaging (blue) and without (green). The blue graph is reset to zero at $t = 2, 4, 6, \ldots$ according to (19). Without resetting, the inter-agent errors diverge and destabilize the system.

TABLE I

RMS ESTIMATION ERRORS AND COMMUNICATION RATES FOR SIMULATION WITH AND WITHOUT PACKET DROPS.

Simulation scenario Synchronous averaging (19)	5% packet drops yes	no packet drops no
RMS estimation error e_1	$5.53 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$
RMS estimation error e_2	$5.34 \cdot 10^{-3}$	$5.31 \cdot 10^{-3}$
RMS inter-agent error ϵ_{21}	$1.47 \cdot 10^{-3}$	0
Total communication rate $\mathcal{R}$	0.158	0.145

Table I also shows for both scenarios the total communication rate $\mathcal{R}$, which is defined as the number of communicated units (measurements) averaged over the duration of the experiment and the number of sensors. That is, $0 \leq \mathcal{R} \leq 1$, and $\mathcal{R} = 1$ means that all sensors communicate at every step. The slight increase in $\mathcal{R}$ for the packet drop case is due to the additional communication required for the synchronous averaging (19) (one element of $\hat{x}_j(k-)$ is counted as one measurement). Apart from this, the communication rates are comparable.

VII. CONCLUDING REMARKS

The key difference of this work compared to the event-based estimation framework presented in [1] is the removal of the assumption that all agents' estimates are identical. This assumption is not practical since it is violated, for example, as soon as a single packet is dropped or delayed, or if initial conditions vary slightly. The simulation example herein shows that stability may actually be lost due to such differences in the agents' estimates. In this paper, we established stability in the presence of inter-agent differences by means of the synchronous averaging mechanism (19); that is, by periodically (but infrequently) resetting all agents' estimates to their joint average.

As a result, there are two types of communication in this approach: event-based communication of measurements according to the event triggers (10), and periodic exchange of estimates for the synchronous averaging (19). Even though it would be possible to design a stable estimation scheme with the synchronous averaging mechanism alone, this typically requires periodic updates at a relatively high frequency, which would be against the event-based communication paradigm. Instead, the results herein suggest a different design philosophy: first, an event-based design is carried out making the idealizing assumption of zero inter-agent errors (i.e. according to [1]). In a second stage, a synchronous averaging mechanism is introduced in order to keep inter-agent errors small and guarantee stability under practical circumstances (proposed herein). In a scenario, where inter-agent differences occur infrequently, the synchronous resetting frequency can be small and most communication is due to the event-triggering. Moreover, one can expect the estimation performance to be comparable to the idealized design (see simulation example in Sec. VI, esp. Table I).

Stability is established for bounded disturbances d_j driving the inter-agent error (18). The assumption of bounded d_j seems realistic when representing errors from initialization or computational accuracy, for example. In the simulation experiment, the signals d_j are used to model random packet drops (cf. Sec. VI-A). Even though boundedness of d_j cannot be established in a deterministic sense for this case, the method still proves effective in stabilizing the inter-agent error in this example.

Alternative ways of stabilizing the inter-agent errors (18) than the synchronous averaging proposed herein, as well as avoiding the periodic communication of control inputs (Assumption 1), are some topics for future work.

Acknowledgments

The author would like to thank Michael Muehlebach and the reviewers for valuable comments on this work.

References

[1] S. Trimpe, "Event-based state estimation with switching static-gain observers," in *Proc. of the 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Sep. 2012, pp. 91–96.

[2] M. Lemmon, "Event-triggered feedback in control, estimation, and optimization," in *Networked Control Systems*, ser. Lecture Notes in Control and Information Sciences. Springer, 2010, vol. 406, pp. 293–358.

[3] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proc. of the IEEE 51st Conference on Decision and Control*, 2012, pp. 3270–3285.

[4] L. Grüne, S. Hirche, O. Junge, P. Koltai, D. Lehmann, J. Lunze, A. Molin, R. Sailer, M. Sigurani, C. Stöcker *et al.*, "Event-based control," in *Control Theory of Digitally Networked Dynamic Systems*, J. Lunze, Ed. Springer, 2014, pp. 169–261.

[5] C. G. Cassandras, "The event-driven paradigm for control, communication and optimization," *Journal of Control and Decision*, vol. 1, no. 1, pp. 3–17, 2014.

[6] 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.

[7] K.-D. Kim and P. Kumar, "Cyber-physical systems: A perspective at the centennial," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1287–1308, May 2012.

[8] O. Imer and T. Basar, "Optimal estimation with limited measurements," in *Proc. of the 44th IEEE Conference on Decision and Control and the European Control Conference*, Dec. 2005, pp. 1029–1034.

[9] L. Li, M. Lemmon, and X. Wang, "Event-triggered state estimation in vector linear processes," in *Proc. of the American Control Conference*, Jul. 2010, pp. 2138–2143.

[10] M. Rabi, G. Moustakides, and J. Baras, "Multiple sampling for estimation on a finite horizon," in *Proc. of the 45th IEEE Conference on Decision and Control*, Dec. 2006, pp. 1351–1357.

[11] Y. Xu and J. Hespanha, "Estimation under uncontrolled and controlled communications in networked control systems," in *Proc. of the 44th IEEE Conference on Decision and Control and the European Control Conference*, Seville, Spain, Dec. 2005, pp. 842–847.

[12] J. Sijs and M. Lazar, "On event based state estimation," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2009, pp. 336–350.

[13] J. Wu, Q.-S. Jia, K. Johansson, and L. Shi, "Event-based sensor data scheduling: Trade-off between communication rate and estimation quality," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 1041–1046, Apr. 2013.

[14] D. Shi, T. Chen, and L. Shi, "Event-based state estimation of linear dynamical systems: Communication rate analysis," in *American Control Conference*, Jun. 2014, pp. 4665–4670.

[15] J. Weimer, J. Araujo, and K. H. Johansson, "Distributed event-triggered estimation in networked systems," in *Proc. of the 4th IFAC conference on Analysis and Design of Hybrid Systems*, Jun. 2012, pp. 178–185.

[16] S. Trimpe and R. D'Andrea, "An experimental demonstration of a distributed and event-based state estimation algorithm," in *Proc. of the 18th IFAC World Congress*, Aug. 2011, pp. 8811–8818.

[17] ——, "Event-based state estimation with variance-based triggering," *IEEE Transaction on Automatic Control, Special Issue on Control of Cyber-Physical Systems*, 2014, to appear.

[18] P. Millan, U. Tiberi, C. Fischione, D. Dimarogonas, K. H. Johansson, and F. Rubio, "Distributed event-based observers for LTI networked systems," in *Portuguese Conf. on Automatic Control*, Jul. 2012.

[19] L. Yan, X. Zhang, Z. Zhang, and Y. Yang, "Distributed state estimation in sensor networks with event-triggered communication," *Nonlinear Dynamics*, vol. 76, pp. 169–181, 2014.

[20] A. Molin and S. Hirche, "Structural characterization of optimal event-based controllers for linear stochastic systems," in *49th IEEE Conference on Decision and Control*, Dec. 2010, pp. 3227–3233.

[21] ——, "On the optimality of certainty equivalence for event-triggered control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 470–474, Feb. 2013.

[22] L. Li and M. Lemmon, "Weakly coupled event triggered output feedback control in wireless networked control systems," in *Proc. of the 49th annual Allerton Conference on Communication, Control, and Computing*, Sep. 2011.

[23] D. Lehmann and J. Lunze, "Event-based output-feedback control," in *19th Mediterranean Conference on Control Automation*, Jun. 2011, pp. 982–987.

[24] D. P. Borgers and W. P. M. H. Heemels, "Event-separation properties of event-triggered control systems," *IEEE Transactions on Automatic Control*, 2014, to appear.

[25] M. C. F. Donkers and W. P. M. H. Heemels, "Output-based event-triggered control with guaranteed L_∞-gain and improved and decentralized event-triggering," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1362–1376, Jun. 2012.

[26] C. Stoecker and J. Lunze, "Event-based control with incomplete state measurement and guaranteed performance," in *Proc. of the 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Sep. 2012, pp. 49–54.

[27] W. P. M. H. Heemels and M. C. F. Donkers, "Model-based periodic event-triggered control for linear systems," *Automatica*, vol. 49, no. 3, pp. 698–711, 2013.

[28] K. Åström and B. Wittenmark, *Computer-controlled systems: theory and design*. Prentice Hall, 1997.

[29] F. M. Callier and C. A. Desoer, *Linear System Theory*. Springer-Verlag, 1991.

[30] S. Trimpe and R. D'Andrea, "The Balancing Cube: A dynamic sculpture as test bed for distributed estimation and control," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 48–75, Dec. 2012.

[31] ——, "A limiting property of the matrix exponential with application to multi-loop control," in *Proc. of the Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, Dec. 2009, pp. 6419–6425.