A Decoupled Design in Distributed Control of Uncertain Networked Control Systems

Xiaofeng Wang and Naira Hovakimyan

***Abstract*— This paper studies interconnected networked control systems in the presence of communication constraints and uncertainties. We propose a novel control architecture for subsystems that can decouple the design of output-feedback control strategy and the communication scheme. In this architecture, a navigator serves as the bridge between the uncertain, communication-limited system and an ideal model that has perfect communication and no uncertainties. Under this framework, one can integrate the existing communication protocols and robust control techniques into this architecture without worrying for the coupling between the control and the communication. We provide stability condition for the resulting closed-loop system and derive bounds on the error signals between the real system, the navigator and the ideal model. It is shown that these bounds can be arbitrarily reduced by improving the communication condition and tuning the low-pass filter in the controller. The results can significantly improve the predictability of the behavior of uncertain systems, which is important for safety-critical applications.**

I. Introduction

As the scales of control systems become larger and larger, distributed control gets a lot of attention these days for its ability of reducing the complexity in controller synthesis and communication. A distributed control system consists of numerous subsystems, which are controlled by local controllers. Information is exchanged among subsystems through communication networks, based on which the control decisions are made for certain global objectives [1], [2]. These systems are ubiquitous throughout our national infrastructure, such as transportation systems, air traffic systems, power systems, medical systems, to name a few. For the case that subsystems are not physically interconnected, one can refer to cooperative control. Specific cooperative control objectives include consensus [3], flocking [4], formation [5], maximal coverage [6], and so on. A more complete survey on cooperative control can be found in [7].

Most of the distributed control or cooperative control approaches assume that the communication is perfect (or continuous) and the system dynamics are completely known. These assumptions, however, are not realistic in many applications. First, in practice the communication happens in a discrete manner, but not continuous, and the limited network capacity may cause delays in message delivery. Second, modeling errors, uncertainties, and disturbances always exist when using mathematical models to describe the physical process. These factors may lead to a major impact on the overall system performance and cause failure in fulfilling the desired objectives. To address the communication issue in distributed control/cooperative control, time-triggering models and distributed event-triggering models were introduced for different control objectives, such as stabilization [8], consensus [9], coverage [6], optimization [10], to name a few. To handle the uncertainties, distributed control synthesis scheme was provided in [11] for $\mathcal{L}_p$ stability. However, this body of work addresses only one of these two issues. Less work has been done to simultaneously tackle both issues, more importantly, in a distributed manner.

A related work was done in [12], which studied networked control systems (NCSs) over control area network (CAN) buses and derived the *maximum allowable transfer interval* (MATI) between two subsequent message transmissions that ensures input-output stability. However, this is a centralized approach and the periodic nature of the MATI makes MATI conservative in the sense of being shorter than necessary. Realizing these facts, a distributed event-triggering scheme was proposed in [13] to ensure input-to-state stability (ISS). In some applications, especially safety-critical applications, however, ISS may not be enough, because it can only provide an upper bound on the state trajectory, without providing an opportunity to quantify the transient behavior in the sense of performance specifications. To guarantee the system's transient performance, an $\mathcal{L}_1$-adaptation-based distributed event-triggering scheme was developed [14]. It considers state-feedback and provides a distributed communication and control co-design method, where the communication scheme is strongly coupled with the controller design. The real state information is transmitted among subsystems over the network, through which the errors due to the uncertainties can be propagated into the network and affect the system performance.

This paper extends the results from [14] to output-feedback systems. More importantly, by introducing a local navigator that does not contain uncertainties, we propose a control architecture that enables to decouple the design of robust controller and the communication protocl, even in the presence of physical interconnections. With this architecture, one can integrate the existing communication and robust control techniques into the system without worrying about the coupling between the control loop and the communication protocol.

Although all the triggering models will work under this framework, we choose distributed event-triggering as a tool to determine the broadcast time for its ability of saving communication resources. Each agent broadcasts the output

Both authors are with the Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801; e-mail: {wangx, nhovakim}@illinois.edu. Research is supported by AFOSR and ARO grants.

information of its navigator, but not the real data as it is in [14]. It avoids the propagation of the error due to uncertainties into the network. We limit the discussion to a simple class of systems, for which the disturbance observer (DOB) controller is appropriate for compensation of uncertainties [15]. The simple structure of DOB helps to capture the main spirit of the *decoupling idea* without being distracted by the complexity of the controller structure. It is worth mentioning that the decoupling architecture in this paper can be applied to more complex systems, including systems whose behavior is highly nonlinear, which cannot be addressed by DOB controllers. In that case, $\mathcal{L}_1$ adaptive controller can be used in this decoupled architecture for compensation of uncertainties, along with the same communication protocol, to achieve the desired nonlinear behavior for the closed-loop systems. The relationship between the disturbance observers and the $\mathcal{L}_1$ adaptive controllers is clarified in [16].

We derive the stability condition for the resulting closed-loop system. Moreover, we show that the performance of the real-system can be rendered arbitrarily close to the navigator, in the presence of uncertainties, by tuning the bandwidth of the low-pass filter in the local robust controller. Also, the navigator can be arbitrarily close to an ideal mathematical model, given enough communication resources. Simulations verify the theoretical results.

The paper is organized as follows. Section II formulates the problem. The control architecture is introduced in Section III. The distributed robust controller is presented in Section IV and the triggering event design is discussed in Section V. Simulation results are presented in Section VI. Section VII concludes the paper.

II. Problem Formulation

Notation: We denote by $\mathbb{R}^n$ the n-dimensional real vector space, by $\mathbb{R}^+$ the set of real positive numbers, and $\mathbb{R}_0^+ = \mathbb{R}^+ \cup \{0\}$. We use $\|\cdot\|_\infty$ to denote ∞-norm of a vector and $\|\cdot\|_{\mathcal{L}_1}$, $\|\cdot\|_{\mathcal{L}_\infty}$ are the $\mathcal{L}_1$ and $\mathcal{L}_\infty$ norms of a function, respectively. The truncated $\mathcal{L}_\infty$ norm of a function $x : [0,\infty) \to \mathbb{R}^n$ is defined as $\|x\|_{\mathcal{L}_\infty^{[0,\tau]}} \triangleq \sup_{0\le t\le \tau} \|x(t)\|_\infty$. Given a function $x(t)$, let $x(s) = [x(t)]$ denote the Laplace transform of $x(t)$. Given a set $\mathcal{N}$ with finite number of elements, we use $|\mathcal{N}|$ to denote its cardinality. Given a collection of N vectors $\{x_i\}_{i=1}^N$, let $x \triangleq (x_1^\top, \cdots, x_N^\top)^\top$.

Consider a system consisting of N subsystems (or called "agents"). Let $\mathcal{N} = \{1, \cdots, N\}$. For agent i, we use $\mathcal{N}_i \subseteq \mathcal{N}$ to denote its neighboring set, not including i and $\bar{\mathcal{N}}_i = \mathcal{N}_i \cup \{i\}$. In our framework, agent i can broadcast/receive information to/from the agents in the set $\mathcal{N}_i$.

Ideally, we can use a mathematical model to describe agent i's dynamics (also called "ideal model"):

$$\begin{aligned} \dot{x}_i^{\rm id}(t) &= A_i^m x_i^{\rm id}(t) + B_i u_i^{\rm id}(t) + f_i\left(t, y_i^{\rm id}(t), y_{-i}^{\rm id}(t)\right) \\ y_i^{\rm id}(t) &= C_i x_i^{\rm id}(t) \\ x_i^{\rm id}(0) &= x_i^0, \end{aligned} \tag{1}$$

where $x_{\rm id} : \mathbb{R}_0^+ \to \mathbb{R}^n$ is the state of the ideal model, $u_i^{\rm id} : \mathbb{R}^+0 \to \mathbb{R}^m$ is the ideal control input, $y_i^{\rm id} : \mathbb{R}_0^+ \to \mathbb{R}^m$ is the ideal output, $y_{-i}^{\rm id} : \mathbb{R}_0^+ \to \mathbb{R}^{|\mathcal{N}_i|m}$ is the outputs of agent i's neighbors with $y_{-i}^{\rm id}(t) = \left\{y_j^{\rm id}(t)\right\}_{j\in\mathcal{N}_i}$, $x_i^0 \in \mathbb{R}^n$ is the initial condition, $A_i^m \in \mathbb{R}^{n\times n}, B_i \in \mathbb{R}^{n\times m}, C_i \in \mathbb{R}^{m\times n}$ are known matrices with appropriate dimensions, A_i^m is Hurwitz, and $f_i : \mathbb{R}_0^+ \times \mathbb{R}^m \times \mathbb{R}^{|\mathcal{N}_i|m} \to \mathbb{R}^n$ is a known, locally Lipschitz, continuous function describing the interconnections. Assume that B_i has full column rank, i.e. $\text{col rank}(B_i) = m$, and $C_i(s\mathbb{I} - A_i^m)^{-1}B_i$ is invertible and minimal phase.

Remark 2.1: The results in this paper can be applied to subsystems with different dimensions. To simplify the notations and make the paper easier to read, we assume that the dimensions of subsystems are the same. Note that we also assume that the dimension of the output is equal to the dimension of the input. In case of different dimensions, e.g. $dim(y) = l$ and $dim(u) = m$, one needs to assume that $C_i(s\mathbb{I} - A_i^m)^{-1}B_i$ is minimum phase and has full column rank. In that case one can always find a matrix $\bar{C}_i \in \mathbb{R}^{m\times l}$ such that $\bar{C}_i C_i(s\mathbb{I} - A_i^m)^{-1}B_i$ is minimum phase and invertible.

With this mathematical model in equation (1), different distributed control algorithms have been developed for different control purposes, such as stabilization [1], tracking, consensus [3], flocking [4], formation [5], maximal coverage [6], to name a few. Without loss of the generality, those control algorithms can be unified by the following structure:

$$\begin{aligned} \dot{z}_i^{\rm id}(t) &= g_i\left(t, z_i^{\rm id}(t), y_i^{\rm id}(t), y_{-i}^{\rm id}(t)\right) \\ u_i^{\rm id}(t) &= h_i\left(t, z_i^{\rm id}(t), y_i^{\rm id}(t), y_{-i}^{\rm id}(t)\right) \end{aligned} \tag{2}$$

where $z^{\rm id} : \mathbb{R}_0^+ \to \mathbb{R}^p$ is the state of the ideal controller and $g_i : \mathbb{R}_0^+ \times \mathbb{R}^p \times \mathbb{R}^m \times \mathbb{R}^{|\mathcal{N}_i|m} \to \mathbb{R}^p$, $h_i : \mathbb{R}_0^+ \times \mathbb{R}^p \times \mathbb{R}^m \times \mathbb{R}^{|\mathcal{N}_i|m} \to \mathbb{R}^m$ are known, locally Lipschitz functions that describe the control algorithm for a specific control objective denoted by $\mathcal{O}_{\mathbf{1}}$.

Example 1: Although our approach is for output-feedback, to better illustrate the idea, we consider a simple state-feedback consensus algorithm as an example, which can be written in the form of equation (1) with the controller in equation (2):

$$\begin{aligned} \dot{x}_i^{\rm id} &= -x_i^{\rm id} + u_i^{\rm id} + x_i^{\rm id} \\ y_i^{\rm id} &= x_i^{\rm id} \\ \dot{z}_i^{\rm id} &= 0 \\ u_i^{\rm id} &= \sum_{j\in\mathcal{N}_i} (x_j^{\rm id} - x_i^{\rm id}). \end{aligned}$$

If the mathematical model in equation (1) was perfect and the controller in equation (2) could be precisely executed in the real system, we know that the control objective $\mathcal{O}_{\mathbf{1}}$ could be fulfilled. However, in practice, there always exist modeling errors, exogenous disturbances as well as communication limitations in the system, which raise the control challenges. Let us assume that the actual state equation of agent i is given by:

$$\begin{aligned} \dot{x}_i(t) &= A_i x_i(t) + B_i u_i(t) + \Delta_i(t, y(t)) \\ y_i(t) &= C_i x_i(t) \\ x_i(0) &= x_i^0, \end{aligned} \tag{3}$$

where $A_i \in \mathbb{R}^{n\times n}$ is a known matrix, $y = \left(y_1^\top, \cdots, y_N^\top\right)^\top$ is the overall state of the real system, and $\Delta_i : \mathbb{R}_0^+ \times \mathbb{R}^{|\mathcal{N}|m} \to \mathbb{R}^n$ is an unknown, locally Lipschitz, continuous function describing the uncertainties and disturbances in the dynamics and the interconnections associated with agent i. Assume that there exists $L_i \in \mathbb{R}^{n\times m}$, such that $A_i^m = A_i + L_i C_i$. This can be achieved by assuming (A_i, C_i) is observable. Also assume that given a compact set, the Lipschitz constant of Δ_i is known and denoted by L_{Δ_i}. Note that this uncertainty Δ_i might be significantly different from the modeled interconnection term f_i in the ideal model. Consequently, the ideal controller in equation (2) might drive the real system deviating from the desired control objective.

Another challenge in controlling the real system is related to the communication limitations. As we see in equation (2), to accomplish the objective $\mathcal{O}_1$, information from agent i's neighbors is needed. In real-time communication networks, the data transmission is discrete, but not continuous as it is assumed in equation (2). If agents transmit wrong data at the wrong time, the system performance will be largely degraded, or, even worse, the overall system might become unstable.

In this paper, we try to control the system in equation (3) to fulfill the objective $\mathcal{O}_1$, in the presence of uncertainties and communication limitations. The problem can be formally stated as follows:

Problem 1: Assume that the ideal model in equation (1) with the ideal controller in equation (2) can fulfill $\mathcal{O}_1$. To ensure that the system in equation (3) also fulfills $\mathcal{O}_1$, we ask the following questions:

- what is the distributed output-feedback control architecture for agent i?
- what information should be transmitted among agents?
- when should agent i transmit data to its neighbors?

III. Local Control and Communication Architecture

To fulfill the objective $\mathcal{O}_1$, we need to deal with the robust control design and the communication design simultaneously. This section introduces an architecture for agents that enables us to decouple the design of the communication scheme and the local control strategy. It is shown in Figure 1.

In this architecture, agent i's controller consists of two parts: the navigator and the control generator. The navigator generates the signals for the real system to follow. It also takes care of broadcasting/receiving data to/from agent i's neighbors. Note that there is no uncertainty in the navigator. The control generator is to drive the real system to follow the navigator's trajectory, in the presence of uncertainties. It will not be involved in the decision making process of when to broadcast.

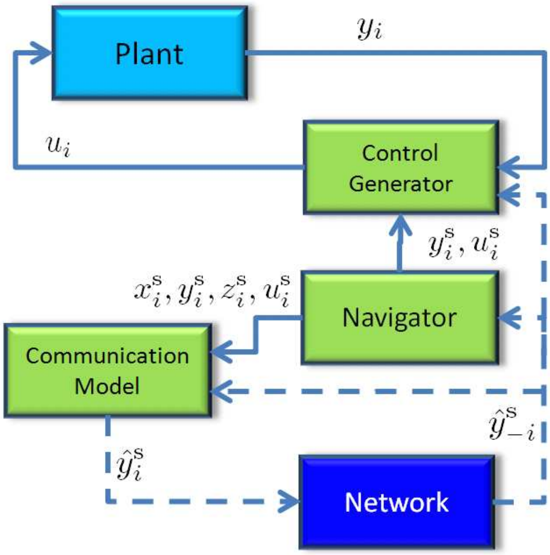

Fig. 1. A local control and communication architecture

The navigator has the following dynamics:

$$
\begin{aligned}
\dot{x}_i^s(t) &= A_i^m x_i^s(t) + B_i u_i^s(t) + f_i\left(t, y_i^s(t), \hat{y}_{-i}^s(t)\right)\\
y_i^s(t) &= C_i x_i^s(t)\\
x_i^s(0) &= x_i^0\\
\dot{z}_i^s(t) &= g_i\left(t, z_i^s(t), y_i^s(t), \hat{y}_{-i}^s(t)\right)\\
u_i^s(t) &= h_i\left(t, z_i^s(t), y_i^s(t), \hat{y}_{-i}^s(t)\right)
\end{aligned}
\tag{4}
$$

where $\hat{y}_i^s(t)$ is the broadcast data of agent i at time t and $\hat{y}_{-i}^s(t) = \{\hat{y}_j^s(t)\}_{j\in\mathcal{N}_i}$. The broadcast data $\hat{y}_i^s(t)$ is the sampled output signal of agent i's navigator. Let $t_i^k \in \mathbb{R}_0^+$ be the time instant when agent i releases its kth broadcast. At each t_i^k, the data $y_i^s(t_i^k)$ will be broadcast to agent i's neighbors in the set $\mathcal{N}_i$. If the transmission delays are neglectable, then $\hat{y}_i^s(t) = y_i^s(t_i^k)$ for any $t \in \left[t_i^k, t_i^{k+1}\right)$, which is piecewise constant. Otherwise, if the delays in the kth broadcast of agent i are D_i^k, then $\hat{y}_i^s(t) = y_i^s(t_i^k)$ for any $t \in \left[t_i^k + D_i^k, t_i^{k+1} + D_i^{k+1}\right)$, provided that $t_i^k + D_i^k < t_i^{k+1} + D_i^{k+1}$.

The communication model in this framework can be either time-triggered or event-triggered. It will not affect the decoupling nature in this work. Without loss of the generality, we can define the broadcast time t_i^{k+1} in an event-triggering way since time-triggering can be viewed as a special case of event-triggering:

$$
t_i^{k+1} = \min_{t > t_i^k} \left\{ t \,\middle|\, \phi\left(t, t_i^k, x_i^s(t), y_i^s(t), z_i^s(t), u_i^s(t), \hat{y}_{-i}^s(t)\right) = 0 \right\}, \tag{5}
$$

where $\phi : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \times \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R}^m \times \mathbb{R}^{|\mathcal{N}_i|m} \to \mathbb{R}_0^+$ is the event-triggering function. The function ϕ must satisfy $\phi\left(t, t_i^k, x_i^s(t), y_i^s(t), z_i^s(t), u_i^s(t), \hat{y}_{-i}^s(t)\right) < 0$ when $t = t_i^k$, which ensures that $t_i^{k+1} > t_i^k$ always holds. For time-trigging model with a period T, the function ϕ is $t - (t_i^k + T)$, which is independent of the states and outputs of the navigator.

Note that in this architecture, t_i^{k+1} is determined only based on the information of the navigator. The real output of agent i will not affect the decision-making process. It provides the possibility of decoupling the control and communication designs. The navigator serves as a bridge between the ideal model in equation (1) and the real system in equation (3), which is the key for decoupling. Between the ideal model and the navigator, it is only the communication that makes the difference. It is obvious that if the communication was perfect, then $\hat{y}_i^{\mathrm{s}}(t)$ would be exactly the same as $y_i^{\mathrm{s}}(t)$ and the navigator would be the same as the ideal model. As to the relation between the navigator and the real system, we need to design the control generator to overcome the uncertainties and enforce the real system to follow the navigator. The inputs of the control generator are $y_i(t)$, $y_i^{\mathrm{s}}(t)$, and $\hat{y}_{-i}^{\mathrm{s}}(t)$, and the output is the control signal $u_i(t)$ to be actuated. Later we will show that the design of control generator can be completely independent of the communication model.

Remark 3.1: One of the advantages of using this architecture is that it transmits the "correct" information among agents. By "correct", it means that the transmitted data $\hat{y}_i^{\mathrm{s}}(t)$ is not affected by uncertainties. The impreciseness of the information over the network only lies in the communication constraints, which is unavoidable when using real-time networks. On the contrary, if the agents directly transmit their real outputs, the error due to the system uncertainties will be propagated into the network, and the system performance may be degraded. Under the current architecture, even if one subsystem temporarily deviates from its expected trajectory, its neighbors will still receive the right information for their navigation.

IV. Control Generator Design

This section studies how to design the control generator that enforces the real dynamics of agent i to follow its navigator. The difficulty lies in the uncertainty $\Delta_i(t,x)$ in the real system (3), which may drive agent i in an unpredictable way. We will discuss how to compensate for such uncertainty in agent i.

Let us split the control input $u_i(t)$ into two terms: the nominal input $v_i^{\mathrm{n}}(t)$ and the compensation input $v_i^{\mathrm{c}}(t)$, where $u_i(t) = v_i^{\mathrm{n}}(t) + v_i^{\mathrm{c}}(t)$. We can rewrite agent i's dynamics in equation (3) as

$$\begin{aligned} \dot{x}_i(t) &= A_i^m x_i(t) + B_i\left(v_i^{\mathrm{n}}(t) + v_i^{\mathrm{c}}(t)\right) \\ &\quad + \bar{\Delta}_i(t,y(t)) + f_i\left(t, y_i^{\mathrm{s}}(t), \hat{y}_{-i}^{\mathrm{s}}(t)\right) \\ y_i(t) &= C_i x_i(t) \\ x_i(0) &= x_i^0, \end{aligned} \tag{6}$$

where

$$\bar{\Delta}_i(t,y(t)) = \Delta_i(t,y(t)) - L_i y_i(t) - f_i\left(t, y_i^{\mathrm{s}}(t), \hat{y}_{-i}^{\mathrm{s}}(t)\right). \tag{7}$$

Recall that $A_i^m = A_i + L_i C_i$.

In the following discussion, we treat $\bar{\Delta}_i(t,y(t))$ as the uncertainty. Transforming this state equation into frequency domain yields

$$y_i(s) = \bar{H}_i(s)(v_i^{\mathrm{n}}(s) + v_i^{\mathrm{c}}(s)) + \hat{H}_i(s)(\bar{\Delta}_i(s) + f_i(s) + x_i^0) \tag{8}$$

where $v_i^{\mathrm{n}}(s) = \mathfrak{L}\left[v_i^{\mathrm{n}}(t)\right]$, $v_i^{\mathrm{c}}(s) = \mathfrak{L}\left[v_i^{\mathrm{c}}(t)\right]$, $\bar{\Delta}_i(s) = \mathfrak{L}[\bar{\Delta}_i(t,y(t))]$, $f_i(s) = \mathfrak{L}\left[f_i\left(t, y_i^{\mathrm{s}}(t), \hat{y}_{-i}^{\mathrm{s}}(t)\right)\right]$, and

$$\begin{aligned} \bar{H}_i(s) &= C_i(s\mathbb{I} - A_i^m)^{-1}B_i, \\ \hat{H}_i(s) &= C_i(s\mathbb{I} - A_i^m)^{-1}. \end{aligned}$$

Ideally, if we could set

$$v_i^{\mathrm{c}}(s) = -\bar{H}_i(s)^{-1}\hat{H}_i(s)\bar{\Delta}_i(s),$$

the uncertainty $\bar{\Delta}_i$ could be completely canceled in equation (8). However, this is not realistic since $\bar{\Delta}_i$ is unknown. Instead, we consider partial cancelation of the uncertainty. Since $\bar{H}_i(s)$ is invertible, we can rewrite equation (8) as

$$\begin{aligned} \bar{H}_i(s)^{-1}\hat{H}_i(s)\bar{\Delta}_i(s) &= \bar{H}_i(s)^{-1}y_i(s) - (v_i^{\mathrm{n}}(s) + v_i^{\mathrm{c}}(s)) \\ &\quad - \bar{H}_i(s)^{-1}\hat{H}_i(s)(f_i(s) + x_i^0). \end{aligned}$$

Letting the signals at both sides go through a low-pass filter $F_i(s)$, we have

$$\begin{aligned} & F_i(s)\bar{H}_i(s)^{-1}\hat{H}_i(s)\bar{\Delta}_i(s) \\ = \; & F_i(s)\bar{H}_i(s)^{-1}y_i(s) - F_i(s)(v_i^{\mathrm{n}}(s) + v_i^{\mathrm{c}}(s)) \\ & -F_i(s)\bar{H}_i(s)^{-1}\hat{H}_i(s)(f_i(s) + x_i^0). \end{aligned} \tag{9}$$

As we mentioned in context, we are interested in partially canceling the uncertainty. To be more accurate, we would like to cancel the low-frequency part of the uncertainty, which means

$$v_i^{\mathrm{c}}(s) = -F_i(s)\bar{H}_i(s)^{-1}\hat{H}_i(s)\bar{\Delta}_i(s). \tag{10}$$

Applying equation (9) into the equation above, we obtain

$$\begin{aligned} v_i^{\mathrm{c}}(s) &= -F_i(s)\bar{H}_i(s)^{-1}y_i(s) + F_i(s)(v_i^{\mathrm{n}}(s) + v_i^{\mathrm{c}}(s)) \\ &\quad + F_i(s)\bar{H}_i(s)^{-1}\hat{H}_i(s)(f_i(s) + x_i^0), \end{aligned}$$

which implies

$$v_i^{\mathrm{c}}(s) = \frac{F_i(s)\left(-\bar{H}_i(s)^{-1}y_i(s) + v_i^{\mathrm{n}}(s) + \bar{H}_i(s)^{-1}\hat{H}_i(s)(f_i(s) + x_i^0)\right)}{1 - F_i(s)}. \tag{11}$$

Meanwhile, letting the nominal input be

$$v_i^{\mathrm{n}}(t) = u_i^{\mathrm{s}}(t), \tag{12}$$

we have the control input

$$u_i(t) = v_i^{\mathrm{n}}(t) + v_i^{\mathrm{c}}(t). \tag{13}$$

Remark 4.1: The low-pass filter in this structure plays a very important role. Without $F_i(s)$, the term at the right side of equation (11) is non-casual, which is then not implementable. The introduction of $F_i(s)$ in fact adopts the spirit of the disturbance-observer (DOB) design [15]. It can be viewed as a local networked version of DOB controller.

Example 2: To better understand the control generator, let us re-visit the consensus algorithm. Assume that the real dynamics of agent i contains uncertainty $\Delta_i(x)$. Then the system dynamics are

$$\dot{x}_i = -x_i + u_i + \underbrace{\Delta_i(x) + x_i - x_i^{\mathrm{s}}}_{\bar{\Delta}_i} + \underbrace{x_i^{\mathrm{s}}}_{f_i}. \tag{14}$$

Let the low-pass filter be $F_i(s) = \frac{\omega}{s+\omega}$. Then the control input is

$$u_i(s) = v_i^{\rm n}(s) + \frac{\omega(s+1)x_i(s) + \omega v_i^{\rm n}(s) + \omega(x_i^{\rm s}(s) + x_i^0)}{s}$$

with $v_i^{\rm n}(t) = \sum_{i\in\mathcal{N}_i}(\hat{x}_j^{\rm s} - \hat{x}_i^{\rm s})$.

The relation between the navigator and the real dynamics is given by the next proposition.

Proposition 4.1: Consider the navigator in equation (4) and the real dynamics in equation (6) with the controller defined in equations (11) – (13). Assume that the navigator inputs and the navigator outputs of all agents are bounded, i.e. there exist $\rho_i^{u^s}$, $\rho_j^{y^s} \in \mathbb{R}_0^+$ such that for any $i \in \mathcal{N}_i$,

$$\|u_i^{\rm s}\|_{\mathcal{L}_\infty} \le \rho_i^{u^s} \quad \text{and} \tag{15}$$

$$\|y_i^{\rm s}\|_{\mathcal{L}_\infty} \le \rho_i^{y^s} \tag{16}$$

hold. If the stability condition

$$\left\|\left(\mathbb{I} - F_i(s)\right)\hat{H}_i(s)\right\|_{\mathcal{L}_1}\left(\max_{i\in\mathcal{N}} L_{\Delta_i} + \max_{i\in\mathcal{N}}\|L_i\|\right) < 1 \tag{17}$$

holds for all $i \in \mathcal{N}$, then $y(t)$, $u(t)$ are uniformly bounded, i.e. there exist ρ_y, $\rho_u \in \mathbb{R}^+$ such that

$$\|y\|_{\mathcal{L}_\infty} \le \rho_y \quad \text{and} \quad \|u\|_{\mathcal{L}_\infty} \le \rho_u. \tag{18}$$

Moreover, the error signal between the navigator output and the real system output is uniformly bounded by

$$\begin{aligned}\|y - y^{\rm s}\|_{\mathcal{L}_\infty} \le\ & \max_{i\in\mathcal{N}}\|(\mathbb{I} - F_i(s))\hat{H}_i(s)\|_{\mathcal{L}_1} \\ & \cdot\left(\rho_y \max_{i\in\mathcal{N}} L_{\Delta_i} + \rho_y \max_{i\in\mathcal{N}}\|L_i\| + f_{\max}\right)\end{aligned} \tag{19}$$

where $f_{\max} = \min_{i\in\mathcal{N},\ y_i^{\rm s},\ \hat{y}_i^{\rm s}\in\{y_i | \|y_i\|_\infty \le \rho_i^{y^s}\}} f_i\left(t, y_i^{\rm s}, \hat{y}_{-i}^{\rm s}\right)$.

Proof: We first prove the boundedness of $\|y(t)\|$ and $\|u(t)\|$. According to agent i's real dynamics in equation (6), we know that equation (9) holds. By the definition of $v_i^{\rm c}$ in equation (11), equation (11) holds. Equations (9) and (11) together imply equation (10). Applying equation (10) into equation (8) yields

$$\begin{aligned}y_i(s) &= \bar{H}_i(s)\left(v_i^{\rm n}(s) - F_i(s)\bar{H}_i(s)^{-1}\hat{H}_i(s)\bar{\Delta}_i(s)\right) \\ &\quad + \hat{H}_i(s)(\bar{\Delta}_i(s) + f_i(s) + x_i^0) \\ &= \bar{H}_i(s)v_i^{\rm n}(s) - F_i(s)\hat{H}_i(s)\bar{\Delta}_i(s) \\ &\quad + \hat{H}_i(s)(\bar{\Delta}_i(s) + f_i(s) + x_i^0)\end{aligned} \tag{20}$$

Define $\hat{\Delta}_i(s) = \bar{\Delta}_i(s) + f_i(s)$. Based on the definition of $\bar{\Delta}_i$ in equation (7), we know

$$\hat{\Delta}_i(s) = \mathfrak{L}[\Delta_i(t, y(t))] - L_i y_i(s).$$

Therefore, equation (20) can be written as

$$\begin{aligned}y_i(s) &= (\mathbb{I} - F_i(s))\hat{H}_i(s)\left(\hat{\Delta}_i(s) - f_i(s)\right) \\ &\quad + \bar{H}_i(s)v_i^{\rm n}(s) + \hat{H}_i(s)(f_i(s) + x_i^0).\end{aligned} \tag{21}$$

Since $\bar{\Delta}_i$ is locally Lipschitz, $\hat{\Delta}_i$ is also locally Lipschitz satisfying

$$\|\hat{\Delta}_i(s)\|_{\mathcal{L}_\infty} \le L_{\Delta_i}\|y\|_{\mathcal{L}_\infty} + \|L_i\|\|y_i\|_{\mathcal{L}_\infty}.$$

With this inequality, equation (21) implies

$$\begin{aligned}&\|y_i(s)\|_{\mathcal{L}_\infty} \\ \le\ & \|\left(\mathbb{I} - F_i(s)\right)\hat{H}_i(s)\|_{\mathcal{L}_1}\|\hat{\Delta}_i(s)\|_{\mathcal{L}_\infty} \\ & + \|F_i(s)\hat{H}_i(s)\|_{\mathcal{L}_1}\|f_i(s)\|_{\mathcal{L}_\infty} + \|\hat{H}_i(s)\|_{\mathcal{L}_1}\|x_i^0\|_\infty \\ & + \|\bar{H}_i(s)\|_{\mathcal{L}_1}\|v_i^{\rm n}(s)\|_{\mathcal{L}_\infty} \\ \le\ & \|\left(\mathbb{I} - F_i(s)\right)\hat{H}_i(s)\|_{\mathcal{L}_1}\left(L_{\Delta_i}\|y\|_{\mathcal{L}_\infty} + \|L_i\|\|y_i\|_{\mathcal{L}_\infty}\right) \\ & + \|F_i(s)\hat{H}_i(s)\|_{\mathcal{L}_1}\|f_i(s)\|_{\mathcal{L}_\infty} + \|\hat{H}_i(s)\|_{\mathcal{L}_1}\|x_i^0\|_\infty \\ & + \|\bar{H}_i(s)\|_{\mathcal{L}_1}\|v_i^{\rm n}(s)\|_{\mathcal{L}_\infty}.\end{aligned}$$

Since this inequality holds for all $i \in \mathcal{N}$, we have

$$\begin{aligned}&\|y(s)\|_{\mathcal{L}_\infty} = \max_{i\in\mathcal{N}}\|y_i(s)\|_{\mathcal{L}_\infty} \\ \le\ & \max_{i\in\mathcal{N}}\left\|\left(\mathbb{I} - F_i(s)\right)\hat{H}_i(s)\right\|_{\mathcal{L}_1} \\ & \cdot\left(\max_{i\in\mathcal{N}} L_{\Delta_i}\|y\|_{\mathcal{L}_\infty} + \max_{i\in\mathcal{N}}\|L_i\|\ \|y\|_{\mathcal{L}_\infty}\right) \\ & + \max_{i\in\mathcal{N}}\|F_i(s)\hat{H}_i(s)\|_{\mathcal{L}_1}\max_{i\in\mathcal{N}}\|f_i(s)\|_{\mathcal{L}_\infty} \\ & + \max_{i\in\mathcal{N}}\left\{\|\hat{H}_i(s)\|_{\mathcal{L}_1}\|x_i^0\|_\infty + \|\bar{H}_i(s)\|_{\mathcal{L}_1}\|v_i^{\rm n}(s)\|_{\mathcal{L}_\infty}\right\}.\end{aligned}$$

Since $\|y_i^{\rm s}\|_{\mathcal{L}_\infty} \le \rho_i^{y^s}$ and f_i is continuous for all $i \in \mathcal{N}$, we know $\max_{i\in\mathcal{N}}\|f_i(s)\|_{\mathcal{L}_\infty}$ is bounded. Because $v_i^{\rm n}(s) = u_i^{\rm s}(s)$ and $\|u_i^{\rm s}\|_{\mathcal{L}_\infty} \le \rho_i^{u^s}$, the term $\max_{i\in\mathcal{N}}\|v_i^{\rm n}(s)\|_{\mathcal{L}_\infty}$ is also bounded. Therefore, the preceding inequality implies that $\|y(s)\|_{\mathcal{L}_\infty}$ is bounded. With the bound on $\|y(s)\|_{\mathcal{L}_\infty}$ and the definition of $u_i(t)$ in equation (13), we can easily verify the uniform boundedness of $\|u_i(t)\|$.

Let us now consider the bound on $\|y_i - y_i^{\rm s}\|_{\mathcal{L}_\infty}$. According to equation (4), the dynamics of the navigator satisfies:

$$y_i^{\rm s}(s) = \bar{H}_i(s)u_i^{\rm s}(s) + \hat{H}_i(s)(f_i(s) + x_i^0).$$

For any $i \in \mathcal{N}$, combining this equation with equation (20) and the fact $v_i^{\rm n}(s) = u_i^{\rm s}(s)$ implies the error $e_i(s) = y_i(s) - y_i^{\rm s}(s)$ satisfying

$$e_i(s) = (\mathbb{I} - F_i(s))\hat{H}_i(s)\bar{\Delta}_i(s),$$

which implies, with the definition of $\bar{\Delta}_i$ in equation (7) and its Lipschitz property,

$$\begin{aligned}\|e\|_{\mathcal{L}_\infty} &\le \max_{i\in\mathcal{N}}\|(\mathbb{I} - F_i(s))\hat{H}_i(s)\|_{\mathcal{L}_1}\max_{i\in\mathcal{N}}\left\|\bar{\Delta}_i(s)\right\|_{\mathcal{L}_\infty} \\ &\le \max_{i\in\mathcal{N}}\|(\mathbb{I} - F_i(s))\hat{H}_i(s)\|_{\mathcal{L}_1}\cdot \\ &\quad \left(\max_{i\in\mathcal{N}} L_{\Delta_i}\|y\|_{\mathcal{L}_\infty} + \max_{i\in\mathcal{N}}\|L_i\|\|y\|_{\mathcal{L}_\infty} + f_{\max}\right) \\ &\le \max_{i\in\mathcal{N}}\|(\mathbb{I} - F_i(s))\hat{H}_i(s)\|_{\mathcal{L}_1}\cdot \\ &\quad \left(\rho_y \max_{i\in\mathcal{N}} L_{\Delta_i} + \rho_y \max_{i\in\mathcal{N}}\|L_i\| + f_{\max}\right)\end{aligned}$$

where the last inequality is obtained by using $\|y\|_{\mathcal{L}_\infty} \le \rho_y$ that has been proven above. ■

Remark 4.2: Consider the stability condition in inequality (17). The term $\left\|\left(\mathbb{I} - F_i(s)\right)\hat{H}_i(s)\right\|_{\mathcal{L}_1}$ goes to zero, as the bandwidth of $F_i(s)$ increases. Therefore, we can always find an appropriate $F_i(s)$ to satisfy this stability condition.

Similarly, we can also arbitrarily reduce the performance bound in inequality (19) by increasing the bandwidth of $F_i(s)$. From another side, however, the higher bandwidth $F_i(s)$ has, the smaller the stability margin is for the system. Therefore, $F_i(s)$ must be chosen in a way that can balance the performance and the robustness with respect to the actuation delays.

Remark 4.3: The assumption of the boundedness on the navigation inputs and outputs, in fact, is not a strong assumption. For example, when considering stabilization or tracking problem, the state of the ideal close-loop stable system will always stay in a compact set. As for cooperative control, most of the approaches ensure that the behavior is limited to a compact region. With careful selection of the triggering events, the outputs and the inputs of the navigators can also be bounded. In the next section, we will show a way to select the events for stabilization. The selected events ensure that the navigator stays close to the ideal model, which also implies that the signals in the navigators are bounded, given bounded signals of the ideal model.

V. Event Design

With this framework, to achieve a control objective, one only needs to ensure that the navigators can fulfill it. Therefore, for a specific objective, the existing time-triggered, event-triggered, and self-triggered approaches, which are designed for completely known system dynamics, can be directly applied to establish the communication between navigators, such as event-triggered consensus [9], event-triggered sensor network [17]. However, although the navigator's dynamics are completely known, the communication models, especially event-triggered approaches, may bring in a certain level of uncertainties. If we want to further improve the predictability of the system, one way is to quantify the difference of the navigators in (4) and the ideal model in (1). In this following discussion, we mainly focus on the objective of system stabilization.

Note that the difference between these two systems is the broadcast outputs $\hat{y}^{\mathrm{s}}_{-i}(t)$. If we replace $\hat{y}^{\mathrm{s}}_{-i}(t)$ by $y^{\mathrm{s}}_{-i}(t)$ in the navigator, then it will be the same as the ideal model. Therefore, to quantify the error between these two systems, the key is to quantify the difference between $\hat{y}^{\mathrm{s}}_{-i}(t)$ by $y^{\mathrm{s}}_{-i}(t)$ and study the impact of such difference on the error dynamics.

Let

$$w^{\mathrm{id}} = \left[x_1^{\mathrm{id}\top}, \cdots, x_N^{\mathrm{id}\top}, z_1^{\mathrm{id}\top}, \cdots, z_N^{\mathrm{id}\top} \right]^\top$$

be the overall state of the ideal model in equation (1). We can re-write the overall closed-loop ideal model as follows:

$$\dot{w}^{\mathrm{id}} = \Psi\left(t, w^{\mathrm{id}}(t)\right), \tag{22}$$

where $\Psi : \mathbb{R}_0^+ \times \mathbb{R}^{|\mathcal{N}|(n+p)} \to \mathbb{R}^{|\mathcal{N}|(n+p)}$ is defined based on the distributed version of the ideal model in equation (1).

Consequently, the overall dynamics of the navigator can be written as

$$\dot{w}^{\mathrm{s}} = \Psi\left(t, w^{\mathrm{s}}(t)\right) + \Phi\left(t, y^{\mathrm{s}}(t), \hat{y}^{\mathrm{s}}(t)\right) \tag{23}$$

where $y^{\mathrm{s}} = \left[y_1^{\mathrm{s}\top}, \cdots, y_N^{\mathrm{s}\top} \right]^\top$, $\hat{y}^{\mathrm{s}} = \left[(\hat{y}_1^{\mathrm{s}})^\top, \cdots, (\hat{y}_N^{\mathrm{s}})^\top \right]^\top$, $\Phi : \mathbb{R}_0^+ \times \mathbb{R}^{|\mathcal{N}|m} \times \mathbb{R}^{|\mathcal{N}|m}$ is defined by

$$\Phi\left(t, y^{\mathrm{s}}, \hat{y}^{\mathrm{s}}\right) = \begin{pmatrix} B_1 \tilde{h}_1 + \tilde{f}_1 \\ \cdots \\ B_N \tilde{h}_N + \tilde{f}_N \\ \tilde{g}_1 \\ \cdots \\ \tilde{g}_N \end{pmatrix},$$

and

$$\begin{aligned} \tilde{h}_i &= h_i\left(t, z_i^{\mathrm{s}}, y_i^{\mathrm{s}}, \hat{y}_{-i}^{\mathrm{s}}\right) - h_i\left(t, z_i^{\mathrm{s}}, y_i^{\mathrm{s}}, y_{-i}^{\mathrm{s}}\right) \\ \tilde{f}_i &= f_i\left(t, y_i^{\mathrm{s}}, \hat{y}_{-i}^{\mathrm{s}}\right) - f_i\left(t, y_i^{\mathrm{s}}, y_{-i}^{\mathrm{s}}\right) \\ \tilde{g}_i &= g_i\left(t, z_i^{\mathrm{s}}, y_i^{\mathrm{s}}, \hat{y}_{-i}^{\mathrm{s}}\right) - g_i\left(t, z_i^{\mathrm{s}}, y_i^{\mathrm{s}}, y_{-i}^{\mathrm{s}}\right). \end{aligned}$$

Assume that the signals in the navigators are bounded. Since f_i, g_i, h_i are locally Lipschitz, we know there exists $L_\Phi \in \mathbb{R}_0^+$ such that

$$\Phi\left(t, y^{\mathrm{s}}, \hat{y}^{\mathrm{s}}\right) \le L_\Phi \|y^{\mathrm{s}} - \hat{y}^{\mathrm{s}}\|_\infty.$$

Therefore, if each agent can enforce

$$\|y_i^{\mathrm{s}}(t) - \hat{y}_i^{\mathrm{s}}(t)\|_\infty \le \epsilon_i \tag{24}$$

for a given constant ϵ_i, then

$$\Phi\left(t, y^{\mathrm{s}}, \hat{y}^{\mathrm{s}}\right) \le L_\Phi \epsilon_{\max}, \tag{25}$$

where $\epsilon_{\max} = \max_{i \in \mathcal{N}} \epsilon_i$. It implies that the perturbation term Φ in equation (23) is uniformly bounded.

With this bound, we can borrow Theorem 9.1 in [18, pp. 356] to establish the following proposition:

Proposition 5.1: Consider the systems in equations (22) and (23). Assume that

- $\Psi(t, w^{\mathrm{id}})$ and its first partial derivatives with respect to x are continuous, bounded, and Lipschitz in w^{id}, uniformly in t, for any $t \in \mathbb{R}_0^+$ and w^{id} in a compact set;
- $\Phi\left(t, y^{\mathrm{s}}, \hat{y}^{\mathrm{s}}\right)$ is piecewise continuous in t, locally Lipschitz in y^{s}, $\hat{y}^{\mathrm{s}}$, and inequality (25) holds;
- there exist continuous, positive definite functions α_1, α_2, $\alpha_3 : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ and a continuously differentiable function $V : \mathbb{R}_0^+ \times \mathbb{R}^{|\mathcal{N}|n} \to \mathbb{R}_0^+$ such that
$$\begin{aligned} &\alpha_1(\|w^{\mathrm{id}}\|) \le V(t, w^{\mathrm{id}}) \le \alpha_2(\|w^{\mathrm{id}}\|) \\ &\frac{\partial V}{\partial t} + \frac{\partial V}{\partial w^{\mathrm{id}}} \Psi\left(t, w^{\mathrm{id}}\right) \le -\alpha_3(\|w^{\mathrm{id}}\|); \end{aligned}$$
- $x = 0$ is an exponentially stable equilibrium point of the ideal system.

Then there exist $\beta, \eta \in \mathbb{R}_0^+$, such that if $L_\Phi \epsilon_{\max} \le \eta$,

$$\|w^{\mathrm{id}} - w^{\mathrm{s}}\|_{\mathcal{L}_\infty} \le \beta L_\Phi \epsilon_{\max} \tag{26}$$

holds.

Proof: The proof is a direct application of Theorem 9.1 in [18, pp. 356] and is therefore omitted. ■

Remark 5.1: Note that no matter the transmission delays exist or not, or what types of communication schemes (time-triggering or event-triggering) that agents are using, as long as the condition in equation (24) holds, the closeness between the two systems can be guaranteed. For example, let us take a look at event-triggering. For the case that the transmission delays are negligible, we can directly use the violation of $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty \leq \epsilon_i$ to trigger the next broadcast time. The time instant t_i^{k+1} is given by equation (5) with ϕ being in this case $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty - \epsilon_i$. For the case where transmission delays exist, the event can be $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty = \underline{\epsilon}_i$ with $\underline{\epsilon}_i < \epsilon_i$. Then the time interval for $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty$ to grow from $\underline{\epsilon}_i$ to ϵ_i is the amount of delay that the system can tolerate. A detailed discussion on how to obtain the bounds on the transmission delays can be found in [8].

Remark 5.2: The event threshold ϵ_i determines how frequently agent i broadcasts. The smaller the ϵ_i is, the more frequent are the broadcasts. On the contrary, based on inequality (26), the smaller ϵ_i is, the smaller the performance bound is. Therefore, when the communication condition improves, the navigator can be arbitrarily close to the ideal model. It suggests a tradeoff between the performance and the communication condition.

Remark 5.3: Although this section discusses the stabilization problem, similar perturbation-based analysis can be applied for the tracking problem. However, in the tracking problem, the magnitude of the tracking signal is usually inverse proportional to the nonlinearity. Therefore, to ensure tracking, one must be careful about the design of the ideal model, where the nonlinearity must be weak enough to ensure a desired level of the magnitude for the tracking signal.

Remark 5.4: For cooperative control, such as consensus, maximal coverage problems, Proposition 5.1 might not be applicable because the third condition might fail. Generally speaking, it is difficult to establish such closeness between the navigator and the ideal model for the control purpose. However, as we mentioned before, to fulfill a global control objective, the navigator does not have to follow the ideal model. As long as one can find the right communication model for the navigator to achieve that objective, our control architecture can ensure that the real dynamics follows the navigator, which implies that the real dynamics will also achieve the same control objective, even in the presence of uncertainties.

VI. Simulations

This section presents simulation results that demonstrate the decoupling nature of the proposed control and communication architecture. We still consider the consensus problem in example 1 and example 2. In the simulations, we set $N = 4$ and $\mathcal{N}_1 = \{2\}$, $\mathcal{N}_2 = \{1, 3\}$, $\mathcal{N}_3 = \{2, 4\}$, and $\mathcal{N}_4 = \{3\}$. The low-pass filter is $F_i(s) = \frac{30}{s+30}$. The initial condition of each agent is randomly generated over the interval $[-5, 5]$.

We first examine the impact of communication models on the closeness between the navigator and the real dynamics. The uncertainty is $\Delta_i(t, x) = \sum_{j \in \mathcal{N}_i} 0.2x_j^2$. Three different types of broadcast triggering are considered: (1) each agent broadcasts every 0.5 seconds; (2) agent i broadcasts when $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty = 0.4$; (3) agents broadcast simultaneously when the centralized event $\|e^s(t)\|_2 = 0.2\frac{\|Lx^s(t)\|_2}{\|L\|_2}$ occurs, where L is the Laplacian of the communication graph (This event-trigger was presented in [9]). The simulations results are plotted in Figure 2 and Figure 3. Figure 2 shows that the state trajectories of the navigators vary under different broadcast schemes. Figure 3 shows the errors between the states in the real system and the navigator ($\|x(t) - x^s(t)\|_\infty$) under these three broadcast triggering schemes. It is obvious that although the state trajectories of the navigators change under different broadcast schemes, the state trajectory of the real system is always close to that of the navigator. When the bandwidth of $F_i(s)$ increases, the error can be further reduced. It verifies our claim that the broadcast schemes will not affect the closeness between the real system and the navigator, provided that the signals in the navigator are bounded.

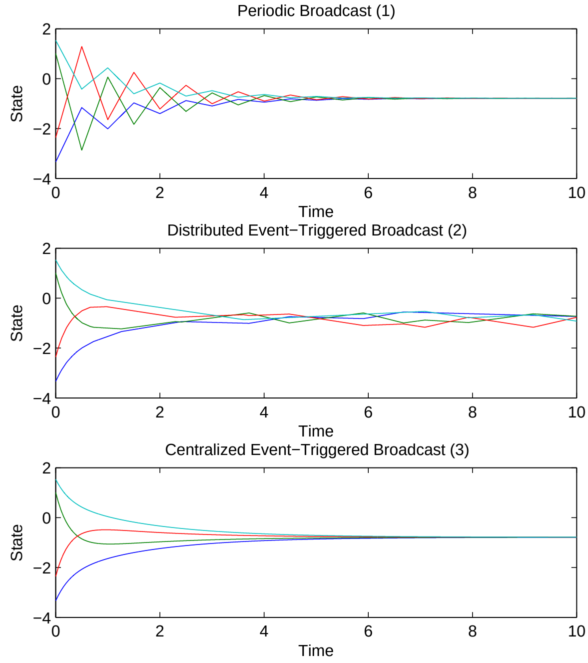

Fig. 2. The state trajectories of the navigators under different broadcast schemes

We then consider two different uncertainties in the system:

$$
\begin{aligned}
\text{Uncertainty I}: \quad & \Delta_1(t,x) = \sin(t)x_1^2 + 0.1e^{0.2x_2} - x_3 \\
& \Delta_2(t,x) = \cos(30t)x_1 + x_2 - x_4^{0.3} \\
& \Delta_3(t,x) = x_2^2 + x_3^2 + x_4 \\
& \Delta_4(t,x) = 0.2 - 3x_3 + x_4 \quad \text{and} \\
\text{Uncertainty II}: \quad & \Delta_1(t,x) = x_1^2 - x_2 \\
& \Delta_2(t,x) = x_2 + x_3^2 - x_4^3 \\
& \Delta_3(t,x) = x_2^2 + x_3^2 + x_4^2 \\
& \Delta_4(t,x) = x_3 + x_4.
\end{aligned}
$$

The controller is the same as the one in the first simulation.

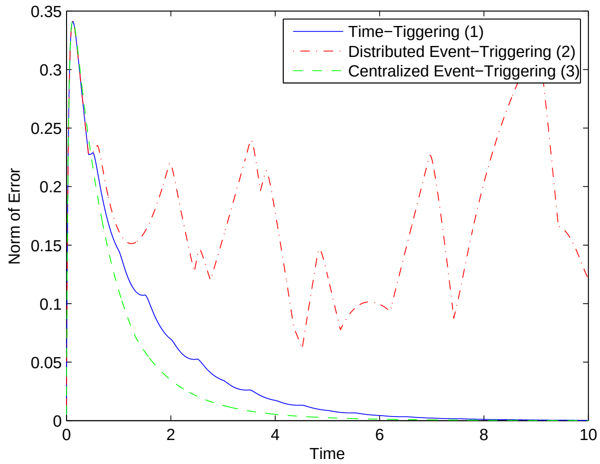

Fig. 3. The state error between the real system and the navigator under different broadcast schemes

We consider event-triggered broadcast, where agent i broadcasts when $\|y_i^s(t) - y_i^s(t_i^k)\|_\infty = 0.2$. The simulations show that the state trajectories of navigators and the broadcast time instants of each agent remain the same for different uncertainties, which is consistent with our theoretical results. Figure 4 plots the error between the states of the real subsystem and its navigator. It shows that even under different uncertainties, the closeness between two systems can still be preserved using the same controller.

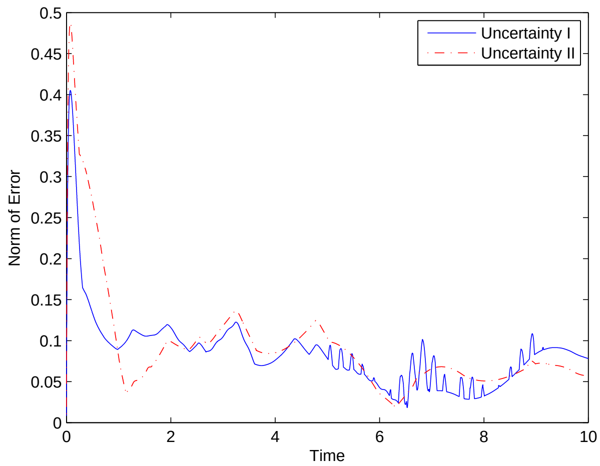

Fig. 4. The state error between the real system and the navigator under different uncertainties

VII. CONCLUSIONS

This paper studies uncertain networked control systems with communication constraints. A navigation-based architecture is provided to decouple the design of control and communication schemes. The closeness between the real system and the navigator is affected only by the uncertainty, and the difference between the navigator and the ideal model is due to the communication constraints. We derive bounds on the difference between the real system, the navigator, and the ideal model, which can be arbitrarily improved by adjusting the design parameters. However, in this work, we only quantify the difference in the outputs. One can also quantify the errors in the states of different systems. Future research will also include using the performance bounds for optimal path design for the purpose of collision avoidance.

REFERENCES

[1] C. Langbort, R. Chandra, and R. D'Andrea, "Distributed control design for systems interconnected over an arbitrary graph," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1502–1519, 2004.

[2] W. Dunbar, "A distributed receding horizon control algorithm for dynamically coupled nonlinear systems," in *Proc. IEEE Conference on Decision and Control*, 2005.

[3] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[4] H. Tanner, A. Jadbabaie, and G. Pappas, "Stable flocking of mobile agents, part II: Dynamic topology," in *Proc. IEEE Conference on Decision and Control*, 2003, pp. 2016–2021.

[5] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 947–950, 2001.

[6] J. Cortes, S. Martinez, T. Karatas, F. Bullo, *et al.*, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[7] R. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, p. 571, 2007.

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

[9] D. V. Dimarogonas and K. H. Johansson, "Event-triggered control for multi-agent systems," in *Proc. IEEE Conference on Decision and Control*, 2009.

[10] P. Wan and M. Lemmon, "An event-triggered distributed primal-dual algorithm for network utility maximization," in *Proc. IEEE Conference on Decision and Control*, 2009.

[11] R. D'Andrea and G. Dullerud, "Distributed control design for spatially interconnected systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1478–1495, 2003.

[12] D. Nesic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1650–1667, 2004.

[13] X. Wang and M. Lemmon, "Finite-gain $\mathcal{L}_2$ stability in distributed event-triggered networked control systems with data dropouts," in *Proc. European Control Conference*, 2009.

[14] X. Wang and N. Hovakimyan, "Performance prediction in uncertain networked control systems using $\mathcal{L}_1$-adaptation-based distributed event-triggering," in *Proc. IEEE Conference on Decision and Control*, 2010.

[15] H. Shim and N. H. Jo, "An almost necessary and sufficient condition for robust stability of closed–loop systems with disturbance observer," *Automatica*, vol. 45, no. 1, pp. 296–299, 2009.

[16] E. Kharisov, K. K. K. Kim, X. Wang, and N. Hovakimyan, "Limiting behavior of $\mathcal{L}_1$ adaptive controllers," in *Proc. AIAA Guidance, Navigation, and Control Conference*, 2011.

[17] M. Mazo and P. Tabuada, "On event-triggered and self-triggered control over sensor/actuator networks," in *Proc. IEEE Conference on Decision and Control*, 2008.

[18] H. Khalil, *Nonlinear systems*. Prentice Hall Upper Saddle River, NJ, 2002.