Stochastic Networked Control Systems with Dynamic Protocols†

Duarte Antunes, João P. Hespanha, and Carlos Silvestre

ABSTRACT

We consider networked control systems in which sensors, controllers, and actuators communicate through a shared network that introduces stochastic intervals between transmissions, delays and packet drops. Access to the communication medium is mediated by a protocol that determines which node (one of the sensors, one of the actuators, or the controller) is allowed to transmit a message at each sampling/actuator-update time. We provide conditions for mean exponential stability of the networked closed loop in terms of matrix inequalities, both for investigating the stability of given protocols, such as static round-robin protocols and dynamic maximum error first-try once discard protocols, and to design new dynamic protocols. The main result entailed by these conditions is that if the networked closed loop is stable for a static protocol then we can provide a dynamic protocol for which the networked closed loop is also stable. The stability conditions also allow for obtaining an observer-protocol pair that reconstructs the state of an LTI plant in a mean exponential sense and less conservative stability results than other conditions that previously appeared in the literature.

Key Words: Networked Control Systems, Dynamic Protocols, Scheduling, Stochastic Systems

I. Introduction

The proliferation of network communication systems in recent years paved the way for important research in the area of networked control systems. This research area addresses control loops closed via a shared network that provides the medium for sensor, actuator and controller nodes to communicate.

Walsh and co-authors [1] made strides in the analysis of control systems closed via a local area network, such as the controller-area network, the ethernet and wireless 802.11 networks. The key assumptions in [1] are that there exists a bound on the interval between transmissions denoted by maximum allowable transfer interval (MATI), and that transmission delays and packet drops are negligible. In [1] an emulation set-up is considered in the sense that the controller for the networked control system is obtained from a previously designed stabilizing continuous-time controller. Two basic types of protocols have been proposed: static protocols, such as the round-robin (RR) protocol where nodes take turns transmitting data in a periodic fashion; and dynamic protocols such as the maximum error first-try once discard (MEF-TOD) protocol, where the node that has the top priority in using the communication medium is the one whose current value to transmit differs the most from the last transmitted value. Under this setup, one can attempt to provide an upper bound on the MATI for which stability can be guaranteed. Since these protocols have been proposed in the papers referenced above, MATI bounds have been...
improved [2], [3], [4]. Although, [1] illustrate through simulations that using the MEF-TOD protocol allows for preserving stability of the networked closed loop for a larger MATI than that obtained when using the RR protocol, and similar conclusions are obtained in [2]-[4] from sufficient stability conditions, no analytical result has been established proving that this holds in general.

As mentioned in [1], the occurrence of transmission events on the network is often more appropriately modeled as a random process. This feature is taken into account in [5], which considers networked control systems with MEF-TOD and RR protocols, and independent and identically distributed (i.i.d.) intervals between transmissions. It is shown that stability can be guaranteed for distributions of the inter-transmission intervals that have a support larger than previously derived deterministic upper bounds for the MATI [1], [2], [3]. The conservativeness of these results for linear networked control systems using the RR protocol was eliminated in [6]. Recently, [7] addresses a model of networked control systems with i.i.d. intervals between transmissions and stochastic delays for a class of quadratic protocols that is more general than MEF-TOD. Through a convex over-approximation approach, sufficient conditions are given for mean exponential stability. In [8] a method is proposed to design an observer-protocol pair to asymptotically reconstruct the states of an LTI plant where the plant outputs are sent through a network with constant intervals between transmissions. The protocol to be designed can be viewed as a weighted version of the MEF-TOD.

In the present paper we follow this line of research considering that the network imposes i.i.d. intervals between transmissions. We also take into account stochastic delays modeled as in [7], and packet drops. We consider that access to the network is mediated by a dynamic protocol specified as follows. Associated to each node there is a set of quadratic state functions, which are evaluated at a given transmission time. The node allotted to transmit is the one corresponding to the least value of these quadratic state functions. These protocols are more general than quadratic protocols considered in [7] and thus more general than the MEF-TOD protocol.

We establish two stability results for the networked control system, both providing conditions in terms of linear matrix inequalities (LMIs) for investigating the stability in a mean exponential sense of given protocols, and conditions in terms of BMIs to design quadratic state functions, specifying the dynamic protocol, that yield the networked closed loop stable. The first stability result allows to prove that if the networked closed loop is stable for a static protocol then we can provide a dynamic protocol for which the networked closed loop is also stable. This is the main contribution of the paper and gives an analytical justification on why one should utilize dynamic protocols rather than static, while, e.g., in [1], this conclusion is only illustrated through simulation. The second stability result allows us to extend the work [8] to the case where transmission intervals are stochastic. We also address the relation of this stability result with the necessity of existence of a quadratic stochastic Lyapunov function that assures stability for the networked control system.

We illustrate through benchmark examples, that the conditions in this paper are significantly less conservative than other conditions that previously appeared in the literature.

A preliminary version of the results presented here appeared in the conference paper [9]. Besides including all the formal proofs of the results, here we establish the connection between the second of our two main stability results and the existence of a quadratic stochastic Lyapunov function that assures stability for the networked control system.

The remainder of the paper is organized as follows. The networked control problem set up is given in Section II. The main results are stated in Section III. In Section IV we compare our results with previous works. Concluding remarks are given in Section V. The proofs of the results are provided in the appendix.

Notation We denote by $I_n$ and $O_n$ the $n \times n$ identity and zero matrices, respectively, and by \text{diag}([A_1 \ldots A_n]) a block diagonal matrix with blocks $A_i$. For dimensionally compatible matrices $A$ and $B$, we define $(A, B) := [A^T B^T]^T$.

II. PROBLEM FORMULATION

We start by introducing the networked control stability problem and then we show that it can be casted into analyzing the stability of an impulsive system.

2.1. Networked Control Set-up

We consider a networked control system for which sensors, actuators, and a controller, are connected through a communication network, possibly shared with other users. The plant and controller are described by the following state-space model:

\begin{align*}
\text{Plant:} \quad & \dot{x}_P = A_P x_P + B_P u, \quad y = C_P x_P \quad (1) \\
\text{Controller:} \quad & \dot{x}_C = A_C x_C + B_C y, \quad u = C_C x_C + D_C y. \quad (2)
\end{align*}
Following an emulation approach, we assume that the controller has been designed to stabilize the closed loop, when the process and the controller are directly connected, i.e., \( u(t) = u(t), \ y(t) = y(t), t \geq 0 \) and we are interested in analyzing the effects of the network on the stability of the closed loop. We denote the times at which a node transmits a message by \( \{ t_k, k \in \mathbb{N} \} \), and assume that \( \hat{u} \) and \( \hat{y} \) are held constant between transmission times, i.e.,

\[
\hat{u}(t) = \hat{u}(t_k), \ y(t) = \hat{y}(t), t \in [t_k, t_{k+1}), \ k \in \mathbb{Z}_{\geq 0}.
\]

We denote by \( e \) the error signal between process output and controller input \( \hat{y} - y \) and between controller output and process input \( \hat{u} - u \). In particular,

\[
e := (\hat{y} - y, \hat{u} - u).
\]

We assume that while \( m \) nodes compete for the network, only one of them is allowed to transmit at each given transmission time. However, in our terminology a single transmitting node could be associated with several entries of the process output \( y \) or with several entries of the controller output \( u \). For simplicity, we assume that the sensors and actuators have been ordered in such a way that we can partition the error vector as \( e = (e_1, \ldots, e_m) \) and is error associated with node \( i \in \mathcal{M} := \{1, \ldots, m \} \). The state of the networked control system is thus defined by the vector \( x := (x_P, x_C, e) \), where \( x_P \in \mathbb{R}^{n_P}, x_C \in \mathbb{R}^{n_C}, e \in \mathbb{R}^{n_e} \), and \( x \in \mathbb{R}^{n_x} \). We are interested in scenarios for which the following assumptions hold:

(i) The time intervals \( \{ h_k := t_{k+1} - t_k \} \) are i.i.d. described by a probability measure \( \mu \) with support on \( [0, \gamma], \gamma \in \mathbb{R}_{\geq 0} \cup \{+\infty\} \), i.e., \( \text{Prob}(a \leq h_k \leq b) = \int_a^b \mu(dr) \) for \( a, b \in [0, \gamma] \).

(ii) Corresponding to a transmission at time \( t_k \) there is a transmission delay \( d_k \) no greater than \( h_k := t_{k+1} - t_k \); A joint stationary probability density \( \chi \) describes \( (h_k, d_k) \), in the sense that

\[
\text{Prob}(a \leq h_k \leq b, c \leq d_k \leq d) = \int_a^b \int_c^d \chi(dr, ds)
\]

where \( a, b \in [0, \gamma] \) and \( \text{Prob}(a \leq h_k \leq b, c \leq d_k \leq d) = 0 \) if \( c > b \). In view of (i) and (ii), we have that \( \mu([a, b]) = \chi([a, b], [0, b]) \).

(iii) At each transmission time there is a probability \( p_{\text{drop}} \) that a packet may not arrive at its destination or that it may arrive corrupted (packet drop).

(iv) The nodes implement one of the two protocols:

Dynamic protocol (DP): This protocol is specified by \( m_D \) symmetric matrices \( \{ R_i, i \in \mathcal{M}_D \}, \mathcal{M}_D := \{1, \ldots, m_D \}, m_D \geq m \). A subset of these matrices \( \{ R_i, i \in I_j \} \) is associated with node \( j \in \mathcal{M} \) where \( I_j := \{i_1^j, i_2^j, \ldots, i_{m_j}^j\} \) is an index subset of \( \mathcal{M}_D \). These subsets are assumed to be nonempty, i.e., \( r_j \geq 1 \), disjoint, and the \( r_j \) are such that \( \sum_{j=1}^{m_D} r_j = m_D \). The node \( j \) allotted to transmit at \( t_k \) is determined by the map \( d : \mathbb{R}^{n_x} \rightarrow \mathcal{M} \),

\[
d(x(t_k)) = d_1 \circ d_2(x(t_k)),
\]

where \( d_2 : \mathbb{R}^{n_x} \rightarrow \mathcal{M}_D \) is given by

\[
d_2(x(t_k)) := \text{argmin}_{i \in \mathcal{M}_D} x(t_k)^T R_i x(t_k),
\]

and \( d_1 : \mathcal{M}_D \rightarrow \mathcal{M} \) is given by

\[
d_1(i) := \{ j : i \in I_j \}.
\]

In case the minimum in (7) is achieved simultaneously for several values of the index \( i \), stability of the networked control system should be guaranteed regardless of the specific choice for the argmin. In view of (6), the error \( e \) is updated at time \( t_k \) according to

\[
e(t_k) = (I_{n_e} - \Lambda_{d_2(x(t_k))})e(t_k^-),
\]

where \( \Lambda_j := \text{diag}(0_{(\sum_{i=1}^{r_j} I_{s_i}, 0_{(\sum_{i=r_j+1}^{m} s_i})}) \), \ j \in \mathcal{M} \). That is, only the components of \( \hat{y} \) or \( \hat{u} \) associated with the node that transmits are updated by the corresponding components of \( y(t_k) \) or \( u(t_k) \). We call a dynamic protocol regular if for every \( j \in \mathcal{M}_D \) there exists a state \( x \) such that \( j \) is the unique index such that \( j = \text{argmin}_{i \in \mathcal{M}_D} x^T R_i x \). A non regular dynamic protocol can always be made regular by discarding unnecessary matrices \( R_i \).

Static Protocol (SP): The nodes transmit in a \( m_S \)-periodic sequence determined by a periodic function

\[
s : \mathbb{N} \rightarrow \mathcal{M}
\]

with period \( m_S \). In this case, the error \( e \) is updated at time \( t_k \) according to

\[
e(t_k) = (I_{n_e} - \Lambda_{s(t_k)})e(t_k^-).
\]

We assume that \( s \) is onto, i.e., each node transmits at least once in a period. When \( m_S = m \), each node transmits exactly once in a period.

As mentioned in Section 1, Assumptions (i)-(iii) are appropriate for networked control systems in which feedback loops are closed via local area networks.
(cf. [1], [10]). In particular, Assumption (i) holds for scenarios in which nodes attempt to do periodic transmissions of data, but these regular transmissions may be perturbed by the medium access. It is typically the case in Carrier Sense Multiple access (CSMA) protocols that nodes may be forced to back-off for a typically random amount of time until the network becomes available. Note that networks with CSMA protocols protocol, such as the Ethernet of the Wireless 802.11 are prevalent in nowadays communication systems. The probability distribution of the time interval between transmissions, which can be estimated experimentally or by running Monte Carlo simulations of the protocol, is determined by two factors: the congestion of the network and the delay introduced by the medium access protocol.

The class of dynamic protocols that we describe in (iv) allow a node to transmit if the state of the networked control system lies on a given region of the state space, partitioned according to quadratic restrictions. This class of protocols boils down to the quadratic control system (cf. [1], [10]). In particular, Assumption (i) holds for scenarios in which nodes attempt to do periodic transmissions of data, but these regular transmissions may be perturbed by the medium access. It is typically the case in Carrier Sense Multiple access (CSMA) protocols that nodes may be forced to back-off for a typically random amount of time until the network becomes available. Note that networks with CSMA protocols protocol, such as the Ethernet of the Wireless 802.11 are prevalent in nowadays communication systems. The probability distribution of the time interval between transmissions, which can be estimated experimentally or by running Monte Carlo simulations of the protocol, is determined by two factors: the congestion of the network and the delay introduced by the medium access protocol.

The class of dynamic protocols that we describe in (iv) allow a node to transmit if the state of the networked control system lies on a given region of the state space, partitioned according to quadratic restrictions. This class of protocols boils down to the quadratic control system (cf. [1], [10]). In particular, Assumption (i) holds for scenarios in which nodes attempt to do periodic transmissions of data, but these regular transmissions may be perturbed by the medium access. It is typically the case in Carrier Sense Multiple access (CSMA) protocols that nodes may be forced to back-off for a typically random amount of time until the network becomes available. Note that networks with CSMA protocols protocol, such as the Ethernet of the Wireless 802.11 are prevalent in nowadays communication systems. The probability distribution of the time interval between transmissions, which can be estimated experimentally or by running Monte Carlo simulations of the protocol, is determined by two factors: the congestion of the network and the delay introduced by the medium access protocol.

The class of dynamic protocols that we describe in (iv) allow a node to transmit if the state of the networked control system lies on a given region of the state space, partitioned according to quadratic restrictions. This class of protocols boils down to the quadratic control system (cf. [1], [10]). In particular, Assumption (i) holds for scenarios in which nodes attempt to do periodic transmissions of data, but these regular transmissions may be perturbed by the medium access. It is typically the case in Carrier Sense Multiple access (CSMA) protocols that nodes may be forced to back-off for a typically random amount of time until the network becomes available. Note that networks with CSMA protocols protocol, such as the Ethernet of the Wireless 802.11 are prevalent in nowadays communication systems. The probability distribution of the time interval between transmissions, which can be estimated experimentally or by running Monte Carlo simulations of the protocol, is determined by two factors: the congestion of the network and the delay introduced by the medium access protocol.

2.2. Impulsive systems

Suppose that there are no delays, i.e. \( d_k = 0 \), and no packet drops, i.e., \( p_{\text{drop}} = 0 \). Then we can write the networked control system (1), (2), (3), (4), in the form of the following impulsive system

\[
\begin{align*}
\dot{x}(t) &= Ax(t), \quad t \in \mathbb{R}_{\geq 0}, t \neq t_k \\
x(t_k) &= J_p(x(t_k^-)), x(t_k^-), k \in \mathbb{N}_0 \tag{12}
\end{align*}
\]

where \( x \in \mathbb{R}^{n_x}, x(0^-) := x_0 \), and \( t_{k+1} - t_k \) are i.i.d. random variables characterized by the probability density \( \mu \), and the map \( p \) takes the following form for dynamic and static protocols

\[
\begin{align*}
\text{DP:} & \quad p(x(t_k^-), k) = d(x(t_k^-)) \tag{13} \\
\text{SP:} & \quad p(x(t_k^-), k) = s(k). \tag{14}
\end{align*}
\]

For example, the following expressions for \( A \) and \( \{ J_i, i \in \mathcal{M} \} \), correspond to the case in which the controller and plant are directly connected and only the outputs are transmitted through the network, i.e., \( \ddot{u}(t) = u(t), x = (x_P, x_C, y - y) \).

\[
\begin{align*}
A &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \\
A_{11} &= \begin{bmatrix} A_P + B_P D_C C_P & B_P C_C \\ B_C C_P & A_C \end{bmatrix} \\
A_{12} &= \begin{bmatrix} B_P D_C \\ B_C \end{bmatrix} \\
A_{21} &= - \begin{bmatrix} C_P & 0 \end{bmatrix} A_{11} \\
A_{22} &= - \begin{bmatrix} C_P & 0 \end{bmatrix} A_{12} \\
J_i &= \text{diag}([I_{n_P + n_C}, I_n - \Lambda_i]), \ i \in \mathcal{M}.
\end{align*}
\]

This case will be considered in Section IV. Expressions for the general case considered in Section II can be easily obtained (see, e.g., [11, p. 5]).

To take into account delays and packet drops modeled as described in Section II, we consider the following impulsive system

\[
\begin{align*}
\dot{x}(t) &= Ax(t), \quad t \in \mathbb{R}_{\geq 0}, t \neq t_k \\
x(t_k) &= K^p_{\text{int}}(x(t_k^-), k) x(t_k^-), k \in \mathbb{N}_0 \\
x(s_k) &= L x(s_k), \ t_k \leq s_k \leq t_{k+1}, \tag{16}
\end{align*}
\]

where \( p(x_k, k) \) is defined as in (13) for dynamic protocols and as in (14) for static protocols, and the initial condition is set to \( x(0^-) := x_0 \). The random variables \( t_k \) and \( s_k \) are completely defined by the inter-sampling times \( h_k := t_{k+1} - t_k \) and by the delays \( d_k := s_k - t_k \). The \( (t_k, d_k) \) are i.i.d., and are as described by (5). The \( q_k \in \{1, \ldots, n_q\} \) are i.i.d., and such that \( \text{Prob}[q_k = j] = w_j \forall j \in \{1, \ldots, n_q\}, k \geq 0 \). We provide below expressions for \( A, L, w_i \) and \( K^p_{\text{int}}, i \in \mathcal{M}, j \in \{1, \ldots, n_q\} \) which model the case where the controller and the plant are directly connected and only the plant outputs are transmitted through the network, i.e., \( \ddot{u}(t) = u(t) \). The state is now considered to be \( x = (x_P, x_C, \tilde{y}, v) \in \mathbb{R}^{n_x} \) where \( v \in \mathbb{R}^{n_v} \) is an auxiliary vector \( (v_1, \ldots, v_m) \) that is updated with the sampled value \( v_j = y_j(t_k) \) at each sampling time \( t_k \) at which node \( j \) is allowed to transmit. However, the update only takes place if a packet sent at \( t_k \) is not dropped and the sampled value \( v_j \) is only used to update the value of \( \tilde{y}_j \) after a transmission delay \( d_k \), at the time \( s_k = t_k + d_k \).
the survivor function. Assuming (16) constants assume that the following condition holds

\begin{equation}
\sum_{j} p_{ji} = 1, \forall i \in \mathcal{M}_D,
\end{equation}

we have that

\begin{equation}
\sum_{j} p_{ji} E(R_j) J_{d_t(i)} = R_i, \forall i \in \mathcal{M}_D,
\end{equation}

where

\begin{equation}
E(R_j) := \int_0^\infty e^{A^T h} R_j e^{Ah} \mu(dh).
\end{equation}

This result can be used to analyze if a given protocol yields the networked control system stable or to synthesize a protocol that achieves this.

**Analysis**: Note first that a given dynamic protocol specified by \( R_i > 0, i \in \mathcal{M}_D \), is equivalent to a dynamic protocol specified by

\begin{equation}
\tilde{R}_i = P + R_i > 0, i \in \mathcal{M}_D,
\end{equation}

where \( P \) can be any symmetric matrix such that \( P + R_i > 0 \). If we replace in (21) the matrices \( R_i \) by \( \tilde{R}_i \), given by (23), we obtain that (21), (23) are LMIs in the variables \( P \) and \( p_{ji} \) (using the fact that \( \sum_{j=1}^{m_D} p_{ji} = 1, \forall i \in \mathcal{M}_D \)).

**Synthesis**: If we allow \( R_i \) to be variables in (21), then (21) are in general BMIs. In fact, if we chose a basis \( B_i \) for the linear space of symmetric matrices, we have \( R_i = \sum_{i=1}^{m_D} p_{bi} B_i \) and (21) depends on the products \( p_{bi} B_i \). In this case the dynamic protocol, determined by the matrices \( R_i \) comes out from the solution to (21).
Remark 2 Stability condition (21) resembles a stability condition for Markov Jump Linear Systems [12]. Indeed, suppose that we choose protocol \( p \) for (12) according to the following stochastic rule:

\[
\text{Prob}[\omega_{k+1} = j | \omega_k = i] = \mu_{ij}, \; i, j \in \mathcal{M}_D
\]

\[
p(x(t_k^i), k) = d_1(\omega_k), \quad k \geq 0, \tag{24}
\]

where \( \omega_k \) is a Markov Chain with \( \mathcal{M}_D \geq m \) states, \( \sum_{j=1}^{\mathcal{M}_D} \mu_{ij} = 1, \forall i \in \mathcal{M}_D, \) and \( d_1 \) is described by (8). Then, it is well-known (cf. [12, Ch. 4]) that (18) with scheduling (24) is Mean Square Stable (\( \lim_{k \to \infty} E[x(t_k^i)^T x(t_k^i)] = 0, \forall x_0 \)) if and only if there exists \( n_x \times n_x \) symmetric matrices \( \{R_i > 0, i \in \mathcal{M}_D\} \) such that (21) holds for \( p_{ji} = \mu_{ij}, \forall i, j \in \mathcal{M}_D. \) Note, however, that contrarily to protocol (24), protocol (6), (8) is a deterministic state-dependent protocol and Theorem 1 assures stability in the deterministic sense (19).

Proof. Since the stability conditions of Theorem 3 are necessary and sufficient, there exists a static protocol with period \( m_S \) that yields the networked control system MES if and only if there exists \( \{R_i, i \in \mathcal{M}_S\} \) such that (25) holds for (12) with matrices defined by (15). This implies that if we consider a dynamic protocol with \( m_D = m_S, I_j = \{k \in \mathcal{M}_S : s(k) = j\}, j \in \mathcal{M}, \) then \( d_1(i) = s(i), \) for \( i \in \mathcal{M}_S \) and (21) holds with

\[
p_{ji} = \begin{cases} 
1, & \text{if } i < m_D \text{ and } j = i + 1, \\
1, & \text{if } i = m_D \text{ and } j = 1, \\
0 & \text{otherwise}
\end{cases}
\]

and with \( \{R_i, i \in \mathcal{M}_S = \mathcal{M}_D\} \) taken to be the solution to (25).

From the proof of Theorem 4 we see that the matrices \( \{R_i, i \in \mathcal{M}_D\} \) that characterize the dynamic protocol mentioned in its statement can be taken to be the solution to (25). Note that in the special case where \( m_D = m = m_S, \) Theorem 4 states that if there exists a round-robin protocol with period \( m_S = m, \) i.e., each node only transmits exactly once in a period, that yields the networked control system MES, then one can find a quadratic protocols as introduced in [7] that also yields the networked control system MES.

Remark 5 The fact that the stability conditions of Theorem 3 are necessary and sufficient is key to obtain Theorem 4. In the work [4] a similar reasoning to Theorem 4 can be used to prove that if the stability conditions provided there for quadratic protocols (cf. [4, Th. 3]) hold then so do the stability conditions for a static protocol in the special case where each node transmits only once in a period (cf. [7, Th. 3]). However, since the conditions provided in [4] are only sufficient for the RR protocol, it does not allow to conclude that if a stabilizing static protocol exists then so does a dynamic protocol, as stated in Theorem 4. Although [7] does not explicitly presents stability conditions for a static protocol, the same remarks should apply, since convex over-approximations introduce conservativeness.

3.2. Stability Result II and observer-protocol design

The following is our second stability result for (12). Similarly to Theorem 1 this theorem provides easy to test sufficient conditions for (12) to be Mean Exponentially Stable. However, it will allow us to obtain different results than Theorem 1.
Theorem 6 The system (12) with dynamic protocol (13) is MES if there exists an $n_x \times n_x$ symmetric matrix $W > 0$, scalars $\{\epsilon_{ij} \geq 0, i, j \in \mathcal{M}_D, i \neq j\}$ and $n_x \times n_x$ matrices $R_i, i \in \mathcal{M}_D$ such that

$$J_{d(i)}^T E(W) J_{d(i)} + \sum_{j=1, j \neq i}^{m_D} \epsilon_{ij} (R_j - R_i) - W < 0, \forall i \in \mathcal{M}_D$$

where $E(W) := \int_0^t (e^{Ah})^T W e^{Ah} \mu(dh)$. \hfill \Box

Given a quadratic protocol, i.e., specific values for the matrices $R_i$, testing if (26) holds is an LMI feasibility problem. To design a protocol for which mean exponential stability of the networked control feasibility problem. To design a protocol for which mean exponential stability of the networked control system is guaranteed, we can take the $\{R_i, i \in \mathcal{M}_D\}$ as additional unknowns and (26) should now be viewed as a BMI feasibility problem.

The proof of Theorem 6 builds upon establishing that if there exists a positive definite matrix $W$, positive constants $\epsilon_{ij}$ and matrices $R_i$ such that (26) holds then the quadratic function

$$V(x) := x^T W x$$

is a stochastic Lyapunov function for system (18) (which models (12) at sampling times) in the sense that the following condition holds for (18)

$$E[V(z_{k+1})] - V(z_k) \leq -x_k^T Z x_k, \forall x_k \in \mathbb{R}^{n_x},$$

for some $Z > 0$. The next result shows that under certain conditions, which include the case $m_D = m = 2$, i.e., only two nodes pertaining to the closed loop access the network, the converse holds. Let $\text{co}(A)$ denote the convex hull of a set $A$, and for each $i \in \mathcal{M}_D$ define the function

$$g_i^j(y) := y^T (W - J_{d(i)}^T E(W) J_{d(i)}) y$$

and the $m_D - 1$ functions

$$g_i^j(y) = y^T (R_j - R_i) y, \quad j \in \mathcal{M}_D - \{i\}.$$ 

Define also the following sets in $\mathbb{R}^{m_D}$

$$\mathcal{K}^i := \{\{g_i^0(y), g_i^1(y), \ldots, g_i^{j-1}(y), g_i^{j+1}(y), \ldots, g_i^{m_D}(y)\} | y \in \mathbb{R}^{m_D} \}, \quad i \in \mathcal{M}_D,$$

and

$$\mathcal{N} := \{(\eta_0, \eta_1, \ldots, \eta_{m_D}) | \eta_0 < 0, \eta_k > 0, 1 \leq k \leq m_D \}.$$

Theorem 7 Suppose that dynamic protocol (13) is regular. Then, if

$$\mathcal{K}^i \cap \mathcal{N} = \emptyset \Rightarrow \text{co}(\mathcal{K}^i) \cap \mathcal{N} = \emptyset, \forall i \in \mathcal{M}_D,$$

there exists an $n_x \times n_x$ symmetric matrix $W > 0$, scalars $\{\epsilon_{ij} \geq 0, i, j \in \mathcal{M}_D, i \neq j\}$ and $n_x \times n_x$ matrices $R_i, i \in \mathcal{M}_D$ such that (26) holds if and only if there exists a quadratic stochastic Lyapunov function taking the form (27) such that (28) holds. In particular (29) holds if $m_D = 2$.

The proof of Theorem 7 relies on the S-Procedure (see, e.g., [13]) which is a relaxation technique that can be used to provide easy to check stability conditions for linear systems with quadratic constraints. In particular, the condition (29) is a condition for the S-Procedure to be lossless, and is always satisfied in the case in which there is only one quadratic constraint ($m_D = 2$).

We show next that Theorem 6 allows to extend the observer-protocol design proposed in [8].

3.2.1. Observer Design

Suppose that we wish to estimate the state of the following plant

$$\dot{x}_P(t) = A_P x_P(t), \quad y(t) = C_P x_P(t), \quad x_P(0) = x_{P0}$$

where the $m$ outputs $y(t) = (y_1, \ldots, y_m), y_i \in \mathbb{R}^{n_i}$ are sent through a network that imposes i.i.d. intervals between transmissions to a remote observer. As in Section II, we denote by $\mu$ the measure that defines the inter-transmissions times $h_k = t_k + t - t_k$ and we let $\mathcal{M} = \{1, \ldots, m\}$. Let also $\Psi := \text{diag}(0, \ldots, 0, I_{n_m}, \ldots, I_{n_m})$, for $j \in \mathcal{M}$. A natural model based linear remote observer for this system is defined by

$$\dot{x}(t) = A_P \dot{x}(t) + L_k \Psi \varepsilon(x_c(t_k^-))(C_P \dot{x}(t_k^-) - y(t_k^-)),$$

where the observer gains $L_k$ to be designed are allowed to depend on the index $k$ and the map

$$c(x_c(t_k^-)) := \text{argmin}_{j \in \mathcal{M}} x_c(t_k^-)^T C_{Pj}^T S_j C_{Pj} x_c(t_k^-)$$

determines which node transmits at $t_k$ based on the estimation error $x_c(t_k^-) := \dot{x}(t_k^-) - x_P(t_k^-)$, where $\{S_j, j \in \mathcal{M}\}$ is a set of $m$ matrices. As argued in [8], the sensors should run a replica of the remote observer to access $\dot{x}(t)$, which allows each node to encode in the message arbitration field $x_c(t_k^-)^T C_{Pj}^T S_j C_{Pj} x_c(t_k^-)$, where $C_{Pj} x_c(t_k^-) = C_P \dot{x}(t_k^-) - y_j(t_k^-), j \in \mathcal{M}$.
The resulting estimation error $x_e := \hat{x} - x_p$ evolves according to

$$\dot{x}_e(t) = A_P x_e(t) + L_k \Psi(x_e(t_{-k})) C_P x_e(t_{-k}).$$  \hspace{1cm} (32)

We can cast this problem into the framework of (12) with dynamic protocol (13) by adding an auxiliary variable $v$ that holds the value of $x_e(t_k)$ between transmission times, considering $x = (x_e, v)$ and

$$A = \begin{bmatrix} A_P & I_{n_P} \\ 0_{n_P} & 0_{n_P} \end{bmatrix},$$

$$J_i = \begin{bmatrix} I_{n_P} & 0_{n_P} \\ 0_{n_P} & 0_{n_P} \end{bmatrix},$$

$$L_k = \begin{bmatrix} 0 \\ [\Psi_i C_P \ 0] \end{bmatrix},$$

$$R_i = \begin{bmatrix} C_P^T \Psi_i C_P & 0 \\ 0 & 0_{n_P} \end{bmatrix}, \quad 1 \leq i \leq m = m_D.$$  \hspace{1cm} (33)

In the following theorem, we propose a method to obtain observer gains $L_k$ that yield the networked control system MES. To state the result we need the following assumption:

$$H(s) := \int_0^s e^{Ap} dr \text{ is invertible for every } s \in [0, \gamma].$$  \hspace{1cm} (34)

While this assumption holds for a large class of matrices $A_P$, it is possible to construct examples where it does not, as in the case where $\gamma > s = 2\pi$ and $A_P = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, in which case $H(s) = 0$.

**Theorem 8** Suppose that (34) hold. If there exists a $n_P \times n_P$ symmetric matrix $P > 0$, a $m \times m$ matrix $Y$, a $n_P \times m$ matrix $M$, a $m \times m$ matrices $\{S_i, i \in M\}$, and scalars $\{c_{ij} \geq 0, i, j \in M\}$, such that

$$F(P) + D M \Psi_i C_P + (D M \Psi_i C_P)^T + C_P^T Y C_P + \sum_{j=1, j\neq i}^m c_{ij} (C_P^T S_j C_P - C_P^T S_i C_P) - P < 0, \quad \forall i \in M,$$

$$\begin{bmatrix} P & M \\ M^T & Y \end{bmatrix} > 0,$$  \hspace{1cm} (35)

$$\text{where } F(P) := \int_0^\gamma e^{Ap} \mu(dr) \text{ and } D := \int_0^\gamma e^{Ap} \mu(dr),$$

then we have that the observer gain $L_k = H(h_k)^{-1} P^{-1} M$ yields (12) with matrices (33) MES.

**Remark 9** When the intervals between transmission are constant, one can show that the stability conditions (35) and (36) are equivalent to the ones given in [8], where such an assumption is made. In this case, the matrices $L_k$ do not depend on $k$, and can therefore be computed off-line.

### 3.3. Extensions to handle delays and packet drops

Theorems 6 and 1 can be extended to the case where the network introduces packet drops and delays modeled by (16) with matrices (17). We state these extensions next.

**Theorem 10** System (16) with dynamic protocol (13) is MES if there exist scalars $\{0 \leq p_{ji} \leq 1, \ i \neq j \in \mathcal{M}_D\}$, with $\sum_{j=1}^{m_D} p_{ji} = 1, \forall i \in \mathcal{M}_D$, and $n_x \times n_x$ symmetric matrices $\{R_i > 0, \ i \in \mathcal{M}_D\}$ such that

$$\sum_{l=1}^{n_q} \omega_l (K_{d_1(l)}^T \sum_{j=1}^{m_D} p_{ji} E(R_j) K_{d_1(l)}) - R_i < 0, \forall i \in \mathcal{M}_D,$$

where

$$E(R_j) := \int_0^\gamma (e^{Ah_j - s Le^{A}\chi})(dh, ds).$$  \hspace{1cm} (37)

**Theorem 11** System (16) with dynamic protocol (13) is MES if there exists an $n_x \times n_x$ symmetric matrix $W > 0$, scalars $\{c_{ij} \geq 0, i, j \in \mathcal{M}_D, i \neq j\}$ and $n_x \times n_x$ matrices $R_i, \ i \in \mathcal{M}_D$ such that

$$\sum_{l=1}^{n_q} \omega_l (K_{d_1(l)}^T E(W) K_{d_1(l)}) + \sum_{j=1, j\neq i}^m c_{ij} (R_j - R_i) - W < 0, \forall i \in \mathcal{M}_D,$$

where $E(W)$ is defined as in (37).

© 2012 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

Prepared using asjcauth.cls
transmissions intervals provided by Theorem obtained with the necessary and sufficient conditions from solving (the maximum support of a uniform distribution that i.e., the protocol is to be designed. Note that Theorem results obtained with Theorem matches well with the results in Table examples. The results in Table as in (and in (and exponential with expected value assures that the values obtained with Theorem allowing in Theorem in Theorem and TRIDENT (Contract No. 248497).

Motivated by recent research (cf., e.g., [14], [15], [16]) on H2 optimal problems in the context of networked control systems a possible line for future research is to consider related optimal control problems in the scenarios addressed in the present paper.

VI. Acknowledgements

This work is supported by the Dutch Science Foundation (STW) and the Dutch Organization for Scientific Research (NWO) under the VICI grant “Wireless controls systems: A new frontier in Automation” (No. 11382), by the European 7th Framework Network of Excellence by the project “Highly-complex and networked control systems (HYCON2-257462), by the NSF CNS-0720842, by project FCT [PEst-OE/EEI/LA0009/2011], and by the EU Project TRIDENT (Contract No. 248497).

A. Proofs

Proof. (of Theorem 1) The discrete-time process zk, described by (18), can be easily shown to be a Markov process due to the i.i.d. assumption on hk. In particular

\[ \mathbb{E}[z_{k+1} | V(z_{k+t+m})] = \mathbb{E}_z[V(z_{k+t+m})] \] (38)

for any bounded measurable function V, where \( \mathbb{E}_z \) denotes expectation given \( z_k \), i.e., \( \mathbb{E}_z[.] := \mathbb{E}[.|z_k] \). If one can find a function V and positive constants \( c_1, c_2, c_3 \), such that

\[ c_1 \| z \|^2 \leq V(z) \leq c_2 \| z \|^2, \forall z \in \mathbb{R}^n \] (39)

and

\[ \mathbb{E}_z[V(z_{k+1})] - V(z_k) \leq -c_3 \| z_k \|^2, \forall z_k \in \mathbb{R}^n \] (40)

then we can prove that

\[ \mathbb{E}[z_{k+1} | z_k] \leq c \alpha^k z_0^\top z_0, \forall k \geq 0 \text{ for some } 0 < \alpha < 1, c > 0. \] (41)
which implies MES for (12) according to the definition (19) since \( x(t_k) = J_z z_k \) for some \( i \in M \). In fact if (39) and (40) hold then

\[
E_{z_k}[V(z_{k+1})] \leq \alpha V(z_k)
\]

where \( 0 < \alpha = 1 - \frac{\delta}{2} < 1 \) must be greater than zero since \( V \) is positive. From (38) and (42) we can conclude that

\[
E_{z_0}[V(z_k)] \leq \alpha^k V(z_0).
\]

From (39) and (43) we obtain

\[
E_{z_0}[\|z_k\|^2] \leq \frac{c_0}{\alpha^k} \|z_0\|^2, \quad k \geq 0.
\]

Take \( V(z_k) := \min_{i \in M_D} z_k^T R_i z_k \), which satisfies (39) since \( R_i > 0 \) \( \forall i \in Q \). Suppose that \( z_k \) is such that \( i = d_2(z_k) = \arg\min_{i \in M_D} z_k^T R_i z_k \), i.e., \( V(z_k) = z_k^T R_i z_k \). Note that, in the case where the minimum is achieved simultaneously for several value of the index \( i \), any of these indexes can be chosen without affecting the present proof. Then, for any \( p_{ji} \geq 0 : \sum_{j=1}^{m_D} p_{ji} = 1 \), we have that

\[
E_{z_k}[V(z_{k+1})] = E_{z_k}[\min_{j \in M_D} z_k^T J_{d_1(i)} e^{A_{th} R_j e^{Ah} J_{d_1(i)} z_k}]
\]

\[
\leq E_{z_k}\left[ \sum_{j=1}^{m_D} p_{ji} z_k^T J_{d_1(i)} e^{A_{th} R_j e^{Ah} J_{d_1(i)} z_k} \right]
\]

\[
= z_k^T J_{d_1(i)} \sum_{j=1}^{m_D} p_{ji} E(R_j) J_{d_1(i)} z_k
\]

Suppose we choose \( p_{ji} \) such that (21) holds, i.e.,

\[
J_{d_1(i)} \sum_{j=1}^{m_D} p_{ji} E(R_j) J_{d_1(i)} = -R_i = -Q_i
\]

for some \( Q_i > 0 \). Then from (44) we conclude that

\[
E_{z_k}[V(z_{k+1})] = V(z_k) \leq -z_k^T Q_i z_k, \quad i = d_2(z_k), \text{ which implies (40) and concludes the proof.}
\]

**Proof.** (of Theorem 6) As in the proof of the Theorem 1 it suffices to find a function \( V \) such that (39), (40) hold. Take \( V(z) = z^T W z \), where \( W \) is the solution to (26) and suppose that \( i = \arg\min_{j \in M_D} z^T R_j z \). Then

\[
E_{z_k}[V(z_{k+1})] - V(z_k) = z_k^T J_{d_1(i)} E(W) J_{d_1(i)} - W] z_k
\]

\[
= -z_k^T \left[ \sum_{j=1, j \neq i}^{m_D} c_{ij}(R_j - R_i) + Q_i \right] z_k
\]

where \( Q_i > 0 \). Since \( i = \arg\min_{j \in M_D} z_k^T R_j z_k \) we have that \( z_k^T [\sum_{j=1, j \neq i}^{m_D} c_{ij}(R_j - R_i)] z_k \geq 0 \) for every \( z_k \in \mathbb{R}^{m_D} \) and therefore from (45) we conclude that \( V \) satisfies (40). It is also clear that \( V \) satisfies (39), which concludes the proof. ■

**Proof.** (of Theorem 7) If for a given \( z_k, i \in M_D \) is such that \( i = \arg\min_{j \in M_D} z_k^T R_j z_k \), which is equivalent to

\[
g_k^i(z_k) \geq 0, \quad \forall j \in M - \{i\},
\]

then we have that \( z_{k+1} = e^{Ah} J_{d_1(i)} z_k \) in which case (28) boils down to

\[
g_0^i(z_k) > 0.
\]

The fact that the dynamic protocol is regular implies that there exists at least one \( y \) such that (46) holds with strict inequality. Then, a straightforward adaptation of the lossless S-Procedure theorem provided in [13, Th.2] to handle strict inequalities in the objective function \( g^i_0(y) \) assures that, under condition (29), (47) holds if and only if there exists \( c_{ij} \geq 0, \quad j \in M_D - \{i\} \) such that

\[
g^i_0(y) - \sum_{j \in M_D - \{i\}} c_{ij} g^j_0(y) > 0, \quad \forall y \in \mathbb{R}^{m_D}.
\]

The result then follows by noticing that (48) is equivalent to (26). The fact that (29) holds if \( m_D = 2 \) (in which case there is only one quadratic constraint) follows from [13, Th.3]. ■

**Proof.** (of Theorem 8) We first prove that if there exists \( P > 0 \) such that

\[
\int_0^1 (e^{A_{ph}} + H(h) L_k \Psi_s C_P) ^T P \ldots
\]

\[
\ldots (e^{A_{ph}} + H(h) L_k \Psi_s C_P) \mu (dh)
\]

\[
+ \sum_{j=1, j \neq i}^m e_{ij}(C_{ij}^P S_j C_P - C_{ij}^P S_i C_P) - P < 0
\]

then (26) holds for (12) with matrices (33). Note that we can assume that \( d_1(i) = i, \forall i \in M \) since if this is not the case we can relabel the sensor nodes in such a way that this holds. For \( A, J_i \) given by (33) we have that

\[
e^{Ah} J_i = \begin{bmatrix}
  e^{A_{ph}} + H(h) L_k \Psi_s C_P \\
  L_k \Psi_s C
\end{bmatrix}
\]

Using this expression and considering \( W = \text{diag}([P e_{1,n}]) \) in (26) where \( P \) satisfies (49) and
\[ \int_0^\infty \begin{bmatrix} B(h) & 0 \\ 0 & 0 \end{bmatrix} \mu(\mathbf{d}h) - \begin{bmatrix} P & 0 \\ 0 & eL_{n_p} \end{bmatrix} + \sum_{j=1, j \neq i}^m c_{ij} \begin{bmatrix} C^T_i P_j S_j P_j - C^T_i P_j S_j P_j & 0 \\ 0 & 0 \end{bmatrix} n_p < 0, \; \forall i \in \mathcal{M}. \]

where

\[
B(h) = \epsilon(L_K \Psi_i C_P)^T (L_K \Psi_i C_P) + \left( e^{A_{wh}} + H(h)L_K \Psi_i C_P \right)^TP(e^{A_{wh}} + H(h)L_K \Psi_i C_P). \]

From this expression we conclude that if (49) holds then (50) holds for sufficiently small \( \epsilon \). Set \( L_K = H(h_k)^{-1} P^{-1} M \) for a \( n_p \times m \) matrix. Then (49) can be written as

\[
F(P) + DMP_i C_P + (DMP_i C_P)^T + C_i^T M P^{-1} M C_P + \sum_{j=1, j \neq i}^m c_{ij} \left( C^T_j S_j C_P - C^T_j S_j C_P \right) - P < 0, \; \forall i \in \mathcal{M}. \]

If we let \( Y > 0 \) be such that \( MT^{-1} M < Y \), which applying the Shur complement can be seen to be equivalent to (36), we see that if (35) holds then (51) holds, which concludes the proof. \Box

**Proof.** (of Theorem 10) The proof is obtained by following the same steps of Theorem 1 and by noticing that using a similar reasoning to (44) one obtains

\[
\mathbb{E}_Z_k [V(z_{k+1})] \leq \sum_{l=1}^{n_z} w_l \left( K^l_{d_{k+1}(i)} \right)^T \mathbb{E} \left( \sum_{j=1}^{m_{P_j}} p_{ij} E(R_j) \right) K^l_{d_{k+1}(i)}. \]

**Proof.** (of Theorem 11) The proof is obtained by following the same steps of Theorem 6 and by noticing that in (45)

\[
\mathbb{E}_Z_k [V(z_{k+1})] = \sum_{l=1}^{n_z} w_l \left( K^l_{d_{k+1}(i)} \right)^T \mathbb{E}(W) K^l_{d_{k+1}(i)}. \]

**REFERENCES**


Table 1. Stability results for the batch reactor example–MEF-TOD and Round Robin protocol. NA stands for Not Available

<table>
<thead>
<tr>
<th></th>
<th>Dynamic Protocol</th>
<th>Static Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>no drops</td>
<td>p = 0.5</td>
</tr>
<tr>
<td>Max. ( \gamma )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( h_k \sim \text{Uni}(\gamma) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Results from [3]</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Results from [5]</td>
<td>0.0372</td>
<td>0.0170</td>
</tr>
<tr>
<td>Results from [7]</td>
<td>0.11</td>
<td>NA</td>
</tr>
<tr>
<td>Ths. 6 and 11</td>
<td>0.0550</td>
<td>0.024</td>
</tr>
<tr>
<td>Th. 3</td>
<td>0.111</td>
<td>NA</td>
</tr>
<tr>
<td>Th. 4</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 2. Stability results for the batch reactor example–Protocol design, no packet drops

<table>
<thead>
<tr>
<th></th>
<th>Dyn. Prot. Design (Th.1)</th>
<th>Static Prot. (Th.3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. ( \gamma )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( h_k \sim \text{Uni}(\gamma) )</td>
<td>0.140</td>
<td>0.112</td>
</tr>
</tbody>
</table>