

# Stability analysis of networked control systems with direct-feedthrough terms: Part I - The nonlinear case

Navid Noroozi   Romain Postoyan   Dragan Nešić   Stefan H. J. Heijmans   W. P. Maurice H. Heemels

**Abstract**—This paper investigates the stability of nonlinear networked control systems (NCSs) with dynamic controllers that possess direct-feedthrough terms (i.e. that are of relative degree zero). The presence of the direct-feedthrough terms obstructs the application of existing stability results for NCSs. Indeed, the uniform global exponential stability (UGES) of an auxiliary system induced by the plant and the network protocol needs to be verified. In prior work, this auxiliary system depends solely on the protocol (and not on the plant or on the controller) and, consequently, the analysis is simpler. Checking UGES of this auxiliary system turns out to be non-trivial when direct-feedthrough terms are present (even for the simplest protocols). Still, we are able to show UGES of the auxiliary system for Round-Robin (RR) and Try-Once-Discard (TOD) network protocols, which, together with other requirements on the maximum allowable transmission intervals (MATIs), ensures the stability of the overall system. We also show that the analysis and proofs can be greatly simplified in cases when the control inputs are sent over one communication channel and the plant outputs over a separate channel.

## I. INTRODUCTION

Networked control systems (NCSs) are control systems in which sensor data and control commands are being communicated over a wired or wireless communication network, such as CAN, FlexRay and WirelessHart [1]-[5]. The recent interest in NCSs is motivated by the many benefits they offer, such as the ease of maintenance and installation, configuration flexibility, reduced weight and volume and lower cost. Moreover, NCSs are applied in a broad range of systems, such as mobile sensor networks, remote surgery, automated highway systems and unmanned aerial vehicles. However, in order to utilise the full potential of this emerging technology, we need novel design and analysis approaches that are attuned to the issues arising in this context.

N. Noroozi is with the Faculty of Computer Science and Mathematics, University of Passau, Innstraße 33, 94032 Passau, Germany email: navidnoroozi@gmail.com.

R. Postoyan with the Université de Lorraine, CRAN, UMR 7039 and the CNRS, CRAN, UMR 7039, France email: romain.postoyan@univ-lorraine.fr. His work was partially supported by the ANR under the grant COMPACS (ANR-13-BS03-0004-02).

D. Nešić with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, 3010, Victoria, Australia email: dnesic@unimelb.edu.au. His work was supported under the Australian Research Council under the Discovery Project DP1094326.

S.H.J. Heijmans and W.P.M.H. Heemels are with the Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands emails: {m.heemels, s.h.j.heijmans}@tue.nl. Their work is supported by the Innovational Research Incentives Scheme under the VICI grant Wireless control systems: A new frontier in automation (No. 11382) awarded by STW (Dutch Science Foundation) and NWO (The Netherlands Organization for Scientific Research).

A popular design approach for NCSs is via the so-called *emulation* method, see [6]-[13]. The idea is to first ignore communication constraints and design a continuous-time controller for a continuous-time plant. Then, the controller is implemented via the network and it is shown (under suitable conditions) that the closed-loop system is stable when the transmission frequency is sufficiently high, i.e. the maximum allowable transmission interval (MATI) is sufficiently small. This approach was shown to work well for a large class of systems whose scheduling protocols are uniformly globally exponentially stable (UGES) in an appropriate sense. The emulation approach enjoys considerable advantages in terms of its simplicity and applicability to a large class of nonlinear NCSs. Indeed, any continuous-time design approach can be used to obtain the controller.

Most existing emulation results on NCSs concentrate on the stabilization using dynamic controllers *without* direct-feedthrough terms, see [7], [8], [12] for example. However, direct-feedthrough terms are essential to model controllers commonly used in the industry such as proportional-integral(-derivative) regulators. Considering dynamic controllers with direct-feedthrough terms in general complicates the analysis as it will be shown in this paper. Some results exist in the NCSs literature when the channel is only used to ensure the communication between the sensors and the controller, or between the controller and the actuators but not both, see [11] for instance. The main purpose of the present paper is to consider NCSs with dynamic controllers that contain direct-feedthrough inputs and for which both the sensor data and the control input are transmitted over a network. In particular, we focus on the effect of scheduling and sampling<sup>1</sup>. First, we extend the modelling framework developed in [7]-[10] to cover NCSs with direct-feedthrough terms. Then, we revisit the notion of uniformly globally exponentially stable (UGES) scheduling protocols as given in [7] in order to incorporate direct-feedthrough terms. In particular, an auxiliary system induced by the plant and the protocol turns out to be instrumental in the stability analysis of NCSs designed via emulation; this auxiliary system depends only on the protocol in the simpler case of dynamic controllers without direct-feedthrough terms, as was considered in [7], [8], [9], [12], but this is no longer true in the NCS setup considered here.

We investigate two cases. In the first case, we assume that all inputs and outputs are sent over one serial communication

<sup>1</sup>We foresee that our results can be extended under minor changes to the case where the transmission delays are not negligible as well as in presence of quantization errors, in view of [10], [12].

channel and that some nodes may contain both inputs and outputs. Using a Lyapunov based approach, we show that the auxiliary system induced by the plant and the protocol is UGES for Round-Robin (RR) and Try-Once-Discard (TOD) protocols. In the second case, we assume that there are two dedicated channels that are respectively used to send the outputs and the inputs over the network. In this case, the analysis is greatly simplified since the auxiliary system induced by the plant and the protocol is a cascaded system. Once UGES of the auxiliary system induced by the plant and the protocol is established, we can use stability results of [7]-[10] to conclude stability of NCSs.

In the companion paper [14], we address the scenario where the plant and the controller dynamics are *linear* and *both* contain direct-feedthrough terms, which is a source of additional difficulties, while here we investigate *nonlinear* systems but *only* the controller (or the plant) has direct-feedthrough terms.

The remainder of this paper is organized as follows. Section II provides the used notation. The problem is stated and the model formulation is developed in Section III. The main results are given in Sections IV-V. Section VI provides the concluding remarks. All the proofs of our results are omitted due to space constraints.

## II. NOTATION

Let  $\mathbb{R} := (-\infty, +\infty)$ ,  $\mathbb{R}_{\geq 0} := [0, +\infty)$ ,  $\mathbb{R}_{> 0} := (0, +\infty)$ ,  $\mathbb{Z}_{\geq 0} := \{0, 1, 2, \dots\}$ ,  $\mathbb{Z}_{\leq 0} := \{\dots - 2, -1, 0\}$  and  $\mathbb{Z}_{> 0} := \{1, 2, \dots\}$ . The Euclidean norm of a vector  $x \in \mathbb{R}^n$  and its 1-norm are respectively denoted  $|x|$  and  $|x|_1$ . We denote the identity matrix of dimension  $n \in \mathbb{Z}_{> 0}$  by  $I_n$ . A function  $f: \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}^n$  belongs to  $\ell_\infty$  if  $\sup\{|f(i)| : i \in \mathbb{Z}_{\geq 0}\}$  is bounded. By convention, for any  $m \in \mathbb{Z}_{\leq 0}$ ,  $\sum_{i=1}^m w(i) = 0$  where  $w: \mathbb{Z}_{> 0} \rightarrow \mathbb{R}$ . The function  $\theta: \mathbb{Z} \rightarrow \{0, 1\}$  is defined by  $\theta(m) := 1$  when  $m \in \mathbb{Z}_{> 0}$  and  $\theta(m) := 0$  when  $m \in \mathbb{Z}_{\leq 0}$ . We write  $(x, y)$  to represent  $[x^T, y^T]^T$  for any  $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ .

## III. PROBLEM STATEMENT

Consider the nonlinear plant model

$$\begin{aligned} \dot{x}_p &= f_p(x_p, u) \\ y &= g_p(x_p), \end{aligned} \quad (1)$$

where  $x_p \in \mathbb{R}^{n_p}$  is the plant state,  $u \in \mathbb{R}^{n_u}$  is the control input and  $y \in \mathbb{R}^{n_y}$  is the plant output. We follow the emulation approach to design the controller. Hence, we assume that we know a continuous-time controller, which stabilizes the origin of system (1) in the absence of network. We focus on dynamic controllers of the form

$$\begin{aligned} \dot{x}_c &= f_c(x_c, y) \\ u &= g_c(x_c, y), \end{aligned} \quad (2)$$

where  $x_c \in \mathbb{R}^{n_c}$  is the controller state. In contrast with e.g. [7]-[13], the controller output map  $g_c$  depends on the plant output  $y$ . This direct-feedthrough term appears for standard

dynamic controllers such as the popular linear proportional-integral(-derivative) controllers. This term prevents the application of the results in [7]-[13] because it changes the protocol equation as we will see later.

*Remark 1:* The forthcoming results apply mutatis mutandis to the case where the plant (1) has direct-feedthrough terms but not the controller (2). The presence of direct-feedthrough terms in both (1) and (2) lead to an algebraic constraint, which requires particular care as shown in [14] where linear NCSs are studied.  $\square$

*Remark 2:* Existing results as in [7]-[13] allow for the controller output map  $g_c$  to depend on  $y$  when the controller is directly connected to the plant, see Section IX in [7]. When this is not the case, substantial differences arise, which prevent to apply the results of [7]-[13].  $\square$

We consider the scenario where the plant and the controller are connected via a digital network that is composed of  $\ell \in \mathbb{Z}_{> 0}$  nodes. A node corresponds to a collection of sensors and/or actuators. Let  $\ell_y \in \mathbb{Z}_{> 0}$  denote the number of nodes which are not associated to any actuators. Similarly, let  $\ell_u \in \mathbb{Z}_{\geq 0}$  denote the number of nodes which are not associated to any sensors. Hence  $\ell_u + \ell_y \leq \ell$  (there is no equality as a node may be associated to both sensors and actuators).

The network generates various constraints on the communication of both  $u$  and  $y$ . In this paper, we concentrate on the effect due to sampling and scheduling. Transmissions occur only at some given time instants  $t_j, j \in \mathbb{Z}_{\geq 0}$ , such that  $v \leq t_{j+1} - t_j \leq \tau_{\text{MATI}}$ , where  $v \in (0, \tau_{\text{MATI}}]$  and  $\tau_{\text{MATI}}$  respectively represent the minimum time and the maximum time between any two transmission instants and  $v$  can be arbitrarily small. Furthermore, at each transmission instant, a single node is granted access to the network. This selection is done by the *scheduling protocol*. As in [7], the overall system can be modelled by the following impulsive system

$$\begin{aligned} \dot{x}_p &= f_p(x_p, \hat{u}) & t \in [t_j, t_{j+1}] \\ y &= g_p(x_p) \\ \dot{x}_c &= f_c(x_c, \hat{y}) & t \in [t_j, t_{j+1}] \\ u &= g_c(x_c, \hat{y}) \\ \hat{y} &= \hat{f}_p(x_p, x_c, \hat{y}, \hat{u}) & t \in [t_j, t_{j+1}] \\ \hat{u} &= \hat{f}_c(x_p, x_c, \hat{y}, \hat{u}) & t \in [t_j, t_{j+1}] \\ \hat{y}(t_j^+) &= y(t_j) + h_y(j, e(t_j)) \\ \hat{u}(t_j^+) &= u(t_j) + h_u(j, e(t_j)) \end{aligned} \quad (3)$$

where  $x := (x_p, x_c) \in \mathbb{R}^{n_x}$ , and  $\hat{u} \in \mathbb{R}^{n_u}$  and  $\hat{y} \in \mathbb{R}^{n_y}$  are, respectively, the vector of the most recently transmitted controller output value and the vector of the most recently transmitted plant output value. These two variables are generated by the holding function  $\hat{f}_p$  and  $\hat{f}_c$  between two successive transmission instants. The use of zero-order-hold devices leads to  $\hat{f}_p = 0$  and  $\hat{f}_c = 0$  for instance. The functions  $h_y$  and  $h_u$  model the network protocol that can, for instance, be RR, TOD, or any other protocol discussed in [7] and [8]. In addition,  $e := (e_y, e_u) \in \mathbb{R}^{n_e}$  denotes the network-induced errors where  $e_y := \hat{y} - y \in \mathbb{R}^{n_y}$  and

$e_u := \hat{u} - u \in \mathbb{R}^{n_u}$ . It is more convenient to rewrite (3) as

$$\dot{x} = f(x, e) \quad (4a)$$

$$\dot{e} = g(x, e) \quad (4b)$$

$$e(t_j^+) = h(j, e(t_j), x(t_j)) \quad (4c)$$

where  $f$  and  $g$  are assumed to be continuously differentiable and are obtained by direct calculations from (3) (cf. [7] for more details).

Contrary to (14) in [7], the function  $h$  not only depends on  $(j, e)$  but also on  $x$  in (4c). As seen below, this extension comes from the fact that  $g_c$  in (3) depends on  $\hat{y}$ . Let us illustrate this point through the RR protocol, which periodically grants access to each node. Let  $t_j, j \in \mathbb{Z}_{\geq 0}$ , according to (3),

$$\begin{aligned} e_y(t_j^+) &= \hat{y}(t_j^+) - y(t_j^+) \\ &= y(t_j) + h_y(j, e(t_j)) - y(t_j) \\ &= h_y(j, e(t_j)) \\ e_u(t_j^+) &= \hat{u}(t_j^+) - u(t_j^+) \\ &= u(t_j) + h_u(j, e(t_j)) - u(t_j^+) \\ &= h_u(j, e(t_j)) + g_c(x_c(t_j), \hat{y}(t_j)) \\ &\quad - g_c(x_c(t_j^+), \hat{y}(t_j^+)) \\ &= h_u(j, e(t_j)) + g_c(x_c(t_j), y(t_j) + e_y(t_j)) \\ &\quad - g_c(x_c(t_j), y(t_j) + h_y(j, e(t_j))) \end{aligned} \quad (5)$$

where  $h_y(j, e) = (I_{n_y} - \Delta_y(j))e_y$  and  $h_u(j, e) = (I_{n_u} - \Delta_u(j))e_u$  with

$$\Delta_y(j) := \text{diag}(\delta_{y_1}(j)I_{n_{y_1}}, \dots, \delta_{y_{\ell'_y}}(j)I_{n_{y_{\ell'_y}}}) \quad (6)$$

$$\Delta_u(j) := \text{diag}(\delta_{u_1}(j)I_{n_{u_1}}, \dots, \delta_{u_{\ell'_u}}(j)I_{n_{u_{\ell'_u}}}). \quad (7)$$

In (6)-(7),  $\ell'_y \in \mathbb{Z}_{\geq 0}$  and  $\ell'_u \in \mathbb{Z}_{\geq 0}$  are respectively the number of nodes associated at least to one sensor and to at least one actuator (hence  $\ell'_y \geq \ell_y$ ,  $\ell'_u \geq \ell_u$  and  $\ell'_y + \ell'_u \geq \ell$ ),  $e_y = (e_{y_1}, \dots, e_{y_{\ell'_y}})$  and  $e_u = (e_{u_1}, \dots, e_{u_{\ell'_u}})$  (after reordering, if needed),  $e_{y_i} \in \mathbb{R}^{n_{y_i}}$  for  $i \in \{1, \dots, \ell'_y\}$ , and  $e_{u_i} \in \mathbb{R}^{n_{u_i}}$  for  $i \in \{1, \dots, \ell'_u\}$ , so that  $n_e = \sum_{i \in \{1, \dots, \ell'_y\}} n_{y_i} + \sum_{i \in \{1, \dots, \ell'_u\}} n_{u_i}$ . To define the  $\delta$  terms in (6)-(7), we need to number the nodes. We use for that purpose the mapping  $\pi$  such that  $\pi(y_i)$  is the number of the node associated to  $y_i$ , for  $i \in \{1, \dots, \ell'_y\}$ , and  $\pi(u_i)$  is the number of the node associated to  $u_i$ , for  $i \in \{1, \dots, \ell'_u\}$ . We consider, for  $i \in \{1, \dots, \ell'_y\}$ ,

$$\delta_{y_i}(j) := \begin{cases} 1 & \text{when } j = \pi(y_i) - 1 + \ell k, k \in \mathbb{Z}_{\geq 0} \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

and we similarly define  $\delta_{u_i}$  for  $i \in \{1, \dots, \ell'_u\}$ .

The dynamics of  $e_u$  at each transmission involve  $y$  and thus  $x$  because of the term  $g_c(x_c, y + e_y) - g_c(x_c, y + h_y(j, e))$  in (5). This is due to the direct-feedthrough term in (2), which implies that  $u(t_j^+) \neq u(t_j)$  in general<sup>2</sup>. On the other hand, the dynamics of  $e_y$  at each transmission is the same as in

<sup>2</sup>When the controller is such that  $u = g_c(x_c)$  (as in [7]),  $u(t_j^+) = u(t_j)$  and we recover the protocol equation studied in [7].

[7] due to the absence of direct-feedthrough terms in (1). To be more precise, for the RR protocol,

$$e_y(t_j^+) = (I_{n_y} - \Delta_y(j))e_y(t_j) \quad (9a)$$

$$\begin{aligned} e_u(t_j^+) &= (I_{n_u} - \Delta_u(j))e_u(t_j) \\ &\quad + \left( I_{n_u} - \Delta_u(j) \right) \left( g_c(x_c(t_j), y(t_j) + e_y(t_j)) \right. \\ &\quad \left. - g_c(x_c(t_j), y(t_j) + (I_{n_y} - \Delta_y(j))e_y(t_j)) \right) \end{aligned} \quad (9b)$$

where the product  $(I_{n_u} - \Delta_u)$ , which multiples  $g_c(x_c, y + e_y) - g_c(x_c, y + (I_{n_y} - \Delta_y)e_y)$ , is required to accommodate transmissions in which a node corresponding to a collection of *both* sensors and actuators grants access to the network. As a result, it is not clear that the stability properties proved for the RR protocol in [7] are preserved in this case. Going back to the general case, the fact that  $h$  depends on both the network protocol model, as well as the plant model is in stark contrast with [7], [8], and introduces significant technical difficulties in the analysis and is the topic of this paper. We will consider in this context commonly used protocols, such as RR, TOD<sup>3</sup> protocols and we will also present results for any other protocols discussed in [7], [8], [10] in the two channel case. In particular, we focus on the following stability definition for system (4c) which extends Definition 7 in [7].

*Definition 1:* The discrete-time system  $e(j+1) = h(j, e(j), x(j))$  is UGES with a Lyapunov functional  $W: \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_e} \times \ell_\infty \rightarrow \mathbb{R}_{\geq 0}$  if there exist  $a_1, a_2 \in \mathbb{R}_{>0}$  and  $\rho \in [0, 1)$  such that for all  $j \in \mathbb{Z}_{\geq 0}, e \in \mathbb{R}^{n_e}, x \in \ell_\infty$  the following holds

$$a_1 |e| \leq W(j, e, x) \leq a_2 |e| \quad (10)$$

$$W(j+1, h(j, e, x), x) \leq \rho W(j, e, x). \quad (11)$$

□

The UGES property is required to combine our results with those developed in [7]-[13] to guarantee that (4) is uniformly globally asymptotically stable (UGAS). Particularly, in [7]-[13], conditions on (4) are provided to ensure asymptotic properties of the origin when the mapping  $h$  only depends on  $j$  and  $e$ . The general idea is to assume that (i) the controller (2) is such that the system  $\dot{x} = f(x, e)$  satisfies a robust asymptotic stability property with respect to  $e$ , (ii) the system  $\dot{e} = g(x, e)$  satisfies an exponential growth condition, (iii) the origin of the system  $e^+ = h(j, e)$  is UGES or UGAS with a Lyapunov function(al)  $W$ , which is locally Lipschitz in  $e$ . Item (iii) is shown to be verified by RR and TOD protocols in [7], and other examples are provided in [8], [10], [13]. Then, by selecting  $\tau_{\text{MATI}}$  sufficiently small, the stability of the overall system can be guaranteed. We can apply the same approach to analyse the stability of system (4). However, it is no longer clear that the  $e$ -system at jumps is still UGES (or UGAS) when the mapping  $h$  depends on  $x$  for standard protocols such as RR and TOD. More generally, if the system

<sup>3</sup>TOD consists in granting access to the node, which has the largest local network-induced error; a mathematical definition is given in Section IV-C.

$e^+ = h(j, e, 0)$  is UGES (or UGAS), it may be the case that this property is lost when  $x \neq 0$ , which arises when the controller (or the plant) has direct-feedthrough terms.

The objective of this paper is to provide conditions under which, if the origin of system  $e^+ = h(j, e, 0)$  is UGES (or UGAS), this property is preserved when  $x \neq 0$ .

#### IV. ONE CHANNEL CASE

In this section, we consider the case of networks in which all nodes are transmitted over one channel and any given node may contain both inputs and outputs. In particular, we provide conditions for system (4c) to be UGES. While results of this paper are applicable to a large class of protocols, for illustration purposes we will concentrate on a special form of protocols including two particular examples that have been studied in [7].

##### A. UGES of system (4c)

Consider the case where system (4c) can be written as

$$\begin{aligned} e_y(t_j^+) &= (I_{n_y} - \Psi_y(e(t_j), j))e_y(t_j) \\ e_u(t_j^+) &= (I_{n_u} - \Psi_u(e(t_j), j))e_u(t_j) \\ &\quad + \left( I_{n_u} - \Psi_u(e(t_j), j) \right) \left( g_c(x_c(t_j), y(t_j) + e_y(t_j)) \right. \\ &\quad \left. - g_c(x_c(t_j), y(t_j)) + (I_{n_y} - \Psi_y(e(t_j), j))e_y(t_j) \right), \end{aligned} \quad (12a) \quad (12b)$$

where the protocol is fully defined by the functions

$$\Psi_y(e, j) := \text{diag} \left( \psi_{y_1}(e, j)I_{n_{y_1}}, \dots, \psi_{y_{\ell_y}}(e, j)I_{n_{y_{\ell_y}}} \right) \quad (13)$$

$$\Psi_u(e, j) := \text{diag} \left( \psi_{u_1}(e, j)I_{n_{u_1}}, \dots, \psi_{u_{\ell_u}}(e, j)I_{n_{u_{\ell_u}}} \right) \quad (14)$$

with  $\psi_{y_i}$  and  $\psi_{u_i}$  are mappings from  $\mathbb{R}^{n_e} \times \mathbb{Z}_{\geq 0}$  to  $\{0, 1\}$ . Equations (12a)-(12b) respectively describe the update of  $e_y$  and  $e_u$  at each transmission. Model (12) encompasses RR and TOD protocol as special cases.

Consider the following auxiliary discrete-time system induced by the plant and the protocol, for  $j \in \mathbb{Z}_{\geq 0}$ ,

$$e_y(j+1) = (I_{n_y} - \Psi_y(e, j))e_y(j) \quad (15a)$$

$$\begin{aligned} e_u(j+1) &= (I_{n_u} - \Psi_u(e, j))e_u(j) \\ &\quad + \left( I_{n_u} - \Psi_u(e, j) \right) \left( g_c(x_c(j), y(j) + e_y(j)) \right. \\ &\quad \left. - g_c(x_c(j), y(j)) + (I_{n_y} - \Psi_y(e, j))e_y(j) \right). \end{aligned} \quad (15b)$$

We refer to system (15) as the *auxiliary discrete-time system induced by the plant and the protocol* (12). For any initial time  $j_0 \in \mathbb{Z}_{\geq 0}$ , any initial condition  $e_0 \in \mathbb{R}^n$  and any input  $x \in \ell_\infty$ ,  $\phi(\cdot, j_0, e_0, x)$  denotes the corresponding solution to (15). As  $e = (e_y, e_u)$ , we partition the solution  $\phi$  into  $\phi := (\phi_y, \phi_u)$ , whenever it is convenient. We use the following assumption.

*Assumption 1:* There exists  $L_g > 0$  such that  $|g_c(x_c, y_1) - g_c(x_c, y_2)|_1 \leq L_g |y_1 - y_2|_1$  for all  $x_c \in \mathbb{R}^{n_c}$  and all  $y_1, y_2 \in \mathbb{R}^{n_y}$ .  $\square$

Assumption 1 means that  $g_c$  is globally Lipschitz in its second argument uniformly in  $x$ . This condition is always verified by linear systems for instance. The next assumption is that the solutions to (15) converge in (a uniform) finite-time to the origin. Examples of protocols that satisfy Assumption 2 are the RR protocol and the TOD protocol, which is the purpose of Sections IV-B and IV-C.

*Assumption 2:* There exists  $\bar{j} \in \mathbb{Z}_{\geq 0}$  such that for any  $e \in \mathbb{R}^{n_e}$ , any  $x \in \ell_\infty$  and  $j_0 \in \mathbb{Z}_{\geq 0}$   $\phi(j, j_0, e_0, x) = 0$  for all  $j - j_0 \geq \bar{j}$ .  $\square$

When Assumptions 1-2 hold, we can construct a Lyapunov function to prove that system (15) is UGES, using a similar construction as in the proof of Theorem 1 in [15].

*Lemma 1:* Let Assumptions 1 and 2 hold. Also, let the functional  $W: \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_e} \times \ell_\infty \rightarrow \mathbb{R}_{\geq 0}$  be given by

$$W(j, e, x) = \sqrt{\sum_{k=j}^{+\infty} |\phi(k, j, e, x)|^2} \quad (16)$$

for any  $j \in \mathbb{Z}_{\geq 0}$ ,  $e \in \mathbb{R}^{n_e}$  and  $x \in \ell_\infty$ . Then the auxiliary discrete-time system induced by the protocol (12) is UGES. In particular,  $W$  satisfies Definition 1 with  $a_1 = 1$ ,  $a_2 = \sqrt{M}$  and  $\rho = \sqrt{\frac{M-1}{M}}$  with  $M := \frac{(1+L_g)^{\bar{j}+1} \sqrt{n_e}^{\bar{j}+1} - 1}{(1+L_g) \sqrt{n_e} - 1}$  and  $\bar{j}$  coming from Assumption 2.  $\square$

It follows from Lemma 1 that protocol (15) is dead-beat stable, which is a stronger property than UGES. However, the UGES property is sufficient for our purposes.

*Remark 3:* Given the system

$$e_y(j+1) = (I_{n_y} - \Psi_y(e, j))e_y(j) \quad (17a)$$

$$e_u(j+1) = (I_{n_u} - \Psi_u(e, j))e_u(j) \quad (17b)$$

is UGES, without consideration of Assumption 2, one can easily show that there exists sufficiently small  $L_g > 0$  such that system (15) is UGES, as well. However, such a condition is not needed for RR and TOD protocols as we show in the remaining part of this section.  $\square$

##### B. Round-Robin Protocol

For the ease of reference, we rewrite the RR protocol model (9)

$$e_y(j+1) = (I_{n_y} - \Delta_y(j))e_y(j) \quad (18a)$$

$$\begin{aligned} e_u(j+1) &= (I_{n_u} - \Delta_u(j))e_u(j) \\ &\quad + \left( I_{n_u} - \Delta_u(j) \right) \left( g_c(x_c(j), y(j) + e_y(j)) \right. \\ &\quad \left. - g_c(x_c(j), y(j)) + (I_{n_y} - \Delta_y(j))e_y(j) \right) \end{aligned} \quad (18b)$$

where  $\Delta_y(j) := \text{diag}(\delta_{y_1}(j)I_{n_{y_1}}, \dots, \delta_{y_{\ell_y}}(j)I_{n_{y_{\ell_y}}})$ ,  $\Delta_u(j) := \text{diag}(\delta_{u_1}(j)I_{n_{u_1}}, \dots, \delta_{u_{\ell_u}}(j)I_{n_{u_{\ell_u}}})$  and the functions  $\delta_{u_i}$  and  $\delta_{y_i}$  are defined in (8). It should be pointed out that, here, we replace the notation  $\Psi_i$ 's (and

$\psi_i$ , respectively) with  $\Delta_i$ 's (and  $\delta_i$ , respectively) to be as consistent as possible with the NCSs context.

To establish UGES for system (18), we only need to show that system (18) satisfies Assumption 2 in view of Lemma 1.

*Proposition 1:* Under Assumption 1, system (18) verifies Assumption 2 with  $\bar{j} = 2\ell$ , where  $\ell$  is the number of nodes.  $\square$

In [7], where no direct-feedthrough term is considered, the RR protocol is dead-beat stable in  $\ell$ -steps. Here, the maximal number of steps to reach 0 has doubled, which will lead to a smaller MATI bound in view of [7], [9]. A similar observation applies in Section IV-C. The following lemma concludes this subsection and is a direct consequence of Proposition 1 and Lemma 1.

*Lemma 2:* Under Assumption 1, system (18) is UGES with the Lyapunov functional (16) and  $M = \frac{(1+L_g)^{2\ell+1}\sqrt{n_e^{2\ell+1}-1}}{(1+L_g)\sqrt{n_e-1}}$ . Moreover, the Lyapunov functional (16) is locally Lipschitz in  $e$  uniformly in  $j$  and  $x$ .  $\square$

### C. Try-Once-Discard Protocol

The TOD protocol grants the node whose network-induced error is the largest access to the network. This gives

$$e_y(j+1) = \left( I_{n_y} - \Psi_y(e(j)) \right) e_y(j) \quad (19a)$$

$$e_u(j+1) = \left( I_{n_u} - \Psi_u(e(j)) \right) e_u(j) + \left( I_{n_u} - \Psi_u(e(j)) \right) \left( g_c(x_c(j), y(j) + e_y(j)) - g_c(x_c(j), y(j) + (I_{n_y} - \Psi_y(e(j)))e_y(j)) \right), \quad (19b)$$

where

$$\Psi_y(e) := \text{diag}(\psi_{y_1}(e)I_{n_{y_1}}, \dots, \psi_{y_{\ell'_y}}(e)I_{n_{y_{\ell'_y}}}) \quad (20)$$

$$\Psi_u(e) := \text{diag}(\psi_{u_1}(e)I_{n_{u_1}}, \dots, \psi_{u_{\ell'_u}}(e)I_{n_{u_{\ell'_u}}}) \quad (21)$$

with

$$\psi_i(e) := \begin{cases} 1 & \text{when } i = \min(\arg \max_{k \in \{1, \dots, \ell\}} |e_k|) \\ 0 & \text{otherwise,} \end{cases} \quad (22)$$

and  $e = (e_1, \dots, e_\ell)$  is the partition of  $e$  according to the  $\ell$  nodes (after reordering, if needed). With the same arguments as those for the RR protocol, the difference with [7] is the term  $g_c(x_c, y + e_y) - g_c(x_c, y + (I_{n_y} - \Psi_y(e))e_y) \neq 0$  on the right-hand side of (19b).

The next proposition shows that Assumption 2 is verified by system (19).

*Proposition 2:* Under Assumption 1, system (19) satisfies Assumption 2 with<sup>4</sup>  $\bar{j} = \tilde{\ell} := \ell - 1 + (\ell - \ell_u)(\ell_u + 1) + \theta(\ell - \ell_u - 2) \frac{(\ell - \ell_u - 1)(\ell - \ell_u - 2)}{2}$ .  $\square$

We conclude this subsection with the following result.

<sup>4</sup>The function  $\theta$  is defined Section II.

*Lemma 3:* Under Assumption 1, system (19) is UGES with the Lyapunov functional (16) with  $M = \frac{(1+L_g)^{\tilde{\ell}+1}\sqrt{n_e^{\tilde{\ell}+1}-1}}{(1+L_g)\sqrt{n_e-1}}$  with  $\tilde{\ell}$  defined in Proposition 2.  $\square$

*Remark 4:* We have not been able to prove that the Lyapunov functional in Lemma 3 obtained from Lemma 1 is locally Lipschitz in  $e$ , uniformly in  $j$  and  $x$ . We will work on this point in future work.

## V. TWO CHANNEL CASE

In this section, we consider NCSs, which have two channels that are respectively dedicated to transmission of  $y$  and  $u$  signals. Hence, all nodes in  $e_y$  are sent over one channel and all nodes in  $e_u$  are sent via a different channel. In other words, there are no nodes in which both inputs and outputs are included and using our notation from the previous sections we have that

$$\ell = \ell_y + \ell_u. \quad (23)$$

Each channel has its own protocol that governs transmissions of inputs only or outputs only. With this stronger assumption, we show in this section that it is possible to obtain different results via proofs and constructions that are simpler than the general case that was considered in the previous section. Hence, a slightly higher cost of implementation (i.e. two channels rather than one channel) simplifies the controller design and subsequent analysis significantly<sup>5</sup>

Under our assumptions, the central point in the analysis is an auxiliary system of the form

$$e_y(j+1) = \underbrace{\left( I_{n_y} - \Psi_y(e_y, j) \right)}_{=: h_y(i, e_y)} e_y(j) \quad (24a)$$

$$e_u(j+1) = \underbrace{\left( I_{n_u} - \Psi_u(e_u, j) \right)}_{=: h_u(i, e_u)} e_u(j) + \left( I_{n_u} - \Psi_u(e_u, j) \right) \left( g_c(x_c(j), y(j) + e_y(j)) - g_c(x_c(j), y(j) + (I_{n_y} - \Psi_y(e_y, j))e_y(j)) \right). \quad (24b)$$

Note the difference with (15) since  $\Psi_y(e_u, j)$  and  $\Psi_y(e_y, j)$  respectively depend on  $e_u$  and  $e_y$  rather than on  $e$  as it is the case in (15). This simplifying but reasonable assumption allows us to consider the system (24) as a cascade consisting of the  $e_y$  subsystem and the  $e_u$  subsystem with inputs  $e_y$  and  $x$ . The main result of this section is given next.

*Lemma 4:* Suppose that the following conditions hold.

- 1) System (24a) is UGES with a Lyapunov function<sup>6</sup>  $W_y: \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_y} \rightarrow \mathbb{R}_{\geq 0}$ , which is locally Lipschitz in its second argument, uniformly in the first one.

<sup>5</sup>For simplicity, we assume here that transmissions in both channels occur at the same time and we use the same MATI for both channels. However, it is possible to further generalise these results to the case when the two channels are not synchronised in this manner and different MATIs are used, see [16] for example.

<sup>6</sup> $W_y$  is a function (and not a functional as in Section IV-A, since it only depends on  $j$  and  $e_y$ , and not on the input  $x$ ). The same comment applies to  $W_u$ .

- 2) System (24b) with  $x = 0, e_y = 0$  is UGES with a Lyapunov function  $W_u: \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}_{\geq 0}$ , which is globally Lipschitz in its second argument, uniformly in the first one.
- 3) Assumption 1 holds.

Then, there exists  $d > 0$  such that the system (24) is UGES with Lyapunov function  $W(j, e) := W_y(j, e_y) + dW_u(j, e_u)$  where  $j \in \mathbb{R}_{\geq 0}$  and  $e \in \mathbb{R}^{n_e}$ , which is locally Lipschitz in  $e$ , uniformly in  $j$ .  $\square$

The advantage of the setup in this section is a simplified analysis that allows us to use known results on UGES protocols to combine them. Indeed, we note that items 1) and 2) of Lemma 4 were shown to hold in [7], [10] for a large class of commonly used protocols, such as the RR and TOD protocols. Hence, results of this section apply to a range of situations when either RR or TOD protocols are used for the output and input channels. Note that we do not have to use the same protocol for the output and input channels. For instance, we can use the RR protocol for the output channel and the TOD protocol for the input channel, or vice versa.

*Remark 5:* A notion of UGAS protocols was introduced in [8]. We can rephrase Lemma 4 so that instead of UGES protocols, we use UGAS protocols for each subsystem. Since the system (24) is a cascade system, we can use results in [17] to construct a UGAS Lyapunov function for the system (24) using UGAS Lyapunov functions for subsystems. In this case, we would need to appeal to stability results in [8] to conclude stability of the overall NCSs.  $\square$

*Remark 6:* The simplified analysis of this section can be applied also to situations when we use one channel for all inputs and outputs but the protocol is such that it yields a cascade system in (12) so that the auxiliary system takes the form as in (24). Note that in this case it is necessary that no node contains both inputs and outputs, i.e.  $\ell = \ell_u + \ell_y$ . This may give rise to new classes of protocols. For instance, suppose that we use a periodically time-varying protocol that on different time intervals acts as TOD for all output nodes and then TOD for all input nodes. This idea can be used to generate a range of novel protocols that fit within our analysis framework. This flexibility is very useful since different protocols perform differently when applied to a given plant.  $\square$

## VI. CONCLUSIONS

We have extended the emulation-based controller design framework for NCSs to systems controlled with dynamic controllers that contain direct-feedthrough terms. The existing stability analysis applies to this case once the stability of the auxiliary system induced by the protocol and the plant is established. We have provided several results that can be used in the case of a single channel or two channels setups.

## REFERENCES

[1] M.H. Mamduhi, D. Tolic, and S. Hirche, Robust event-based data scheduling for resource constrained networked control systems, in *Proc. American Control Conference*, pp. 4695-4701, Chicago: USA, 2015.

[2] J. Hespanha, P. Naghshabrizi, and Y. Xu, A Survey of recent results in networked control systems, in *Proc. IEEE Special Issue on Technology of Networked Control Systems*, 95, pp. 138-162, Jan. 2007.

[3] P. Antsaklis, and J. Baillieul, Special issue on technology of networked control systems, *Proceedings of the IEEE*, vol. 95, pp. 5-8, Jan. 2007.

[4] T.C. Yang, Networked control system: a brief survey, *IEE Proc. Control Theory Appl.*, vol. 153, July 2006.

[5] W. Zhang, M.S. Branicky, and S.M. Phillips, Stability of networked control systems, *IEEE Control Systems Magazine*, pp. 84-99, Feb. 2001.

[6] G. C. Walsh, O. Beldiman, and L. G. Bushnell, Asymptotic behavior of nonlinear networked control systems, *IEEE Trans. Automatic Control*, vol. 46, pp. 1093-1097, 2001.

[7] D. Nešić, and A.R. Teel, Input-output stability properties of networked control systems, *IEEE Trans. Automatic Control*, vol. 49, pp. 1650-1667, 2004.

[8] D. Nešić, and A. R. Teel, Input-to-state stability of networked control systems, *Automatica*, vol. 40, 2121-2128.

[9] D. Carnevale, A. R. Teel, and D. Nešić, A Lyapunov proof of an improved maximum allowable transfer interval for networked control systems, *IEEE Trans. Automatic Control*, vol. 52, pp. 892-897, 2007.

[10] D. Nešić, and D. Liberzon, A unified framework for design and analysis of networked and quantized control systems, *IEEE Trans. Automatic Control*, vol. 54, pp. 732-747, 2009.

[11] D.E. Quevedo, Daniel, J.S. Welsh, G.C. Goodwin, and M. McLeod, Networked PID control, In *Proc. IEEE International Symposium on Intelligent Control*, pp. 1695-1700, 2006.

[12] W.P.M.H. Heemels, A.R. Teel, N. van de Wouw, and D. Nešić, Networked control systems with communication constraints: tradeoffs between transmission intervals, delays and performance, *IEEE Trans. Automatic Control*, vol. 55, pp. 1781-1796, 2010. Dec. 2013, Florence, Italy, pp. 4631-4637.

[13] R. Postoyan, N. van de Wouw, D. Nešić, and W.P.M.H. Heemels, Tracking control for nonlinear networked control systems, *IEEE Trans. Automatic Control*, vol. 59, pp. 1539-1554, 2014.

[14] S.H.J. Heijmans, R. Postoyan, N. Noroozi, D. Nešić, W.P.M.H. Heemels, Stability analysis of networked control systems with direct-feedthrough terms: Part II - The linear case, submitted for publication in *55th IEEE Conference on Decision and Control*.

[15] Z-P. Jiang, and Y. Wang, A converse Lyapunov theorem for discrete-time systems with disturbances, *Systems & Control Letters*, vol. 45, pp. 49-58, 2002.

[16] M. Abdelrahim, R. Postoyan, J. Daafouz, and D. Nešić, Robust event-triggered output feedback controllers for nonlinear systems, *Automatica*, to appear.

[17] D. Nešić and A.R. Teel, Changing supply functions in input-to-state stable systems: the discrete-time case, *IEEE Trans. Automatic Control*, vol. 46, pp. 960-962, 2001.