Emulation-based tracking solutions for nonlinear networked control systems

R. Postoyan, N. van de Wouw, D. Nešić and W.P.M.H. Heemels

Abstract— We investigate emulation-based tracking control for nonlinear networked control systems (NCS) affected by disturbances. We consider a general scenario in which the network is used to ensure the communication between the controller, the plant and the reference system generating the desired trajectory to be tracked. The communication constraints induce non-vanishing errors (in general) on the feedforward term and the output of the reference system. These network-induced errors affect the convergence of the tracking error. As a consequence, available results on the stabilization of equilibrium points for NCS are not applicable. Therefore, we develop an appropriate hybrid model and we give sufficient conditions on the closed-loop system, the communication protocol and an explicit bound on the maximum allowable transmission interval (MATI) guaranteeing that the tracking error converges to the origin up to some errors due to both the external disturbances and the aforementioned non-vanishing network-induced errors. Our results cover a large class of the so-called uniformly globally asymptotically stable protocols which include the well-known round-robin and try-once-discard protocols. We also introduce a new dynamic protocol suitable for tracking control.

I. INTRODUCTION

Networked control systems (NCS) have received considerable research interest these last decades. This is justified by the fact that, nowadays, controllers often communicate with the plant via a network which may be used for other tasks as well. This implementation offers great advantages over classical point-to-point connections in terms of cost, flexibility and ease of maintenance. On the other hand, it requires the development of appropriate control strategies to guarantee the desired stability properties under the communication constraints caused by the use of the network. Most available works on NCS concentrate on the stabilization of *equilibrium points*, while very few studies address the *tracking control* of NCS, see [2], [9], [11]. The latter references have shown that tracking control exhibits characteristic difficulties not present in the stabilization of equilibria of NCS. Indeed, tracking controllers are often composed of a feedback term (to ensure

R. Postoyan is with the Université de Lorraine, CRAN, UMR 7039 and the CNRS, CRAN, UMR 7039, France romain.postoyan@univ-lorraine.fr

N. van de Wouw and W.P.M.H. Heemels are with the Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands, {n.v.d.wouw,m.heemels}@tue.nl. W.P.M.H. Heemels is supported by the Innovational Research Incentives Scheme under the VICI grant "Wireless control systems: A new frontier in automation" (No. 11382) awarded by NWO (The Netherlands Organization for Scientific Research) and STW (Dutch Science Foundation), and the European 7th Framework Network of Excellence "Highly-complex and networked control systems" (HYCON2) (grant agreement no. 257462).

D. Nešić is with the Department of Electrical and Electronic Engineering, the University of Melbourne, Parkville, VIC 3010, Australia d.nesic@ee.unimelb.edu.au

the convergence to the desired solution) and a feedforward term (which induces the desired solution in the closed-loop system). The authors of [2], [9], [11] have shown that the errors induced by the network on the feedforward term lead to *approximate* tracking. Similarly, the fact that the reference signals are transmitted via the communication channel may also be a source of errors that affect the convergence of the tracking error.

The main purpose of the paper is to propose a method to design controllers which ensure a state tracking objective for NCS affected by exogenous perturbations. Compared to [2], [9], [11], we consider nonlinear systems affected by disturbances (as opposed to linear systems) and we study the effect of scheduling. We follow an emulation-like approach as in [10], [5] which consists in first designing a controller that solves the problem in the absence of communication constraints. Afterwards, we implement the controller over a network and study the conditions that allow us to maintain the tracking property up to some errors caused by the network. We propose a general scenario where the channel is used to ensure the communication between the controller, the plant and the reference system. This allows us to encompass the architectures studied in [2], [9], [11] as particular cases and to investigate new ones. At each transmission instant, the network is such that only a single *node* (i.e. a group of sensors and / or actuators) is granted access to the network according to a rule called *protocol*. The class of protocols we consider include the round-robin (RR) protocol, the try-once-discard (TOD) protocol [10] and more generally the protocols which are Lyapunov uniformly globally asymptotically stable (UGAS) as defined in [6]. We also propose a new TOD-like protocol for tracking control which may ensure better performances compared to the RR and TOD protocols.

The paper is organized as follows. The tracking control problem is formalized in Section II. Next, we propose a suitable NCS model in Section III and the assumptions we adopt are given in Section IV. The main stability result is stated in Section V. In Section VI, we give examples of protocols suitable in the scope of tracking. An illustrative example is presented in Section VII. The proofs are omitted for space reasons.

Notation. Let $\mathbb{R} = (-\infty, \infty)$, $\mathbb{R}_{\geq 0} = [0, \infty)$, $\mathbb{R}_{>0} = (0, \infty)$, $\mathbb{Z}_{\geq 0} = \{0, 1, 2, \ldots\}$, $\mathbb{Z}_{>0} = \{1, 2, \ldots\}$. A function $\gamma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is of class $\mathcal{K}$ if it is continuous, zero at zero and strictly increasing, and it is of class $\mathcal{K}_\infty$ if, in addition, it is unbounded. A continuous function $\gamma : \mathbb{R}_{\geq 0}^2 \longrightarrow \mathbb{R}_{\geq 0}$ is of class $\mathcal{KL}$ if for each $t \in \mathbb{R}_{\geq 0}$, $\gamma(\cdot, t)$ is of class $\mathcal{K}$, and, for

each $s \in \mathbb{R}_{>0}$, $\gamma(s, \cdot)$ is decreasing to zero. Additionally, a function $\beta : \mathbb{R}^3_{\geq 0} \to \mathbb{R}_{\geq 0}$ is of class $\mathcal{KLL}$, if $\beta(\cdot, \cdot, t) \in \mathcal{KL}$ and $\beta(\cdot, t, \cdot) \in \mathcal{KL}$ for any $t \in \mathbb{R}_{\geq 0}$. For $(x, y) \in \mathbb{R}^{n+m}$, the notation (x, y) stands for $[x^{\mathrm{T}}, y^{\mathrm{T}}]^{\mathrm{T}}$.

II. PROBLEM STATEMENT

A. The tracking problem

Consider the nonlinear plant

$$\dot{x}_p = \mathbf{f}_p(x_p, u, w_p), \qquad y_p = \mathbf{g}_p(x_p), \qquad (1)$$

where $x_p \in \mathbb{R}^{n_{x_p}}$ is the state, $u \in \mathbb{R}^{n_u}$ the control input, $y_p \in \mathbb{R}^{n_y}$ the measured output and $w_p \in \mathbb{R}^{n_{w_p}}$ is an external perturbation. The reference x_d that the system (1) has to track is given by the solution to the system

$$\dot{x}_d = \mathbf{f}_p(x_d, u_{ff}, w_d), \qquad y_d = \mathbf{g}_p(x_d), \qquad (2)$$

where $u_{ff} \in \mathbb{R}^{n_u}$ is the (feedforward) input, $y_d \in \mathbb{R}^{n_y}$ denotes the measured output and $w_d \in \mathbb{R}^{n_{w_d}}$ is a vector of external disturbances. When x_d is a reference trajectory, we assume that we know how to compute u_{ff} so that (2) holds with $w_d = 0$. System (2) may also model a *master* system that the plant (1) has to synchronize with. In this scenario, the master system (2) may be affected by external disturbances which justifies the presence of w_d in (2). We assume that the reference system (2) has a unique solution for any initial condition $x_d(0)$ and any inputs u_{ff} and w_d of interest. Both u_{ff} and y_d are available for the purpose of control.

We consider the following controller decomposition

$$u = u_{fb} + u_{ff}, \qquad (3)$$

where the feedforward term u_{ff} comes from (2) and the feedback term u_{fb} is an output of a dynamic controller given by

$$\dot{x}_c = \mathbf{f}_c(x_c, y_p, y_d, w_c), \qquad u_{fb} = \mathbf{g}_c(x_c, y_p, y_d), \qquad (4)$$

where $x_c \in \mathbb{R}^{n_{x_c}}$ is the controller state and $w_c \in \mathbb{R}^{n_{w_c}}$ is a vector of perturbations which may affect the controller dynamics.

B. Controller implementation over the network

We investigate the scenario where a network is used to ensure the communication between the plant's sensors and the controller and between the controller and the plant's actuators. We also allow for the case where the communication channel is used to transmit the output and the input of the reference system (2), i.e. y_d and u_{ff}. We consider a general setting because we can then capture, in a unified manner, specific scenarios in which the network is only used to realize some relevant subsets of the aforementioned communications, such as e.g. the cases in:

- [2], [11] where the reference and plant outputs, y_d and y_p respectively, are sent together to the controller and u_{ff} is not transmitted.
- [9] where the output y_d is directly available to the controller and u_{ff} is generated by the controller (note that $y_d = x_d$ in [9]), see Figure 1.

Our approach also allows us to study the scenario depicted in Figure 2, for instance, where the reference output y_d and the feedforward term u_{ff} are transmitted via the network. In that case, it is reasonable to set up the network in such a way that the feedforward term u_{ff} is directly transmitted to the plant's actuators.
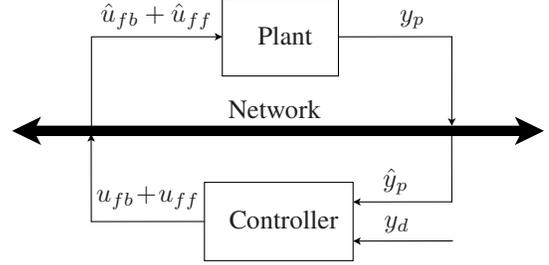

Fig. 1. Block diagram of the tracking control of NCS studied in [9].
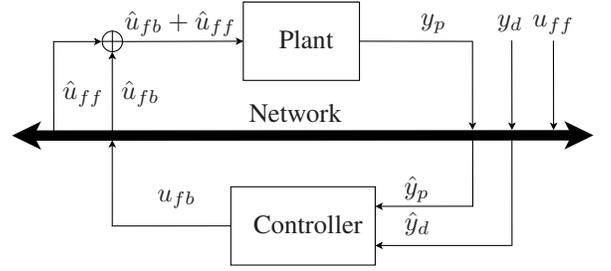

Fig. 2. Block diagram of the tracking control of NCS when u_{ff} and y_d are sent over the network.

The sensors and the actuators of the plant (1) and of the reference system (2) are grouped into l nodes (depending on their spatial location) which are connected to the network. At each transmission instant t_i, $i \in \mathbb{Z}_{\geq 0}$, only one node is granted access to the network by the scheduling protocol. The transmission sequence $\{t_i\}_{i \in \mathbb{Z}_{\geq 0}}$ is such that $\upsilon \leq t_i - t_{i-1} \leq \tau^*$ for $i \in \mathbb{Z}_{\geq 0}$, where $\tau^* \in \mathbb{R}_{>0}$ is the maximum allowable transmission interval (MATI) and υ is the lower bound on the achievable transmission interval given by the hardware constraints (see [5]). Notice that the transmission intervals $t_i - t_{i-1}$ may be time-varying and uncertain.

The plant (1) no longer receives $u = u_{fb} + u_{ff}$ but $\hat{u} = \hat{u}_{fb} + \hat{u}_{ff}$ which is generated from the most recently transmitted feedback and feedforward terms. We distinguish the feedback term u_{fb} from the feedforward term u_{ff} because these may be transmitted via distinct nodes (see Figure 2 for instance). The dynamics of the plant now becomes

$$\begin{aligned} \dot{x}_p &= \mathbf{f}_p(x_p, \hat{u}_{fb} + \hat{u}_{ff}, w_p) \qquad \forall t \in [t_{i-1}, t_i] \\ y_p &= \mathbf{g}_p(x_p). \end{aligned} \qquad (5)$$

Similarly, the controller (4) no longer receives y_p and y_d but their networked versions $\hat{y}_p$ and $\hat{y}_d$

$$\begin{aligned} \dot{x}_c &= \mathbf{f}_c(x_c, \hat{y}_p, \hat{y}_d, w_c) \qquad \forall t \in [t_{i-1}, t_i] \\ u_{fb} &= \mathbf{g}_c(x_c, \hat{y}_p, \hat{y}_d). \end{aligned} \qquad (6)$$

The variables $\hat{u}_{fb}, \hat{u}_{ff}, \hat{y}_p, \hat{y}_d$ have the following dynamics

$$\left.\begin{array}{rl} \dot{\hat{u}}_{fb} & = \hat{\mathbf{f}}_{fb}(x_p, x_c, x_d, \hat{y}_p, \hat{y}_d, \hat{u}_{fb}, \hat{u}_{ff}) \\ \dot{\hat{u}}_{ff} & = \hat{\mathbf{f}}_{ff}(x_p, x_c, x_d, \hat{y}_p, \hat{y}_d, \hat{u}_{fb}, \hat{u}_{ff}) \\ \dot{\hat{y}}_p & = \hat{\mathbf{f}}_p(x_p, x_c, x_d, \hat{y}_p, \hat{y}_d, \hat{u}_{fb}, \hat{u}_{ff}) \\ \dot{\hat{y}}_d & = \hat{\mathbf{f}}_d(x_p, x_c, x_d, \hat{y}_p, \hat{y}_d, \hat{u}_{fb}, \hat{u}_{ff}) \end{array}\right\} \forall t \in [t_{i-1}, t_i],$$

and

$$\begin{aligned} \hat{u}_{fb}(t_i^+) &= u_{fb}(t_i) + \mathbf{h}_{fb}(i, e_p(t_i), e_d(t_i), e_{fb}(t_i), e_{ff}(t_i)) \\ \hat{u}_{ff}(t_i^+) &= u_{ff}(t_i) + \mathbf{h}_{ff}(i, e_p(t_i), e_d(t_i), e_{fb}(t_i), e_{ff}(t_i)) \\ \hat{y}_p(t_i^+) &= y_p(t_i) + \mathbf{h}_p(i, e_p(t_i), e_d(t_i), e_{fb}(t_i), e_{ff}(t_i)) \\ \hat{y}_d(t_i^+) &= y_d(t_i) + \mathbf{h}_d(i, e_p(t_i), e_d(t_i), e_{fb}(t_i), e_{ff}(t_i)), \end{aligned}$$

where $e_{fb} = \hat{u}_{fb} - u_{fb} \in \mathbb{R}^{n_{e_u}}$, $e_{ff} = \hat{u}_{ff} - u_{ff} \in \mathbb{R}^{n_{e_u}}$, $e_p = \hat{y}_p - y_p \in \mathbb{R}^{n_{e_p}}$, $e_d = \hat{y}_d - y_d \in \mathbb{R}^{n_{e_d}}$ ($n_{e_p} = n_{e_d}$) denote the network-induced errors on the feedback and the feedforward terms and the plant and the reference outputs, respectively. The functions $\hat{\mathbf{f}}_{fb}, \hat{\mathbf{f}}_{ff}, \hat{\mathbf{f}}_p, \hat{\mathbf{f}}_d$ represent the in-network processing algorithms, i.e. the way the variables $\hat{u}_{fb}, \hat{u}_{ff}, \hat{y}_p, \hat{y}_d$ are generated between two successive transmission instants. In practice, it is common to use zero-order-hold devices, i.e. the functions $\hat{\mathbf{f}}_{fb}, \hat{\mathbf{f}}_{ff}, \hat{\mathbf{f}}_p, \hat{\mathbf{f}}_d$ are equal to 0. Other algorithms may also be implemented such as model-based algorithms as explained in [7] for example. We let $\hat{\mathbf{f}}_{fb}, \hat{\mathbf{f}}_{ff}, \hat{\mathbf{f}}_p, \hat{\mathbf{f}}_d$ depend on x_p, x_c and x_d for the sake of generality to capture the cases where they depend on a part of these vector variables. Functions $\mathbf{h}_{fb}, \mathbf{h}_{ff}, \mathbf{h}_p, \mathbf{h}_d$ model the scheduling mechanism which governs the transmissions at each instant t_i between the controller on one hand and the plant and the reference system on the other hand. Following the terminology of [5], we refer to the equation below as the protocol

$$\mathbf{e}(t_i^+) = \mathbf{h}(i, \mathbf{e}(t_i)), \tag{7}$$

where $\mathbf{e} = (e_p, e_d, e_{fb}, e_{ff}) \in \mathbb{R}^{n_{\mathbf{e}}}$, $n_{\mathbf{e}} = n_{e_p} + n_{e_d} + 2n_{e_u}$, and $\mathbf{h} = (\mathbf{h}_p, \mathbf{h}_d, \mathbf{h}_{fb}, \mathbf{h}_{ff})$. Since the network is composed of l nodes, we partition $\mathbf{e}$ as $\mathbf{e} = (\mathbf{e}_1, \ldots, \mathbf{e}_l)$ (after reordering, if necessary). The protocol (7) is such that at each transmission instant t_i, if node j gets access to the network, the corresponding error $\mathbf{e}_j$ experiences a jump while the other components of $\mathbf{e}$ remain unchanged; usually $\mathbf{e}_j(t_i^+) = 0$ but this is not needed in general. It has been shown in [5] that several common protocols can be modeled by (7) such as the round-robin (RR) protocol which grants access to each node at a fixed period, or the maximum error first try-once-discard (TOD) protocol introduced in [10] which gives access to the node where the norm of the local network-induced error, $|\mathbf{e}_j|$ with $j \in \{1, \ldots, l\}$, is the largest. Model (7) also captures standard sampled-data systems by setting $\mathbf{h}$ to 0.

Our objective is to provide conditions on the system (1)-(4) and on the network to guarantee the *approximate* convergence of the plant state x towards the reference state x_d in the presence of network-induced communication constraints.

III. HYBRID MODEL FOR NCS

Before presenting the hybrid model, we need to define new coordinates. As we are interested in the convergence

of x_p towards x_d, we introduce the tracking error $\xi := x_p - x_d \in \mathbb{R}^{n_\xi}$ ($n_\xi = n_{x_p}$). We also define the error $e := (e_\xi, e_{fb}) \in \mathbb{R}^{n_e}$ where $e_\xi := e_p - e_d \in \mathbb{R}^{n_{e_\xi}}$. The idea is to show that the ξ- and the e-system dynamics satisfy some robust asymptotic stability properties w.r.t. the external perturbation vector $w := (w_p, w_d, w_c) \in \mathbb{R}^{n_w}$ and the network-induced errors (e_d, e_{ff}) which are regarded as external disturbances similarly to [9]. This choice is motivated by the fact that e_d and e_{ff} typically depend on the reference system (2) and there is a priori no reason why they should satisfy some asymptotic stability properties even for very fast transmissions (recall that the MATI τ^* cannot be infinitely small as it needs to be such that $\tau^* \geq \upsilon > 0$), contrary to e as we will show in Section V. For instance, when zero-order-hold devices are implemented, $\dot{e}_d = -\dot{y}_d$ and $\dot{e}_{ff} = -\dot{u}_{ff}$ so that the origin is not an equilibrium point of the systems in e_d and e_{ff} when $\dot{y}_d \neq 0$ and $\dot{u}_{ff} \neq 0$ (which is generally the case when tracking time-varying trajectories).

We write the overall NCS as a hybrid system using the framework and the notation of [3]. We use the coordinates $(\xi, x_c, x_d, e, e_d, e_{ff}, \kappa, \tau_1, \tau_2)$ where $\kappa \in \mathbb{Z}_{\geq 0}$ is a counter variable which may be used to describe protocols such as the RR protocol (see Example 1 in [5]) and $\tau_1, \tau_2 \in \mathbb{R}_{\geq 0}$ are clock variables:

$$\left.\begin{array}{rl} \dot{\xi} &= f_\xi(\tau_2, \xi, x_c, x_d, e, e_d, e_{ff}, w) \\ \dot{x}_c &= f_c(\tau_2, \xi, x_c, x_d, e, e_d, w) \\ \dot{x}_d &= f_d(\tau_2, x_d, w) \\ \dot{e} &= g_e(\tau_2, \xi, x_c, x_d, e, e_d, e_{ff}, w) \\ \dot{e}_d &= g_d(\tau_2, \xi, x_c, x_d, e, e_d, e_{ff}, w) \\ \dot{e}_{ff} &= g_{ff}(\tau_2, \xi, x_c, x_d, e, e_d, e_{ff}, w) \\ \dot{\kappa} &= 0 \\ \dot{\tau}_1 &= 1 \\ \dot{\tau}_2 &= 1 \end{array}\right\} \tau_1 \in [0, \tau^*]$$

$$\left.\begin{array}{rl} \xi^+ &= \xi \\ x_c^+ &= x_c \\ x_d^+ &= x_d \\ e^+ &= h_e(\kappa, e, e_d, e_{ff}) \\ e_d^+ &= h_d(\kappa, e, e_d, e_{ff}) \\ e_{ff}^+ &= h_{ff}(\kappa, e, e_d, e_{ff}) \\ \kappa^+ &= \kappa + 1 \\ \tau_1^+ &= 0 \\ \tau_2^+ &= \tau_2 \end{array}\right\} \tau_1 \in [\upsilon, \tau^*]. \tag{8}$$

The variable τ_1 represents the time elapsed since the last transmission and τ_2 models the time. The vector fields and mappings $f_\xi, f_c, f_d, g_e, g_d, g_{ff}, h_e, h_d$ and h_{ff} are obtained by direct calculations from the developments in Section II (the τ_2-argument captures their dependency on u_{ff} or $\dot{u}_{ff}$) and are assumed to be continuous. We similarly write $e_p^+ = h_p(\kappa, e, e_d, e_{ff})$ and $e_{fb}^+ = h_{fb}(\kappa, e, e_d, e_{ff})$ which will be used in the sequel.

For the sake of convenience, we introduce $q_x := (\xi, x_c, x_d) \in \mathcal{R}_x$ and $q_e := (e, e_d, e_{ff}) \in \mathcal{R}_e$, where $\mathcal{R}_x := \mathbb{R}^{n_\xi + n_{x_c} + n_{x_d}}$ and $\mathcal{R}_e := \mathbb{R}^{n_e + n_{e_d} + n_{e_{ff}}}$. In that way, we write $\dot{q}_x = f(\tau_2, q_x, q_e, w)$, $\dot{q}_e = g(\tau_2, q_x, q_e, w)$ and $q_e^+ = h(\kappa, q_e)$ (note that $q_x^+ = q_x$).

IV. Assumptions

Inspired by [1], we present the assumptions we adopt which can be used as guidelines to design and implement the controller (3)-(4) for the robust stabilisation of the desired trajectory.

The protocol has to be such that Assumption 1 holds.

Assumption 1: There exist a function $W : \mathbb{Z}_{\geq 0} \times \mathcal{R}_e \to \mathbb{R}_{\geq 0}$ that is locally Lipschitz in q_e, $\underline{\alpha}_W, \overline{\alpha}_W \in \mathcal{K}_\infty$, $\rho \in [0,1)$ and $\mu^d, \mu^{ff} \in \mathcal{K}_\infty$ such that for any $(\kappa, q_e) \in \mathbb{Z}_{\geq 0} \times \mathcal{R}_e$, it holds that

$$\underline{\alpha}_W(|e|) \leq W(\kappa, q_e) \leq \overline{\alpha}_W(|q_e|),$$
$$W(\kappa+1, h(\kappa, q_e)) \leq \rho W(\kappa, q_e) + \mu^d(|e_d|) + \mu^{ff}(|e_{ff}|).$$
$$(9)$$
$\square$

In Section VI, we give examples of protocols that verify Assumption 1. Note that, contrary to similar conditions in [5], [1], [4], the second inequality in (9) holds with the additional perturbation terms μ^d and μ^{ff}. This difference is due to the fact that Assumption 1 does not apply to the protocol (7) but to the q_e-system at jumps which, although related, are different dynamical systems. Indeed, the jumps of q_e are governed by the vector field $h = (h_p - h_d, h_{fb}, h_{ff})$ while the protocol concerns the variable $\mathbf{e}$ whose jumps are dictated by $\mathbf{h} = (h_p, h_d, h_{fb}, h_{ff})$. It can be noticed that analogous conditions to (9) are considered in [8] where input-to-state stable (ISS) protocols have been defined (except that here e_d and e_{ff} are parts of the overall state q_e, while in [8] there are exogenous disturbances and a similar dissipation inequality).

We assume that the following exponential growth condition on the q_e-dynamics between two transmission instants holds, which thus depends on the continuous-time dynamics of y_p, y_d, u_{fb}, u_{ff} and on the choice of the in-network processing algorithms.

Assumption 2: There exist $L \geq 0$ and a continuous function $H : \mathcal{R}_x \to \mathbb{R}_{\geq 0}$ and $\nu^d, \nu^{ff}, \nu^w \in \mathcal{K}_\infty$ such that for all $q_x \in \mathcal{R}_x$, $\kappa \in \mathbb{Z}_{\geq 0}$, $\tau_2 \in \mathbb{R}_{\geq 0}$, $w \in \mathbb{R}^{n_w}$ and almost all $q_e \in \mathcal{R}_e$

$$\left\langle \frac{\partial W(\kappa, q_e)}{\partial q_e}, g(\tau_2, q_x, q_e, w) \right\rangle \leq LW(\kappa, q_e) + H(q_x)$$
$$+ \nu^d(|e_d|) + \nu^{ff}(|e_{ff}|) + \nu^w(|w|),$$

where W comes from Assumption 1. $\square$

The controller (3)-(4) needs to be designed so that the condition below is valid.

Assumption 3: There exist a locally Lipschitz function $V : \mathcal{R}_x \to \mathbb{R}_{\geq 0}$, $\underline{\alpha}_V, \overline{\alpha}_V \in \mathcal{K}_\infty$, $\varepsilon \in \mathbb{R}_{>0}$, $\gamma \in \mathbb{R}_{\geq 0}$ and $\sigma^d, \sigma^{ff}, \sigma^w \in \mathcal{K}_\infty$ such that for any $q_x \in \mathcal{R}_x$

$$\underline{\alpha}_V(|\xi|) \leq V(q_x) \leq \overline{\alpha}_V(|q_x|), \qquad (10)$$

and for all $q_e \in \mathcal{R}_e$, $\tau_2 \in \mathbb{R}_{\geq 0}$, $w \in \mathbb{R}^{n_w}$ and almost all $q_x \in \mathcal{R}_x$

$$\langle \nabla V(q_x), f(\tau_2, q_x, q_e, w) \rangle \leq -\varepsilon V(q_x) - \varepsilon W^2(\kappa, q_e)$$
$$- H^2(q_x) + \gamma^2 W^2(\kappa, q_e) + \sigma^d(|e_d|)$$
$$+ \sigma^{ff}(|e_{ff}|) + \sigma^w(|w|),$$
$$(11)$$

where W and H come from Assumptions 1-2. $\square$

According to (10) and (11), the emulated controller does ensure an ISS-like property for the tracking error dynamics (i.e. the ξ-system) with W, e_d, e_{ff}, w as inputs. Assumption 3 also implies that the ξ-system is $\mathcal{L}_2$ stable from W to H when there is no error e_d, e_{ff} and no disturbance w. The constant ε in (11) is usually taken sufficiently small.

The last condition is on the MATI. As in [1], we need to have a network that has a sufficiently high bandwidth so that the assumption stated below is satisfied.

Assumption 4: The MATI τ^* satisfies $\tau^* < \mathcal{T}(\rho, \gamma, L)$ where

$$\mathcal{T}(\rho, \gamma, L) := \begin{cases} \frac{1}{Lr} \arctan\left(\frac{r(1-\rho)}{2\frac{\rho}{1+\rho}(\frac{\gamma}{L}-1)+1+\rho}\right) & \text{if } \gamma > L \\ \frac{1}{L}\frac{1-\rho}{1+\rho} & \text{if } \gamma = L \\ \frac{1}{Lr} \text{arctanh}\left(\frac{r(1-\rho)}{2\frac{\rho}{1+\rho}(\frac{\gamma}{L}-1)+1+\rho}\right) & \text{if } \gamma < L, \end{cases}$$
$$(12)$$

with $r := \sqrt{\left|\left(\frac{\gamma}{L}\right)^2 - 1\right|}$ and $\rho \in [0,1)$ and $\gamma, L \geq 0$ come from Assumptions 1-3. $\square$

V. Main results

We are ready to state the main result. Its proof is based on the proof of Theorem 1 in [1] and requires some essential modifications to handle the effect of the perturbations induced by e_d, e_{ff} and w.

Theorem 1: Consider system (8) and suppose Assumptions 1-4 hold. Then there exist $\beta \in \mathcal{KLL}$, $\delta^d, \delta^{ff}, \delta^w \in \mathcal{K}_\infty$ such that for any initial condition $q_x(0,0) \in \mathcal{R}_x$, $q_e(0,0) \in \mathcal{R}_e$, $\tau_1(0,0), \tau_2(0,0) \in \mathbb{R}_{\geq 0}$ and $\kappa(0,0) \in \mathbb{Z}_{\geq 0}$ and each corresponding solution $(q_x, q_e, \tau_1, \tau_2, \kappa, w)$ of (8) it holds that

$$|(\xi(t,j), e(t,j))| \leq \beta(|(q_x(0,0), q_e(0,0))|, t, j)$$
$$+ \delta^d(\|e_d\|_{(t,j)}) + \delta^{ff}(\|e_{ff}\|_{(t,j)}) + \delta^w(\|w\|_{(t,j)}),$$
$$(13)$$

for all (t, j) in the solution's domain. Moreover, $\delta^d(s)$ and $\delta^{ff}(s)$ can be written as $(1 + \varphi(\tau^*))\psi(v^{-1})\delta(s)$ for $s \geq 0$ where $\delta, \varphi, \psi \in \mathcal{K}_\infty$. $\square$

Remark 1: The property (13) is obtained by constructing a hybrid Lyapunov function U which satisfies an ISS-like property on flows but not at jumps. Thus, we use the fact that U flows for some time (at least v seconds, see Section II-B) before jumping in order for the decreasing property of U on flows to compensate, in some sense, the potential increase of U at jumps. $\square$

Remark 2: The norms of the errors $\|e_d\|_{(t,j)}, \|e_{ff}\|_{(t,j)}$ and the functions δ^d, δ^{ff} in (13) depend on the MATI τ^*. We may find upper bounds for $\|e_d\|_{(t,j)}$ and $\|e_{ff}\|_{(t,j)}$ on a case-by-case basis. For instance, when zero-order-hold devices are implemented and the RR protocol is selected, we can proceed like in (31) in [9] (where delays are taken into account but not scheduling). On the other hand, the functions δ^d, δ^{ff} also depend on the minimum time v between two jumps. We see that δ^d, δ^{ff} go to infinity as v tends to 0. This fact is due to our stability analysis which requires to decrease for some time v during flows in order to guarantee stability, see Remark 1. We think that a different analysis inspired by

the small gain arguments used in [8] may help to avoid this issue. Nevertheless, our approach is justified by the fact that we do not aim at estimating these gains and that we rely on a Lyapunov-based proof which allows us to derive easily computable MATI bounds. □

Theorem 1 shows that (ξ, e) tends to a ball centered at the origin and of radius $\delta^d(\|e_d\|_{(t,j)}) + \delta^{ff}(\|e_{ff}\|_{(t,j)}) + \delta^w(\|w\|_{(t,j)})$ as (t, j) grows. Thus, ξ indeed converges to the origin up to some errors due to w, as expected, but also due to e_{ff} and e_d which are induced by the network, similar to [9]. In practice, we want these errors to be sufficiently small and it might then be convenient to have some estimates of $\delta^d(\|e_d\|_{(t,j)})$ and $\delta^{ff}(\|e_{ff}\|_{(t,j)})$. While it may be possible to bound the $\mathcal{L}_\infty$-norm of e_d and e_{ff} (see Remark 2), we know that the expressions for δ^d and δ^{ff} we can deduce from the proof of Theorem 1 are subject to some conservatism. Nevertheless, the result in Theorem 1 provides the following qualitative insights on how to reduce the impact of the network-induced errors e_{ff} and e_d on the tracking errors:

- For $\delta^{ff}(\|e_{ff}\|_{(t,j)})$: first, when u_{ff} can be directly implemented at the actuators' stage, we have $e_{ff} \equiv 0$. When this is not possible, some previews of u_{ff} might be considered as in [9] to reduce the error due to e_{ff}.
- For $\delta^d(\|e_d\|_{(t,j)})$: it can be shown that δ^d can be written as $\delta^d(s) = \alpha\big(\mu^d(s) + \nu^d(s) + \sigma^d(s)\big)$ for $s \geq 0$, where α is some class-$\mathcal{K}_\infty$ function (which depends on V, W, τ^* and υ) and μ^d, ν^d, σ^d come from Assumptions 1-3. We show in Section VI that it is possible to set $\mu^d = 0$ by selecting an appropriate protocol or by appropriately implementing the emulated controller.

VI. ON THE CHOICE OF THE PROTOCOL

In this section, we give examples of protocols that ensure the satisfaction of Assumption 1. We first show that this assumption is verified when the protocol (7) is Lyapunov UGAS (as defined below) under mild conditions.

Definition 1 ([6]): The protocol (7) is said to be *Lyapunov uniformly globally asymptotically stable (UGAS)* if there exist $\mathbf{W} : \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_\mathbf{e}} \to \mathbb{R}_{\geq 0}$, $\underline{\alpha}_\mathbf{W}, \overline{\alpha}_\mathbf{W} \in \mathcal{K}_\infty$ and $\rho \in [0, 1)$ such that for all $\kappa \in \mathbb{Z}_{\geq 0}$ and $\mathbf{e} \in \mathbb{R}^{n_\mathbf{e}}$ the following is satisfied (recall $\mathbf{e} = (e_p, e_d, e_{fb}, e_{ff})$):

$$\underline{\alpha}_\mathbf{W}(|\mathbf{e}|) \leq \mathbf{W}(\kappa, \mathbf{e}) \leq \overline{\alpha}_\mathbf{W}(|\mathbf{e}|) \quad (14)$$
$$\mathbf{W}(\kappa + 1, \mathbf{h}(\kappa, \mathbf{e})) \leq \rho \mathbf{W}(\kappa, \mathbf{e}). \quad (15)$$

□

We are now ready to state the main result of this section.

Proposition 1: Consider the protocol (7) and suppose the following conditions hold:

(i) For any $j \in \{1, \ldots, n_\mathbf{e}\}$ and $i \in \mathbb{Z}_{\geq 0}$, $|\mathbf{h}_j(i, \mathbf{e}(t_i))| \leq |\mathbf{e}_j(t_i)|$ with $\mathbf{h} = (\mathbf{h}_1, \ldots, \mathbf{h}_{n_\mathbf{e}})$ where $\mathbf{h}$ is given in (7).

(ii) The protocol (7) is Lyapunov UGAS with a function $\mathbf{W} : \mathbb{Z}_{\geq 0} \times \mathbb{R}^{n_\mathbf{e}} \to \mathbb{R}_{\geq 0}$ which is differentiable almost everywhere in $\mathbf{e}$ and satisfies for all $\kappa \in \mathbb{Z}_{\geq 0}$ and almost all $\mathbf{e} \in \mathbb{R}^{n_\mathbf{e}}$, $\left|\frac{\partial \mathbf{W}(\kappa, \mathbf{e})}{\partial \mathbf{e}}\right| \leq M$, where $M \geq 0$.

Then Assumption 1 is verified with $W(\kappa, e) = \mathbf{W}(\kappa, e_\xi, 0, e_{fb}, 0)$, $\underline{\alpha}_W(s) = \underline{\alpha}_\mathbf{W}(s)$, $\overline{\alpha}_W(s) = \overline{\alpha}_\mathbf{W}(s)$, $\mu^d(s) = 2M(1 + \boldsymbol{\rho})s$, $\mu^{ff}(s) = M(1 + \boldsymbol{\rho})s$ for $s \geq 0$ and $\rho = \boldsymbol{\rho}$. □

Note that item (i) in Proposition 1 simply states that the local errors do not increase at each transmission which is the case for all relevant protocols. The conditions of Proposition 1 are satisfied by the RR and the TOD protocol in view of Section IV in [5].

Since we are interested in a different stability property for the e-system at jumps than in [5], we can propose an alternative Lyapunov function for the RR protocol, based on Proposition 4 in [5], which ensures stronger properties and may lead to less conservative MATI bounds.

Lemma 1: Suppose the protocol (7) is the RR protocol, then Assumption 1 is satisfied with: $W(\kappa, e) = \sqrt{\sum_{i=\kappa}^{\infty} |\phi(i, \kappa, e)|^2}$, where $\phi(i, \kappa, e)$ is the solution to[1] $e^+ = (h_p(\kappa, e_\xi), h_{fb}(\kappa, e_{fb}))$ at time i starting at time κ with initial condition e, $\underline{\alpha}_W(s) = s$, $\overline{\alpha}_W(s) = \sqrt{l}s$, $\mu^d(s) = \sqrt{l}s$ and $\mu^{ff}(s) = 0$ for $s \geq 0$ and $\rho = \sqrt{\frac{l-1}{l}}$. Moreover, $\mu^d = 0$ if and only if $h_p = h_d$. □

We now propose a new TOD-like protocol, that we call the *TOD-tracking* protocol. Consider the scenarios where each corresponding components of y_p and y_d are assigned to the same nodes[2]. In that way, a subvector $(e, e_{ff})_j$ of (e, e_{ff}), $j \in \{1, \ldots, l\}$, can be associated to each of the l nodes of the network. The idea is to grant access to the node where $|(e, e_{ff})_j|$ is the biggest (and not $|\mathbf{e}_j|$, $j \in \{1, \ldots, l\}$, as in the classical TOD protocol). We define the function $\mathbf{h}$ in (7) as $\mathbf{h}(\kappa, \mathbf{e}) = (\mathbb{I} - \Psi(\mathbf{e}))\mathbf{e}$ where $\Psi(\mathbf{e}) = (\delta_1(\mathbf{e})\mathbb{I}_{n_1}, \ldots, \delta_l(\mathbf{e})\mathbb{I}_{n_l})$ where $n_1 + \ldots + n_l = n_\mathbf{e}$ and $\delta_j(\mathbf{e}) = 1$ if $j = \min(\arg\max_j |(e, e_{ff})_j|)$ and $\delta_j(\mathbf{e}) = 0$ otherwise. The lemma below shows that the TOD-tracking protocol satisfies Assumption 1. It directly follows from Proposition 5 in [5].

Proposition 2: Suppose the protocol (7) is the TOD-tracking protocol, then Assumption 1 is satisfied with $W(q_e) = |(e, e_{ff})|$, $\underline{\alpha}_W(s) = s$, $\overline{\alpha}_W(s) = s$, $\mu^d(s) = \mu^{ff}(s) = 0$ for $s \geq 0$ and $\rho = \sqrt{\frac{l-1}{l}}$. □

The TOD-tracking protocol ensures Assumption 1 holds with $\mu^d = \mu^{ff} = 0$, which is a priori not the case for the TOD protocol according to Proposition 1. Thus, the TOD-tracking protocol may reduce the error of (ξ, e), and hence improve tracking performance in view of the discussion in Section V.

Remark 3: When the control input is sent over the network as $u_{fb} + u_{ff}$, like in the example in Section VII, we can set the protocol to grant access to the node where $|(e_\xi, e_{fb} + e_{ff})_j|$ is the largest (and not $|(e_\xi, e_{fb}, e_{ff})_j|$ as

---

[1]It has to be noted that h_p (respectively h_d) only depends on κ and e_p (respectively e_d) for the RR protocol, see Example 1 in [5].

[2]The TOD-tracking protocol can also be used when the nodes which transmit y_p (equivalently y_d) have access to y_d (equivalently y_p). That is typically the case when y_d is a given trajectory which can be implemented on the nodes.

	RR	TOD	TOD-tracking
Assumption 4	0.0061	0.0105	0.0105
Simulations	0.150	0.170	0.170

TABLE I

MATI BOUNDS IN SECTION VII.

above). We then take $W(q_e) = |(e_\xi, e_{fb} + e_{ff})|$ which satisfies Assumption 1 with the same functions $\underline{\alpha}_W$, $\overline{\alpha}_W$, μ^d, μ^{ff} and constant ρ. □

VII. ILLUSTRATIVE EXAMPLE

We apply the results developed in the previous sections to the tracking control of a single-link robot arm whose dynamics can be written as $\dot{x}_1 = x_2$, $\dot{x}_2 = -a\sin(x_1) + bu$, where x_1 is the angle, x_2 is the rotational velocity which are both measured, u is the input torque and $a, b > 0$ are fixed parameters. The robot arm has to track the reference system $\dot{x}_{1,d} = x_{2,d}$, $\dot{x}_{2,d} = -a\sin(x_{1,d}) + bu_{ff}$, where $x_{1,d}$ and $x_{2,d}$ are measured and $u_{ff}(t) = 10\sin(50t)$. When there is no communication constraint, the asymptotic convergence of (x_1, x_2) towards $(x_{1,d}, x_{2;d})$ is ensured using the control input $u = u_{fb} + u_{ff}$ where $u_{fb} = b^{-1}\big(a(\sin(x_1) - \sin(x_{1,d})) - (x_1 - x_{1,d}) - (x_2 - x_{2,d})\big)$. We consider the case where the controller is implemented using zero-order-hold devices and communicates with the robot arm via a network composed of 3 nodes for x_1, x_2 and u, respectively ($l = 3$). Thus, we assume that[3] $x_{1,d}, x_{2,d}, u_{ff}$ are directly available to the controller as in Figure 1. The protocol is either the RR, the TOD or the TOD-tracking. We consider the function W in Lemma 1 for the RR protocol, $W(e) = |e|$ for the TOD protocol and $W(q_e) = |(e_\xi, e_{fb} + e_{ff})|$ for the TOD-tracking protocol (see Remark 3). In that way, Assumption 1 is valid, see Section VI. By taking $a = 9.81 \cdot 0.5$ and $b = 2$, we also have that Assumptions 2 and 3 hold with $V(\xi) = \alpha\xi_1^2 + \beta\xi_1\xi_2 + \delta\xi_2^2$ where $\alpha = 3.05$, $\beta = 1.05$, $\delta = 5.05$. The obtained MATI bounds are summarized and compared to the bounds estimated via simulations in Table I. It has to be emphasized that our method strongly relies on the choice of the Lyapunov functions V and W and that other functions may lead to larger bounds. We notice that the bounds for the TOD and the TOD-tracking protocol are the same according to Assumption 4 and in simulations. Interest in the TOD-tracking is justified by the fact that it may reduce the impact of the errors e_d and e_{ff} on the tracking error as discussed below Proposition 2 and illustrated by Figure 3. On the other hand, we see in Figure 4 that the convergence error is of the same order of magnitude when using the TOD-tracking and the RR protocol; the advantage of the TOD-tracking is that we can consider larger transmission intervals (see Table I).

VIII. CONCLUSIONS

The Lyapunov-based emulation approach investigated in [1] for the stabilization of equilibrium points of NCS has

---

³We make this assumption in order to be able to consider the TOD-tracking protocol (see Section VI).
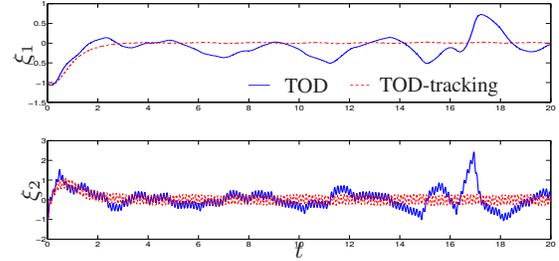

Fig. 3.   Tracking error for $\tau^* = 0.006$.
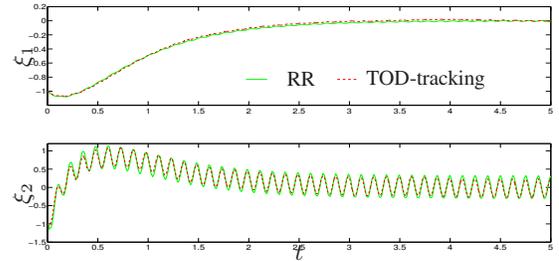

Fig. 4.   Tracking error for $\tau^* = 0.006$.

been extended to tracking control of time-varying trajectories. To handle the specific features of tracking control for NCS, we have proposed an appropriate hybrid model. We have presented sufficient conditions under which an approximate tracking control objective is achieved. We have explained how the controller can be implemented and how the protocol can be set up in order to reduce the impact of some of the network-induced errors on the tracking error.

REFERENCES

[1] D. Carnevale, A.R. Teel, and D. Nešić. A Lyapunov proof of an improved maximum allowable transfer interval for networked control systems. *IEEE Trans. on Automatic Control*, 52(5):892–897, 2007.

[2] H. Gao and T. Chen. Network-based H_∞ output tracking control. *IEEE Trans. on Aut. Control*, 53(3):655–667, 2008.

[3] R. Goebel, R.G. Sanfelice, and A.R. Teel. Hybrid dynamical systems: robust stability and control for systems that combine continuous-time and discrete-time dynamics. *IEEE Control Systems Magazine*, April:28–83, 2009.

[4] 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. on Aut. Control*, 55(8):1781–1796, 2010.

[5] D. Nešić and A.R. Teel. Input-output stability properties of networked control systems. *IEEE Trans. on Aut. Control*, 49:1650–1667, 2004.

[6] D. Nešić and A.R. Teel. Input-to-state stability of networked control systems. *Automatica*, 40:2121–2128, 2004.

[7] R. Postoyan and D. Nešić. A framework for the observer design for networked control systems. *IEEE Transactions on Automatic Control*, 57(5):1309–1314, 2012.

[8] M. Tabbara and D. Nešić. Input-output stability with input-to-state stable protocols for quantized and networked control systems. In *CDC (IEEE Conference on Decision and Control), Cancun, Mexico*, pages 2680–2685, 2008.

[9] N. van de Wouw, P. Naghshtabrizi, M.B.G. Cloosterman, and J.P. Hespanha. Tracking control for sampled-data systems with uncertain sampling intervals and delays. *Int. J. of Robust and Nonlinear Control*, 20(4):387–411, 2010.

[10] G.C. Walsh, O. Beldiman, and L.G. Bushnell. Asymptotic behavior of nonlinear networked control systems. *IEEE Transactions on Automatic Control*, 46:1093–1097, 2001.

[11] S. Yüksel, H. Hindi, and L. Crawford. Optimal tracking with feedback-feedforward control separation over a network. In *ACC (American Control Conference) Minneapolis, U.S.A.*, 2006.