



Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:

Wang, W;Nesic, D;Postoyan, R

Title:

Emulated controller design for networked control systems implemented on FlexRay

Date:

2014-01-01

Citation:

Wang, W., Nesic, D. & Postoyan, R. (2014). Emulated controller design for networked control systems implemented on FlexRay. 53rd IEEE Conference on Decision and Control, 2015-February, (February), pp.5272-5277. IEEE. <https://doi.org/10.1109/CDC.2014.7040213>.

Persistent Link:

<https://hdl.handle.net/11343/299377>

# Emulated controller design for networked control systems implemented on FlexRay

Wei Wang, Dragan Nešić and Romain Postoyan

**Abstract**—We design stabilizing controllers for nonlinear networked control systems (NCS) whose transmissions are scheduled by FlexRay. FlexRay protocol has been developed by the automotive industry to provide high bandwidth and deterministic communications. It works with communication cycles which consist of a static segment and a dynamic segment during which different scheduling rules are employed. We generalize existing emulated controller designs to be applicable to NCS with FlexRay. We start from a feedback law which stabilizes the origin of the plant when there is no network. We then present a novel hybrid model of the closed-loop system when the controller is implemented over a network scheduled by FlexRay. Afterwards, we provide conditions on the network under which the stability of the NCS is ensured. In particular, we consider segments of arbitrary lengths and we provide segment-dependent maximal allowable transmission interval bounds. The analysis relies on the construction of a new hybrid Lyapunov function. We believe that this work demonstrates the flexibility of the emulation approach and that it can be used to investigate other control problems for NCS with switched protocols.

## I. INTRODUCTION

Networked control systems (NCS) are systems which integrate computer networks into the control loops. The communication constraints induced by the network must be carefully taken into account when designing the controller for NCS. A common approach for the stabilization of NCS is emulation. The idea is to design a controller that stabilizes the plant in the absence of communication constraints, to implement the controller over the network and then to give conditions on the communication channel under which the stability of the NCS is preserved, see [1], [9], [13] for example. The emulation approach provides great freedom in the choice of continuous-time design tools and pertains to a large class of nonlinear NCS. On the other hand, the mathematical models of NCS are sometimes hard to relate to real physical networks. In this paper, we address this gap by tailoring emulation results so that they are applicable to NCS with FlexRay [3]. We need for that purpose to solve non-trivial technical issues notably in terms of modeling and Lyapunov analysis.

This work is supported by the Australian Research Council under the Discovery Project and Future Fellow program, the ANR under the grant COMPACS (ANR-13-BS03-0004-02) and the European 7th Framework Network of Excellence “Highly-complex and networked control systems” (HYCON2 No. 257462).

W. Wang and D. Nešić are with the Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, 3010, Victoria, Australia [wei.wang@unimelb.edu.au](mailto:wei.wang@unimelb.edu.au), [dnesic@unimelb.edu.au](mailto:dnesic@unimelb.edu.au)

R. Postoyan is with the Université de Lorraine, CRAN, UMR 7039 and the CNRS, CRAN, UMR 7039, France [romain.postoyan@univ-lorraine.fr](mailto:romain.postoyan@univ-lorraine.fr)

FlexRay was developed by BMW, Daimler-Chrysler, Philips and Freescale in 2000 [3], [12] to provide appropriate communications for implementing x-by-wire technology in automotive control. FlexRay consists of pre-set communication cycles where each cycle is composed of a static segment and a dynamic segment for data transmissions. In the static segment, the network is assigned to the nodes in a prefixed manner: we say that the scheduling rule is *static*. The dynamic segment enables messages to be sent whenever it is required which helps meet varying bandwidth requirements that can emerge at system run time. A *dynamic* policy schedules transmissions of the nodes in this case based on the online information. Because of the coexistence of these two scheduling rules, available results on the stabilization of NCS in [1], [7], [9] are not directly applicable.

We first propose a novel hybrid model for NCS implemented on FlexRay using the formalism of [4]. The main challenge comes from the fact that these systems are subject to discontinuities due to the transmissions on the one hand and the switches between the static and the dynamic segments on the other hand, which we both model as jumps. We resort to a modeling technique proposed in [11] to interconnect hybrid systems for that purpose and we obtain a hybrid model which satisfies the basic assumptions of [4]. We consider static and dynamic segments of any given lengths for which the transmissions are governed by uniformly globally exponentially stable protocols, which cover the round robin (RR) and the try-once-discard (TOD) protocols as particular cases, see [9]. We then construct a novel Lyapunov function such that the asymptotic stability property of the system in the absence of communication constraints is maintained when the controller is implemented over FlexRay provided that the *maximal allowable transmission interval* (MATI) of each segment satisfies a given bound. In contrast with available results on NCS with FlexRay, see [5], [8] for instance, we address nonlinear systems (as opposed to linear systems). Moreover, we derive segment-dependent MATI bounds which is more adequate in practice. We believe that the results of this paper, in particular the hybrid model and the Lyapunov function, can be used as a starting point to investigate NCS implemented over other types of hybrid protocols.

The paper is organized as follows. Preliminaries and definitions related to hybrid systems are given in Section II. FlexRay is described in more detail in Section III. We present the hybrid model in Section IV. The main stability results are stated in Section V and an example is proposed in Section VI. Conclusions are given in Section VII. All proofs

are omitted due to space limitations.

## II. PRELIMINARIES

Let  $\mathbb{Z}_{>0} := \{1, 2, \dots\}$ ,  $\mathbb{Z}_{\geq 0} := \{0, 1, 2, \dots\}$  and  $\mathbb{R}_{\geq 0} := [0, \infty)$ . Given a closed set  $\mathcal{A} \subset \mathbb{R}^n$  and  $x \in \mathbb{R}^n$ , we define the distance of a vector  $x$  to  $\mathcal{A}$  as  $|x|_{\mathcal{A}} := \inf_{y \in \mathcal{A}} |x - y|$ . A set-valued mapping  $M : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$  is outer semi-continuous (OSC) if and only if its graph  $\{(y, z) : y \in \mathbb{R}^m, z \in M(y)\}$  is closed, see Lemma 5.10 in [4]. A continuous function  $\gamma : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is of class- $\mathcal{K}$  if it is zero at zero and strictly increasing and it is of class- $\mathcal{K}_{\infty}$  if, in addition, it is unbounded. A continuous function  $\gamma : \mathbb{R}_{\geq 0}^2 \rightarrow \mathbb{R}_{\geq 0}$  is of class- $\mathcal{KL}$  if for each  $r \in \mathbb{R}_{\geq 0}$ ,  $\gamma(\cdot, r)$  is of class- $\mathcal{K}$ , and, for each  $s \in \mathbb{R}_{\geq 0}$ ,  $\gamma(s, \cdot)$  is decreasing to zero.

Consider a hybrid system using the formalism of [4]

$$H : \begin{cases} \dot{\xi} \in \mathcal{F}(\xi) & \xi \in C \\ \xi^+ \in \mathcal{G}(\xi) & \xi \in D, \end{cases} \quad (1)$$

with state  $\xi \in \mathbb{R}^n$  and where  $C, D \subset \mathbb{R}^n$  are respectively the flow and the jump sets. Typically, the system flows on  $C$  and experiences a jump on  $D$ . When  $\xi \in C \cap D$ , the system can either jump or flow, the latter only if flowing keeps  $\xi$  in  $C$ . The system is assumed to be well-posed, see Chapter 6 in [4] for more details.

We recall that a set  $S \subset \mathbb{R}_{\geq 0} \times \mathbb{Z}_{\geq 0}$  is called a *compact hybrid time domain* if  $S = \bigcup_{j=0}^{J-1} ([t_j, t_{j+1}], j)$  for some finite sequence of times  $0 = t_0 \leq t_1 \leq t_2 \leq \dots \leq t_J$ . The set  $S$  is a *hybrid time domain* if for all  $(T, J) \in S$ ,  $S \cap ([0, T] \times \{0, 1, \dots, J\})$  is a compact hybrid time domain. A function  $\xi : S \rightarrow \mathbb{R}^n$  is a *hybrid arc* if  $S$  is a hybrid time domain and  $\xi(\cdot, j)$  is locally absolutely continuous for each  $j$ . A hybrid arc  $\xi : \text{dom } \xi \rightarrow \mathbb{R}^n$  is a solution to the hybrid system  $H$  if  $\xi(0, 0) \in C \cup D$  and

- 1) for all  $j \in \mathbb{Z}_{\geq 0}$  and almost all  $t$  such that  $(t, j) \in \text{dom } \xi$ ,  $\xi(t, j) \in C$  and  $\dot{\xi}(t, j) \in \mathcal{F}(\xi(t, j))$ ;
- 2) for all  $(t, j) \in \text{dom } \xi$  such that  $(t, j+1) \in \text{dom } \xi$ ,  $\xi(t, j) \in D$  and  $\xi(t, j+1) \in \mathcal{G}(\xi(t, j))$ .

A solution is *maximal* if it cannot be extended and it is *complete* when  $\text{dom } \xi$  is unbounded. We also recall the following set stability definition.

*Definition 1:* Consider system (1), the closed set  $\mathcal{A} \subset \mathbb{R}^n$  is *uniformly globally asymptotically stable* (UGAS) if there exists  $\beta \in \mathcal{KL}$  such that all solutions  $\xi$  satisfy

$$|\xi(t, j)|_{\mathcal{A}} \leq \beta(|\xi(0, 0)|_{\mathcal{A}}, t + j) \quad \forall (t, j) \in \text{dom } \xi \quad (2)$$

and all maximal solutions are complete.  $\square$

## III. FLEXRAY NETWORK

### A. General description

We investigate the scenario where a FlexRay network is used to connect the plant and the controller. The sensors/actuators signals are grouped into nodes which transmit their data in a single packet. Only one node is granted access to the communication channel at each transmission instant. Data is coded into packets and a packet is sent at each transmission instant  $t_i^s$  and is received at reception instant

$t_i^r$  for  $i \in \mathbb{Z}_{>0}$ , as illustrated in Figure 1. The transmission time  $t_i^r - t_i^s$  depends on the size of the packet: the larger the packet, the longer the transmission time.

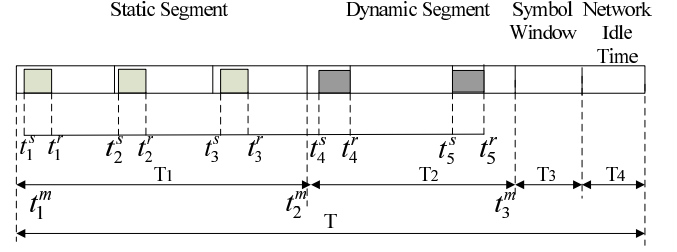


Fig. 1. FlexRay communication cycle

FlexRay is characterized by pre-set communication cycles of length  $T > 0$ . As shown in [3], each cycle contains a static segment, a dynamic segment and two protocol segments called symbol window and network idle time, which are of respective lengths  $T_1, T_2, T_3$  and  $T_4 > 0$ , see Figure 1. Distinct network access techniques are applied for the static and the dynamic segments ([3]). In particular, the static segment relies on a *time division multiple access* (TDMA) approach and the network is assigned to the nodes by a static protocol such as RR<sup>1</sup>. The dynamic segment employs the *flexible time division multiple access* (FTDMA) technique [2], which enables nodes to compete for accessing the network and a dynamic protocol such as TOD<sup>2</sup> from [13] can be implemented to select nodes to transmit messages based on the online information. The symbol window is a time slot for transmitting data for network management purposes and the network idle time is a communication-free period that is required for clock synchronization.

We assume that  $T = T_1 + T_2$  since  $T_3$  and  $T_4$  are usually negligible compared to  $T_1$  and  $T_2$ , see [3]. Note that the length of the cycle as well as  $T_1$  and  $T_2$  can be configured but are fixed once the network is implemented. We next focus on the communications during the static and the dynamic segments.

### B. Static segment

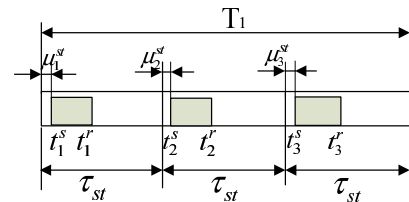


Fig. 2. Static segment

We assume that FlexRay cycles start with the static segment without loss of generality, as in Figure 1. Let  $t_j^m$ ,

<sup>1</sup>RR protocol assigns access to network in a predetermined and cyclic manner.

<sup>2</sup>TOD protocol gives access to the node with the largest mismatch between the current signal value and the last transmitted one.

$j \in \mathbb{Z}_{>0}$ , denote the segments switching instants. The static segments correspond to the following time intervals

$$[t_j^m, t_j^m + T_1], \quad j = 2n - 1 \text{ and } n \in \mathbb{Z}_{>0}. \quad (3)$$

The static segment consists of time slots of equal length  $\tau_{st} > 0$ , as illustrated in Figure 2. Hence,  $T_1 = N\tau_{st}$  for some  $N \in \mathbb{Z}$ . At the beginning of each time slot, a single node is chosen by the protocol, codes its data and sends it over the communication channel. The coding time is denoted by  $\mu_i^{st}$ ,  $i \in \mathbb{Z}_{>0}$ .

### C. Dynamic segment

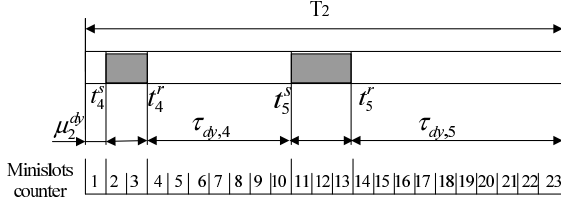


Fig. 3. Dynamic segment and minislots

In view of the definition of  $t_j^m$ ,  $j \in \mathbb{Z}_{>0}$ , in Section III-B, the dynamic segments correspond to the intervals

$$[t_j^m, t_j^m + T_2], \quad j = 2n \text{ and } n \in \mathbb{Z}_{>0}. \quad (4)$$

These intervals are composed of minislots, which are substantially shorter than static time slots and do not necessarily correspond to the transmission of a packet. Data is coded into packets before being sent to the physical channel. To transmit a packet requires more than one minislot. Nodes compete to transmit their packets and the scheduling rule associated to the dynamic segment grants the priority based on the online information. The node with highest priority is assigned with the ‘earliest’ minislot, that is the minislot with the lowest number. On the other hand, the minislots are idle when no nodes compete for access. We denote by  $\tau_{dy,i}$ ,  $i \in \mathbb{Z}_{>0}$ , the times during which the network is idle and we introduce  $\mu_j^{dy}$ ,  $j \in \mathbb{Z}_{>0}$ , to represent the time elapsed between the beginning of the dynamic segment and the first transmission.

### D. Modeling assumptions

For modeling purposes, we summarize the assumptions made so far and present additional ones.

#### Assumption 1:

- The lengths of the symbol window and the network idle time are negligible, *i.e.*  $T_3 = T_4 = 0$ .
- Data are transmitted instantaneously, *i.e.*  $t_i := t_i^s = t_i^r$  for all  $i \in \mathbb{Z}_{>0}$ .
- Data coding time is negligible and a transmission occurs at the beginning of the two segments<sup>3</sup>, *i.e.*  $\mu_i^{st} = \mu_i^{dy} = 0$  for all  $i \in \mathbb{Z}_{>0}$ .

<sup>3</sup>Note that two packets cannot be transmitted before  $\varepsilon$  units of time have elapsed during the dynamic segment, in view of item (d) of Assumption 1. The assumption of transmissions occurring at the beginning of the segments therefore exclude the case when a packet is transmitted during the last minislot of the dynamic segment. This point is important when we construct the hybrid model for the NCS in Section IV.

- The static time slots are of length  $\tau_{MATI,1} > 0$  and the dynamic segments satisfy  $\tau_{dy,i} \in [\varepsilon, \tau_{MATI,2}]$  for all  $i \in \mathbb{Z}_{>0}$ , where  $\varepsilon > 0$  refers to the length of a minislot.  $\square$

We have already mentioned that  $T_3$  and  $T_4$  are typically negligible compared to the lengths of the static and of the dynamic segments, which justifies item (a) of Assumption 1. The instantaneous transmission assumption in item (b) is reasonable when the packets are short. Otherwise, the effects of transmission time  $t_i^r - t_i^s$  can be modeled as a time-delay and we believe that the results presented hereafter can be extended to cover this case using [6]. Item (c) is also reasonable as the time needed to code the data into frames is usually very small. In addition, for given  $\tau_{MATI,1}$  and  $\tau_{MATI,2}$ , we can configure  $\tau_{st}$  such that it is equal to  $\tau_{MATI,1}$  and design the dynamic segment such that at least one node competes for transmissions in each time interval of length  $\tau_{MATI,2}$  to satisfy item (d).

Let  $\tau_{MATI,1} > 0$  be such that  $T_1 = N\tau_{MATI,1}$  for some  $N \in \mathbb{Z}$  (see Section III-B). Under Assumption 1, we have that for any given  $T_1, T_2 > 0$  the transmission instants sequence  $\{t_i\}_{i \in \mathbb{Z}_{>0}}$  satisfies

$$t_{i+1} - t_i = \tau_{MATI,1} \quad \forall t_i, t_{i+1} \in [t_{2n-1}^m, t_{2n}^m] \quad (5)$$

$$t_{i+1} - t_i \in [\varepsilon, \tau_{MATI,2}] \quad \forall t_i, t_{i+1} \in [t_{2n}^m, t_{2n+1}^m] \quad (6)$$

where  $n \in \mathbb{Z}_{>0}$  and the segment switching instants sequence  $\{t_j^m\}_{j \in \mathbb{Z}_{>0}}$  satisfies (3) and (4).

## IV. NCS MODEL

We now model NCS with FlexRay. Consider the plant model

$$\dot{x}_p = f_p(x_p, u) \quad y = g_p(x_p), \quad (7)$$

where  $x_p \in \mathbb{R}^{n_p}$  is the state,  $u \in \mathbb{R}^{n_u}$  is the control input and  $y \in \mathbb{R}^{n_y}$  is the measured output. We suppose that the following controller has been designed while ignoring the effect of the network

$$\dot{x}_c = f_c(x_c, y) \quad u = g_c(x_c), \quad (8)$$

where  $x_c \in \mathbb{R}^{n_c}$  is the state of the controller. The plant and the controller are connected via a FlexRay network composed of  $\ell$  nodes. Let  $\hat{u} \in \mathbb{R}^{n_u}$  and  $\hat{y} \in \mathbb{R}^{n_y}$  respectively denote the networked versions of  $u$  and  $y$  available at the plant and the controller. The transmissions of signals  $\hat{u}$  and  $\hat{y}$  are organized in periodic cycles as described in Section III. The implementation of the controller (8) over FlexRay leads to the following dynamics for almost all time between two successive transmissions, as in [1], [9],

$$\begin{aligned} \dot{x}_p &= f_p(x_p, \hat{u}) & t \in [t_i, t_{i+1}] \\ \dot{x}_c &= f_c(x_c, \hat{y}) & t \in [t_i, t_{i+1}] \\ \hat{y} &= \hat{f}_p(x_p, x_c, \hat{y}, \hat{u}) & t \in [t_i, t_{i+1}] \\ \hat{u} &= \hat{f}_c(x_p, x_c, \hat{y}, \hat{u}) & t \in [t_i, t_{i+1}] \\ y &= g_p(x_p) \\ u &= g_c(x_c) \end{aligned} \quad (9)$$

where the transmission instants  $t_i$ ,  $i \in \mathbb{Z}_{>0}$ , satisfy (5)-(6). The implementation of the controller using zero-order-hold

devices gives  $\hat{f}_p = 0$  and  $\hat{f}_c = 0$ , however other choices are possible. The specificity of NCS with FlexRay comes from the fact that the system variables experience two sorts of jumps which are either due to a transmission or to a segment switch. We introduce the variable  $q$  to denote the active segment:  $q(t) = 1$  denotes the static segment and  $q(t) = 2$  represents the dynamic segment. We also introduce the network-induced error  $e := (e_y, e_u)$  with  $e_y := \hat{y} - y$  and  $e_u := \hat{u} - u$ .

The error vector  $e$  is partitioned as  $e = (e_1, e_2 \dots e_\ell)$  according to the  $\ell$  nodes (after reordering, if necessary). If the  $s$ -th node, for  $s \in \{1, \dots, \ell\}$ , gets access to the network at some transmission time  $t_i$ ,  $e_s$  jumps. We typically have  $e_s(t_i^+) = 0$ . However, this condition is not needed for our results to be applicable, like in [1], [9]. The dynamics of the  $e$  variable at jumps is governed by the following rule, for  $i \in \mathbb{Z}_{>0}$ ,

$$e(t_i^+) = h(i, e(t_i), q(t_i)). \quad (10)$$

The mapping  $h$  specifies how errors at transmission times  $t_i$  are mapped to errors at  $t_i^+$ . We see that  $h$  depends on the segment through the variable  $q$  and it can be written as

$$h(i, e(t_i), q(t_i)) := (2 - q(t_i))h_1(i, e(t_i)) + (q(t_i) - 1)h_2(i, e(t_i)), \quad (11)$$

where  $h_1$  and  $h_2$  are defined by the scheduling rule of the static and the dynamic segments, respectively. In that way, the variable  $q$  generates switches of the scheduling policy between  $h_1$  and  $h_2$ . It is interesting to note that (11) generalizes existing scheduling protocol models for which we typically have  $q = 1$  and  $h = h_1$ , see [1], [6], [9] to mention a few. For instance, suppose that the static segment is governed by a RR protocol and the dynamic segment by a TOD protocol. The induced protocol is defined by (10) and (11) with functions  $h_1 := h_{RR}$  and  $h_2 := h_{TOD}$ , which are defined by, for  $e \in \mathbb{R}^{n_e}$  and  $i \in \mathbb{Z}_{>0}$ ,

$$h_{RR}(i, e) := (I - \Delta(i))e, \quad (12)$$

$$h_{TOD}(e) := (I - \Psi(e))e, \quad (13)$$

with  $\Delta(i) = \text{diag}\{\delta_1(i)I_{n_1}, \dots, \delta_\ell(i)I_{n_\ell}\}$ ,  $\Psi(e) := \text{diag}\{\psi_1(e)I_{n_1}, \psi_2(e)I_{n_2}, \dots, \psi_\ell(e)I_{n_\ell}\}$ ,  $I_{n_s}$  is the identity matrix of dimension  $n_s$  with  $\sum_{s=1}^\ell n_s = n_e$ . The functions  $\delta_s$  and  $\psi_s$  satisfy  $\delta_s(i) = 1$  when  $s - 1 = i \bmod \ell$  and  $\delta_s(i) = 0$  otherwise,  $\psi_s(e) = 1$  when  $s = \min(\arg \max_{j \in \{1, \dots, \ell\}} |e_j|)$  and  $\psi_s(e) = 0$  otherwise, for  $s \in \{1, \dots, \ell\}$  and  $i \in \mathbb{Z}_{>0}$ .

We now write the model as a hybrid system using the formalism of [4] so that we can resort to the analytical tools of [4] to infer about the stability properties of the system. We introduce for this purpose two clock variables  $\tau_1, \tau_2 \in \mathbb{Z}_{\geq 0}$ . The variable  $\tau_1$  represents the time elapsed since the last transmission and  $\tau_2$  denotes the time elapsed since the last segment switch. We also introduce  $\kappa \in \mathbb{Z}_{\geq 0}$  to count the number of transmissions. In that way, we obtain

the following hybrid system

$$\left. \begin{aligned} \dot{x} &= f(x, e) \\ \dot{\xi} &= \mathcal{F}(\xi) \\ x^+ &= x \\ \xi^+ &\in \mathcal{G}(\xi) \end{aligned} \right\} \begin{aligned} (x, \xi) &\in \mathbb{R}^{n_x} \times C \\ (x, \xi) &\in \mathbb{R}^{n_x} \times D \end{aligned} \quad (14)$$

where  $x := (x_p, x_c) \in \mathbb{R}^{n_x}$ ,  $\xi := (e, \kappa, \tau_1, \tau_2, q)$ ,  $C := \mathbb{R}^{n_e} \times \mathbb{Z}_{\geq 0} \times (C_1 \cup C_2)$ ,  $D := \mathbb{R}^{n_e} \times \mathbb{Z}_{\geq 0} \times (D_1 \cup D_2)$ , and

$$\begin{aligned} C_1 &:= [0, \tau_{\text{MATI},1}] \times [0, T_1] \times \{1\} \\ C_2 &:= [0, \tau_{\text{MATI},2}] \times [0, T_2] \times \{2\} \\ D_1 &:= ((\{\tau_{\text{MATI},1}\} \times [0, T_1]) \cup ([\varepsilon, \tau_{\text{MATI},2}] \times \{0\}) \\ &\quad \cup ([0, \tau_{\text{MATI},1}] \times \{T_1\})) \times \{1\} \\ D_2 &:= (([\varepsilon, \tau_{\text{MATI},2}] \times [0, T_2]) \cup (\{\tau_{\text{MATI},1}\} \times \{0\}) \\ &\quad \cup ([0, \varepsilon] \times \{T_2\})) \times \{2\}. \end{aligned} \quad (15)$$

The definition of the set  $C$  means that system (14) is allowed to flow in the static segment ( $q = 1$ ) when  $\tau_1 \leq \tau_{\text{MATI},1}$  and in the dynamic segment ( $q = 2$ ) when  $\tau_1 \leq \tau_{\text{MATI},2}$ . The system experiences a jump, which corresponds to a transmission when  $\tau_1 = \tau_{\text{MATI},1}$  and  $\tau_2 \leq T_1$  for  $q = 1$ , and when  $\tau_1 \in [\varepsilon, \tau_{\text{MATI},2}]$  and  $\tau_2 \leq T_2$  for  $q = 2$ , which is in agreement with Assumption 1. The other kind of jumps correspond to segment switches. The transmissions are assumed to occur at the beginning of the two segments, *i.e.* when  $\tau_1 \in [\varepsilon, \tau_{\text{MATI},2}]$  at the beginning of the static segments, where the lower bound  $\varepsilon$  is the minislot length (see item (d) of Assumption 1), and when  $\tau_1 = \tau_{\text{MATI},1}$  at the beginning of the dynamic segment.

The mapping  $\mathcal{F}$  in (14) is defined as

$$\mathcal{F}(\xi) := [g^T(x, e) \ 0 \ 1 \ 1 \ 0]^T. \quad (16)$$

The expressions of the continuous vector fields  $g$  and  $f$  in (14) are obtained by direct calculations from (9), see [9]. The mapping  $\mathcal{G}$  is given by

$$\mathcal{G}(\xi) := \begin{cases} \mathcal{G}_1(\xi) & \xi \in \Omega_1 \\ \mathcal{G}_2(\xi) & \xi \in \Omega_2 \\ \{\mathcal{G}_1(\xi), \mathcal{G}_2(\xi)\} & \xi \in \Omega_3 \end{cases} \quad (17)$$

where  $\mathcal{G}_1(\xi) := [h(\kappa, e, q)^T \ \kappa + 1 \ 0 \ \tau_2 \ q]^T$  corresponds to a transmission jump,  $\mathcal{G}_2(\xi) := [e^T \ \kappa \ \tau_1 \ 0 \ 3 - q]^T$  corresponds to a segment switching jump, and  $\Omega_i := \mathbb{R}^{n_e} \times \mathbb{Z}_{\geq 0} \times \bar{\Omega}_i$ ,  $i \in \{1, 2, 3\}$ , are given by

$$\begin{aligned} \bar{\Omega}_1 &:= (((\{\tau_{\text{MATI},1}\} \times [0, T_1]) \cup ([\varepsilon, \tau_{\text{MATI},2}] \times \{0\})) \times \{1\}) \\ &\quad \cup ((([\varepsilon, \tau_{\text{MATI},2}] \times [0, T_2]) \cup (\{\tau_{\text{MATI},1}\} \times \{0\})) \times \{2\}) \\ \bar{\Omega}_2 &:= ([0, \tau_{\text{MATI},1}] \times \{T_1\} \times \{1\}) \cup ([0, \varepsilon] \times \{T_2\} \times \{2\}) \\ \bar{\Omega}_3 &:= (\{\tau_{\text{MATI},1}\} \times \{T_1\} \times \{1\}) \cup ([\varepsilon, \tau_{\text{MATI},2}] \times \{T_2\} \times \{2\}). \end{aligned}$$

Note that the mapping  $\mathcal{G}$  is defined on the set  $D$  since  $\Omega_1 \cup \Omega_2 \cup \Omega_3 = D$ . The set  $\Omega_1$  is the set where only transmission jumps are allowed. The set  $\Omega_2$  corresponds to the jumps according to segment switches only. The set  $\Omega_3$  refers to the region of the state space where both kind of jumps can occur. The mapping  $\mathcal{G}$  is defined in such a way that, when  $\xi \in \Omega_3$ , the state may either jump according to  $\mathcal{G}_1$  or  $\mathcal{G}_2$ . In this case,

another jump will immediately occur as the state will enter either in  $\Omega_2$  or  $\Omega_1$ . This captures the fact that two kinds of jumps occur simultaneously at  $t_j^m$  for each  $j \in \mathbb{Z}_{>0}$ , see item (c) of Assumption 1. The construction of the jump map in (17) ensures that  $\mathcal{G}$  is OSC. This together with the continuity of  $(f, \mathcal{F})$  and the closedness of sets  $C$  and  $D$  guarantee that system (14) is well-posed, see Chapter 6 in [4].

## V. MAIN RESULTS

### A. Assumptions

We focus on the segment scheduling rules that satisfy the following properties.

*Assumption 2:* For each  $m \in \{1, 2\}$ , there exist  $W_m : \mathbb{R}^{n_e} \times \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ , which is locally Lipschitz in its first argument, a continuous function  $H_m : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_{\geq 0}$ , constants  $\underline{\alpha}_{W_m}, \bar{\alpha}_{W_m} > 0$ ,  $L_m \geq 0$  and  $\rho_m \in (0, 1)$  such that for any  $\kappa \in \mathbb{Z}_{\geq 0}$  and  $e \in \mathbb{R}^{n_e}$

- 1)  $\underline{\alpha}_{W_m}|e| \leq W_m(e, \kappa) \leq \bar{\alpha}_{W_m}|e|$ ,
- 2)  $W_m(h_m(\kappa, e), \kappa + 1) \leq \rho_m W_m(e, \kappa)$ ,

and for all  $x \in \mathbb{R}^{n_x}$ ,  $\kappa \in \mathbb{Z}_{\geq 0}$  and almost all  $e \in \mathbb{R}^{n_e}$

- 3)  $\left\langle \frac{\partial W_m(e, \kappa)}{\partial e}, g(x, e) \right\rangle \leq L_m W_m(e, \kappa) + H_m(x)$ .  $\square$

Items 1)-2) of Assumption 2 hold when the scheduling rules of each segment are *uniformly globally exponentially stable* (UGES), see the definition of UGES protocols in Section IV of [9]. Item 3) of Assumption 2 is verified when  $g$  is globally Lipschitz, vanishes at the origin and  $\frac{\partial W}{\partial e}$  is bounded, which is the case for RR and TOD protocols in view of Section V in [9]. Consequently, a possible choice is the RR protocol for the static segment and the TOD protocol for the dynamic segment, as given in (10)-(13).

We also assume that the controller (8) is designed such that the  $x$ -system in (14) is  $\mathcal{L}_2$  stable from  $W_m$  to  $H_m$  for  $m \in \{1, 2\}$ . Examples of systems which satisfy it can be found in [1], [9].

*Assumption 3:* There exist a continuously differentiable function  $V : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_{\geq 0}$  and  $\underline{\alpha}_V, \bar{\alpha}_V \in \mathcal{K}_\infty$  such that for each  $m \in \{1, 2\}$  there exist a positive definite continuous function  $\varrho_m$  and  $\gamma_m > 0$  such that for all  $x \in \mathbb{R}^{n_x}$ ,  $e \in \mathbb{R}^{n_e}$  and  $\kappa \in \mathbb{Z}_{\geq 0}$

$$\begin{aligned} \underline{\alpha}_V(|x|) &\leq V(x) \leq \bar{\alpha}_V(|x|) \\ \langle \nabla V(x), f(x, e) \rangle &\leq -\varrho_m(|x|) - \varrho_m(|e|) \\ &\quad - H_m(x) + \gamma_m^2 W_m(e, \kappa) \end{aligned} \quad (18)$$

with functions  $H_m$  and  $W_m$  coming from Assumption 2.  $\square$

Now that we have explained how to select the scheduling rules and presented the property the controller (8) has to ensure, we provide explicit bounds on  $\tau_{\text{MATI},1}$  and  $\tau_{\text{MATI},2}$ . A feature of the problem is that these bounds depend on the length of the segments. We first introduce some notations for the ease of presentation. Let  $\rho_m \in (0, 1)$  and  $\gamma_m > 0$  come from Assumptions 2-3,  $\pi_m := \bar{\alpha}_{W_{3-m}}/\underline{\alpha}_{W_m}$  for  $m \in \{1, 2\}$  and the function  $c_m : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{>0}$  be defined by, for

<sup>4</sup>Note that  $c_m(s)$  in (19) is a strictly positive function since it decreases when  $s$  grows and  $c_m(s) \rightarrow k_m + \delta_m > 0$  when  $s \rightarrow \infty$ .

$s > 0$ ,

$$c_m(s) := \frac{\gamma_m}{\gamma_m/k_m - d_m(s)} + \delta_m + \frac{1}{\rho_m s}, \quad (19)$$

where  $\delta_m := \frac{(M-1)\gamma_m}{\rho_m^2}$ ,  $d_m(s) := \frac{k_m M - \gamma_m}{k_m \exp(k_m \rho_m s)}$  with  $M := \max \left\{ 1, \frac{\gamma_2 \rho_1 \rho_2}{\gamma_1 \pi_2^2}, \frac{\gamma_1 \rho_1 \rho_2}{\gamma_2 \pi_1^2} \right\}$  and  $k_m := \max \left\{ \gamma_m, \frac{\gamma_m^2 \pi_{3-m}^2}{\gamma_{3-m} \rho_1 \rho_2} \right\}$ .

*Assumption 4:* For each  $T_1, T_2 > 0$ ,

$$\tau_{\text{MATI},1} = \frac{T_1}{\lceil T_1/\bar{\tau}_{\text{MATI},1} \rceil} \quad \tau_{\text{MATI},2} = \bar{\tau}_{\text{MATI},2}, \quad (20)$$

where, for  $m \in \{1, 2\}$ ,

$$\begin{aligned} \bar{\tau}_{\text{MATI},m} &= \begin{cases} \frac{1}{L_m r_m(T_m)} \arctan(\lambda(T_m)) & \sqrt{\gamma_m c_m(T_m)} > L_m \\ \frac{\rho_m \gamma_m + L_m}{L_m r_m(T_m)} - \frac{\rho_m}{\rho_m L_m + \gamma_m} & \sqrt{\gamma_m c_m(T_m)} = L_m \\ \frac{1}{L_m r_m(T_m)} \operatorname{arctanh}(\lambda(T_m)) & \sqrt{\gamma_m c_m(T_m)} < L_m, \end{cases} \end{aligned} \quad (21)$$

with  $L_m, \gamma_m > 0$  and  $\rho_m \in (0, 1)$  coming from Assumptions 2 and 3,  $\lambda(s) := \frac{r_m(s)(1-\rho_m)}{2\frac{\rho_m}{1+\rho_m}(\frac{\gamma_m}{L_m}-1)+1+\rho_m}$ ,  $r_m(s) :=$

$\sqrt{\left| \frac{\gamma_m c_m(s)}{L_m^2} - 1 \right|}$  and  $c_m : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{>0}$  given by (19).  $\square$

The MATIs in (20) adapt to the segment lengths and the latter can be arbitrarily selected. The MATIs in Assumption 4 depend on the length of the corresponding segment through the function  $c_m$ . The smaller  $T_m$  the larger  $c_m$ , which leads to a smaller  $\tau_{\text{MATI},m}$ . In particular, when  $T_1$  or  $T_2$  goes to infinity which corresponds to the case there exists only one segment, the MATI bounds in (20) tend to finite values which agree with the MATI bounds in [1], as explained in the following section. On the other hand, note that (20) is justified by the fact that  $T_1$  has to be a multiple of  $\tau_{\text{MATI},1}$  according to Section III-D.

### B. Hybrid Lyapunov function and stability results

For the sake of convenience, we write the system (14) as

$$\dot{\psi} = F(\psi) \quad \psi \in \mathcal{C}, \quad \psi^+ \in G(\psi) \quad \psi \in \mathcal{D}, \quad (22)$$

where  $\psi := (x, \xi)$ ,  $\mathcal{C} := \mathbb{R}^{n_x} \times C$ ,  $\mathcal{D} := \mathbb{R}^{n_x} \times D$ ,  $F(\psi) := [f(x, e)^T, \mathcal{F}(\xi)^T]^T$  and  $G(\psi) := [x^T, \mathcal{G}(\xi)^T]^T$ . Based on the assumptions listed above, we can study the stability of system (22) using the following Lyapunov function

$$\begin{aligned} U(\psi) &:= V(x) + \gamma_1 \phi_1(\tau_1) \theta_1(\tau_2) W_1^2(e, \kappa) (2 - q) \\ &\quad + \gamma_2 \phi_2(\tau_1) \theta_2(\tau_2) W_2^2(e, \kappa) (q - 1), \end{aligned} \quad (23)$$

where functions  $\phi_1, \phi_2, \theta_1$  and  $\theta_2$  are designed in a way such that the following proposition holds. Note that  $U$  in (23) is a generalization of the hybrid Lyapunov function in [1], which is constructed for NCS with non-switched protocols. Compared with [1], we introduce new functions  $\theta_1$  and  $\theta_2$  in (23) to accommodate the influence of segment switches. It is interesting to note that when there is a single segment ( $T_1 = \infty$ ), we can let the function  $\theta_1$  satisfy  $\theta_1(s) \equiv 1$  for all  $s \geq 0$  (since  $\theta_1$  is irrelevant in this case) and  $U$  becomes  $U(\psi) = V(x) + \gamma_1 \phi_1(\tau_1) W_1^2(e, \kappa)$  as in [1]. Furthermore, we can recover the MATI bounds in [1] by taking  $M = 1$  and  $k_1 = \gamma_1$ , which is possible since we can choose parameters

$\gamma_2, \pi_1, \pi_2, \rho_2$  in an arbitrary way as the segment  $q = 2$  will never be activated.

*Proposition 1:* Consider system (22), let  $\mathcal{A} := \{\psi : x = 0, e = 0\}$  and suppose Assumptions 2-4 hold. Then, there exist  $\underline{\alpha}_U, \bar{\alpha}_U \in \mathcal{K}_\infty$  and a continuous positive definite function  $\varrho : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  such that the following holds.

- 1)  $U$  is locally Lipschitz in  $x, e, \tau_1$  and  $\tau_2$ .
- 2) For all  $\psi \in \mathcal{C} \cup \mathcal{D}$ ,  $\underline{\alpha}_U(|\psi|_{\mathcal{A}}) \leq U(\psi) \leq \bar{\alpha}_U(|\psi|_{\mathcal{A}})$ .
- 3) For almost all  $\psi \in \mathcal{C}$ ,  $\langle \nabla U(\psi), F(\psi) \rangle \leq -\varrho(|\psi|_{\mathcal{A}})$ .
- 4) For all  $\psi \in \mathcal{D}$ ,  $U(G(\psi)) \leq U(\psi)$ .  $\square$

With the conclusion in Proposition 1, we can derive that the set  $\mathcal{A}$  is UGAS for system (22) by using similar arguments as in Proposition 3.27 in [4].

*Theorem 1:* Consider system (22) and suppose Assumptions 2-4 hold, then the set  $\mathcal{A}$  is UGAS.  $\square$

## VI. ILLUSTRATIVE EXAMPLE

In this section, we apply our results to stabilize the origin of a batch reactor in [9] over FlexRay. Consider the batch reactor plant of the form

$$\dot{x}_p = A_p x_p + B_p u \quad y = C_p x_p,$$

and the PI controller given by

$$\dot{x}_c = A_c x_c + B_c y \quad u = C_c x_c + D_c y,$$

where  $x_p \in \mathbb{R}^4$ ,  $x_c \in \mathbb{R}^2$ ,  $y \in \mathbb{R}^2$  and  $u \in \mathbb{R}^2$ ,  $A_p, B_p, C_p, A_c, B_c, C_c$  and  $D_c$  are constant matrices that can be found in Section 4 of [9]. We consider the scenario where the controller is directly connected to the actuator and the sensor measurements are transmitted via a FlexRay network which is scheduled by the mixed RR-TOD protocol in (10)-(13).

Let  $e := \hat{y} - y$ . As shown in [1], Assumptions 2-3 hold for  $H_1(x) = |\sqrt{2}A_{21}x|$  and  $H_2(x) := |A_{21}x|$  with  $\rho_1 = 0.7071$ ,  $L_1 = 15.73$ ,  $\gamma_1 = 21.5275$ ,  $\rho_2 = 0.7071$ ,  $L_2 = 15.73$  and  $\gamma_2 = 15.9222$ . Through calculations according to the definitions of  $\delta_i$  and  $k_i$  for  $i = 1, 2$  below (19), we obtain that  $\delta_1 = \delta_2 = 0$ ,  $k_1 = 232.853$ ,  $k_2 = 31.844$ .

The MATI bounds given by (20) are summarized in Table I for different values of  $T_1$  and  $T_2$ . Compared with the MATI bounds for non-switching protocols RR and TOD from [1], the NCS over FlexRay requires more frequent transmissions, which is not surprising as these adapt to the segment lengths. This becomes remarkable when the lengths of segments are particularly small. For instance, simulations have shown that for  $T_1 = 0.0186$  and  $T_2 = 0.01084$ , the system becomes unstable when we use the MATI bounds given by [1], contrary to those of (20).

## VII. CONCLUSION

We have used the emulation method to design controllers for NCS with FlexRay. We have presented a hybrid model of the NCS using the formalism of [4] and we have constructed a novel type of hybrid Lyapunov function to investigate its stability. The latter generalizes the Lyapunov function in [1] to NCS with switched protocols. We provide MATI bounds for each segment which adapt to the corresponding segment

TABLE I

	RR	TOD
MATI for RR from [1]	0.0093	-
MATI for TOD from [1]	-	0.01084
MATI for RR-TOD via (20) $T_1 = 0.0186, T_2 = 0.01084$	0.0062	0.01057
MATI for RR-TOD via (20) $T_1 = 0.5, T_2 = 1.5$	0.0082	0.01079

length, under which we have ensured a global asymptotic stability property for the system. We believe that this work, in particular the hybrid model and the Lyapunov function, can be used to address other challenges, such as tracking control or observer design for NCS with FlexRay using the results in [10], as well as other types of NCS with switched protocols.

## 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 Transactions on Automatic Control*, 52:892–897, 2007.
- [2] G. Cena and A. Valenzano. On the properties of the flexible time division multiple access technique. *IEEE Transactions on Industrial Informatics*, 2:86–94, 2006.
- [3] Flexray Consortium. *FlexRay Communications System - Protocol Specification V2.1 Rev.A*, 2005.
- [4] R. Goebel, R. G. Sanfelice, and A. R. Teel. *Hybrid dynamical systems: Modeling, Stability, and Robustness*. Princeton University Press, 2012.
- [5] D. Goswami, R. Schneider, and S. Chakraborty. Re-engineering cyber-physical control applications for hybrid communication protocols. In *Proceedings of the Design, Automation & Test in Europe Conference & Exhibition*, pages 2571–2576, 2011.
- [6] W.P.M.H. Heemels, A.R. Teel, N. van de Wouw, and D. Nešić. Networked control systems with communication constraints: Trade-offs between transmission intervals, delays and performance. *IEEE Transactions on Automatic Control*, 55:1781–1796, 2010.
- [7] W.P.M.H. Heemels and N. van de Wouw. Stability and stabilization of networked control systems. In A. Bemporad, Maurice Heemels, and Mikael Johansson, editors, *Networked Control System*, volume 406 of *Lecture notes in control and information sciences*, pages 203–253. Springer-Verlag, London, 2010.
- [8] P. Naghshtabrizi and J. P. Hespanha. Analysis of distributed control systems with shared communication and computation resources. In *Proceedings of the American Control Conference*, pages 3384–3389, Saint Louis, USA, 2009.
- [9] D. Nešić and A. R. Teel. Input-output stability properties of networked control systems. *IEEE Transactions on Automatic Control*, 49:1650–1667, 2004.
- [10] R. Postoyan, N. van de Wouw, D. Nešić, and W.P.M.H. Heemels. Tracking control for nonlinear networked control systems. *IEEE Transactions on Automatic Control*, 59:1539–1554, 2014.
- [11] R. G. Sanfelice, J. J. B. Biemond, N. van de Wouw, and W.P.M.H. Heemels. An embedding approach for the design of state-feedback tracking controllers for references with jumps. *International Journal of Robust and Nonlinear Control*, Published online in Wiley Online Library, DOI: 10.1002/rnc.2944, 2013.
- [12] E. G. Schmidt and K. Schmidt. Message scheduling for the Flexray protocol: the dynamic segment. *IEEE Transactions on Vehicular Technology*, 58:2160–2169, 2009.
- [13] G. C. Walsh and H. Ye. Scheduling of networked control systems. *IEEE Control Systems Magazine*, 21:57–65, 2001.