

# State-feedback event-holding control for nonlinear systems

W. Wang, D. Nešić, R. Postoyan, I. Shames, W.P.M.H. Heemels,

**Abstract**— We propose a novel triggering policy to implement state-feedback controllers for nonlinear systems via packet-based communication networks. The idea is to generate transmissions between the plant and the controller only when a state-dependent rule has been satisfied for a given amount of time. We refer to this new paradigm as *event-holding control*, in which a clock variable is thus only running when a state-dependent criterion is verified. This is different from time-regularized event-triggered control, where the clock variable keeps running after each transmission instant until it is reset to zero at the moment a state-based condition is verified. We approach the problem of designing an event-holding controller via emulation. We first synthesize a state-feedback law, which stabilizes the closed-loop system in the absence of the communication network. We then design the event-holding triggering mechanism under a set of general assumptions. The results are applied to two case studies consisting of linear systems and a class of nonlinear systems controlled by backstepping. We also provide a numerical backstepping control example, which demonstrates that the event-holding behaviour can reduce the number of transmissions.

## I. INTRODUCTION

Event-triggered control (ETC) refers to strategies for which the communication between the plant and the controller is orchestrated by a state-dependent rule. This implementation paradigm is well-suited for networked control systems and embedded systems, as the communication and/or the computational resources are used only when it is needed in view of the system current state, see, e.g., [8] and the references therein. In this paper, we propose a novel ETC strategy, which consists in generating transmissions only after a state-dependent criterion has been satisfied for a given time, instead of triggering a transmission instantaneously when the criterion is satisfied as in [12], [13], [18], [21] for instance. We coin the term *event-holding control* (EHC) for this novel paradigm as the transmission events are allowed to be held for a while.

The event generators of EHC are dynamic, since they depend not only on the states (the criterion) but also on

their dynamics (the holding time). Other ETC works using dynamic event generators, can be found in [5], [6], [17] for example. In fact, [6] adopts a low-pass filtered version of the static event-generator in [18], which may extend the inter-transmission times. On the other hand, [17] generalizes the time-triggering mechanism in [16] so that the dynamics of the clock variable from [16] are state-dependent, while [5] extends the time-triggered counterpart in [4], also to enlarge the inter-execution intervals.

EHC has the following important features. First, it has better robustness properties in presence of exogenous disturbances contrary to static triggering rules [13], [18], [21], as it avoids Zeno phenomenon by enforcing a fixed amount of time between any two transmissions. Second, compared to time-regularized ETC works in [1], [5], which also ensure that two consecutive transmissions are spaced, at least, by a fixed amount of time, the time variable used in EHC “freezes” when the triggering criterion is not satisfied. In contrast, the clock variable used in [1], [5] keeps running since the previous transmission until it is reset to zero. This difference, for example, the triggering mechanism relying on the event-holding time, may help enlarging the inter-transmission time, as we will show via a numerical example. Finally, EHC is well-suited for practical set-ups such as those operated by *supervisory control and data acquisition* (SCADA), where a “*hold time parameter*” is used to adjust the maximal period that a slave allows to holds an event, the detection of a condition which generally requires some responses, before reporting to the master, see the details in [9]. The “master” in SCADA refers to a central computer which gathers data and transmit instructions to “slaves”, and “slaves” are remote terminal units, which gather local data and work under the supervision of the “master”.

The setup we investigate in this paper considers state-feedback control for nonlinear systems affected by exogenous disturbances. We apply the emulation approach, and hence, we first assume the availability of a state-feedback controller, which robustly stabilizes the plant in the absence of the communication network. We then implement the controller over the network and design the triggering criterion, as well as explicit bounds on the *maximum allowable event-holding time* (MAET), based on the assumptions we make on the original closed-loop system without network. For the analysis in the second step of emulation approach, we model the overall system as a hybrid system using the formalism of [3], [7], and investigate the problem using similar techniques as the ones in [19], [20]. In [19], [20], periodic event-triggered control is addressed, which is a different implementation paradigm, requiring a different model and

This work is supported by the Australian Research Council under the Discovery Project DP170104099, HANDY project ANR-18-CE40-0010-02, and the Innovational Research Incentives Scheme under the VICI grant “Wireless control systems: A new frontier in automation” (No. 11382) awarded by NWO (The Netherlands Organisation for Scientific Research) and STW (Dutch Technology Foundation).

W. Wang, D. Nešić, I. Shames are with the Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, Australia, wei.wang@unimelb.edu.au, dnesic@unimelb.edu.au, iman.shames@unimelb.edu.au

R. Postoyan is with the Université de Lorraine, CNRS, CRAN, F-54000 Nancy, France, romain.postoyan@univ-lorraine.fr

W.P.M.H. Heemels is with the Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands, m.heemels@tue.nl

a different triggering mechanism design.

We show that the EHC system satisfies an input-to-state stability (ISS) property. The estimated ISS gain grows for a larger event-holding time, while the latter may be helpful to reduce transmission times. This shows that there is a tradeoff between the robust control performance and the network usage. We apply the results to two classes systems comprising of linear time-invariant systems and a class of nonlinear systems that are amenable to controller design via backstepping. A numerical nonlinear backstepping control example is also provided to demonstrate the effectiveness of the design.

We omit the description of the notation and definitions used throughout this paper and refer the reader to Section II in [19]. The proofs are omitted due to the space limit.

## II. EVENT-HOLDING CONTROL SETUP

### A. Problem statement

We consider the plant model

$$\dot{x}_p = f_p(x_p, u, w) \quad (1)$$

where  $x_p \in \mathbb{R}^{n_p}$  is the state,  $w \in \mathbb{R}^{n_w}$  is a vector of exogenous disturbances, and  $u \in \mathbb{R}^{n_u}$  is the control input, which is generated by the controller

$$\dot{x}_c = f_c(x_c, x_p) \quad u = g_c(x_c, x_p), \quad (2)$$

with the controller state  $x_c \in \mathbb{R}^{n_c}$ . When (2) is static, it is replaced by  $u = g_c(x_p)$  and there is no need to introduce the state  $x_c$ . The controller (2) is assumed to robustly stabilize the origin of (1), as formally stated in Section III-A, and it can be designed using any synthesis procedure. The functions  $f_p$  and  $f_c$  are assumed to be continuous, and  $g_p$  and  $g_c$  are continuously differentiable and zero at zero.

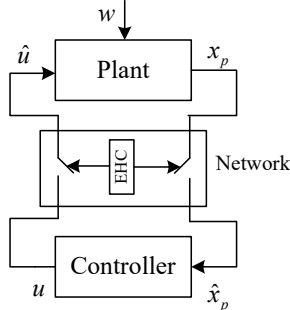


Fig. 1: Block diagram of the setup

We consider the scenario where plant (1) and controller (2) communicate with each other via a packet-based communication network, see Fig. 1. We assume that transmission delays and quantization effects are negligible and that the triggering mechanism has access to both  $x_p$  and  $u$ . We adopt this formulation because it allows covering in a unified way the cases where only  $x_p$  or  $u$  is transmitted over the network, as explained in Section 3.1 in [2].

In the present set-up, plant (1) has access to  $\hat{u}$ , the networked version of  $u$ , and controller (2) has access to the networked version  $\hat{x}_p$  of the plant state  $x_p$ . We implement

these networked variables using zero-order-hold devices, and hence,  $\hat{x}_p$  and  $\hat{u}$  are governed by  $\dot{\hat{x}}_p(t) = 0$  and  $\dot{\hat{u}}(t) = 0$  when  $t \in (t_i, t_{i+1})$ ,  $\hat{x}_p(t_i^+) = x_p(t_i)$  and  $\hat{u}(t_i^+) = u(t_i)$ , where  $t_i, i \in \mathbb{Z}_{>0}$ , denote the transmissions instants, which are defined by the triggering mechanism described next.

### B. Event-holding

The event-holding triggering policy generates transmissions only when a state-dependent rule has been verified for a given amount of time  $\tau^H > 0$ . The rule takes the form  $\Gamma(x, e) \geq 0$ , where  $x := (x_p, x_c) \in \mathbb{R}^{n_x}$ ,  $e := (e_1, e_2) \in \mathbb{R}^{n_e}$  with  $e_1 := x_p - \hat{x}_p \in \mathbb{R}^{n_p}$  the network-induced error on the state measurement and  $e_2 := u - \hat{u} \in \mathbb{R}^{n_u}$  the network-induced error on the control signal,  $n_x := n_p + n_c$  and  $n_e := n_p + n_u$ . We also need to introduce a time variable  $\tau$  to keep track of the accumulated time of  $\Gamma(x, e) \geq 0$  being satisfied, which has the dynamics

$$\begin{aligned} \dot{\tau} &\in \eta(\Gamma(x, e)) & \tau &\in [0, \tau^H] \\ \tau^+ &= 0 & \tau &\in [\tau^H, \infty), \end{aligned} \quad (3)$$

where

$$\eta(s) = \begin{cases} \{1\}, & s > 0 \\ \{0, 1\}, & s = 0 \\ \{0\}, & s < 0. \end{cases} \quad (4)$$

Then, the transmission instants are defined, for  $i \in \mathbb{Z}_{\geq 0}$ , by

$$t_0 = 0, t_{i+1} = \inf\{t > t_i \mid \Gamma(x, e) \geq 0 \wedge \tau \geq \tau^H\}. \quad (5)$$

Compared with the clock variable commonly encountered in the sampled-data literature and in time-regularized ETC,  $\tau$  grows only when  $\Gamma(x, e)$  is non-negative, and not all the time as in [?], [1], [5]. Hence, when  $\Gamma(x, e) < 0$ ,  $\tau$  freezes as  $\dot{\tau} = 0$ . Note that  $\tau$  is allowed to either grow or remain unchanged when  $\Gamma(x, e) = 0$ . This construction ensures that the map  $\eta$  in (4) is outer semi-continuous, which is important for the hybrid model presented below to be (nominally) well-posed, see Chapter 6 in [7] for more details. From (5), it is clear that a transmission is generated when  $\tau \geq \tau^H$  and  $\Gamma(x, e) \geq 0$ , obviously. The Zeno phenomenon is avoided as two successive transmissions are spaced by at least  $\tau^H$  time units.

Our objective is to design the triggering condition  $\Gamma$  in (3) and the bound  $\tau^H$  to ensure a robust stability property of the system in Fig. 1 when controller (2) is implemented via a network.

### C. System model

We implement controller (2) over the network. We derive from (1), (2) and the definitions of  $e_1$  and  $e_2$  that

$$\begin{aligned} \dot{x}_p &= f_p(x_p, u - e_2, w) \\ \dot{x}_c &= f_c(x_c, x_p - e_1) \\ u &= g_c(x_c, x_p - e_1). \end{aligned} \quad (6)$$

We then, in view of (3), model the overall system as a hybrid system, using the formalism of [3], [7], given by

$$\begin{aligned} \dot{q} &\in \mathcal{F}(q, w) & q &\in C \\ q^+ &\in \mathcal{G}(q) & q &\in D, \end{aligned} \quad (7)$$

where  $q := (x, e, \tau)$  is the state,  $C, D \subset \mathbb{R}^{n_x+n_e+1}$  are respectively the flow and the jump sets defined by

$$\begin{aligned} C &:= (\Upsilon^{\leq 0} \times \mathbb{R}_{\geq 0}) \cup (\Upsilon^{\geq 0} \times [0, \tau^H]) \\ D &:= \Upsilon^{\geq 0} \times [\tau^H, \infty), \end{aligned} \quad (8)$$

with  $\Upsilon^{\leq 0} := \{(x, e) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_e} : \Gamma(x, e) \leq 0\}$  and  $\Upsilon^{\geq 0} := \{(x, e) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_e} : \Gamma(x, e) \geq 0\}$ . The choice of set  $C$  indicates that solutions to the considered hybrid system are allowed to flow when  $\Gamma(x, e)$  is non-positive or when  $\tau$  is less than or equal to  $\tau^H$ . When  $\tau$  is larger or equal to  $\tau^H$  and  $\Gamma(x, e)$  is non-negative, a jump is enforced, according to the definition of set  $D$ . The set-valued maps  $\mathcal{F}$  and  $\mathcal{G}$  are defined as

$$\begin{aligned} \mathcal{F}(q, w) &:= \{(f(x, e, w), g(x, e, w), \eta(\Gamma(x, e)))\} \\ \mathcal{G}(q) &:= \{(x, 0, 0)\}, \end{aligned} \quad (9)$$

where  $f(x, e, w) := (f_p(x_p, g_c(x_c, x_p - e_1) - e_2, w), f_c(x_c, x_p - e_1))$ ,  $g(x, e, w) := (f_p(x, g_c(x_c, x_p - e_1) - e_2, w), \frac{\partial g_c}{\partial x}(f_p(x_p, g_c(x_c, x_p - e_1) - e_2, w), f_c(x_c, x_p - e_1)))$ .

### III. MAIN RESULTS

We now make assumptions on controller (2) such that the closed-loop system (1)-(2) satisfies some robustness properties in the absence of the network, and then design the event-holding triggering mechanism so that robustness properties are preserved when the network is involved.

#### A. Assumptions

We assume that controller (2) has been designed to robustly stabilize system (1) in the following sense.

*Assumption 1:* There exist locally Lipschitz functions  $V : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_{\geq 0}$  and  $W : \mathbb{R}^{n_e} \rightarrow \mathbb{R}_{\geq 0}$ ,  $\underline{\alpha}_V, \bar{\alpha}_V, \underline{\alpha}_W, \bar{\alpha}_W, \alpha_W, \varrho_V, \varrho_W \in \mathcal{K}_\infty$ ,  $L_V \geq 0$ ,  $L_W, a_V, \gamma > 0$  and  $\Delta \in \mathbb{R}_{>0} \cup \{\infty\}$  such that

- i-a) For all  $x \in \mathbb{R}^{n_x}$ ,  $\underline{\alpha}_V(|x|) \leq V(x) \leq \bar{\alpha}_V(|x|)$ .
- i-b) For almost all  $x \in \mathbb{R}^{n_x}$ , all  $e \in \mathbb{R}^{n_e}$  and  $w \in \mathbb{R}^{n_w}$  satisfying  $\max\{|x|, |e|, |w|\} \leq \Delta$ ,

$$\begin{aligned} \langle \nabla V(x), f(x, e, w) \rangle &\leq -a_V V(x) - \alpha_W(|e|) \\ &\quad + \gamma^2 W^2(e) + \varrho_V(|w|). \end{aligned}$$

- ii-a) For all  $e \in \mathbb{R}^{n_e}$ ,  $\underline{\alpha}_W(|e|) \leq W(e) \leq \bar{\alpha}_W(|e|)$ .
- ii-b) For almost all  $e \in \mathbb{R}^{n_e}$ , all  $x \in \mathbb{R}^{n_x}$  and  $w \in \mathbb{R}^{n_w}$  satisfying  $\max\{|x|, |e|, |w|\} \leq \Delta$ ,

$$\begin{aligned} \langle \nabla W(e), g(x, e, w) \rangle &\leq L_W W(e) + L_V \sqrt{V(x)} \\ &\quad + \varrho_W(|w|). \quad \square \end{aligned}$$

Assumption 1 states properties on the flow of the  $x$ - and the  $e$ -system of system (7). Item i) of Assumption 1 implies that  $V$  is positive definite and radially unbounded, and  $\dot{x} = f(x, e, w)$  is locally-ISS with respect to input  $(e, w)$  when  $\Delta$  is finite. It is a property of the closed-loop system (1)-(2) and is independent of the communication network. Similar assumptions are made in the emulation-based NCS literature, see, e.g., [2], [4], [5]. At this step, any control design method, such as backstepping, forwarding, feedback

linearization, high-gain techniques etc., can be applied to design the control law (2), to ensure that Assumption 1 holds.

Item ii) of Assumption 1 is an exponential growth condition of  $W(e)$ , along the solutions to  $\dot{e} = g(x, e, w)$ . Function  $W$  is required to be positive definite and radially unbounded in item ii-a). Item ii) is always feasible when  $W$  is locally Lipschitz in  $e$  and  $g$  satisfies a linear growth condition, in view of Remark 11 in [15]. Note that items i-b) and ii-b) are required to hold only in a compact set when  $\Delta$  is finite. In that case, we derive *local input-to-state stability* (LISS) when  $\Delta = \infty$ , not ISS, see Theorem 1 in Section III-C.

#### B. Event-holding mechanism design

Given Assumption 1, we define the map  $\Gamma$  in (3), for any  $x \in \mathbb{R}^{n_x}$  and  $e \in \mathbb{R}^{n_e}$ , as

$$\Gamma(x, e) := \gamma W^2(e) - \lambda \rho(\lambda) V(x) \quad (10)$$

where  $\rho(\lambda) := \lambda \frac{\gamma}{a_V}$ ,  $\lambda \in (0, \lambda^*)$  is a free design parameter with

$$\lambda^* := \min \left\{ 1, \frac{a_V}{\gamma} \right\}, \quad (11)$$

$a_V$  and  $\gamma > 0$  come from Assumption 1. Note that  $\lambda \in (0, \lambda^*)$  ensures that  $\rho(\lambda) \in (0, 1)$ .

We now design the *maximum allowable event-holding time* (MAET),  $T^H$ . We fix  $\lambda \in (0, \lambda^*)$  and let a last design parameter  $\bar{\mu}$  be such that  $\bar{\mu} \in \left[ \rho(\lambda), \frac{1}{\rho(\lambda)} \right]$ . Then, select  $\tau^H$  in (8) such that  $\tau^H \in (0, T^H(\lambda, \bar{\mu})]$ , where

$$T^H(\lambda, \bar{\mu}) := \begin{cases} \frac{1}{L_W r} \arctan(\vartheta), & \text{when } \gamma L_V > L_W, \\ \frac{1}{L_W} \frac{\bar{\mu} - \rho(\lambda)}{(\rho(\lambda) + 1)(\bar{\mu} + 1)}, & \text{when } \gamma L_V = L_W, \\ \frac{1}{L_W r} \operatorname{arctanh}(\vartheta), & \text{when } \gamma L_V < L_W, \end{cases} \quad (12)$$

$L_W$ ,  $L_V$  and  $\gamma$  come from Assumption 1,  $r := \sqrt{\left| \left( \frac{2\gamma L_V}{L_W \sqrt{a_V}} \right)^2 - 1 \right|}$ ,  $\vartheta := \frac{r(\bar{\mu} - \rho(\lambda))}{\frac{\gamma}{L_W} \frac{4L_V^2 + a_V}{2a_V} (1 + \bar{\mu}\rho(\lambda)) + \bar{\mu} + \rho(\lambda)}$ .

We see that  $\Gamma$  and  $\tau^H$  are related through the design parameter  $\lambda$ . In view of (12), for a fixed  $\lambda > 0$ ,  $T^H(\lambda, \bar{\mu})$  decreases with  $\bar{\mu}$ . For a given  $\bar{\mu} \in \Lambda^* := \left[ \rho(\lambda^*), \frac{1}{\rho(\lambda^*)} \right]$ , where  $\Lambda^* \subset \left[ \rho(\lambda), \frac{1}{\rho(\lambda)} \right]$  for any  $\lambda \in (0, \lambda^*)$  in view of the definition of  $\lambda^*$  in (11), the larger  $\lambda$  the smaller  $T^H(\lambda, \bar{\mu}(\lambda))$ . In general, a larger  $\lambda$  means less transmissions when  $w = 0$ , and a larger event-holding time  $\tau^H \in (0, T^H(\lambda, \bar{\mu}(\lambda))]$  helps to reduce the transmissions caused by disturbances, as demonstrated by the simulations results in Section V. Hence, there exists a tradeoff when selecting  $\lambda$  and  $\tau^H$  and we use  $\bar{\mu}$  to balance between  $\lambda$  and  $T^H$ .

#### C. Stability guarantee

We are ready to state the main result, which ensures that the ISS property of the closed-loop system in continuous-time, guaranteed by Assumption 1, is preserved in the presence of a network thanks to the proposed triggering mechanism.

*Theorem 1:* Consider system (7), (8) and suppose that Assumption 1 holds. Let  $\lambda \in (0, \lambda^*)$ ,  $\bar{\mu} \in [\rho(\lambda), 1/\rho(\lambda)]$  and  $\tau^H \in (0, T^H(\lambda, \bar{\mu})]$ , where  $\lambda^*$  and  $T^H$  come from (11) and (12), respectively. Then, there exist  $\beta \in \mathcal{KL}$  and  $\chi, \psi_1, \psi_2 \in \mathcal{K}_\infty$  such that for any solution pair  $(\varphi, w)$  to (7) satisfying  $|\varphi(0, 0)|_{\mathcal{A}} \leq \Delta$  and  $\|w\|_\infty \leq \Delta$ , and  $(t, j) \in \text{dom } \varphi$ ,<sup>1</sup>

$$|\varphi(t, j)|_{\mathcal{A}} \leq \beta(|\varphi(0, 0)|_{\mathcal{A}}, t + j) + \chi(\bar{\mu})\psi_1(\|w\|_\infty) + \psi_2(\|w\|_\infty), \quad (13)$$

where  $\mathcal{A} := \{q \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_e} \times \mathbb{R}_{\geq 0} | x = 0; e = 0\}$  and  $\Delta > 0$  comes from Assumption 1.  $\square$

Theorem 1 ensures that the set  $\mathcal{A}$  is LISS in general and ISS when  $\Delta = \infty$ . It also shows that solutions to system (7) locally/globally converge to a neighbourhood of the set  $\mathcal{A}$ , and the “size” of the area increases for a larger  $\bar{\mu}$  and some given  $w$ . On the other hand,  $T^H(\lambda, \bar{\mu})$  increases in  $\bar{\mu}$ , according to (12) and see also Fig. 2 given below, and the latter might help to reduce transmission times, as illustrated later via a numerical example. This implies that we can balance between the robust control performance and the network usage by adjusting the event-holding mechanism.

#### IV. CASE STUDY

In this section, we illustrate how to apply the results of Section III to two case studies consisting of linear time-invariant (LTI) systems and a class of nonlinear systems controlled by backstepping.

##### A. LTI systems

We consider the LTI system

$$\dot{x}_p = A_p x_p + B_p u + E_p w, \quad (14)$$

where  $x_p \in \mathbb{R}^{n_p}$  is the state,  $w \in \mathbb{R}^{n_w}$  is the disturbance,  $A_p, B_p$  and  $E_p$  are matrices of appropriate dimensions, and  $(A_p, B_p)$  is assumed to be stabilizable. The controller is  $u = K x_p \in \mathbb{R}^{n_u}$ , where  $K$  is a real matrix of appropriate dimension such that  $A := A_p + B_p K$  is Hurwitz, which is always possible since  $(A_p, B_p)$  is stabilizable.

We consider the scenario where the controller is co-located with the actuators and communicates with the sensors via a network. We then only consider  $e_1 = x_p - \hat{x}_p$  ( $e_2 := u - \hat{u}$  is not needed) and obtain the system defined by (7) and (8) with  $f(x, e, w) = g(x, e, w) := Ax - Be + Ew$ , where  $x := x_p$ ,  $e := e_1$ ,  $B := B_p K$  and  $E := E_p$ .

We now verify Assumption 1 and formalize it in the proposition. We state the next lemma before that.

*Lemma 1:* Let set  $A$  be Hurwitz. Then, there exist a positive definite symmetric matrix  $P$ ,  $a_V, a_W, \theta > 0$ ,  $\tilde{\eta} > a_W$  such that the LMI

$$\begin{bmatrix} \Sigma_{11} & \star & \star \\ B^T P & -(\tilde{\eta} - a_W)I_{n_e} & \star \\ E^T P & 0 & -\theta I_{n_w} \end{bmatrix} < 0. \quad (15)$$

holds, where  $\Sigma_{11} := A^T P + PA + a_V P$ .  $\square$

<sup>1</sup>See [3] for the definition of  $\|w\|_\infty$ .

The next proposition follows from Lemma 1, which ensures that Assumption 1 holds.

*Proposition 1:* Let matrix  $P$  and  $a_V, a_W, \theta > 0$  and  $\tilde{\eta} > a_W$  be generated by Lemma 1. Then

- item i) of Assumption 1 holds with  $a_V > 0$ ,

$$\begin{aligned} V(x) &= x^T P x, \quad W(e) = |e|, \\ \underline{\alpha}_V(s) &= \lambda_{\min}(P)s^2, \quad \bar{\alpha}_V(s) = \lambda_{\max}(P)s^2, \\ \alpha_W(s) &= a_W s^2, \quad \varrho_V(s) = \theta s^2, \quad \gamma = \sqrt{\tilde{\eta} - a_W}, \end{aligned}$$

for all  $s \geq 0$ ,  $x \in \mathbb{R}^{n_p}$  and  $e \in \mathbb{R}^{n_u}$ .

- item ii) of Assumption 1 holds with  $\Delta = \infty$ ,

$$\begin{aligned} \underline{\alpha}_W(s) &= \bar{\alpha}_W(s) = s, \\ \varrho_W(s) &= |E|s, \quad L_W = |B|, \quad L_V = \frac{|A|}{\sqrt{\lambda_{\min}(P)}}, \end{aligned}$$

for all  $s \geq 0$ .  $\square$

Based on Proposition 1, we derive the triggering condition  $\Gamma(x, e) = \gamma|e|^2 - \lambda\rho(\lambda)x^T P x$ , where  $\lambda \in (0, \lambda^*)$ ,  $\lambda^* = \frac{a_V}{\gamma}$ ,  $P$  and  $a_V > 0$  come from Lemma 1 and  $\gamma$  from Proposition 1, and we can apply Theorem 1 to ensure that the set  $\mathcal{A}$  defined in (13) is ISS, by noting that  $\Delta = \infty$  in this case.

##### B. Backstepping control

We consider plants of the form

$$\begin{aligned} \dot{x}_{p_1} &= f_1(x_{p_1}) + g_1(x_{p_1})x_{p_2} + w \\ \dot{x}_{p_2} &= u \end{aligned} \quad (16)$$

where  $x_{p_1} \in \mathbb{R}^n$ ,  $x_{p_2} \in \mathbb{R}$ ,  $(x_{p_1}, x_{p_2}) =: x_p$  is the state,  $w \in \mathbb{R}^n$  is the disturbance,  $u \in \mathbb{R}$  is the control input,  $f_1, g_1 : \mathbb{R}^n \rightarrow \mathbb{R}^n$  are differentiable continuous and satisfy  $f_1(0) = 0$ . The control law  $u = g_c(x_p)$  is designed by following the standard approach [10].

1) *Design of  $u = g_c(x_p)$  in the absence of networks:*

Step 1, we assume the existence of a stabilizing virtual control law  $u_1 = \phi_1(x_{p_1})$  for system  $\dot{x}_{p_1} = f_1(x_{p_1}) + g_1(x_{p_1})u_1$ , according to the next assumption.

*Assumption 2:* There exist continuously differentiable functions  $V_1 : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ ,  $\phi_1 : \mathbb{R}^n \rightarrow \mathbb{R}$  with  $\phi_1(0) = 0$ ,  $\underline{\alpha}_{V_1}, \bar{\alpha}_{V_1}, \varrho_{V_1}, \alpha_{\phi_1} \in \mathcal{K}_\infty$ ,  $a_{V_1}, L_{\phi_1} > 0$  and  $L_{\bar{\phi}_1} \geq 0$  such that the following hold:

- (i) For all  $x_{p_1}, w \in \mathbb{R}^n$ ,

$$\begin{aligned} \underline{\alpha}_{V_1}(|x_{p_1}|) &\leq V_1(x_{p_1}) \leq \bar{\alpha}_{V_1}(|x_{p_1}|), \\ \frac{\partial V_1}{\partial x_{p_1}}(x_{p_1})(f_1(x_{p_1}) + g_1(x_{p_1})\phi_1(x_{p_1}) + w) &\leq \\ &\quad - a_{V_1} V_1(x_{p_1}) + \varrho_{V_1}(\|w\|). \end{aligned}$$

- (ii) For all  $x_{p_1} \in \mathbb{R}^n$ ,  $|\phi_1(x_{p_1})| \leq \alpha_{\phi_1}(|x_{p_1}|)$ ,  $\left| \frac{\partial \phi_1}{\partial x_{p_1}}(x_{p_1}) \right| \leq L_{\bar{\phi}_1} \sqrt{V_1(x_{p_1})} + L_{\phi_1}$ .  $\square$

Item (i) of Assumption 2 says that  $V_1$  is positive definite and radially unbounded, and implies that system  $\dot{x}_{p_1} = f_1(x_{p_1}) + g_1(x_{p_1})\phi_1(x_{p_1}) + w$  is (globally) input-to-state stable with respect to  $w$ . Item (ii) of Assumption 2 states that  $|\phi_1(x_{p_1})|$  and  $\left| \frac{\partial \phi_1}{\partial x_{p_1}}(x_{p_1}) \right|$  are upper-bounded by a function of class- $\mathcal{K}_\infty$  and a function of  $\sqrt{V_1(x_{p_1})}$ , respectively.

Step 2, we construct  $u = g_c(x_p)$ . In view of Lemma 2.8 in [10], the backstepping-based controller is of the form

$$g_c(x_p) = \frac{\partial \phi_1}{\partial x_{p_1}}(x_{p_1})(f_1(x_{p_1}) + g_1(x_{p_1})x_{p_2}) - c(x_{p_2} - \phi_1(x_{p_1})) - \frac{\partial V_1}{\partial x_{p_1}}(x_{p_1})g_1(x_{p_1}), \quad (17)$$

where  $c > 0$  is a design parameter. It is designed based on

$$V(x_p) = V_1(x_{p_1}) + \frac{1}{2}(x_{p_2} - \phi_1(x_{p_1}))^2, \quad (18)$$

for any  $x_p \in \mathbb{R}^{n+1}$ . It verifies  $\underline{\alpha}_V(|x_p|) \leq V(x_p) \leq \bar{\alpha}_V(|x_p|)$  for each  $x_p \in \mathbb{R}^{n+1}$  with

$$\begin{aligned} \bar{\alpha}_V(s) &:= \bar{\alpha}_{V_1}(s) + s^2 + \alpha_{\phi_1}^2(s) \\ \underline{\alpha}_V(s) &:= \min \left\{ \alpha_{V_1}(s), \alpha_{V_1} \circ \alpha_{\phi_1}^{-1} \left( \frac{s}{2} \right), \frac{1}{8}s^2 \right\} \\ \alpha_{\phi_1}(s) &:= \max \{s, \alpha_{\phi_1}(s)\} \end{aligned} \quad (19)$$

for all  $s \geq 0$ , where  $\underline{\alpha}_V$  is generated by Proposition 1 in [14] and  $\alpha_{V_1}, \bar{\alpha}_{V_1}, \alpha_{\phi_1} \in \mathcal{K}_\infty$  come from Assumption 2. We also have that, for all  $x_p \in \mathbb{R}^{n+1}$  and  $w \in \mathbb{R}^n$ ,

$$\begin{aligned} &\langle \nabla V(x), (f_1(x_{p_1}) + g_1(x_{p_1})x_{p_2} + w, g_c(x_p)) \rangle \\ &\leq -a_{V_1}V_1(x_{p_1}) - c(x_{p_2} - \phi_1(x_{p_1}))^2 + \left| \frac{\partial \phi_1}{\partial x_{p_1}}(x_{p_1}) \right| |w| \\ &\quad + \rho_{V_1}(|w|) \end{aligned} \quad (20)$$

with  $g_c(x_p)$  from (17). By now, we have shown that the proposed controller  $u = g_c(x_p)$  robustly stabilizes plant (16) in sense of item i) of Assumption 1 with  $V$  from (18) when there is no network.

2) *Control over a network:* We consider the case where a communication network transmits data from the controller to the actuators. We follow the modeling technique in Section II and obtain the system defined by (7) and (8) with  $x := x_p$ ,  $e := u - \hat{u}$ , and

$$\begin{aligned} f(x, e, w) &:= (f_1(x_{p_1}) + g_1(x_{p_1})x_{p_2} + w, g_c(x) - e) \\ g(x, e, w) &:= \frac{\partial g_c}{\partial x}(x)f(x, e, w). \end{aligned} \quad (21)$$

We now verify the conditions of Theorem 1 using the following Lipchitz properties.

*Assumption 3:* There exist  $\Delta_{\text{bs}} \in \mathbb{R}_{>0} \cup \{\infty\}$ ,  $L_{\Delta,1}, L_{\Delta,2} > 0$  such that  $|g(x, e, w)| \leq L_{\Delta,1}(|e| + |w|) + L_{\Delta,2}\sqrt{V(x)}$  for all  $x \in \mathbb{R}^{n+1}$ ,  $e \in \mathbb{R}$  and  $w \in \mathbb{R}^n$  satisfying  $\max\{|x|, |e|, |w|\} \leq \Delta_{\text{bs}}$ .  $\square$

The following proposition ensures the satisfaction of the conditions of Theorem 1 with  $V$  from (18) and  $W(e) = |e|$  for any  $e \in \mathbb{R}$ .

*Proposition 2:* Suppose that Assumptions 2-3 hold for the system defined in (7), (8) and (21). Then, the following hold.

- Item (i) of Assumption 1 holds with  $V$  from (18),  $\bar{\alpha}_V$ ,  $\underline{\alpha}_V$  from (19),  $a_V := \min\{a_{V_1}, 2c\} - 2\nu$  and  $\nu \in (0, \min\{a_{V_1}, 2c\}/2)$ ,  $\gamma = \frac{1}{\sqrt{\nu}}$  and  $\varrho_V(s) = \varrho_{V_1}(s) + \frac{1}{\nu}L_{\phi_1}^2 s^2 + L_{\phi_1}s$  for  $s \geq 0$ , where  $a_{V_1}, L_{\phi_1}, L_{\phi_1}, \varrho_{V_1}$  from Assumption 2,  $c > 0$  from (17).

- Item (ii) of Assumption 1 holds with  $\Delta = \Delta_{\text{bs}}$ ,  $W(e) = |e|$  for all  $e \in \mathbb{R}$ ,  $\underline{\alpha}_W(s) = \bar{\alpha}_W(s) := s$  for all  $s \geq 0$ ,  $L_W = L_{\Delta,1}$ ,  $L_V = L_{\Delta,2}$ , and  $\varrho_W(s) = L_{\Delta,1}s$  for  $s \geq 0$ , where  $\Delta_{\text{bs}}, L_{\Delta,1}$  and  $L_{\Delta,2} > 0$  from Assumption 3.  $\square$

A direct consequence of Proposition 2 is that the set  $\mathcal{A}$  is LISS when  $\Delta_{\text{bs}}$  is finite and ISS when  $\Delta_{\text{bs}} = \infty$ , according to Theorem 1.

## V. NUMERICAL EXAMPLE

We apply the results of Section V.B to the “no-stall” model<sup>2</sup> of a jet engine compressor considered in [11], which has the form of (16) with

$$f_1(x_{p_1}) = -\frac{3}{2}x_{p_1}^2 - \frac{1}{2}x_{p_1}^3, \quad g_1(x_{p_1}) = -1, \quad (22)$$

where  $x_{p_1} \in \mathbb{R}$  is the mass flow and  $x_{p_2} \in \mathbb{R}$  is the pressure rise.

In this case, Assumption 2 is satisfied with the virtual control law  $u_1 = \phi_1(x_{p_1}) = c_0 x_{p_1}$ , where  $c_0 = (c_1 + 9/8) > 0$ ,  $c_1 > 0$ . In particular, for some  $c_1, c_2 > 0$ ,

- item (i) of Assumption 2 holds with  $V_1(x_{p_1}) = \frac{1}{2}x_{p_1}^2$ ,  $\alpha_{V_1}(s) = \bar{\alpha}_{V_1}(s) = \frac{1}{2}s^2$ ,  $a_{V_1} = c_1 - \nu$ ,  $\nu \in (0, \min\{\frac{1}{2}c_0c_1^2, c_1, c_2\})$  and  $\varrho_{V_1} = \frac{s^2}{\nu}$  for all  $s \geq 0$ ;
- item (ii) of Assumption 2 holds with  $\alpha_{\phi_1}(s) = c_0s$  for all  $s \geq 0$ ,  $L_{\phi_1} = c_0$ ,  $L_{\bar{\phi}_1} = 0$ .

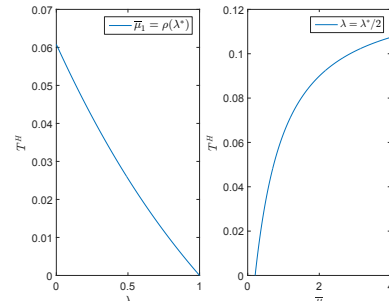


Fig. 2:  $\lambda$  vs  $T^H$  and  $\bar{\mu}$  vs  $T^H$ .

Then, we derive that  $u = g_c(x) = -(c_2 + c_0)(x_{p_2} - \phi_1(x_{p_1}))$  and  $V(x) = c_0((c_1 + \frac{9}{16})x_{p_1}^2 + \frac{1}{2}x_{p_1}^3 + \frac{1}{8}x_{p_1}^4) + \frac{1}{2}(x_{p_2} - c_0x_{p_1})^2$  in view of (17) and (18). We now implement the controller  $u = -(c_2 + c_0)(x_{p_2} - \phi_1(x_{p_1})) = k_1x_{p_1} - k_2x_{p_2}$ , where  $k_1 = (c_1 + \frac{9}{8})k_2$  and  $k_2 = c_1 + c_2 + \frac{9}{8}$ , over the network, and obtain the system defined by (7) and (8) with  $f(x, e, w) := (-\frac{3}{2}x_{p_1}^2 - \frac{1}{2}x_{p_1}^3 - x_{p_2} + w, k_1x_{p_1} - k_2x_{p_2} - e)$ ,  $g(x, e, w) := -k_1(\frac{3}{2}x_{p_1}^2 + \frac{1}{2}x_{p_1}^3 + x_{p_2} - w) - k_2(k_1x_{p_1} - k_2x_{p_2} - e)$ .

We let  $\Delta_{\text{bs}} = 1$  to verify Assumption 3, and consider the case when  $c_1 = c_2 > 0$  to simplify the computation. With the help of Matlab, we have that Assumption 3 holds with  $L_{\Delta,1} = k_2$  and  $L_{\Delta,2} = c_1/2$ . We select  $c_1 = 2.2$  and  $\nu = 1$ , and obtain that  $a_V = 2.4$ ,  $\gamma = \frac{1}{\sqrt{\nu}} = 1$  to satisfy

<sup>2</sup>The model is obtained when the stall initial condition is zero, see Section 3 in [11] for more details.

Proposition 2. We fix  $\lambda = \lambda^*/2$  and calculate  $T^H(\lambda, \bar{\mu})$  for each  $\bar{\mu} \in [\rho(\lambda), 1/\rho(\lambda)]$ . We then let  $\bar{\mu} = \rho(\lambda^*)$ , and calculate the maximal event-holding times  $T^H(\lambda, \bar{\mu})$  for each  $\lambda \in (0, \lambda^*)$ . Fig. 2 illustrates the dependency of  $T^H$  as a function of  $\lambda$  and  $\bar{\mu}$ . We can see from Fig. 2 that the larger  $\lambda$  the smaller  $T^H$  for a fixed  $\bar{\mu}$ , and  $T^H$  increases with  $\bar{\mu}$  for a fixed  $\lambda$ .

We have also considered different values of  $\lambda$  and  $\tau^H$ , with  $\lambda \in (0, \lambda^*)$  and  $\tau^H \in (0, T^H(\lambda, 1/\rho(\lambda))]$ , to illustrate the impact of  $\lambda$  and  $\tau^H$  on the transmissions times. We have run 50 simulations over 10 seconds with initial conditions randomly selected in  $\mathcal{M} := \{x \in \mathbb{R}^2 : x_{p_2} \geq \frac{k_1}{k_2} x_{p_1} \wedge |x| \leq 1\}$  and  $w$  being Gaussian white noise with variance 0.01, where  $\mathcal{M}$  is a forward invariant set of the system and  $x = (x_{p_1}, x_{p_2})$ . Simulation results show that the system might lose stability through divergence when it starts from the exterior of the set  $\mathcal{M}$ , and states of the system asymptotically converge to a neighbourhood of the origin when  $w \neq 0$  and to the origin when  $w = 0$  with initial condition selected in  $\mathcal{M}$ . The obtained average inter-transmission times are reported in Table I.

TABLE I: Event-holding Approach

$w = 0$	Average inter-transmission time		
	$\lambda = 0.01$	$\lambda = 0.9$	$\lambda = 0.95$
$\tau^H = 0.01$	0.313	0.345	0.357
$\tau^H = 0.05$	0.313	0.37	×
$\tau^H = 0.1$	0.313	×	×
$w \neq 0$	$\lambda = 0.01$	$\lambda = 0.5$	$\lambda = 0.9$
$\tau^H = 0.01$	0.204	0.238	0.263
$\tau^H = 0.05$	0.213	0.286	×
$\tau^H = 0.1$	0.303	×	×

In Table I, boxes with  $\times$  denote the case that the condition  $\tau^H \leq T^H(\lambda, 1/\rho(\lambda))$  is violated. We see that the average inter-transmission times increase when  $\lambda$  grows for a given  $\tau^H$ . When  $w = 0$ , adjusting the event-holding time  $\tau_H$  does not have a significant influence on the inter-transmission times. In contrast, when  $w$  is Gaussian white noise with variance 0.01, the average inter-transmission times grow when selecting a larger  $\tau^H$ .

TABLE II: Time-regularized Static Event-triggering

$w \neq 0$	Average inter-transmission time		
	$\lambda = 0.01$	$\lambda = 0.9$	$\lambda = 0.95$
$T = 0.01$	0.1	0.1136	0.123

We also compare the results in Table I to the time-regularized static event-triggering rule, which triggers a transmission when  $\Gamma(x, e) \geq 0$  and the time elapsed from the previous transmission reaches  $T$  times units. We simulate similarly as above, run 50 simulations over 10 seconds with initial conditions randomly selected in the set  $\mathcal{M}$  and  $w$  being Gaussian white noise with variance 0.01. The obtained data is given in Table II. We can see that the average inter-transmission time for the time-regularized static event-triggering method is about a half for the event-holding strategy in Table I, which reflects that the event-holding behavior helps to reduce transmission times.

## VI. CONCLUSIONS

We proposed an event-holding method as a new paradigm to implement full state-feedback controllers for nonlinear systems with exogenous disturbances via communication networks. The event-holding strategy uses a dynamic event-triggering policy and generates transmissions between the plant and the controller only when a state-dependent rule has been satisfied for a given amount of time, which distinguishes itself from other event-triggering methods. This novel approach may behave better on saving network bandwidth when compared with time-regularized static event-triggered controllers, as also shown by a numerical example.

## REFERENCES

- [1] M. Abdelrahim, R. Postoyan, J. Daafouz, and D. Nešić. Stabilization of nonlinear systems using event-triggered output feedback controllers. *IEEE Transactions on Automatic Control*, 61:2682–2687, 2016.
- [2] M. Abdelrahim, R. Postoyan, J. Daafouz, and D. Nešić. Robust event-triggered output feedback controllers for nonlinear systems. *Automatica*, 75:96–108, 2017.
- [3] C. Cai and A. R. Teel. Characterizations of input-to-state stability for hybrid systems. *Systems & Control Letters*, 58:47–53, 2009.
- [4] 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.
- [5] V.S. Dolk, D.P. Borgers, and W.P.M.H. Heemels. Output-based and decentralized dynamic event-triggered control with guaranteed  $L_p$ -gain performance and Zeno-freeness. *IEEE TAC*, 62:34–49, 2017.
- [6] A. Girard. Dynamic triggering mechanisms for event-triggered control. *IEEE Transactions on Automatic Control*, 60:1992–1997, 2015.
- [7] R. Goebel, R.G. Sanfelice, and A.R. Teel. *Hybrid dynamical systems: Modeling, Stability, and Robustness*. Princeton University Press, 2012.
- [8] W.P.M.H. Heemels, K.H. Johansson, and P. Tabuada. *Event-Triggered and Self-Triggered Control*, pages 384–391. Springer, 2015.
- [9] Control Microsystems Inc. *DNP3: User and Reference Manual*, 2007. [https://scadahacker.com/library/Documents/ICS\\_Protocols/Control%20Microsystems%20%20DNP3%20User%20and%20Reference%20Manual.pdf](https://scadahacker.com/library/Documents/ICS_Protocols/Control%20Microsystems%20%20DNP3%20User%20and%20Reference%20Manual.pdf).
- [10] M. Krstić, I. Kanellakopoulos, and P. V. Kokotović. *Nonlinear and adaptive control design*. New York: Wiley, 1995.
- [11] M. Krstić and P. V. Kokotović. Lean backstepping design for a jet engine compressor model. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 1047–1052, Albany, USA, 1995.
- [12] T. F. Liu and Z. P. Jiang. A small-gain approach to robust event-triggered control of nonlinear systems. *IEEE TAC*, 60:2071–2085, 2015.
- [13] J. Lunze and D. Lehmann. A state-feedback approach to event-based control. *Automatica*, 46:211–215, 2010.
- [14] D. Nešić and A. R. Teel. Stabilization of sampled-data nonlinear systems via backstepping on their Euler approximate model. *Automatica*, 46:1801–1808, 2006.
- [15] D. Nešić and A.R. Teel. Input-output stability properties of networked control systems. *IEEE TAC*, 49:1650–1667, 2004.
- [16] D. Nešić, A.R. Teel, and D. Carnevale. Explicit computation of the sampling period in emulation of controllers for nonlinear sampled-data systems. *IEEE Transactions on Automatic Control*, 54:619–624, 2009.
- [17] R. Postoyan, P. Tabuada, D. Nešić, and A. Anta. A framework for the event-triggered stabilization of nonlinear systems. *IEEE TAC*, 60:982–996, 2015.
- [18] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52:1680–1685, 2007.
- [19] W. Wang, R. Postoyan, D. Nešić, and W.P.M.H. Heemels. Periodic event-triggered output feedback control of nonlinear systems. In *Proceedings of the 57th IEEE CDC*, Florida, USA, 2018.
- [20] W. Wang, R. Postoyan, D. Nešić, and W.P.M.H. Heemels. Robust periodic event-triggered control of nonlinear systems. *IEEE Transactions on Automatic Control*, accepted.
- [21] H. Yu and P. J. Antsaklis. Event-triggered output feedback control for networked control systems using passivity: Achieving stability in the presence of communication delays and signal quantization. *Automatica*, 49:30–38, 2013.



Minerva Access is the Institutional Repository of The University of Melbourne

**Author/s:**

Wang, W;Nesic, D;Postoyan, R;Shames, I;Heemels, WPMH

**Title:**

State-feedback event-holding control for nonlinear systems

**Date:**

2020-03-12

**Citation:**

Wang, W., Nesic, D., Postoyan, R., Shames, I. & Heemels, W. P. M. H. (2020). State-feedback event-holding control for nonlinear systems. Proceedings of the 2019 IEEE 58th Conference on Decision and Control (CDC), 2019-December, pp.1650-1655. IEEE. <https://doi.org/10.1109/CDC40024.2019.9029241>.

**Persistent Link:**

<http://hdl.handle.net/11343/249533>