

Self-Triggered Feedback Control Systems With Finite-Gain \mathcal{L}_2 Stability

Xiaofeng Wang, *Member, IEEE*, and Michael D. Lemmon, *Member, IEEE*

Abstract—This paper examines a class of real-time control systems in which each control task triggers its next release based on the value of the last sampled state. Prior work [1] used simulations to demonstrate that self-triggered control systems can be remarkably robust to task delay. This paper derives bounds on a task's sampling period and deadline to quantify how robust the control system's performance will be to variations in these parameters. In particular we establish inequality constraints on a control task's period and deadline whose satisfaction ensures that the closed-loop system's induced \mathcal{L}_2 gain lies below a specified performance threshold. The results apply to linear time-invariant systems driven by external disturbances whose magnitude is bounded by a linear function of the system state's norm. The plant is regulated by a full-information \mathcal{H}_∞ controller. These results can serve as the basis for the design of soft real-time systems that guarantee closed-loop control system performance at levels traditionally seen in hard real-time systems.

Index Terms—Finite-gain \mathcal{L}_2 stability, real-time control systems, self-triggered.

I. INTRODUCTION

COMPUTER-CONTROLLED systems are often implemented using periodic tasks satisfying hard real-time constraints. Under a periodic task model, consecutive invocations (also called jobs) of a task are released in a periodic manner. If the task model satisfies a hard real-time constraint, then each job completes its execution by a specified deadline. Hard real-time periodic task models allow the control system designer to treat the computer-controlled system as a discrete-time system, for which there are a variety of mature controller synthesis methods.

Periodic task models may be undesirable in many situations. Traditional approaches for estimating task periods and deadlines are very conservative, so the control task may have greater utilization than it actually needs. This results in significant overprovisioning of the real-time system hardware. With such high utilization, it may be difficult to schedule other tasks on the same processing system. Secondly, it should be noted that real-time scheduling over networked systems may be poorly served by the periodic task model. In many networked systems, tasks are fin-

ished only after information has been successfully transported across the network. It can be expensive to provide hard real-time guarantees on message delivery in communication networks. This is particularly true for wireless sensor-actuator networks. In these applications, it may be better to consider alternatives to periodic task models that can more effectively balance the real-time system's computational cost against the control system's performance as suggested in [2], [3].

This paper considers a **self-triggered** task model in which each task determines the release of its next job. In reality, one might consider periodic task models as self-triggered tasks since many implementations release tasks upon expiration of a one-shot timer that was started by the previous invocation of the task. Under a periodic task model, the period of this one-shot timer is always a constant value. This paper, however, considers a more adaptive form of self-triggering in which the value loaded into the one-shot timer is actually a function of the system state sampled by the current job. Under this "state-based" self-triggering, each task releases its next job based on the system state. We can therefore consider "state-based" self-triggering as a closed-loop form of releasing tasks for execution, whereas periodic task models release their jobs in an open-loop fashion. For simplicity, this paper refers to a "state-based" self-triggered task model as "self-triggered".

Self-triggering provides a more flexible way of adjusting task periods. Since task periods are based on the system's current state, it is possible to reduce control task utilization during periods of time when the system is sitting happily at its equilibrium point. The question here is precisely how much freedom do we have in adjusting task periods in response to variations in the system state. This paper answers that question by providing bounds on the task periods and deadlines required to assure a specified level of \mathcal{L}_2 stability. Our results pertain to linear time-invariant systems with state feedback. Since our controller seeks to ensure \mathcal{L}_2 stability, we use a full-information \mathcal{H}_∞ controller in our analysis. We also assume that the system has a process noise whose magnitude is bounded by a linear function of the norm of the system state. Under these assumptions we obtain the bounds for the task periods and deadlines as functions of the system state. Taking advantage of these bounds, a self-triggered scheme is presented where the periods and deadlines are uniformly bounded from below by a positive constant. On the basis of simulation results, these bounds appear to be tight and relatively easy to compute, so it may be possible to use them in actual real-time control systems.

The remainder of this paper is organized as follows. Section II reviews the prior work related to self-triggered feedback. Section III introduces the system model. The main contributions

Manuscript received July 31, 2007; revised January 30, 2008. Current version published March 11, 2009. This work was supported in part by the National Science Foundation under Grants NSF-ECS-0400479 and NSF-CNS-0720457. Recommended by Associate Editor A. Astolfi.

The authors are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: xwang13,lemmon@nd.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2009.2012973

of the paper are summarized in Section IV and their development is detailed in Sections V and VI. Section V derives a sufficient threshold condition that can serve as an event trigger for state sampling. Section VI presents a self-triggering scheme and proves that it is \mathcal{L}_2 stable. Simulations are shown in Section VII. Finally, conclusions and future work are presented in Section VIII.

II. PRIOR WORK

To the best of our knowledge there is relatively little prior work examining state-based self-triggered feedback control. A self-triggered task model was introduced by Velasco *et al.* [4] in which a heuristic rule was used to adjust task periods. A self-triggered task model was also introduced by Lemmon *et al.* [1] which chose task periods based on a Lyapunov-based technique. But the authors did not provide analytic bounds for task periods and the task delays were considered only in the simulation results. Other than these two papers, we are aware of no other serious work looking at self-triggered feedback schemes. There is, however, a great deal of related work dealing with so-called event-triggered feedback, sample period selection, and real-time control system co-design. We will review each of these areas in more detail below and then discuss their relationship to the self-triggered task models.

Traditional methods for sample period selection [5] are usually based on Nyquist sampling. Nyquist sampling ensures that the sampled signal can be perfectly reconstructed from its samples. In practice, however, feedback within the control system means the system's performance will be somewhat insensitive to errors in the feedback signal, so that perfect reconstruction is much more than we require in a feedback control system. An alternative approach to the sample period selection problem makes use of Lyapunov techniques. This was done in Zheng *et al.* [6] for a class of nonlinear sampled-data system. Nesic *et al.* [7] used input-to-state stability (ISS) techniques to bound the inter-sample behavior of nonlinear systems. \mathcal{L}_p stability of sampled-data systems was considered in [8]. Further work was done in [9], [10] where upper bounds on the sampling periods were provided, known as the maximal allowable transfer interval (MATI).

The sampling periods determined by the aforementioned methods can be conservative because they are essentially "open-loop." Sample periods are selected before the system is deployed, so this selection must ensure adequate behavior over a wide range of possible input disturbances. As a result, these selected periods may be shorter than necessary. This fact was demonstrated by Tabuada *et al.* [11] where sampling instants were determined on-line using the current system state. In this case, the average sampling periods of Tabuada's event-triggered scheme appeared to be significantly longer than what one would have chosen using traditional estimates of the MATI.

Another related research direction viewed sample period selection as a "co-design" problem that involves both the control system and the real-time system. In this case, sample periods are selected to minimize some penalty on control system performance subject to a schedulability condition. Early statements of this problem may be found in Seto *et al.* [12] with more recent studies in [13] and [14]. The penalty function is often a perfor-

mance index for an infinite horizon optimal control problem. It has, however, been demonstrated [15] that under slow sampling such indices may not be monotone functions of the sampling period. As a result, it only appears to be feasible to do off-line determination of these "optimal" sampling periods. Instead of considering quadratic cost functions, [16] presented an approach to maximize the stability radius subject to a schedulability condition. Although this approach can enlarge the family of stabilizing controllers, it did not provide a direct relationship between the sampling periods and the control performance of the systems.

The prior work on co-design really focuses on optimizing performance subject to scheduling constraints. The scheduling constraints are Liu-Layland [17] schedulability conditions for earliest deadline first (EDF) scheduling. It is not always clear, however, that these are the best set of constraints to be using. This paper actually derives a set of constraints on both the periods and deadlines that we can then use as a quality-of-service (QoS) constraint that the real-time scheduler needs to meet. We do not address the schedulability of these QoS constraints in this paper, though that is an important research issue that we are still studying.

In recent years, a number of researchers have proposed aperiodic and sporadic task models in which tasks are event-triggered [18]. By event-triggering, we usually mean that the system state is sampled when some function of the system state exceeds a threshold. The idea of event-triggered feedback has appeared under a variety of names, such as interrupt-based feedback [19], Lebesgue sampling [20], or state-triggered feedback [11]. Event triggering usually requires some form of hardware event detector to generate a hardware interrupt to release the control task. This can be done using either custom analog integrated circuits (ASIC's) or floating point gate array (FPGA) processors. Event triggering provides a useful way of adaptively adjusting task periods at run time, provided the cost associated with using ASIC/FPGA hardware is acceptable. In some applications, however, it may be unreasonable or impractical to retrofit an existing system with such "event detectors". In these cases, a software approach such as the self-triggered scheme presented in this paper may be more appropriate.

The prior work on event-triggered feedback is probably most closely related to this paper's work. In particular, the bounds we derive are based on variations of the event-triggering conditions used by Tabuada *et al.* [11]. The techniques used in this paper are similar to the input-to-state stability (ISS) methods used in [9], [10] for bounding the MATI. Compared with the prior work in [1], this paper derives explicit bounds on the task sampling periods and deadlines, which appear to be tight and computationally efficient. Based on these bounds, we present a practical self-triggered scheme, where the sampling periods and deadlines are uniformly bounded from below by a positive constant. These bounds provide less conservative sampling times than those obtained using the MATI estimates in [9]. Moreover, in many cases the average sampling periods generated by our approach are less conservative than those generated by [11]. Finally, a major contribution in this paper is an explicit state-dependent bound on the acceptable delay, something which is not found in either [9], [10], or [11].

III. SYSTEM MODEL

Consider a linear time-invariant system whose state $x : [0, \infty) \rightarrow \mathfrak{R}^n$ satisfies the initial value problem,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 w(t) \\ x(0) &= x_0 \end{aligned}$$

where $x_0 \in \mathfrak{R}^n$ is the non-zero initial state, $u : [0, \infty) \rightarrow \mathfrak{R}^m$ is a control input and $w : [0, \infty) \rightarrow \mathfrak{R}^l$ is an exogenous disturbance function in \mathcal{L}_2 . In the above equation, $A \in \mathfrak{R}^{n \times n}$, $B_1 \in \mathfrak{R}^{n \times m}$, and $B_2 \in \mathfrak{R}^{n \times l}$ are real matrices of appropriate dimensions.

Since we are interested in controllers that are finite-gain \mathcal{L}_2 stable, we assume there exists a full-information \mathcal{H}_∞ controller that asymptotically stabilizes the unforced system. In particular, we assume there exists a symmetric positive semi-definite matrix P such that

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u(t) \\ u(t) &= -B_1^T P x(t) \end{aligned} \quad (1)$$

has an asymptotically stable equilibrium. The matrix P satisfies the \mathcal{H}_∞ algebraic Riccati equation (ARE) [21, p. 138],

$$0 = PA + A^T P - Q + R \quad (2)$$

where

$$Q = PB_1 B_1^T P \quad (3)$$

$$R = I + \frac{1}{\gamma^2} PB_2 B_2^T P \quad (4)$$

for some real constant $\gamma > 0$. For notational convenience the system matrix of the closed-loop system ((1)) will be denoted as $A_{cl} = A - B_1 B_1^T P$. The state feedback gain matrix is $K = -B_1^T P$.

If we consider the standard \mathcal{L}_2 storage function $V : \mathfrak{R}^n \rightarrow \mathfrak{R}$ given by $V(x) = x^T P x$ for all $x \in \mathfrak{R}^n$ then the preceding assumptions about P allow us to show that the storage function's directional derivative satisfies the dissipative inequalities,

$$\dot{V}(x(t)) \leq -\|x(t)\|_2^2 - \|u(t)\|_2^2 + \gamma^2 \|w(t)\|_2^2 \quad (5)$$

for all t . Recall that a linear system, \mathbf{T} , is said to be finite-gain \mathcal{L}_2 stable from w to $\mathbf{T}w$ if \mathbf{T} is a linear operator from \mathcal{L}_2 back into \mathcal{L}_2 . The induced gain of \mathbf{T} is

$$\|\mathbf{T}\| = \sup_{\|w\|_{\mathcal{L}_2}=1} \|\mathbf{T}w\|_{\mathcal{L}_2}. \quad (6)$$

Satisfaction of the dissipative inequality in (5) is sufficient to show that the system \mathbf{T} characterized by the state equation

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 w(t) \\ u(t) &= -B_1^T P x(t) \end{aligned} \quad (7)$$

is finite-gain \mathcal{L}_2 stable from the disturbance w to $(x^T, u^T)^T$ with an induced gain less than γ .

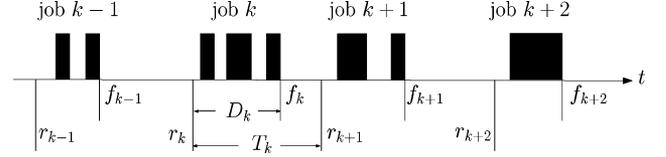


Fig. 1. Relationship between task period (T_k), delay (D_k), release time (r_k), and finishing time (f_k).

This paper considers a sampled-data implementation of the closed-loop system in (7). This means that the plant's control, u , is computed by a computer task. This task is characterized by two monotone increasing sequences of time instants; the release time sequence $\{r_k\}_{k=0}^\infty$ and the finishing time sequence $\{f_k\}_{k=0}^\infty$. We say these two sequences are admissible if $r_k \leq f_k \leq r_{k+1}$ for all $k = 0, \dots, \infty$. The time r_k denotes the time when the k th invocation of a control task (also called a job) is released for execution on the computer's central processing unit (CPU). At this time, we assume that the system state is sampled so that r_k also represents the k th sampling time instant. The time f_k denotes the time when the k th job has finished executing. Each job of the control task computes the control u based on the last sampled state. Upon finishing, the control job outputs this control to the plant. The control signal used by the plant is held constant by a zero-order hold (ZOH) until the next finishing time f_{k+1} . This means that the sampled-data system under study satisfies the following set of state equations,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 w(t) \\ u(t) &= -B_1^T P x(r_k) \end{aligned} \quad (8)$$

for $t \in [f_k, f_{k+1})$ and all $k = 0, \dots, \infty$. The state trajectories x satisfying (8) are continuous so that the initial state at time f_k is simply $x(f_k) = \lim_{t \uparrow f_k} x(t)$. In the following discussion, we will present a self-triggering scheme which ensures finite-gain \mathcal{L}_2 stability of the sampled-data system in (8) from w to x .

We let $T_k = r_{k+1} - r_k$ denote the k th inter-release time ($k = 0, \dots, \infty$). T_k can therefore be interpreted as a time-varying "sampling" period by control engineers and a time-varying "task" period by real-time system engineers. We let $D_k = f_k - r_k$ denote the time interval between the k th job's release and finishing time. Control engineers would view D_k as the "delay" of the k th job whereas real-time system engineers would view D_k as the "jitter" of the k th job. If the control task satisfies a hard real-time constraint, then the delay D_k is required to lie below a specified "deadline". Fig. 1 illustrates the relationship between the task period, T_k , the delay D_k , the task finishing time f_k , and the release time r_k . The x -axis in Fig. 1 is time with the period (T_k), delay (D_k), finishing time (f_k), and release (r_k) marked on the axis. The black rectangles above the time axis mark intervals over which the task is executing.

IV. SUMMARY OF CONTRIBUTIONS

The main contribution of this paper is a detailed analysis of self-triggering schemes that were first presented in [1]. The findings of this analysis are summarized in theorem 6.10. This "self-triggering" theorem states that the closed-loop sampled-

data system is \mathcal{L}_2 stable for the disturbance w to the state x if the task's $k + 1$ st release time is generated by

$$r_{k+1} = f_k + L_2(x(r_k), x(r_{k-1}); D_k, \delta) \quad (9)$$

and the delay D_{k+1} satisfies

$$D_{k+1} \leq \xi(x(r_k); \epsilon, \delta). \quad (10)$$

In the above equations, L_2 is a function of the system state at release times r_k, r_{k-1} , the delay D_k , and a parameter δ . The function ξ can be taken as a requirement on the task's deadline and it is a function of the state at time r_k and two parameters (ϵ and δ). These two parameters can be used to control the relative length of the job's period and deadline.

The self-triggering theorem shows that the bounding functions L_2 and ξ have the following forms,

$$\begin{aligned} & L_2(x(r_k), x(r_{k-1}); D_k, \delta) \\ &= \frac{1}{\alpha} \ln \left(1 + \alpha \frac{\delta \rho(x(r_k)) - \phi(x(r_k), x(r_{k-1}); D_k)}{\mu_0(x(r_k)) + \alpha \phi(x(r_k), x(r_{k-1}); D_k)} \right) \\ \xi(x(r_k); \epsilon, \delta) &= \frac{1}{\alpha} \ln \left(1 + \epsilon \alpha \frac{(1 - \delta) \rho(x(r_k))}{\alpha \delta \rho(x(r_k)) + \mu_0(x(r_k))} \right) \end{aligned}$$

where α is a real constant, ρ and μ_0 are class \mathcal{K} functions, and ϕ bounds a function of the system state as it evolves over the delay time D_k . Note that the form of these results is reminiscent of the MATI bounds obtained in [9]. The main difference is that our bounds are now functions of the previously sampled states.

The other major contributions of the paper are simulation results showing that the bounds in theorem 6.10 do not appear to be overly conservative. These simulations demonstrate that the sampling period generated by the self-triggered controller can be significantly longer than the bound on MATI in [9] for the same system. For systems that evolve over a wide set of time scales, the self-triggered scheme has longer sampling periods than the event-triggered scheme proposed in [11]. Simulation results demonstrate that a self-triggered scheme with an average sampling period of \bar{T} has better disturbance rejection than periodically triggered systems with the same sampling period of \bar{T} . Self-triggered schemes, of course, have a higher computational cost (as measured by floating point operations per update) than periodically triggered schemes with the same sampling period. But for systems with similar disturbance rejection levels, computer utilization (as measured by the ratio of computation time and task period) is comparable between self-triggered and periodically triggered systems. All of these empirical findings appear to support the conjecture that the bounds presented in theorem 6.10 are not overly conservative.

V. \mathcal{L}_2 STABILITY

Consider the sampled-data system in (8) with a set of admissible release and finishing time sequences. For all k , define the k th job's error function $e_k : [r_k, f_{k+1}) \rightarrow \mathfrak{R}^n$ by $e_k(t) = x(t) - x(r_k)$. This error represents the difference between the current system state and the system state at the last re-

lease time, r_k . This section presents two inequality constraints on $e_k(t)$ (see theorem 5.1 and corollary 5.2 below) whose satisfaction is sufficient to ensure that the sampled-data system's \mathcal{L}_2 gain is less than γ/β for some parameter $\beta \in (0, 1]$.

The following theorem states that if a function of the state error $e_k(t)$ and state $x(t)$ satisfies a certain inequality constraint, then the closed-loop system in (8) is finite-gain \mathcal{L}_2 stable.

Theorem 5.1: Consider the sampled-data system in (8) with admissible release and finishing time sequences. Let $x(r_0) = x_0$ and β be any real constant in the interval $(0, 1]$ with the matrix Q as given in (3). If

$$e_k^T(t) Q e_k(t) \leq (1 - \beta^2) \|x(t)\|_2^2 + x^T(r_k) Q x(r_k) \quad (11)$$

holds for all $t \in [f_k, f_{k+1})$ and any $k = 0, \dots, \infty$, then the sampled-data system is finite-gain \mathcal{L}_2 stable from w to x with a gain less than γ/β .

Proof: Consider the storage function $V : \mathfrak{R}^n \rightarrow \mathfrak{R}$ given by $V(x) = x^T P x$ for $x \in \mathfrak{R}^n$ where P is a symmetric positive semi-definite matrix satisfying the algebraic Riccati equation ((2)). The directional derivative of V for $t \in [f_k, f_{k+1})$ is

$$\begin{aligned} \dot{V} &= \frac{\partial V}{\partial x} (Ax(t) - B_1 B_1^T P x(r_k) + B_2 w(t)) \\ &= -x(t)^T \left(I - Q + \frac{1}{\gamma^2} P B_2 B_2^T P \right) x(t) \\ &\quad - 2x(t)^T Q x(r_k) + 2x(t)^T P B_2 w(t) \\ &= -x(t)^T (I - Q) x(t) - \left\| \gamma w(t) - \frac{1}{\gamma} B_2^T P x(t) \right\|_2^2 \\ &\quad + \gamma^2 \|w(t)\|_2^2 - 2x(t)^T Q x(r_k) \\ &\leq -x(t)^T (I - Q) x(t) + \gamma^2 \|w(t)\|_2^2 - 2x(t)^T Q x(r_k). \end{aligned}$$

Insert $x(t) = e_k(t) + x(r_k)$ into the above equation to obtain

$$\begin{aligned} \dot{V} &\leq -\|x(t)\|_2^2 + [e_k(t) + x(r_k)]^T Q [e_k(t) + x(r_k)] \\ &\quad - 2[e_k(t) + x(r_k)]^T Q x(r_k) + \gamma^2 \|w(t)\|_2^2 \\ &= -\|x(t)\|_2^2 + e_k(t)^T Q e_k(t) - x^T(r_k) Q x(r_k) \\ &\quad + \gamma^2 \|w(t)\|_2^2. \end{aligned} \quad (12)$$

By the assumption in (11), we know that (12) can be rewritten as

$$\dot{V} \leq -\beta^2 \|x(t)\|_2^2 + \gamma^2 \|w(t)\|_2^2 \quad (13)$$

which holds for all t and is sufficient to ensure the sampled-data system is finite-gain \mathcal{L}_2 stable from w to x with a gain less than γ/β . ■

In our following work, we will find it convenient to use a slightly weaker sufficient condition for \mathcal{L}_2 stability which is only a function of the state error $e_k(t)$. The following corollary states this result.

Corollary 5.2: Consider the sampled-data system in (8) with admissible sequences of release and finishing times. Let $x(r_0) = x_0$ and Q be a real matrix that satisfies (3). For any $\beta \in (0, 1]$, let

$$M = (1 - \beta^2)I + Q \quad (14)$$

$$N = \frac{1}{2}(1 - \beta^2)I + Q. \quad (15)$$

If the state error trajectory satisfies

$$e_k(t)^T M e_k(t) \leq x^T(r_k) N x(r_k) \quad (16)$$

for $t \in [f_k, f_{k+1})$ and all $k = 0, \dots, \infty$, then the sampled data system is finite-gain \mathcal{L}_2 stable from w to x with a gain less than γ/β .

Proof: Equation (16) can be rewritten as

$$\begin{aligned} e_k(t)^T M e_k(t) &= (1 - \beta^2) \|e_k(t)\|_2^2 + e_k(t)^T Q e_k(t) \\ &\leq \frac{1}{2}(1 - \beta^2) \|x(r_k)\|_2^2 + x(r_k)^T Q x(r_k). \end{aligned}$$

This can be rewritten to obtain

$$\begin{aligned} &e_k(t)^T Q e_k(t) \\ &\leq (1 - \beta^2) (\|x(r_k)\|_2^2 + \|e_k(t)\|_2^2) + x^T(r_k) Q x(r_k) \\ &\quad - (1 - \beta^2) \left(\frac{1}{2} \|x(r_k)\|_2^2 + 2 \|e_k(t)\|_2^2 \right) \\ &= (1 - \beta^2) (\|x(r_k)\|_2^2 + \|e_k(t)\|_2^2) + x^T(r_k) Q x(r_k) \\ &\quad - (1 - \beta^2) \left(\frac{1}{2} \|x(r_k)\|_2^2 + 2 \|e_k(t)\|_2^2 \right) \\ &\quad - (1 - \beta^2) (2x^T(r_k)e_k(t) - 2x^T(r_k)e_k(t)) \\ &= (1 - \beta^2) \|x(r_k) + e_k(t)\|_2^2 + x^T(r_k) Q x(r_k) \\ &\quad - (1 - \beta^2) \left\| \frac{1}{\sqrt{2}}x(r_k) + \sqrt{2}e_k(t) \right\|_2^2 \\ &\leq (1 - \beta^2) \|x(t)\|_2^2 + x(r_k)^T Q x(r_k). \end{aligned}$$

This inequality is the sufficient condition in theorem 5.1 so we can conclude that the sampled-data system is \mathcal{L}_2 stable from w to x with a gain less than γ/β . ■

Remark 5.3: The inequalities in (11) or (16) can both be used as the basis for an event-triggered feedback control system (Section II). Note that both inequalities are trivially satisfied at $t = r_k$. If we let the delay, D_k , be zero for each job, then by triggering the release times $\{r_k\}_{k=0}^{\infty}$ anytime before the inequalities in (11) or (16) are violated, we will ensure the sampled-data system's induced \mathcal{L}_2 gain remains below γ/β . The resulting event-triggered feedback system is very similar to the state-triggering scheme proposed by Tabuada *et al.* [11] for asymptotic stability. The main difference between that result and this one is that our proposed event-triggering condition provides a stronger assurance on the sampled-data system's performance as measured by its induced \mathcal{L}_2 gain.

VI. ADMISSIBLE RELEASE AND FINISHING TIMES

This section establishes sufficient conditions for the existence of admissible sequences of release and finishing times that ensure the sampled data system in (8) is \mathcal{L}_2 stable with a specified gain. These conditions take the form of admissible bounds on the task sampling periods, T_k , and task delays, D_k . Based on these bounds, we present a self-triggered scheme, where the sampling periods and deadlines are uniformly bounded from below by a positive constant. The following assumption

is placed on the disturbance $w(t)$ to ensure these bounds are nonzero.

Assumption 6.1: Consider the sampled-data system in (8). Assume that there exists a positive real constant $W > 0$ so that $\|w(t)\|_2 \leq W \|x(t)\|_2$ for all $t \geq 0$.

Remark 6.2: Assumption 6.1 consists of a restricted class of signals whose norm is bounded by a linear function of the state's norm. The more precise way to state this assumption is $\|w(t, x(t))\|_2 \leq W \|x(t)\|_2$ for all $t \geq 0$, which means that w depends on x as well as t . But, to make the notation consistent, we still use $w(t)$ to denote the disturbance instead of $w(t, x)$. Such disturbances may arise in uncertain systems when there are unmodeled dynamics caused by fluctuations in plant parameters.

For notational convenience let $z_k : [r_k, f_{k+1}) \rightarrow \mathfrak{R}^n$ be given as

$$z_k(t) = \sqrt{(1 - \beta^2)I + Q} e_k(t) = \sqrt{M} e_k(t) \quad (17)$$

where \sqrt{M} is a matrix square root and M is defined in (14). We refer to z_k as the k th job's "trigger signal". Note that M is dependent on the parameter β . In the following discussion, we assume M has full rank by properly choosing β . It also implies that \sqrt{M} has full rank. Notice that $M \geq N$ always holds and, if M has full rank, M, N will be both positive definite, where N is defined in (15).

We define the function $\rho : \mathfrak{R}^n \rightarrow \mathfrak{R}$ given by

$$\rho(x) = \sqrt{x^T N x} \quad (18)$$

where $x \in \mathfrak{R}^n$. So if we can guarantee for any $\delta \in (0, 1]$ that

$$\|z_k(t)\|_2 \leq \delta \rho(x(r_k)) \quad (19)$$

for all $t \in [f_k, f_{k+1})$ for any $k = 0, \dots, \infty$, then the hypotheses in corollary 5.2 are satisfied and we can conclude that the sampled-data system is finite-gain \mathcal{L}_2 stable from w to x with a gain less than γ/β .

The first major result examines what happens if we use (19) as the basis for an event-triggered feedback control system. In particular, let us assume that the k th job's release, r_k , is precisely that time when $\|z_k(t)\|_2 = \delta \rho(x(r_k))$ under the assumption that the k th job's delay, D_k , is zero. The following theorem states a lower bound on the sampling period for which a sampled-data system with zero delay (i.e., $D_k = 0$) has an induced \mathcal{L}_2 gain less than γ/β .

Theorem 6.3: Consider the sampled-data system in (8) satisfying assumption 6.1. Assume that M has full rank and for some $\delta \in (0, 1]$ that the sequence of release times $\{r_k\}_{k=0}^{\infty}$ satisfy

$$\|z(r_{k+1})\|_2 = \delta \rho(x(r_k)) \quad (20)$$

where $f_k = r_k$ for all $k = 0, \dots, \infty$.

The sequence of release and finishing times is admissible and the sampled-data system is finite-gain \mathcal{L}_2 stable from w to x with a gain less than γ/β . Furthermore, the task sampling periods satisfy

$$T_k \geq \frac{1}{\alpha} \ln \left(1 + \delta \alpha \frac{\rho(x(r_k))}{\mu_0(x(r_k))} \right) \quad (21)$$

where α is a real constant

$$\alpha = \left\| \sqrt{M}A\sqrt{M}^{-1} \right\| + W \left\| \sqrt{M}B_2 \right\| \left\| \sqrt{M}^{-1} \right\| \quad (22)$$

and $\mu_0 : \mathfrak{R}^n \rightarrow \mathfrak{R}$ is a real-valued function given by

$$\mu_0(x(r_k)) = \left\| \sqrt{M}A_{cl}x(r_k) \right\|_2 + W \left\| \sqrt{M}B_2 \right\| \left\| x(r_k) \right\|_2. \quad (23)$$

Proof: Let $\Phi = \{t \in [r_k, f_{k+1}) \mid \|z_k(t)\|_2 = 0\}$. The time derivative of $\|z_k(t)\|_2$ for $t \in [r_k, f_{k+1}) \setminus \Phi$ satisfies

$$\begin{aligned} \frac{d}{dt} \|z_k(t)\|_2 &\leq \left\| \sqrt{M}\dot{e}_k(t) \right\|_2 = \left\| \sqrt{M}\dot{x}(t) \right\|_2 \\ &= \left\| \sqrt{M} (Ax(t) - B_1B_1^T Px(r_k) + B_2w(t)) \right\|_2 \\ &\leq \left\| \sqrt{M}Ae_k(t) \right\|_2 + \left\| \sqrt{M}A_{cl}x(r_k) \right\|_2 \\ &\quad + \left\| \sqrt{M}B_2 \right\| \|w(t)\|_2, \end{aligned} \quad (24)$$

where the righthand sided derivative is used when $t = r_k$. Since $\|w(t)\|_2 \leq W\|x(t)\|_2$, $x(t) = e_k(t) + x(r_k)$, and $z_k(t) = \sqrt{M}e_k(t)$, we can bound the preceding (24) as

$$\begin{aligned} \frac{d}{dt} \|z_k(t)\|_2 &\leq \left\| \sqrt{M}A\sqrt{M}^{-1} \right\| \|z_k(t)\|_2 + \left\| \sqrt{M}A_{cl}x(r_k) \right\|_2 \\ &\quad + W \left\| \sqrt{M}B_2 \right\| \left\| \sqrt{M}^{-1} z_k(t) \right\|_2 \\ &\quad + W \left\| \sqrt{M}B_2 \right\| \left\| x(r_k) \right\|_2 \\ &\leq \left(\left\| \sqrt{M}A\sqrt{M}^{-1} \right\| + W \left\| \sqrt{M}B_2 \right\| \left\| \sqrt{M}^{-1} \right\| \right) \|z_k(t)\|_2 \\ &\quad + \left\| \sqrt{M}A_{cl}x(r_k) \right\|_2 + W \left\| \sqrt{M}B_2 \right\| \left\| x(r_k) \right\|_2 \\ &= \alpha \|z_k(t)\|_2 + \mu_0(x(r_k)). \end{aligned} \quad (25)$$

where α and $\mu_0 : \mathfrak{R}^n \rightarrow \mathfrak{R}$ are defined in (22) and (23), respectively.

The initial condition is $\|z_k(r_k)\|_2 = 0$. Using this in the differential inequality in (25) yields,

$$\|z_k(t)\|_2 \leq \frac{\mu_0(x(r_k))}{\alpha} \left(e^{\alpha(t-r_k)} - 1 \right) \quad (26)$$

for all $t \in [r_k, f_{k+1})$ since $\|z_k(t)\|_2 = 0$ for all $t \in \Phi$.

By assumption $r_{k+1} = f_{k+1}$ (i.e., no task delay) and $\delta\rho(x(r_k)) = \|z_k(r_{k+1})\|_2$, so we can conclude that

$$\delta\rho(x(r_k)) = \|z_k(r_{k+1})\|_2 \leq \frac{\mu_0(x(r_k))}{\alpha} \left(e^{\alpha T_k} - 1 \right) \quad (27)$$

where $T_k = r_{k+1} - r_k$ is the task sampling period for job k . Solving (27) for T_k yields (21). The righthand side of inequality (21) is clearly strictly greater than zero, which implies that $r_{k+1} - r_k > 0$. Therefore $r_k = f_k \leq r_{k+1}$ which implies that the sequence of finishing and release times is admissible. Finally we know that $\|z_k(t)\|_2 \leq \delta\rho(x(r_k))$ for all $t \in [r_k, f_{k+1}) = [f_k, f_{k+1})$ and all $k = 0, \dots, \infty$, which by corollary 5.2 implies that the system is \mathcal{L}_2 stable from w to x with a gain less than γ/β . ■

Remark 6.4: Note that the righthand side of (21) will always be strictly greater than zero. We can therefore conclude that if

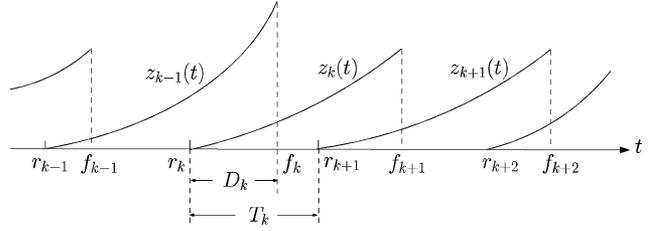


Fig. 2. Time history of $z_k(t)$ with non-zero task delay.

we trigger release times when $\delta\rho(x(r_k)) = \|z_k(r_{k+1})\|$, then the sampling period T_k can never be zero.

Remark 6.5: The admissibility of sequences $\{r_k\}_{k=0}^{\infty}$ and $\{f_k\}_{k=0}^{\infty}$ can be restated in terms of the sequences $\{D_k\}_{k=0}^{\infty}$ and $\{T_k\}_{k=0}^{\infty}$. By definition, the release and finishing time sequences are admissible if and only if $r_k \leq f_k \leq r_{k+1}$ for all k . Clearly this holds if and only if $0 \leq D_k \leq T_k$ for all k .

The previous theorem presumes there is no task delay (i.e., $D_k = 0$). Under this assumption, theorem 6.3 states that triggering release times when (20) holds assures the closed-loop system's induced \mathcal{L}_2 gain. This theorem, however, also provides a lower bound on the task sampling period, which suggests that we can also use theorem 6.3 as the basis for state-based self-triggered feedback. In this scenario, if the k th job would set the next job's release time as

$$r_{k+1} = r_k + \frac{1}{\alpha} \ln \left(1 + \delta\alpha \frac{\rho(x(r_k))}{\mu_0(x(r_k))} \right) \quad (28)$$

then we are again assured that the system's induced \mathcal{L}_2 gain is less than γ/β .

The problem faced in using (28) for self-triggering is the assumption of no task delay. In many applications, the task delay may not be small enough to neglect. If we consider non-zero delay, then the triggering signals appear as shown in Fig. 2. This figure shows the time history for the triggering signals, z_{k-1} , z_k , and z_{k+1} . With non-zero delay, we can partition the time interval $[r_k, f_{k+1})$ into two subintervals $[r_k, f_k)$ and $[f_k, f_{k+1})$. The differential equations associated with subintervals $[r_k, f_k)$ and $[f_k, f_{k+1})$ are

$$\begin{aligned} \dot{x}(t) &= Ax(t) - B_1B_1^T Px(r_{k-1}) + B_2w(t) \quad \text{and} \\ \dot{x}(t) &= Ax(t) - B_1B_1^T Px(r_k) + B_2w(t), \end{aligned}$$

respectively. In a manner similar to the proof of theorem 6.3, we can use differential inequalities to bound $z_k(t)$ for all $t \in [r_k, f_{k+1})$ and thereby determine sufficient conditions assuring the admissibility of the release/finishing times while preserving the closed-loop system's \mathcal{L}_2 -stability. The next two lemmas (lemma 6.6 and 6.8) characterize the behavior of $z_k(t)$ over these two subintervals. We then use lemma 6.8 to establish sufficient conditions assuring the \mathcal{L}_2 -stability of the sampled-data system with non-zero delay. The proofs of these lemmas have been moved to the paper's appendix.

Lemma 6.6: Consider the sampled-data system in (8) satisfying assumption 6.1. Assume that M has full rank and for some k , $r_{k-1} \leq f_{k-1} \leq r_k$. Given some $\epsilon \in (0, 1)$, let

$L_1 : \mathfrak{R}^n \times \mathfrak{R}^n \times (0, 1) \rightarrow \mathfrak{R}$, $\phi : \mathfrak{R}^n \times \mathfrak{R}^n \times \mathfrak{R} \rightarrow \mathfrak{R}$, and $\mu_1 : \mathfrak{R}^n \times \mathfrak{R}^n \rightarrow \mathfrak{R}$ be real-valued functions given by

$$L_1(x(r_k), x(r_{k-1}); \epsilon) = \frac{1}{\alpha} \ln \left(1 + \frac{\epsilon \alpha \rho(x(r_k))}{\mu_1(x(r_k), x(r_{k-1}))} \right), \quad (29)$$

$$\phi(x(r_k), x(r_{k-1}); t - r_k) = \frac{\mu_1(x(r_k), x(r_{k-1}))}{\alpha} \left(e^{\alpha(t-r_k)} - 1 \right), \text{ and} \quad (30)$$

$$\begin{aligned} & \mu_1(x(r_k), x(r_{k-1})) \\ &= W \left\| \sqrt{M} B_2 \right\| \|x(r_k)\|_2 \\ &+ \left\| \sqrt{M} (Ax(r_k) - B_1 B_1^T P x(r_{k-1})) \right\|_2, \quad (31) \end{aligned}$$

respectively, where α is a positive real constant given by (22) and $\rho : \mathfrak{R}^n \rightarrow \mathfrak{R}$ is given by (18). If the k th finishing time f_k satisfies

$$0 \leq D_k = f_k - r_k \leq L_1(x(r_k), x(r_{k-1}); \epsilon) \quad (32)$$

for all $t \in [r_k, f_k]$, then the k th trigger signal, z_k , satisfies

$$\|z_k(t)\|_2 \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \leq \epsilon \rho(x(r_k)) \quad (33)$$

for all $t \in [r_k, f_k]$.

Remark 6.7: In lemma 6.6, $L_1(x(r_k), x(r_{k-1}); \epsilon)$ serves as a bound on the maximal allowable delay. Since we are considering non-zero delays, we would like $L_1(x(r_k), x(r_{k-1}); \epsilon)$ to be strictly away from zero. To ensure that, we need to guarantee $\|x(r_{k-1})\|_2 / \|x(r_k)\|_2$ is bounded by a positive constant from above. The upper bound of $\|x(r_{k-1})\|_2 / \|x(r_k)\|_2$ can be obtained provided that $\|z_{k-1}(r_k)\|_2 \leq \delta \rho(x(r_{k-1}))$ and $x(r_{k-1}) \neq 0$ hold. This is because

$$\begin{aligned} 0 &\geq \|z_{k-1}(r_k)\|_2^2 - \delta^2 \rho(x(r_{k-1}))^2 \\ &= x(r_k)^T M x(r_k) - 2x(r_k)^T M x(r_{k-1}) \\ &\quad + x(r_{k-1})^T (M - \delta^2 N) x(r_{k-1}) \\ &\geq -2\|x(r_k)\|_2 \|M\| \|x(r_{k-1})\|_2 \\ &\quad + \underline{\lambda}(M - \delta^2 N) \|x(r_{k-1})\|_2^2. \end{aligned}$$

Because $\|x(r_{k-1})\|_2 > 0$, the inequality above implies

$$0 \geq -2\|x(r_k)\|_2 \|M\| + \underline{\lambda}(M - \delta^2 N) \|x(r_{k-1})\|_2$$

which means $\|x(r_{k-1})\|_2 / \|x(r_k)\|_2 \leq 2\|M\| / \underline{\lambda}(M - \delta^2 N)$ since $M \geq N > \delta^2 N > 0$. With the fact that $\|x(r_{k-1})\|_2 / \|x(r_k)\|_2$ is bounded by a positive constant from above, it is easy to show that $L_1(x(r_k), x(r_{k-1}); \epsilon)$ is greater than a positive constant, if $\|z_{k-1}(r_k)\|_2 \leq \delta \rho(x(r_{k-1}))$ holds.

Lemma 6.8: Consider the sampled-data system in (8) satisfying assumption 6.1. Assume M has full rank. For a given integer k and some $\epsilon \in (0, 1)$, assume that $r_{k-1} \leq f_{k-1} \leq r_k$. For any $\eta \in (\epsilon, 1]$, let

$$d_\eta = f_k + L_2(x(r_k), x(r_{k-1}); D_k, \eta) \quad (34)$$

where $L_2 : \mathfrak{R}^n \times \mathfrak{R}^n \times \mathfrak{R} \times (0, 1] \rightarrow \mathfrak{R}$ is given by

$$\begin{aligned} & L_2(x(r_k), x(r_{k-1}); D_k, \eta) \\ &= \frac{1}{\alpha} \ln \left(1 + \alpha \frac{\eta \rho(x(r_k)) - \phi(x(r_k), x(r_{k-1}); D_k)}{\mu_0(x(r_k)) + \alpha \phi(x(r_k), x(r_{k-1}); D_k)} \right). \quad (35) \end{aligned}$$

if

$$0 \leq D_k \leq L_1(x(r_k), x(r_{k-1}); \epsilon) \quad (36)$$

then

$$d_\eta > f_k \text{ and} \quad (37)$$

$$\|z_k(t)\|_2 \leq \eta \rho(x(r_k)) \text{ for all } t \in [f_k, d_\eta]. \quad (38)$$

According to lemma 6.8, for a positive constant $\delta \in (\epsilon, 1)$, if $r_{k+1} = f_k + L_2(x(r_k), x(r_{k-1}); D_k, \delta)$ and $f_{k+1} \leq f_k + L_2(x(r_k), x(r_{k-1}); D_k, 1)$ hold, we will always have $\|z_k(r_{k+1})\|_2 \leq \delta \rho(x(r_k))$ and $\|z_k(f_{k+1})\|_2 \leq \rho(x(r_k))$. We will use this fact below to characterize a self-triggering scheme that preserves the sampled-data system induced \mathcal{L}_2 gain. Theorem 6.10 formally states this self-triggering scheme. The proof of theorem 6.10 requires the following lemma which shows that the upper bound for delays given in lemma 6.6 is bounded below by a positive function of $x(r_{k-1})$. In that case, the deadline for D_k can be predicted at time r_{k-1} . The proof of this lemma will be found in the paper's appendix .

Lemma 6.9: Consider the sampled-data system in (8) satisfying assumption 6.1. Assume that M has full rank and for a constant $\delta \in (0, 1)$, the release time r_{k-1} and r_k satisfy

$$\|z_{k-1}(r_k)\|_2 \leq \delta \rho(x(r_{k-1})) \quad (39)$$

for any given k . Then L_1 given by (29) satisfies

$$L_1(x(r_k), x(r_{k-1}); \epsilon) \geq \xi(x(r_{k-1}); \epsilon, \delta) > 0 \quad (40)$$

where $\epsilon \in (0, \delta)$ and $\xi : \mathfrak{R}^n \times (0, 1) \times (0, 1) \rightarrow \mathfrak{R}$ is a real-valued function given by

$$\xi(x(r_{k-1}); \epsilon, \delta) = \frac{1}{\alpha} \ln \left(1 + \frac{\epsilon(1-\delta)\rho(x(r_{k-1}))}{\delta\rho(x(r_{k-1})) + \frac{\mu_0(x(r_{k-1}))}{\alpha}} \right). \quad (41)$$

With the preceding technical lemma we can now state a self-triggered feedback scheme which can guarantee the sampled-data system's induced \mathcal{L}_2 gain. The basis for this self-triggering scheme will be found in the following theorem.

Theorem 6.10: Consider the sampled-data system in (8) satisfying assumption 6.1. Assume M has full rank. For given $\epsilon \in (0, 1)$ and $\delta \in (\epsilon, 1)$, we assume that

- The initial release and finishing times satisfy

$$r_{-1} = r_0 = f_0 = 0$$

- For any non-negative integer k , the release times are generated by the following recursion,

$$r_{k+1} = f_k + L_2(x(r_k), x(r_{k-1}); D_k, \delta) \quad (42)$$

and the finishing times satisfy

$$r_{k+1} \leq f_{k+1} \leq r_{k+1} + \xi(x(r_k); \epsilon, \delta) \quad (43)$$

where L_2 is given in (35) and ξ is given in (41). Then the sequence of release times, $\{r_k\}_{k=0}^\infty$, and finishing time, $\{f_k\}_{k=0}^\infty$, will be admissible and the sampled-data system is finite-gain \mathcal{L}_2 stable from w to x with an induced gain less than γ/β .

Proof: From the definition of ξ in (41), we can easily see that $\xi(x(r_k); \epsilon, \delta) > 0$ for any non-negative integer k . We can therefore use (43) to conclude that the interval $[r_{k+1}, r_{k+1} + \xi(x(r_k); \epsilon, \delta)]$ is nonempty for all k .

Next, we insert (42) into (43) to show that

$$\begin{aligned} f_{k+1} &\leq r_{k+1} + \xi(x(r_k); \epsilon, \delta) \\ &\leq f_k + L_2(x(r_k), x(r_{k-1}); D_k, \delta) + \xi(x(r_k); \epsilon, \delta) \\ &\leq f_k + L_2(x(r_k), x(r_{k-1}); D_k, 1) \end{aligned} \quad (44)$$

for all non-negative integers k .

With the preceding two preliminary results, we now consider the following statement about the k th job. This statement is that

- 1) $r_k \leq f_k \leq r_{k+1}$,
- 2) $\|z_k(t)\|_2 \leq \delta\rho(x(r_k))$ for all $t \in [f_k, r_{k+1}]$,
- 3) and $\|z_k(t)\|_2 \leq \rho(x(r_k))$ for all $t \in [f_k, f_{k+1}]$.

We now use mathematical induction to show that under the theorem's hypotheses, this statement holds for all non-negative integers k .

First consider the base case when $k = 0$. According to the definition of L_2 ((35)) we know that

$$L_2(x_0, x_0; D_0, \delta) = L_2(x_0, x_0; 0, \delta) > 0.$$

We can therefore combine (43) and (42) to obtain

$$r_0 = f_0 \leq f_0 + L_2(x_0, x_0; D_0, \delta) = r_1 \quad (45)$$

which establishes the first part of the inductive statement when $k = 0$.

Next note that

$$D_0 = 0 \leq L_1(x(r_0), x(r_{-1}); \epsilon). \quad (46)$$

If we use the fact that $\delta \in (\epsilon, 1) \subset (0, 1]$ in (42) and (46), we can see that the hypotheses of lemma 6.8 are satisfied. This means that $\|z_0(t)\|_2 \leq \delta\rho(x(r_0))$ for all $t \in [f_0, r_1]$ which completes the second part of the inductive statement for $k = 0$.

Now define the time

$$d_1^0 = f_0 + L_2(x(r_0), x(r_{-1}); D_0, 1)$$

Equation (46) again implies that the hypotheses of lemma 6.8 are satisfied, so that

$$\|z_0(t)\|_2 \leq \rho(x(r_0)) \text{ for all } t \in [f_0, d_1^0]. \quad (47)$$

From (44), we know that $f_1 \leq d_1^0$. We can also combine (43) and (45) to conclude that $f_0 \leq f_1$. We therefore know that $[f_0, f_1] \subseteq [f_0, d_1^0]$ which combined with (47) implies that

$$\|z_0(t)\|_2 \leq \rho(x(r_0)) \text{ for all } t \in [f_0, f_1].$$

This therefore establishes the last part of the inductive statement for $k = 0$.

We now turn to the general case for any k . For a given k let us assume that the statement holds. This means that

$$r_k \leq f_k \leq r_{k+1} \quad (48)$$

$$\|z_k(t)\|_2 \leq \delta\rho(x(r_k)) \text{ for all } t \in [f_k, r_{k+1}] \quad (49)$$

$$\|z_k(t)\|_2 \leq \rho(x(r_k)) \text{ for all } t \in [f_k, f_{k+1}]. \quad (50)$$

Now consider the $k + 1$ st job. Because (49) is true, the hypothesis of lemma 6.9 is satisfied which means there exists a function ξ (given by (41)) such that

$$0 < \xi(x(r_k); \epsilon, \delta) \leq L_1(x(r_{k+1}), x(r_k); \epsilon).$$

We can use this in (43) to obtain

$$0 \leq D_{k+1} \leq \xi(x(r_k); \epsilon, \delta) \leq L_1(x(r_{k+1}), x(r_k); \epsilon). \quad (51)$$

From (51) and the fact that $\delta \in (0, 1)$ we know that the hypotheses of lemma 6.8 hold and we can conclude that

$$f_{k+1} \leq r_{k+2} \text{ and} \quad (52)$$

$$\|z_{k+1}(t)\|_2 \leq \delta\rho(x(r_{k+1})) \text{ for all } t \in [f_{k+1}, r_{k+2}]. \quad (53)$$

Combining (43) with (52) yields $r_{k+1} \leq f_{k+1} \leq r_{k+2}$ which establishes the first part of the statement for the case $k + 1$. Equation (53) is the second part of the statement.

Finally let

$$d_1^{k+1} = f_{k+1} + L_2(x(r_{k+1}), x(r_k); D_{k+1}, 1).$$

Following our prior argument for the case when $k = 0$, we know that the validity of (51) satisfies the hypotheses of lemma 6.8. We can therefore conclude that

$$\|z_{k+1}(t)\|_2 \leq \rho(x(r_{k+1})) \text{ for all } t \in [f_{k+1}, d_1^{k+1}]. \quad (54)$$

According to (44), $f_{k+2} \leq d_1^{k+1}$. We can therefore combine (43) and (52) to show that $f_{k+1} \leq f_{k+2}$ and therefore conclude that $[f_{k+1}, f_{k+2}] \subseteq [f_{k+1}, d_1^{k+1}]$. Combining this observation with (54) yields $\|z_{k+1}(t)\|_2 \leq \rho(x(r_{k+1}))$ for all $t \in [f_{k+1}, f_{k+2}]$ which completes the third part of the inductive statement for case $k + 1$.

We may therefore use mathematical induction to conclude that the inductive statement holds for all non-negative integers k . The first part of the statement, of course, simply means that the sequences $\{r_k\}_{k=0}^\infty$ and $\{f_k\}_{k=0}^\infty$ are admissible. The third part of the inductive statement implies that the hypotheses of corollary 5.2 are satisfied, thereby ensuring that the system's induced \mathcal{L}_2 gain is less than γ/β . ■

Remark 6.11: $\xi(x(r_k); \epsilon, \delta)$ serves as the deadline for the delay D_{k+1} in theorem 6.10.

Remark 6.12: As is evident from the way it was constructed, δ controls the next job's release time. We might therefore expect to see a larger δ result in larger sampling periods. This is indeed confirmed by the analysis. Since

$$T_k \geq r_{k+1} - f_k = L_2(x(r_k), x(r_{k-1}); D_k, \delta)$$

and since L_2 is an increasing function of δ we can see that larger δ result in larger sampling periods.

Remark 6.13: By our construction of the parameter ϵ , we see that it controls the current job's finishing time. Since this

$$D_k = f_k - r_k \leq \xi(x(r_k); \epsilon, \delta)$$

and since ξ is an increasing function of ϵ , we can expect to see the allowable delay increase as we increase ϵ . Note also that ξ is a decreasing function of δ so that adopting a longer sampling period by increasing δ will have the effect of reducing the maximum allowable task delay.

Remark 6.14: From the previous two remarks we see that the parameters δ and ϵ can be used to control the task's deadline and period. One way to choose ϵ and δ is to enforce real-time schedulability constraints such as those discussed in [22]. As a "rule of thumb" a reasonable strategy is to choose δ and ϵ so that L_2 and ξ are as large as possible; as this makes the task easier to schedule under an earliest-deadline first (EDF) scheduling discipline. This suggests that ϵ and δ may be seen as parameters in a scheduling-controller co-design method similar in philosophy to the approach introduced in [12]. We are currently working to see if this idea indeed provides a useful formalism for the systematic co-design of real-time self-triggered control systems.

Remark 6.15: The prior techniques can also be applied to self-triggered systems in which $\|w(t)\|_2 \leq W$, thereby relaxing assumption 6.1. In this case, however, it is easy to see that the bounds on sampling periods and deadlines will asymptotically go to zero, thereby leading to "chattering" behavior. A topic for future research is how best to address this issue when we can only guarantee the disturbance is uniformly bounded by a constant.

The following corollary to the above theorem shows that the task periods and deadlines generated by our self-triggered scheme are all bounded away from zero. This is important in establishing that our scheme does not generate infinite sampling frequencies.

Corollary 6.16: Let the assumptions in theorem 6.10 hold. Then there exist two positive constants $\zeta_1, \zeta_2 > 0$ such that $T_k \geq \zeta_1$ and $\xi(x(r_k); \epsilon, \delta) \geq \zeta_2$.

Proof: From theorem 6.10, we know

$$f_k - r_k \leq \xi(x(r_{k-1}); \epsilon, \delta) \leq L_1(x(r_k), x(r_{k-1}); \epsilon).$$

Therefore, by lemma 6.6,

$$\|z_k(f_k)\|_2 \leq \phi(x(r_k), x(r_{k-1}); D_k) \leq \epsilon \rho(x(r_k)).$$

Let us first take a look at T_k . From (42), we have

$$\begin{aligned} T_k &\geq r_{k+1} - f_k = L_2(x(r_k), x(r_{k-1}); D_k, \delta) \\ &\geq \frac{1}{\alpha} \ln \left(1 + \alpha \frac{\delta \rho(x(r_k)) - \epsilon \rho(x(r_k))}{\mu_0(x(r_k)) + \alpha \epsilon \rho(x(r_k))} \right) \\ &\geq \frac{1}{\alpha} \ln \left(1 + \frac{\alpha(\delta - \epsilon)\lambda(\sqrt{N})}{\|\sqrt{M}A_{cl}\| + W\|\sqrt{M}B_2\| + \alpha\epsilon\bar{\lambda}(\sqrt{N})} \right) \\ &= \zeta_1 > 0. \end{aligned}$$

It is easy to show that

$$\begin{aligned} &\xi(x(r_k); \epsilon, \delta) \\ &\geq \frac{1}{\alpha} \ln \left(1 + \frac{\epsilon\alpha(1 - \delta)\lambda(\sqrt{N})}{\|\sqrt{M}A_{cl}\| + W\|\sqrt{M}B_2\| + \delta\alpha\bar{\lambda}(\sqrt{N})} \right) \\ &= \zeta_2 > 0 \end{aligned}$$

holds. \blacksquare

VII. SIMULATION

This section presents the results of simulation studies that empirically compare the performance of self-triggered controllers against periodically triggered and event-triggered controllers. This section's main finding is that self-triggered systems appear to generate longer sampling periods than the bound on MATI presented in [9]. The simulation results suggest that periodically-triggered systems with a sampling period T have worse disturbance rejection abilities (as measured by energy in the tracking error) than self-triggered systems whose average sampling period is equal to T . Finally we provide examples illustrating that the self-triggering's computational cost (as measured by the ratio of the task's execution time over period) is comparable and sometimes better than the computational cost of periodically triggered systems using the the bound on MATI in [9].

The remainder of this section is organized as follows. Section VII-A describes the system under study. Simulations of the system's self-triggered controller will be found in Section VII-B. The performance of the self-triggered system is then compared against comparable event-triggered schemes (Section VII-C) and periodically-triggered schemes (Section VII-D). A discussion of self-triggering's computational cost is found in Section VII-E.

A. System Model

The following simulation results were generated for event-triggered and self-triggered feedback systems. The plant was an inverted pendulum on top of a moving cart. The plant's linearized state equations were

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g}{\ell} & 0 \end{bmatrix} x(t) \\ &+ \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{(M\ell)} \end{bmatrix} u(t) + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} w(t) \end{aligned}$$

where M was the cart mass, m was the mass of the pendulum bob, ℓ was the length of the pendulum arm, and g was gravitational acceleration. For these simulations, we let $m = 1$, $M = 10$, $\ell = 3$, and $g = 10$. The system state was the vector $x = [y \ \dot{y} \ \theta \ \dot{\theta}]^T$ where y was the cart's position and θ was the pendulum bob's angle with respect to the vertical. The control input $u(t)$ was generated by either an event-triggered or

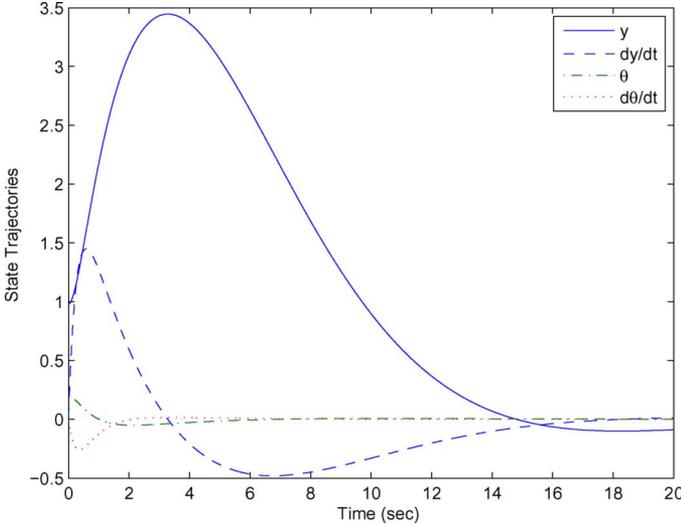


Fig. 3. State trajectories of continuous-time closed-loop system ((1)).

self-triggered controller. The function w was an external disturbance to the system. The system's initial state was the vector $x_0 = [0.98 \ 0 \ 0.2 \ 0]^T$.

We designed a continuous-time state feedback control system ((1)) in which the performance level, γ , was set to 200. Solving the Riccati equation in (2) yielded a positive definite matrix P such that the state-feedback gains were

$$B_1^T P = [-2 \quad -12 \quad -378 \quad -210]. \quad (55)$$

The state trajectory of the resulting closed-loop system is denoted below as x_c . Fig. 3 plots the system states as a function of time under the assumption that $w(t) = 0$ for all t . Fig. 3 is therefore the impulse response of the inverted pendulum system.

B. Self-Triggered Feedback

The simulations in this subsection are for the self-triggered feedback scheme associated with (42) and (43) in theorem 6.10 with $\beta = 0.5$. In this case, the task release times were generated at time f_k using the equation

$$r_{k+1} = f_k + L_2(x(r_k), x(r_{k-1}), D_k, \delta)$$

and the finishing times were assumed to satisfy

$$f_{k+1} = r_{k+1} + \xi(x(r_k); \epsilon, \delta)$$

which means the delays are equal to the deadlines. The plant is the inverted pendulum plant of the preceding subsection in which the external disturbance $w(t)$ was again zero. The ϵ and δ parameters were chosen to be 0.65 and 0.7, respectively.

In comparing the performance of the self-triggered versus the continuous-time system, we examine the “normalized” error. Let x_s denote the self-triggered system's response and let x_c denote the continuous-time system's response. The normalized self-triggered system's error, $E(t; x_s)$ is defined by

$$E(t; x_s) = \frac{|\sqrt{V(x_s(t))} - \sqrt{V(x_c(t))}|}{\sqrt{V(x_c(t))}} \quad (56)$$

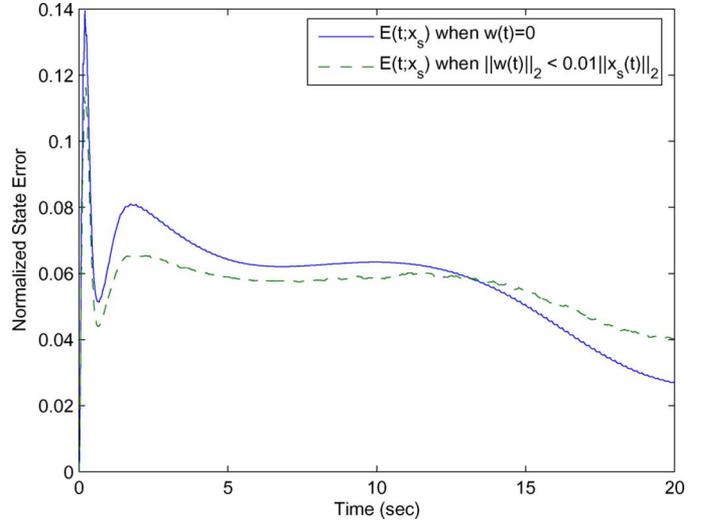


Fig. 4. Normalized state error versus time for a self-triggered systems with $w(t) = 0$ and a self-triggered system with $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$ ($\delta = 0.7$, $\epsilon = 0.65$).

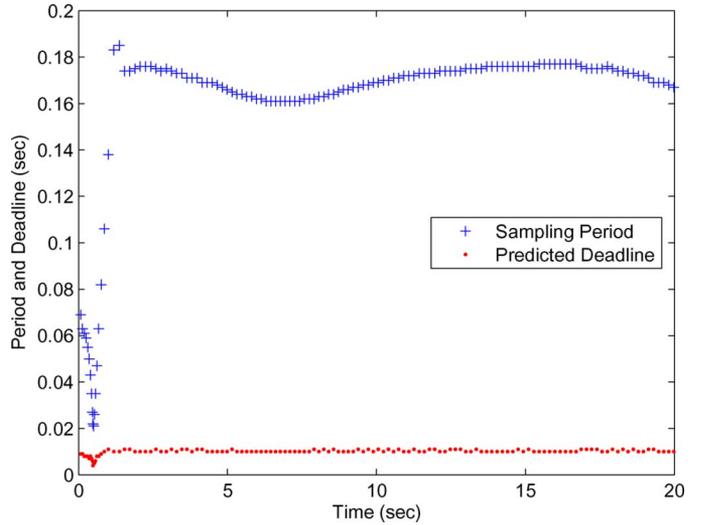


Fig. 5. Sampling period and predicted deadline for a self-triggered system in which $\delta = 0.7$ and $\epsilon = 0.65$.

where $V(x) = x^T P x$ and P is the positive definite matrix satisfying the algebraic Riccati (2). This normalization of the state error allows us to fairly measure those states (i.e., the pendulum bob angle) that are most directly affected when input disturbances exist.

Fig. 4 plots the normalized error, $E(t; x_s)$ of the self-triggered system assuming $w(t) = 0$ and $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$. For both cases, the normalized error is small over time, thereby suggesting that the continuous-time and self-triggered systems have nearly identical impulse responses.

Fig. 5 plots the task periods, T_k , (crosses) and deadlines, ξ , (dots) generated by the self-triggered scheme assuming $w(t) = 0$. The sampling periods range between 0.021 and 0.185 seconds. Note that these sampling periods show significant variability. The shortest and most aggressive sampling periods occurred in response to the system's non-zero initial condition. Longer and relatively constant sampling periods were generated

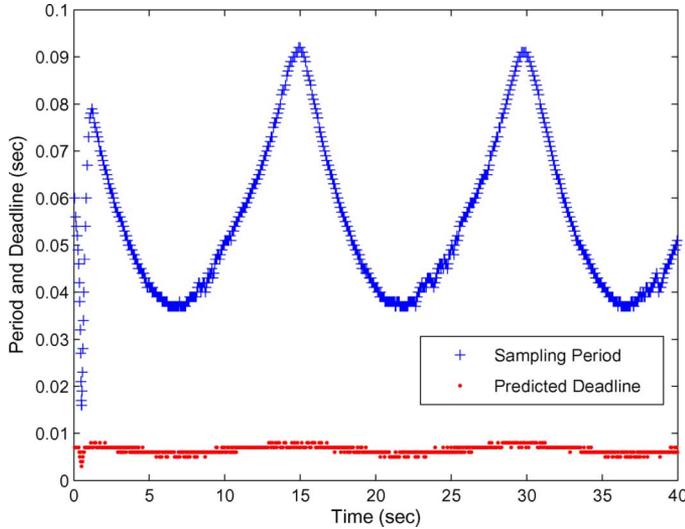


Fig. 6. Sample periods and predicted deadlines versus time for a self-triggered system ($\delta = 0.7$, $\epsilon = 0.65$, and $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$).

once the system state has returned to the neighborhood of the system's equilibrium point. This seems to confirm the conjecture that self-triggering can effectively adjust sampling periods in response to changes in the control system's external inputs.

Fig. 6 plots the sample periods, T_k (crosses), and predicted deadlines (dots), generated by the self-triggered system when it is driven by the disturbance w where $\|w(t)\| \leq 0.01\|x_s(t)\|$. After the initial transient in the system's response, the sampling periods converge to a periodic signal in which the sample periods range between 0.037 and 0.092. It is interesting to note that T_k shows significant periodic variation. Other simulations have shown similar results. These observations suggest that the choice of "optimal" sampling period has its own dynamic that leads to a periodic variation in sampling periods. One interesting issue for future research is whether or not we can take advantage of this variability in scheduling multiple real-time control tasks.

Figs. 7 and 8 show what happens to task periods and deadlines when we varied δ and ϵ . In Fig. 7, $\delta = 0.7$ and ϵ was varied between 0.1, 0.4 and 0.65. The top two plots show histograms of the sampling period (left) and deadline (right) for $\epsilon = 0.65$. The middle two plots are histograms of the sampling periods and deadlines for $\epsilon = 0.4$. The bottom two plots display results when $\epsilon = 0.1$. Examining the three histograms on the left side of Fig. 7 shows little change in sampling period as a function of ϵ . The three histograms on the right side of Fig. 7 show significant variation in deadline as a function of ϵ . These results are consistent with our earlier discussion in remark 6.13. Recall that ϵ controls the time when the k th task finishes. So by changing ϵ we expect to see a large impact on the predicted deadline (ξ) and little impact on the task period.

Fig. 8 is similar to Fig. 7 except that we keep ϵ fixed at 0.1 and vary δ from 0.15 (bottom) to 0.4 (middle) to 0.9 (top). These histograms show that as we increase δ we also enlarge the task periods. Recall that δ controls the time interval $f_{k+1} - f_k$ so that what we observe in the simulation is again consistent with our comments in remark 6.12. As we increase the sampling period, however, we can expect smaller predicted deadlines because the

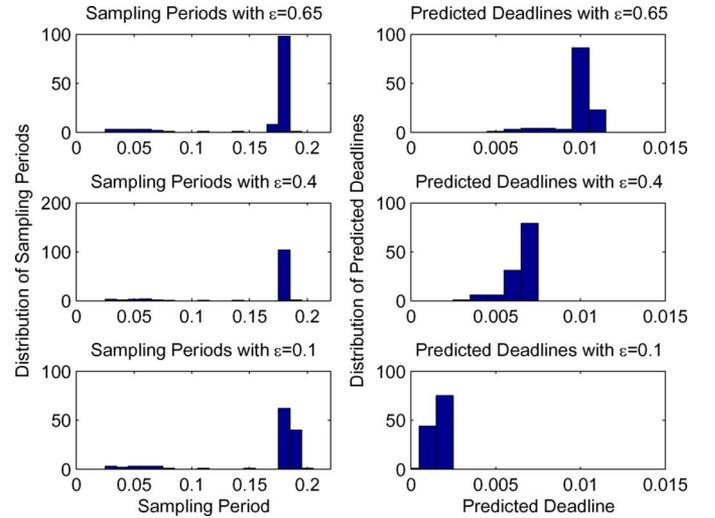


Fig. 7. Histogram of sample period and predicted deadline for a self-triggered system in which $\delta = 0.7$ and $\epsilon \in \{0.1, 0.4, 0.65\}$.

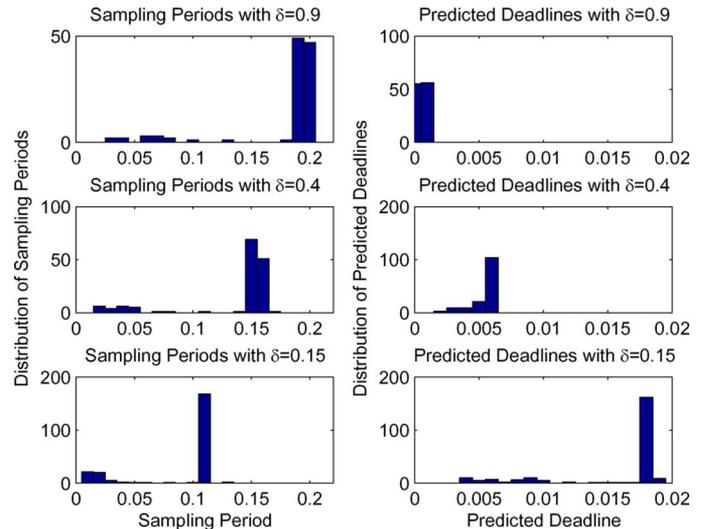


Fig. 8. Histogram of sample period and predicted deadline for a self-triggered system in which $\epsilon = 0.1$ and $\delta \in \{0.15, 0.4, 0.9\}$.

average sampling frequency is lower. This too is seen in the histograms on the righthand side of Fig. 8.

The results in this subsection clearly show that we can effectively bound the task periods and deadlines in a way that preserves the closed-loop system's \mathcal{L}_2 stability. An interesting future research topic concerns how we might use these bounds on period and deadline in a systematic manner to schedule multiple real-time control tasks.

C. Comparison Against Event-Triggered Feedback

This subsection compares the self-triggered scheme against two event-triggered schemes; our own event-triggered scheme in theorem 6.3 and the event-triggered scheme in [11]. To make a fair comparison, we set $\epsilon = 0$ and $\delta = 1$ so self-triggering occurs with zero delay.

Let $x_e(t)$ denote the state trajectory of the event-triggered system based on theorem 6.3's threshold test. Let $E(t; x_e)$ denote the normalized error (see (56)) of the event-triggered tra-

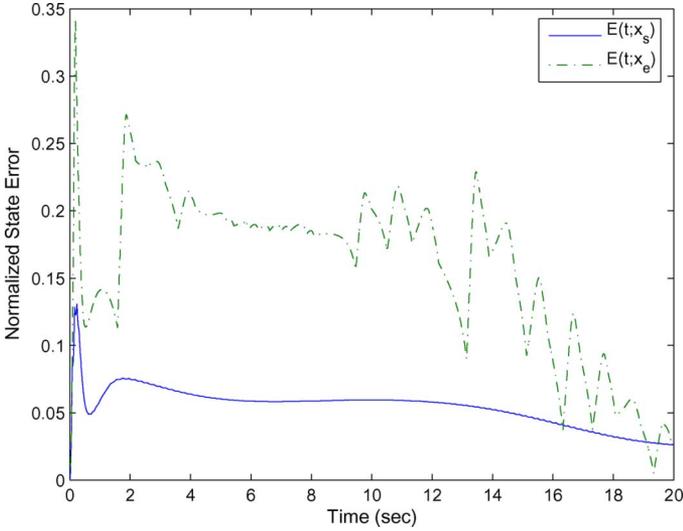


Fig. 9. Normalized state errors versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = 0$).

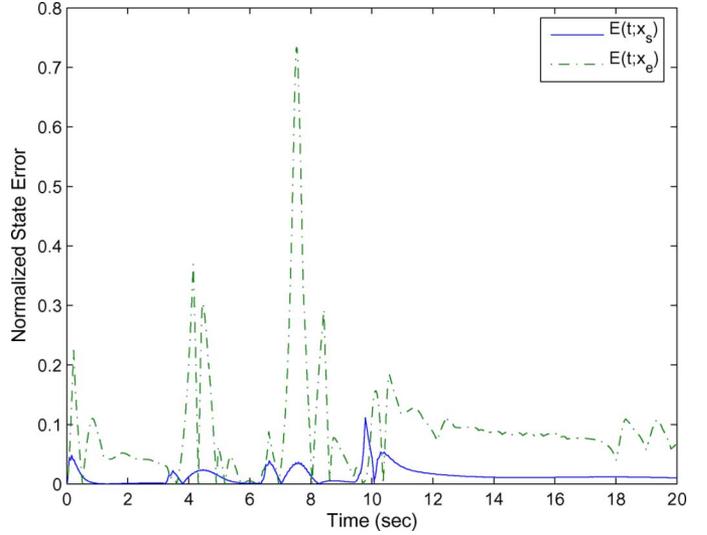


Fig. 11. Normalized error versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = \mu(t)$).

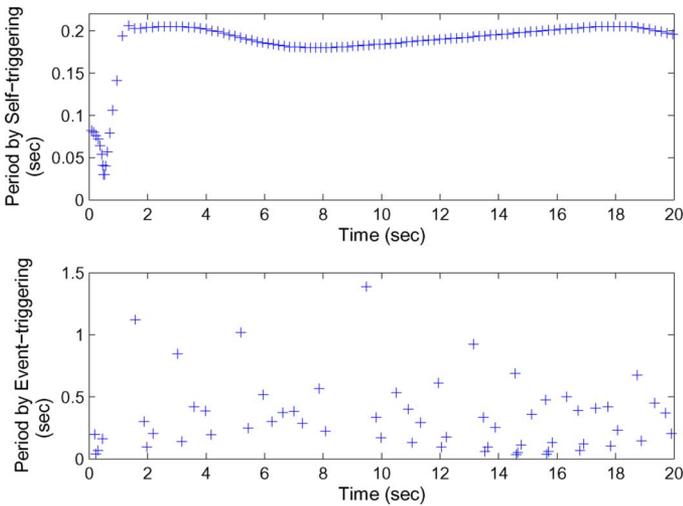


Fig. 10. Sampling period versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = 0$).

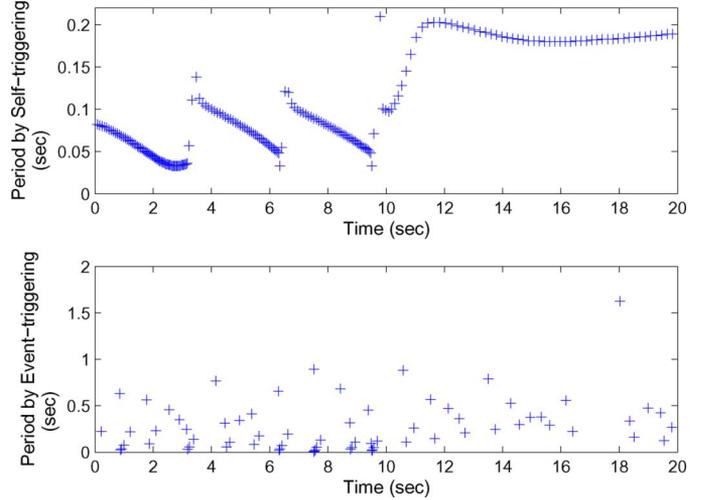


Fig. 12. Sampling period versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$ and $w(t) = \mu(t)$).

jectory. Fig. 9 plots the normalized error for the self-triggered system, $E(t; x_s)$ (solid line), and the event-triggered system, $E(t; x_e)$ (dashed line) as functions of time for $w(t) = 0$. In both cases the normalized errors are small, though the event-triggered system has a slightly larger error.

Fig. 10 plots the sampling periods generated by the self-triggered scheme (top plot) and the event-triggered scheme (bottom plot). The self-triggered sampling periods range between 0.0300 and 0.2060 with an average period of 0.1782. The event-triggered sampling periods range between 0.0340 and 1.3890 with an average period of 0.3375. Note that the self-triggered sampling periods are an order of magnitude smaller than the periods of the event-triggered scheme. These results suggest that even-triggered feedback is better able to reduce sampling period frequency than the self-triggered feedback.

We then added a square wave input to the system to see how the self-triggered and event-triggered systems react to external

disturbances. The results from this comparison are shown in Fig. 11. This figure plots the time history of the normalized error signals, $E(t; x_s)$ (solid line) and $E(t; x_e)$ (dashed line), for the inverted pendulum using the input signal, $w(t) = \mu(t)$ where $\mu : \mathbb{R} \rightarrow \mathbb{R}$ takes the values

$$\mu(t) = \begin{cases} \text{sgn}(\sin t) & \text{if } 0 \leq t < 10 \\ 0 & \text{otherwise} \end{cases} . \quad (57)$$

Again, the error in the self-triggered system is smaller than that in the event-triggered system.

Fig. 12 plots the sampling periods generated by the self-triggered (top plot) and event-triggered (bottom plot) systems when $w(t) = \mu(t)$. The top plot shows that the sampling periods in the self-triggered system readjust and get smaller when the square wave input hits the system over the time interval $[0, 10]$. In the event-triggered system, as shown in the bottom plot, the average period, 0.2830, also get smaller compared with the periods in

the bottom plot of Fig. 10, although the decrease is not very obvious. These results again demonstrate the ability of self-triggering and event-triggering to successfully adapt to changes in the system's input disturbances.

It is instructive to compare the sampling periods generated by self-triggering (see the top plot in Fig. 10) against the periods that would have been generated by the event-triggering scheme in [11]. The event-triggering scheme in [11] samples the state when

$$e_k^T(t)G e_k(t) = p^2 x^T(t)G x(t).$$

G is a positive definite matrix associated with a control Lyapunov function $V(x) = x^T G x$ for the closed-loop system with state feedback gains K . Since V is a control Lyapunov function, we can find a matrix H such that the directional derivative of the unforced closed-loop system satisfies the inequality $\dot{V} \leq -x^T H x$. In the above equation, p is the real constant

$$p = \frac{\lambda_m(G)}{2\lambda_M(G)} \frac{\lambda_m(H)}{\|GBK\|}$$

where $\lambda_m(G)$ and $\lambda_M(G)$ denote the minimum and maximum eigenvalues of matrix G , respectively. For this particular simulation, we set G equal to the P associated with our controller to obtain $p = 4.17 \times 10^{-11}$. This event-triggering threshold generates sampling periods less than 10^{-5} . This is much smaller than the sampling periods generated by the self-triggering scheme.

The reason for this difference is that the condition number of the particular G matrix is extremely large due to the great difference in the time constants associated with the dynamics of the cart and pendulum bob. Such a matrix leads to a very small p , which limits the size of the sampling periods generated by the approach in [11]. In fact, for the inverted pendulum model with the control gain K given by (55), the smallest condition number of G matrix is 409.05. The resulting p is equal to 1.35×10^{-11} . If we directly consider the value of p , the largest p we can get is 1.20×10^{-7} . The resulting sampling periods are still less than 10^{-5} . Therefore, for the inverted pendulum model, our event-triggering threshold generates much longer sampling periods.

However, for different systems which allow G with a small condition number, the approach in [11] may also generate large sampling periods. For example, for a scalar system, $\dot{x} = -x + u + w$, with $\gamma = 1/\sqrt{2}$, we get $P = 1$ and $K = -1$. The average sampling period generated by self-triggering is 0.3670 associated with $\beta = 0.5$, $\delta = 1$, $\epsilon = 0$ and $w(t) = 0$. The threshold condition in [11] is $e_k^2(t) < 4x(t)^2$ for the same P and K . The minimal, average, maximal sampling periods by the approach in [11] are the same, 0.4060, which are longer than the average period generated by our approach.

D. Comparison Against Periodically-Triggered Feedback

The simulations in this subsection compare the performance of self-triggered and "comparable" periodically triggered feedback control systems using the inverted pendulum system described above. Again to make a fair comparison, we enforce zero delays by setting $\delta = 1$ and $\epsilon = 0$ in the self-triggered controller.

We first compare the sampling period in the self-triggered system with the bound on MATI given by [9]. The bound on MATI ensuring an \mathcal{L}_2 gain of γ is,

$$T_{\text{MATI}} = \frac{1}{L} \ln \frac{L + \gamma_1}{\bar{\rho}L + \gamma_1} \quad (58)$$

where, in the inverted pendulum model, $\bar{\rho} = 0$, $L = \max(0.5\lambda_{\max}(-B_1K - K^T B_1^T), 0)$, $\gamma_1 \geq 0$ satisfies

$$\gamma = \frac{1 + \max\{\gamma_1, \gamma_2\}}{1 - \gamma_1\gamma_2}$$

and γ_2 is the \mathcal{L}_2 gain for the closed-loop system ($\dot{x} = A_c x + B_1 K e + B_2 w$) from (e, w) to $-A_c x$.

From (58), we compute the bound on MATI consistent with an \mathcal{L}_2 gain $\gamma = 400$. This results in $T_{\text{MATI}} = 0.0092$. The corresponding average sampling period for a self-triggered system with gain $\gamma = 400$ is equal to 0.1782 (see Fig. 10). Clearly the average period generated by the self-triggered scheme is longer than the estimate of MATI for systems with the same induced \mathcal{L}_2 gain. Note that the bound on MATI obtained assuming an infinite-gain for γ is still only 0.0112 which is still much smaller than the average sampling period generated by the self-triggered controller.

Note that the above self-triggered system generated sampling periods under the assumption that the noise magnitude $W = 0$. For non-zero W the average sampling period will decrease. For instance if $W = 0.01$, then the average self-triggered period shrinks to 0.0629. Though this is still larger than the bound on MATI, it is apparent that as W increases, the average period will continue shrinking until it is less than the MATI. This appears to be one weakness of the current result in theorem 6.10. We believe this can be relaxed, but that will need to be addressed in future work.

One thing worth mentioning is that the self-triggering scheme is compared with a theoretically derived bound on the MATI [9]. This bound may be conservative due to the sampling scheme and the conservatism of the proof techniques. It does not mean that the actual maximal allowable transfer interval is conservative.

We should also notice that the bound on MATI can be predicted before the system is deployed by methods such as the one in [9]. So when designers try to build physical devices, they know exactly what the requirements are on the sampling rate. On the contrary, it is difficult to predict ahead of time the minimal sampling period in self-triggered feedback systems. It may be possible that for a short interval, the controller insists on a sampling/control rate that the physical device cannot provide. Therefore, how to handle such unexpected delays would be an interesting direction to follow in the future.

We then compared the performance of the self-triggered system and a periodically triggered system with "comparable" task period, which is the average sampling period over the entire time zone, 0.1782, generated by the example in Fig. 10. The results from this comparison are shown in Fig. 13. This figure plots the time history of the normalized errors for the self-triggered system, $E(t; x_s)$ (solid line), and the periodically triggered system, $E(t; x_T)$ (dash-dot line) for an inverted pendulum with input signal $w(t) = \mu(t)$ where μ is defined in (57) and x_T denotes the periodically triggered system's response.

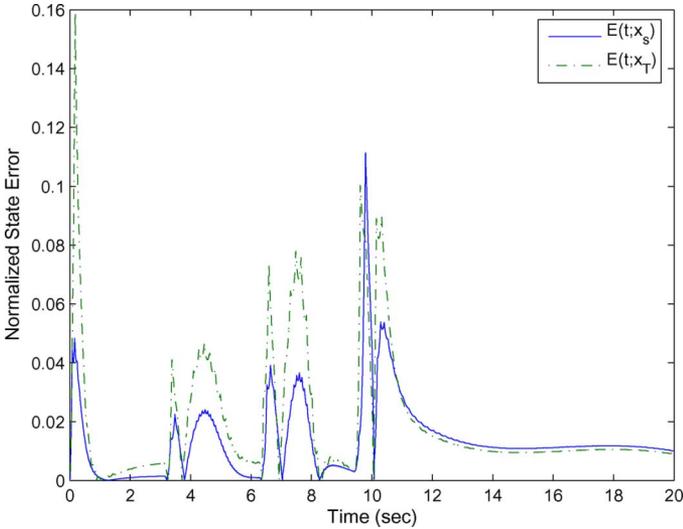


Fig. 13. Normalized error versus time for a self-triggered system ($\delta = 1$ and $\epsilon = 0$) and a periodically triggered system whose period was chosen from the sample periods shown in the top plot of Fig. 10.

Fig. 13 clearly shows that the self-triggered error is significantly smaller than the error of the periodically triggered system. This error is a direct result of the self-triggered system's ability to adjust its sample period as shown in the top plot of Fig. 12.

E. Self-Triggered System's Computational Cost

This subsection compares the computational cost in the self-triggered system by comparing the average utilization of the self-triggered system against a periodically triggered system with period equal to the MATI, $T_{\text{MATI}} = 0.0092$. The average utilization is the quotient of the execution time over the average sampling period. In this section, we set $\delta = 1$ and $\epsilon = 0$ with $w(t) = 0$ for the self triggered system. In this case the average period is 0.1782.

The computational cost for one task will be measured by the number of multiplies required for a single update of the control. We focus on "multiplies" since they represent the most expensive floating point computation. Since other parameters (such as α and \sqrt{M}) can be computed off-line, the computational cost for one task comes from the computation of the control, which uses n multiplies, and the prediction of the next release time, which requires about $2n^2 + 2n$ multiplies, where n is the state dimension. The total computational cost of the self-triggered scheme is therefore $2n^2 + 3n$, whereas the computational cost of the periodically triggered controller is only n multiplies.

While the computational cost of self-triggering is higher than that of periodically-triggered systems, we generally see that self-triggered systems have longer average periods. So a more appropriate comparison of each method's resource usage is provided by their "utilization" which we take here as the quotient of the computational cost (number of multiplies) over the sampling period. For the inverted pendulum system the

periodically triggered system's utilization, U_T , may therefore be taken to be n/T_{MATI} where we normalized out the cost of the control computation. For this example (with $n = 4$) we earlier computed the MATI to be $T_{\text{MATI}} = 0.0092$ so that the periodic system's utilization is $U_T = 434.7828$. The self-triggered system's utilization, U_s , is given by $(2n^2 + 3n)/\bar{T}$ where \bar{T} is the average period generated by self-triggering. In our simulations $\bar{T} = 0.1782$, so that the self-triggered system's utilization becomes $U_s = 246.9136$. The main conclusion to be drawn here is that even though periodically-triggered feedback has lower computational cost per job, the average utilization of both methods appears to be comparable.

VIII. CONCLUSION

This paper has presented a state-dependent threshold inequality whose satisfaction assures the induced \mathcal{L}_2 gain of a sampled-data linear state feedback control system. We derive state-dependent bounds on the task periods and deadlines enforcing this threshold inequality based on an event-triggered feedback scheme. These results were used to present a self-triggered feedback scheme with guaranteed \mathcal{L}_2 stability. Simulation results show that the proposed event- and self-triggered feedback schemes perform better than comparable periodically triggered feedback controllers. The results in this paper, therefore, appear to provide a solid analytical basis for the development of aperiodic sampled-data control systems that adjust their periods and deadlines to variations in the system's external inputs.

There are a number of open directions for future study. The bounds derived in this paper can be thought of as quality-of-control (QoC) constraints that a real-time scheduler must enforce to assure the application's (i.e., control system's) performance level. This may be beneficial in the development of soft real-time systems for controlling multiple plants. The bounds on task period and deadline suggest that real-time engineers can adjust both task period and task deadline to assure task set schedulability while meeting application performance requirements. It would be interesting to see whether such bounds can be used in generalizations of elastic scheduling algorithms [23], [24]. This might allow us to finally build soft real-time systems providing guarantees on application performance that have traditionally been found only in hard real-time control systems.

To our best knowledge, this is the first rigorous examination of what might be required to implement self-triggered feedback control systems. Self-triggering on single processor systems may not be very useful since event-triggers can often be implemented in an inexpensive manner using FPGAs or custom ASICs. If, however, we are controlling multiple plants over a wireless network, then the inability of such networks to provide deterministic guarantees on message delivery make the use of self-triggered feedback much more attractive. An interesting future research direction would explore the use of self-triggered feedback over wireless sensor-actuator networks, which is partially addressed in [9], [10], [25], and [26].

APPENDIX

Proof of Lemma 6.6: Let $\Phi = \{t \in [r_k, f_k] \mid \|z_k(t)\|_2 = 0\}$. For $t \in [r_k, f_k] \setminus \Phi$, the derivative of $\|z_k(t)\|_2$ satisfies the differential inequality,

$$\begin{aligned} & \frac{d}{dt} \|z_k(t)\|_2 \\ & \leq \|\dot{z}_k(t)\|_2 = \left\| \sqrt{M} \dot{e}_k(t) \right\|_2 = \left\| \sqrt{M} \dot{x}(t) \right\|_2 \\ & = \left\| \sqrt{M} (Ax(t) - B_1 B_1^T P x(r_{k-1}) + B_2 w(t)) \right\|_2 \\ & = \left\| \sqrt{M} A e_k(t) + \sqrt{M} A x(r_k) \right. \\ & \quad \left. - \sqrt{M} B_1 B_1^T P x(r_{k-1}) + \sqrt{M} B_2 w(t) \right\|_2 \\ & \leq \left(\left\| \sqrt{M} A \sqrt{M}^{-1} \right\| + W \left\| \sqrt{M} B_2 \right\| \left\| \sqrt{M}^{-1} \right\| \right) \|z_k(t)\|_2 \\ & \quad + \left\| \sqrt{M} (A x(r_k) - B_1 B_1^T P x(r_{k-1})) \right\|_2 \\ & \quad + W \left\| \sqrt{M} B_2 \right\| \|x(r_k)\|_2 \\ & = \alpha \|z_k(t)\|_2 + \mu_1(x(r_k), x(r_{k-1})), \end{aligned}$$

where we use the righthand sided derivative when $t = r_k$. The differential inequality in (59) along with the initial condition $z_k(r_k) = 0$, allows us to conclude that

$$\|z_k(t)\|_2 \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \quad (59)$$

for all $t \in [r_k, f_k]$ since $\|z_k(t)\|_2 = 0$ for all $t \in \Phi$.

The assumption in (32) can be rewritten as

$$\phi(x(r_k), x(r_{k-1}); D_k) \leq \epsilon \rho(x(r_k)) \quad (60)$$

$\phi(x(r_k), x(r_{k-1}); t - r_k)$ is a monotone increasing function of $t - r_k$. Combining this fact with (59) and (60) yields

$$\begin{aligned} \|z_k(t)\|_2 & \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \\ & \leq \phi(x(r_k), x(r_{k-1}); D_k) \leq \epsilon \rho(x(r_k)) \end{aligned}$$

which leads to (33) holding for all $t \in [r_k, f_k]$. ■

Proof of Lemma 6.8: The hypotheses of this lemma also satisfy the hypotheses of lemma 6.6 so we know that

$$\begin{aligned} \|z_k(f_k)\|_2 & \leq \phi(x(r_k), x(r_{k-1}); D_k) \\ & \leq \epsilon \rho(x(r_k)) \leq \eta \rho(x(r_k)). \end{aligned} \quad (61)$$

By (35) and (61), we have

$$L_2(x(r_k), x(r_{k-1}); D_k, \eta) > 0$$

which implies

$$d_\eta > f_k.$$

Assume the system state $x(t)$ satisfies the differential equation

$$\dot{x}(t) = Ax(t) - B_1 B_1^T P x(r_k) + B_2 w(t)$$

for $t \in [f_k, d_\eta]$. Using an argument similar to that in lemma 6.6, we can show that $\|z_k(t)\|_2$ satisfies the differential inequality

$$\frac{d}{dt} \|z_k(t)\|_2 \leq \alpha \|z_k(t)\|_2 + \mu_0(x(r_k)). \quad (62)$$

Equation (61) can be viewed as an initial condition on the differential inequality in (62). Solving the differential inequality, we know for all $t \in [f_k, d_\eta]$,

$$\begin{aligned} \|z_k(t)\|_2 & \leq e^{\alpha(t-f_k)} \phi(x(r_k), x(r_{k-1}); D_k) \\ & \quad + \frac{\mu_0(x(r_k))}{\alpha} (e^{\alpha(t-f_k)} - 1). \end{aligned} \quad (63)$$

Because the right side of (63) is an increasing function of t , we get

$$\begin{aligned} \|z_k(t)\|_2 & \leq e^{\alpha(d_\eta-f_k)} \phi(x(r_k), x(r_{k-1}); D_k) \\ & \quad + \frac{\mu_0(x(r_k))}{\alpha} (e^{\alpha(d_\eta-f_k)} - 1) \\ & = \eta \rho(x(r_k)) \end{aligned} \quad (64)$$

for all $t \in [f_k, d_\eta]$, where the equivalence in the right side of (64) is achieved according to the definition of d_η in (34). ■

Proof of Lemma 6.9: First note that $x(r_k) = e_{k-1}(r_k) + x(r_{k-1})$ implies that

$$- \|e_{k-1}(r_k)\|_2 \leq \|x(r_k)\|_2 - \|x(r_{k-1})\|_2 \leq \|e_{k-1}(r_k)\|_2.$$

We now use this inequality to bound $\mu_1(x(r_k), x(r_{k-1}))$ and $\rho(x(r_k))$ as a function of $x(r_{k-1})$.

An upper bound on $\mu_1(x(r_k), x(r_{k-1}))$ can be obtained by noting that

$$\begin{aligned} & \mu_1(x(r_k), x(r_{k-1})) \\ & = \left\| \sqrt{M} (A x(r_k) - B_1 B_1^T P x(r_{k-1})) \right\|_2 \\ & \quad + W \left\| \sqrt{M} B_2 \right\| \|x(r_k)\|_2 \\ & = \left\| \sqrt{M} (A_{cl} x(r_{k-1}) + A e_{k-1}(r_k)) \right\|_2 \\ & \quad + W \left\| \sqrt{M} B_2 \right\| \|x(r_{k-1}) + e_{k-1}(r_k)\|_2 \\ & \leq \left\| \sqrt{M} A_{cl} x(r_{k-1}) \right\|_2 + W \left\| \sqrt{M} B_2 \right\| \|x(r_{k-1})\|_2 \\ & \quad + \left\| \sqrt{M} A \sqrt{M}^{-1} z_{k-1}(r_k) \right\|_2 \\ & \quad + W \left\| \sqrt{M} B_2 \right\| \left\| \sqrt{M}^{-1} z_{k-1}(r_k) \right\|_2 \\ & \leq \left\| \sqrt{M} A_{cl} x(r_{k-1}) \right\|_2 + W \left\| \sqrt{M} B_2 \right\| \|x(r_{k-1})\|_2 \\ & \quad + \left(\left\| \sqrt{M} A \sqrt{M}^{-1} \right\| + W \left\| \sqrt{M} B_2 \right\| \left\| \sqrt{M}^{-1} \right\| \right) \\ & \quad \cdot \delta \rho(x(r_{k-1})) \\ & = \mu_0(x(r_{k-1})) + \alpha \delta \rho(x(r_{k-1})). \end{aligned}$$

A lower bound on $\rho(x(r_k))$ is obtained by noting that

$$\begin{aligned}\rho(x(r_k)) &= \left\| \sqrt{N}x(r_k) \right\|_2 \\ &= \left\| \sqrt{N}(e_{k-1}(r_k) + x(r_{k-1})) \right\|_2 \\ &\geq \left\| \sqrt{N}x(r_{k-1}) \right\|_2 - \left\| \sqrt{N}e_{k-1}(r_k) \right\|_2 \\ &= \left\| \sqrt{N}x(r_{k-1}) \right\|_2 - \sqrt{e_{k-1}^T(r_k)Ne_{k-1}(r_k)}.\end{aligned}$$

We know $M \geq N$ by the definitions of M and N in (14) and (15), respectively. So the inequality above can be further reduced as

$$\begin{aligned}\rho(x(r_k)) &\geq \left\| \sqrt{N}x(r_{k-1}) \right\|_2 - \sqrt{e_{k-1}^T(r_k)Me_{k-1}(r_k)} \\ &\geq \rho(x(r_{k-1})) - \delta\rho(x(r_{k-1})) \\ &= (1 - \delta)\rho(x(r_{k-1})).\end{aligned}$$

Putting both inequalities together we see that

$$\begin{aligned}L_1(x(r_k), x(r_{k-1}); \epsilon) &= \frac{1}{\alpha} \ln \left(1 + \epsilon\alpha \frac{\rho(x(r_k))}{\mu_1(x(r_k), x(r_{k-1}))} \right) \\ &\geq \frac{1}{\alpha} \ln \left(1 + \epsilon\alpha \frac{(1 - \delta)\rho(x(r_{k-1}))}{\alpha\delta\rho(x(r_{k-1})) + \mu_0(x(r_{k-1}))} \right) \\ &\equiv \xi(x(r_{k-1}); \epsilon, \delta) > 0,\end{aligned}$$

which completes the proof. ■

REFERENCES

- [1] M. Lemmon, T. Chantem, X. Hu, and M. Zyskowski, "On self-triggered full information h-infinity controllers," in *Hybrid Systems: Computation and Control*. Berlin, Germany: Springer, 2007.
- [2] R. Brockett, "Stabilization of motor networks," in *Proc. 34th IEEE Conf. Decision Control*, 1995, pp. 1484–1488.
- [3] R. Brockett, "Minimum attention control," in *Proc. 36th IEEE Conf. Decision Control*, 1997, pp. 2628–2632.
- [4] M. Velasco, P. Marti, and J. Fuertes, "The self triggered task model for real-time control systems," in *Proc. Work-in-Progress Session 24th IEEE Real-Time Syst. Symp. (RTSS'03)*, 2003, pp. 1–4.
- [5] K. Astrom and B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1997.
- [6] Y. Zheng, D. Owens, and S. Billings, "Fast sampling and stability of nonlinear sampled-data systems: Part 2. Sampling rate estimations," *IMA J. Math. Control Inform.*, vol. 7, pp. 13–33, 1990.
- [7] D. Nescic, A. Teel, and E. Sontag, "Formulas relating $\mathcal{K}\mathcal{L}$ stability estimates of discrete-time and sampled-data nonlinear systems," *Syst. Control Lett.*, vol. 38, pp. 49–60, 1999.
- [8] L. Zaccarian, A. Teel, and D. Nescic, "On finite gain \mathcal{L}_p stability of nonlinear sampled-data systems," *Syst. Control Lett.*, vol. 49, pp. 201–212, 2003.
- [9] D. Nescic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Trans. Automat. Control*, vol. 49, no. 10, pp. 1650–1667, Oct. 2004.
- [10] D. Carnevale, A. R. Teel, and D. Nescic, "Further results on stability of networked control systems: A Lyapunov approach," *IEEE Trans. Automat. Control*, vol. 52, no. 5, pp. 892–897, May 2007.
- [11] P. Tabuada and X. Wang, "Preliminary results on state-triggered scheduling of stabilizing control tasks," in *Proc. 45th IEEE Conf. Decision Control*, Dec. 2006, pp. 282–287.
- [12] D. Seto, J. Lehoczky, L. Sha, and K. Shin, "On task schedulability in real-time control systems," in *Proc. IEEE Real-Time Technol. Appl. Symp. (RTAS)*, 1996, pp. 13–21.
- [13] A. Cervin, J. Eker, B. Bernhardtsson, and K.-E. Arzen, "Feedback-feed-forward scheduling of control tasks," *Real-Time Syst.*, vol. 23, no. 1-2, pp. 25–53, 2002.
- [14] P. Marti, C. Lin, S. Brandt, M. Velasco, and J. Fuertes, "Optimal state feedback resource allocation for resource-constrained control tasks," in *Proc. IEEE Real-Time Syst. Symp. (RTSS'04)*, 2004, pp. 161–172.
- [15] B. Bamieh, "Intersample and finite wordlength effects in sampled-data problems," *IEEE Trans. Automat. Control*, vol. 48, no. 4, pp. 639–643, Apr. 2003.
- [16] L. Palopoli, C. Pinello, A. Bicchi, and A. Sangiovanni-Vincentelli, "Maximizing the stability radius of a set of systems under real-time scheduling constraints," *IEEE Trans. Automat. Control*, vol. 50, no. 11, pp. 1790–1795, Nov. 2005.
- [17] C. Liu and J. Layland, "Scheduling for multiprogramming in a hard-real-time environment," *J. Assoc. Comput. Mach.*, vol. 20, no. 1, pp. 46–61, 1973.
- [18] K. Arzen, "A simple event-based PID controller," in *Proc. 14th IFAC World Congress*, 1999, vol. 18, pp. 423–428.
- [19] D. Hristu-Varvakelis and P. Kumar, "Interrupt-based feedback control over a shared communication medium," in *Proc. IEEE Conf. Decision Control*, Dec. 2002, pp. 3223–3228.
- [20] K. Astrom and B. Bernhardtsson, "Comparison of Riemann and Lebesgue sampling for first order stochastic systems," in *Proc. IEEE Conf. Decision Control*, Dec. 2002, pp. 2011–2016.
- [21] T. Basar and P. Bernhard, *H_∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Boston, MA: Birkhauser, 1995.
- [22] T. Chantem, X. Hu, and M. Lemmon, "Period and deadline selection problem for real-time systems," in *Proc. Real Time Syst. Symp. (Work-in-Progress Track)*, 2007, [CD ROM].
- [23] G. Buttazzo, G. Lipari, M. Caccamo, and L. Abeni, "Elastic scheduling for flexible workload management," *IEEE Trans. Computers*, vol. 51, no. 3, pp. 289–302, Mar. 2002.
- [24] T. Chantem, X. Hu, and M. Lemmon, "Generalized elastic scheduling," in *Proc. IEEE Real Time Syst. Symp.*, Dec. 2006, pp. 236–245.
- [25] M. Tabbara, D. Nescic, and A. Teel, "Stability of wireless and wireline networked control systems," *IEEE Trans. Automat. Control*, vol. 52, no. 9, pp. 1615–1630, Sep. 2007.
- [26] X. Wang and M. Lemmon, "Decentralized event-triggering broadcast over networked systems," in *Hybrid Systems: Computation and Control*. New York: Springer-Verlag, 2008.



systems.

Xiaofeng Wang (M'04) was born in Nantong, China, in 1978. He received the B.S. and M.S. degrees in mathematics from East China Normal University, Shanghai, China, in 2000 and 2003, respectively, and is currently pursuing the Ph.D. degree in the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN.

From 2003 to 2004, he was a Software Engineer with IBM, China. His current research interests are event-triggering and self-triggering in embedded systems, networked control systems, and distributed



Michael D. Lemmon (M'81) received the B.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1979 and the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, in 1990.

He is a Professor of electrical engineering at the University of Notre Dame, Notre Dame, IN. Most recently he helped forge a consortium of academic, private and public sector partners to build one of the first metropolitan scale sensor-actuator networks (CSOnet) used in monitoring and controlling combined-sewer overflow events. His research deals with real-time networked control systems with an emphasis on understanding the impact that reduced feedback information has on overall system performance. This work has been funded by a variety of state and federal agencies that include the National Science Foundation, Army Research Office, Defense Advanced Research Project Agency, and Indiana's 21st Century Technology Fund.

Dr. Lemmon served as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS and the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY. He chaired the first IEEE working group on hybrid dynamical systems and was the Program Chair for a hybrid systems workshop in 1997.

Dr. Lemmon served as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS and the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY. He chaired the first IEEE working group on hybrid dynamical systems and was the Program Chair for a hybrid systems workshop in 1997.