Towards decentralized event-triggered implementations of centralized control laws

Manuel Mazo Jr. and Paulo Tabuada

Department of Electrical Engineering, UCLA.
{mmazo,tabuada}@ee.ucla.edu

Abstract. The recent advances in communication technologies have lead to an increasing number of feedback control loops closed over networks interconnecting sensors, computation devices, and actuators. In these systems the communication network is a shared resource and event-triggered implementations of control laws offer a flexible way to reduce network utilization. In a typical event-triggered implementation, the control signals are kept constant until the violation of a condition on the plant state triggers the recomputation of the control signals. Several event-triggered controller implementations have been recently proposed in [1], [2], [3], [4], [5] and references therein. In this paper we present a decentralized event-triggered implementation for centralized nonlinear controllers. Our technique complements the results in [5], which apply only to systems composed by weakly-coupled subsystems, by also decentralizing the triggering condition at the local subsystem level.

1 Decentralized event-triggered control

We start by summarizing the basic results in [2] where a centralized event-triggering condition is proposed. We then propose a simple decentralization of this condition complemented with parameters that are used to fine tune the conservatism introduced by the decentralization. We conclude by proposing some heuristics for the on-line adjustment of these parameters.

1.1 Centralized event-triggering conditions

Let us consider a nonlinear control system:

$$\dot{x}(t) = f(x(t), u(t)) \tag{1}$$

with $x : \mathbb{R}_0^+ \to \mathbb{R}^n$ and $u : \mathbb{R}_0^+ \to \mathbb{R}^m$ and assume that a feedback control law $k : \mathbb{R}^n \to \mathbb{R}^m$ rendering $\dot{x}(t) = f(x(t), k(x(t))$ asymptotically stable has been designed. A sample-and-hold implementation of the control law $u = k(x)$ requires the input signal to be held constant in between update times, *i.e.*, $u(t) = k(x(t_k))$ for $t \in [t_k, t_{k+1}[$ where $t_1, t_2, t_3, \ldots$ is a sequence of update times at which new measurements are acquired. We now consider the signal $e : \mathbb{R}_0^+ \to \mathbb{R}^n$ defined by $e(t) = x(t_k) - x(t)$ for $t \in [t_k, t_{k+1}[$ and regard it as a measurement error. Under the assumption that $\dot{x} = f(x, u)$ is *input-to-state stable* (ISS) [6] with respect to measurement errors e, it is shown in [2] that if the update times from the event-triggered mechanism satisfy:

$$|e(t)|^2 \leq \sigma |x(t)|^2, \quad t \in [t_k, t_{k+1}[, \quad \sigma > 0 \tag{2}$$

where $|\cdot|$ is the l_2 vector norm, the sample-and-hold implementation is guaranteed to render the closed loop system asymptotically stable. Condition (2) defines an event-triggered implementation that consists of continuously checking (2) and triggering the recomputation of the control law as soon as the inequality (2) becomes an equality. Note that at the update time $t = t_k$ the error $e(t_k) = x(t_k) - x(t_k)$ is reset to zero which enforces (2).

1.2 Decentralized event-triggering conditions

We consider now scenarios where each state is measured by a different sensor, and sensors, controller, and actuators are connected through a communication network. In this setting no sensor can evaluate condition (2), since (2) requires the knowledge of the full state vector $x(t)$. Our goal is to provide a set of simple conditions that each sensor can check locally to decide when to trigger a controller update, thus triggering also the transmission of fresh measurements from sensors to the controller.

Using a set of parameters $\theta_1, \theta_2, \ldots, \theta_n \in \mathbb{R}$ such that $\sum_{i=1}^n \theta_i = 0$, we can rewrite inequality (2) as:

$$\sum_{i=1}^{n} \left(e_i^2(t) - \sigma x_i^2(t) \right) \leq 0 = \sum_{i=1}^{n} \theta_i.$$

Hence, the following implication holds:

$$\bigwedge_{i=1}^{n} \left(e_i^2(t) - \sigma x_i^2(t) \leq \theta_i \right) \Rightarrow |e|^2 \leq \sigma |x|^2, \tag{3}$$

which suggests the use of:

$$e_i^2(t) - \sigma x_i^2(t) \leq \theta_i \tag{4}$$

as the local event-triggering conditions. In this decentralized scheme, whenever any of the local conditions (4) becomes an equality, the controller is recomputed. Not having an equivalence in (3) entails that this decentralization approach is in general conservative: meaning shorter times between updates than in the centralized case. The parameter vector $\theta = [\theta_1\, \theta_2\, \ldots \theta_n]^T$ can be used to reduce the mentioned conservatism and thus reduce utilization of the communication network. It is important to note that the vector θ can be changed every time the control input is updated, which can help to reduce the conservatism introduced by the decentralization of the triggering conditions. From here on we show explicitly this time dependence of θ by writing $\theta(k)$ to denote its value between the update instants t_k and t_{k+1}. Following the approach presented, as long as θ satisfies $\sum_{i=1}^n \theta_i = 0$, the stability of the closed-loop is guaranteed regardless of the specific value that θ takes.

1.3 Decentralized event-triggering with on-line adaptation

We present now a family of heuristics to adjust the vector θ whenever the control input is updated. We define the *decision gap* at sensor i at time $t \in [t_k, t_{k+1}[$ as:

$$G_i(t) = e_i^2(t) - \sigma x_i^2(t) - \theta_i(k).$$

The heuristic aims at equalizing all the decision gaps at some future time. We propose a family of heuristics parametrized by an *equalization time* t_e and an *approximation order* q.

For the *equalization time* $t_e : \mathbb{N} \to \mathbb{R}^+$ we present the following two choices: constant and equal to the minimum time[1] between controller updates $t_e = \tau_{min}$; the previous time between updates $t_e(k) = t_k - t_{k-1}$.

The *approximation order* is the order of the Taylor expansion used to estimate the decision gap at the equalization time t_e:

$$\hat{G}_i(t_k + t_e) = \hat{e}_i^2(t_k + t_e) - \sigma \hat{x}_i^2(t_k + t_e) - \theta_i(k).$$

where for $t \in [t_k, t_{k+1}[$:

$$\begin{aligned} \hat{x}_i(t) &= x_i(t_k) + \dot{x}_i(t_k)(t - t_k) + \frac{1}{2}\ddot{x}_i(t_k)(t - t_k)^2 + \ldots + \frac{1}{q!}x_i^{(q)}(t_k)(t - t_k)^q, \\ \hat{e}_i(t) &= \quad 0 \quad - \dot{x}_i(t_k)(t - t_k) - \frac{1}{2}\ddot{x}_i(t_k)(t - t_k)^2 - \ldots - \frac{1}{q!}x_i^{(q)}(t_k)(t - t_k)^q, \end{aligned}$$

using the fact that $\dot{e} = -\dot{x}$ and $e(t_k) = 0$.

Finally, once an *equalization time* t_e and an *approximation order* q are chosen, the update vector $\theta(k) \in \mathbb{R}^n$ is computed so as to satisfy:

$$\forall i, j \in \{1, 2, \ldots, n\}, \quad \hat{G}_i(t_k + t_e) = \hat{G}_j(t_k + t_e), \quad \sum_{i=1}^{n} \theta_i(k) = 0.$$

Note that finding such θ, after $\hat{x}$ and $\hat{e}$ have been computed, amounts to solving a system of n linear equations.

2 Example

We illustrate the proposed decentralized event-triggered control method on a single input linear control system of dimension 7 modeling the distance between a system of 3 wagons and a head train as described in [7]. The obtained results are shown in Figure 1 where the ratio $\frac{|e|}{|x|}$ and the times between updates $t_k - t_{k-1}$ can be found. From left to right we show the results for an implementation where $\theta(k) = 0$, $\forall\, k \in \mathbb{N}$, for an implementation with $q = 1$ and $t_e(k) = \tau_{min}$ and for an implementation with $q = 1$ and $t_e(k) = t_k - t_{k-1}$. The thick horizontal line in the upper plots denotes the value of σ. The closer the ratio (blue line) gets to the value of σ when an update happens (ratio reset to zero), the closer the performance of the system is to the centralized implementation, and thus the longer the times between controller updates.

[1] It was proved in [2] that such a minimum time exists for the centralized condition, and that lower bounds can be explicitly computed.

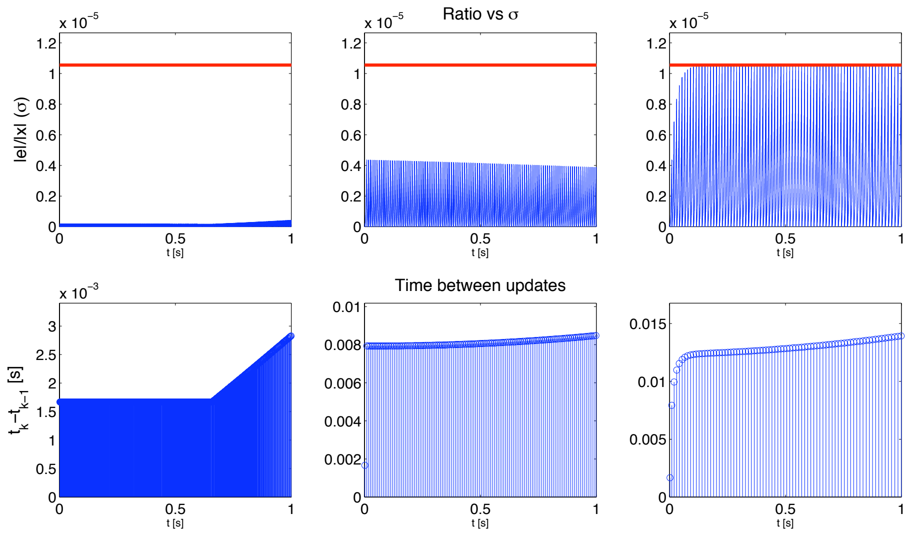

Fig. 1. Different decentralized strategies. From left to right: no adaptation; adaptation with $q = 1$ and $t_e = \tau_{min}$, and adaptation with $q = 1$ and $t_e = t_k - t_{k-1}$. From top to bottom: $\frac{|e|}{|x|}$ (blue) vs. σ (red), time between controller updates.

3 Conclusions

We have presented a decentralized event-triggered implementation of centralized control laws. This implementation can easily be extended to more general scenarios, *e.g.* , systems with several collocated sensors, and combined with previous work of Wang and Lemmon [5] for weakly decoupled subsystems, to update inputs independently of each other.

References

1. Åström, K., Bernhardsson, B.: Comparison of Riemann and Lebesgue sampling for first order stochastic systems. Proceedings of the 41st IEEE Conference on Decision and Control **2** (2002) 2011–2016
2. Tabuada, P.: Event-triggered real-time scheduling of stabilizing control tasks. IEEE Transactions on Automatic Control **52(9)** (2007) 1680–1685
3. Mazo Jr., M., Tabuada, P.: On event-triggered and self-triggered control over sensor/actuator networks. Proceedings of the 47th IEEE Conference on Decision and Control (2008) 435–440
4. Heemels, W., Sandee, J., van den Bosch, P.: Analysis of event-driven controllers for linear systems. Int. J. of Control (2008) 81(4), 571–590
5. Wang, X., Lemmon, M.: Event-Triggering in Distributed Networked Systems with Data Dropouts and Delays. Hybrid Systems: computation and control (HSCC) (2009) 366–380
6. Sontag, E.: Input to state stability: Basic concepts and results. Lecture Notes in Mathematics, Springer-Verlag (Jan 2008)
7. McLane, P., Peppard, L., Sundareswaran, K.K.: Decentralized Feedback Controls for the Brakeless Operation of Multilocomotive Powered Trains. IEEE Transactions on Automatic Control (June 1976) 358–363