Optimising the Decentralisation of Periodic Event-Triggered Control

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

H.J.N. Smit

January 17, 2023

Faculty of Mechanical, Maritime and Materials Engineering (3mE)
Delft University of Technology

European Research Council
Established by the European Commission

The work in this thesis is supported by the European Research Council through the SENTIENT project, Grant No. 755953.

TU Delft Delft University of Technology

Delft Center for Systems and Control

Abstract

Event-Triggered Control (ETC) is a control method where the controller is only updated when necessary. The control inputs are kept fixed until a state-dependent event triggers their re-computation. The triggering condition is designed to guarantee the stability and desired performance of the control system. This prevents the excessive use of scarce communication and energy resources. To fully take advantage of these savings when dealing with a (physically) distributed system, it is vital to decentralise the triggering condition. A decentralised triggering condition can be checked locally at the nodes of the system using the locally available states, eliminating the need to send updated sensor measurements to a central location constantly. However, the decentralisation of the triggering condition needs to be optimised for the state of the system to fully take advantage of the possible resource savings.

This thesis provides a framework to optimally decentralise quadratic triggering conditions for Periodic Event-Triggered Control (PETC) systems in the absence and presence of bounded disturbances. In addition, a method of constructing a region-based map of such optimisations is provided, making it feasible to implement the optimisation approach on larger systems without needing more powerful hardware.

Table of Contents

List of Figures

List of Tables

List of Theorems

Proofs in Appendix

Chapter 1

Introduction

The vast majority of control systems are implemented using microprocessors. The decision of when to sample the plant and update the control inputs inherently becomes part of the design process as a result of using such digital hardware. The standard solution is to update the system periodically, measuring the state and recomputing the control inputs after a fixed period has elapsed. The analysis of the stability and performance of such periodic systems is relatively simple, as the theory and methods are well-developed. Although this approach simplifies the design process and system analysis, there are also significant drawbacks. The control inputs are continually updated even when there is no or minimal added benefit to the performance and stability of the system. This leads to inefficient use of the available computation, communication, and energy resources. Such inefficiencies can be problematic, for example, when dealing with limited bandwidth or when the power source needed for wireless communication has restrictions, like when using a battery.

Aperiodic control extends the feedback control loop to not only compute the control inputs based on the system state but also to decide when to update these control inputs based on the state of the system. Therefore, the system only updates the control inputs when necessary to guarantee stability or to achieve the desired performance. The triggering mechanism makes the decision on when to update the control inputs, which functions in either a reactive or proactive manner. An Event-Triggered Control (ETC) mechanism monitors the state of the system and reacts by triggering an update of the system when required to guarantee stability and performance (Hendricks et al., 1994; Heemels et al., 1999; Åarzén, 1999; Åström and Bernhardsson, 1999; Tabuada, 2007). On the other hand, Self-Triggered Control (STC) uses the measured state of the plant to proactively compute when the following control update should occur whenever the system is updated (Velasco et al., 2003). Such aperiodic control also occurs naturally, such as in your nervous system or when dealing with sensors that only detect changes, like an accelerometer.

The simplest method of implementing an ETC mechanism is to constantly measure the state of the system to verify that it has not met the triggering condition. This approach is referred to as Continuous Event-Triggered Control (CETC). The main drawback of continuous monitoring is that it requires dedicated hardware to run. The fact that the system may trigger at any time also leads to inefficient use of the communication resources for systems that need to communicate over a network, as bandwidth has to be available at all times.

Periodic Event-Triggered Control (PETC) was introduced by Heemels et al. (2011) to overcome the limitations of continuous monitoring and to take advantage of the possibility of reducing resource utilisation. The triggering condition is designed such that it only has to be verified periodically to guarantee the stability and performance of the system. Only needing to check the system periodically eliminates the need for special hardware and significantly reduces resource usage.

Decentralised systems where the sensors are physically distributed and are controlled from a central location are particularly demanding on the communication resources, as all the different parts of the system need to communicate over the (wireless) network frequently. Such a system can greatly benefit from using aperiodic control to substantially reduce the amount of traffic needed to guarantee the stability and performance of the system. However, standard ETC schemes use information from all sensors in the system to make a decision on the sampling time of the system. Such a triggering mechanism would, therefore, still require all sensors to send their measurements to the central controller constantly. The solution is introduced by Mazo Jr. and Tabuada (2011). It works by decentralising the triggering condition to a set of triggering conditions, one for each sensor node, depending only on locally available information.

The main downside of decentralising the triggering condition is the added conservatism introduced by having to decide without having any information on the states of the system, which are measured elsewhere. This conservatism leads to more frequent triggering of the system and, thus, more traffic over the network. However, it is possible to almost entirely eliminate this additional conservatism by intelligently adjusting the balance between different triggering conditions to adapt the range in which the local state of the system is allowed to operate before an update to the control inputs is needed.

My work aims to develop efficient methods to optimise the balance between the different local triggering conditions to achieve system performance as close as possible to the centralised version. As this optimisation needs to happen in real time, it must also be fast to allow for implementation on a wide range of different systems. To achieve this, it is vital to shift as much of the computational burden as possible from online to pre-computing the optimum offline.

The remainder of this thesis is structured as follows: chapter 2 will provide an overview of ETC methods and introduce fundamental concepts associated with their use. The goal of the thesis will be addressed in chapter 3, starting with the problem statement followed by the development of techniques to address those problems. Examples that demonstrate the developed methods are showcased in chapter 4. Finally, chapter 5 concludes this thesis with a brief discussion of the keys findings and highlights opportunities for future improvements,

1-1 Notation

Natural numbers are denoted by $\mathbb{N}$, and the natural numbers including zero by $\mathbb{N}_0 = N \cup \{0\}$. The set of positive real numbers is indicated as $\mathbb{R}^+$ and $\mathbb{R}_0^+ = \mathbb{R}^+ \cup \{0\}$ is used for the set also including zero. Given a matrix M, $M \preceq 0$ (or $M \succeq 0$) indicates that M is a negative (or positive) semi-definite matrix and $M \prec 0$ (or $M \succ 0$) denotes that M is a negative (or positive) definite matrix. The Euclidean l_2 vector norm, or l_2 induced norm for matrices, is represented as $|\cdot|$. The largest integer not greater than $x \in \mathbb{R}$ is denoted by $\lfloor x \rfloor$.

Additionally, the following definitions are used:

Definition 1.1 (Lipschitz Continuity):
A function $f : X \to Y$ is Lipschitz continuous on $S \subseteq X$ if there exists a positive real constant $L > 0$ such that the following is satisfied for each $x_1, x_2 \in S$

$$|f(x_1) - f(x_2)| \leq L|x_1 - x_2|$$

Any such L is referred to as a Lipschitz constant.

Definition 1.2 (class K and K_∞ functions (Khalil, 2002, § 4.4)):
A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class K if it is strictly increasing and $\alpha(0) = 0$. It is said to belong to class K_∞ if $a = \infty$ and $\alpha(r) \to \infty$ as $r \to \infty$.

Definition 1.3 (class KL function (Khalil, 2002, § 4.4)):
A continuous function $\beta : [0, a)x[, \infty) \to [0, \infty)$ is said to belong to class KL if, for each fixed s, the mapping $\beta(r, s)$ belongs to class K with respect to r and, for each fixed r, the mapping $\beta(r, s)$ is decreasing with respect to s and $\beta(r, s) \to 0$ as $s \to \infty$.

Definition 1.4 ($\mathcal{L}_p$-norm and $\mathcal{L}_p$ spaces(Khalil, 2002, § 5.1)):
Given a piece-wise continuous signal $x : \mathbb{R}_0^+ \to \mathbb{R}^n$ the $\mathcal{L}_p$-norm for $1 \leq p < \infty$ is defined as:

$$\|x\|_p = \left(\int_0^\infty |x(t)|^p dt \right)^{\frac{1}{2}} < \infty$$

And for $p = \infty$ the norm is defined as:

$$\|f\|_\infty = \text{ess} \sup_{t \in \mathbb{R}_0^+} |x(t)| < \infty$$

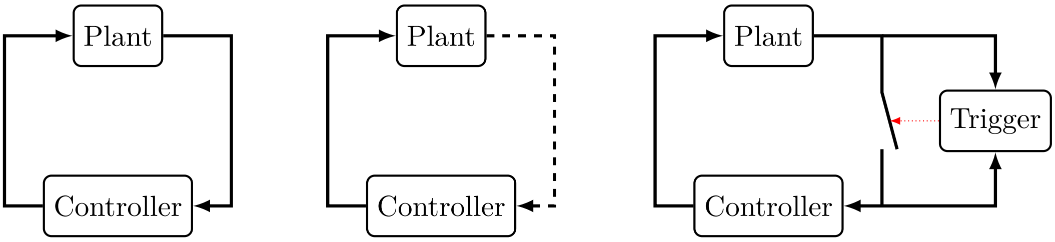
Chapter 2

Event-Triggered Control

Feedback control is generally implemented on a digital platform by periodically updating the control inputs. After a preset amount of time has elapsed, the sensor measurements of the system are used to calculate new inputs to control the system. Regardless of the state of the system, the control inputs are updated at the same frequency, also when it is not needed to keep the system stable or even when the sensor measurements have not changed at all. The chosen sampling time has to be short enough to handle the worst-case scenario for how quickly the system can change. However, a much lower sampling time would suffice most of the time. Such an approach leads to a waste of computation, communication, and energy resources. This is not a problem in many situations as these resources are not particularly limited, and other design considerations are more crucial. On the other hand, there are also many scenarios in which these resources are scarce and reducing their usage is desired. The most common situations arise either when the control system shares its computation resources, when a (wireless) communication network is used to transfer information between different parts of the system or when (part of) the system operates on battery power.

Event-Triggered Control (ETC) aims only to recompute the control actions when needed to meet the control objectives. The idea is to adjust the timing of the controller updates to the state of the system. This is done by triggering an update of the control input whenever a certain condition on the state of the system is violated. This triggering condition is constructed in such a way that it guarantees the stability of the resulting closed-loop system. The ETC mechanism creates a second feedback loop in the controller. Not only does the control input depend on the system's output, but the system's sampling time also does (Heemels et al., 2012).

Such event-based aperiodic feedback also occurs naturally in many systems, such as in our body or when dealing with sensors that do not measure a quantity but detect an event, like encoders (Åarzén, 1999).



(a) Continuous **(b)** Periodic **(c)** Event-Triggered

Figure 2-1 Comparison of different methods to manage when the feedback controller is updated with new information on the state of the system.

2-1 Control setup

To start consider a non-linear control system with states $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$, control input $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ and time $t \in \mathbb{R}_0^+$

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) \tag{2-1}$$

The states of the system are only measured at specific sampling times $\{t_i\}_{i \in \mathbb{N}}$. The resulting state measurements $\hat{\boldsymbol{x}}(t) \in \mathbb{R}^{n_x}$ are thus only updated at these sampling times and are kept at the same value in between samples, resulting in a sample-and-hold mechanism.

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_i), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N}_0 \tag{2-2}$$

The control inputs are calculated using the state measurements and are therefore only updated at sampling times and are maintained at the same level until the next sampling time.

$$\boldsymbol{u}(t) = g(\hat{\boldsymbol{x}}(t)) \tag{2-3}$$

Where $g(\boldsymbol{x})$ represents the controller. This results in the closed-loop system

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, g(\boldsymbol{x}(t_i))), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N}_0 \tag{2-4}$$

The sequence of sampling times is determined by a triggering condition $\mathcal{C}$. When $\mathcal{C}$ is negative, no measurement update is required. Whenever $\mathcal{C}$ reaches zero, the measurements are updated. The triggering condition is a function of the current state of the system and the last measured state of the system $\mathcal{C}(\boldsymbol{x}(t), \hat{\boldsymbol{x}}(t))$. The sequence of triggering times is defined by the triggering condition through the following recursive relation

$$t_0 = 0, \quad t_{i+1} = \inf\{t > t_i \mid \mathcal{C}(\boldsymbol{x}(t), \boldsymbol{x}(t_i)) = 0\} \tag{2-5}$$

The sampling error as a result of the discontinuous sampling of the system is given by $\boldsymbol{e}(t) \in \mathbb{R}^{n_x}$, which is the difference between the last sampled state and the current state of the system.

$$\boldsymbol{e}(t) := \hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t) \tag{2-6}$$

Using this sampling error, the overall dynamics of the system can be represented as

$$\dot{\boldsymbol{x}} = f\left(\boldsymbol{x}, g(\boldsymbol{x} + \boldsymbol{e})\right) \tag{2-7}$$

The idea of ETC is to choose the triggering condition $\mathcal{C}$ in such a manner that it ensures stability for the closed-loop event-triggered system. However, some early works, such as Åarzén (1999), did not yet include any guarantees of the stability of the resulting system.

Most triggering conditions guarantee the stability of the closed-loop by constraining the rate of change of the Lyapunov function. However, it is also possible to achieve such guarantees in other ways, such as with bounds directly on the Lyapunov function (see Wang and Lemmon (2008)).

2-2 Trigger monitoring

There are different approaches to monitoring the triggering conditions, which also influence how to construct a triggering condition $\mathcal{C}$ that ensures stability. The simplest method is to monitor at all times if the triggering condition is met, which is referred to as Continuous Event-Triggered Control (CETC). Additionally, it is possible to save some resources and only check if the triggering condition is met at regular intervals, which is accomplished with Periodic Event-Triggered Control (PETC). Finally, it is also possible to achieve similar aperiodic sampling without constantly monitoring the system. This is done with Self-Triggered Control (STC), which samples the system at the first instance when it would be possible for the system to have met the triggering condition. The following sections will introduce these different triggering monitoring methods.

2-2-1 Continuous Event-Triggered Control

When using CETC, the triggering conditions are monitored constantly, and the system will react as soon as the triggering condition is met. One of the most common techniques to construct guarantees for the stability of the resulting closed-loop event-triggered system is built around the notion of Lyapunov functions. Following along the lines of Tabuada (2007), it is shown how the stability of a closed-loop event-triggered system can be imposed using a relative triggering condition. Other possible triggering conditions will be discussed in section 2-3.

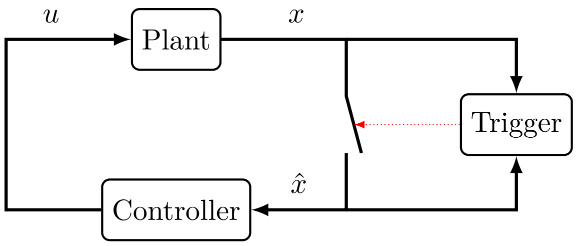


Figure 2-2 CETC control loop.

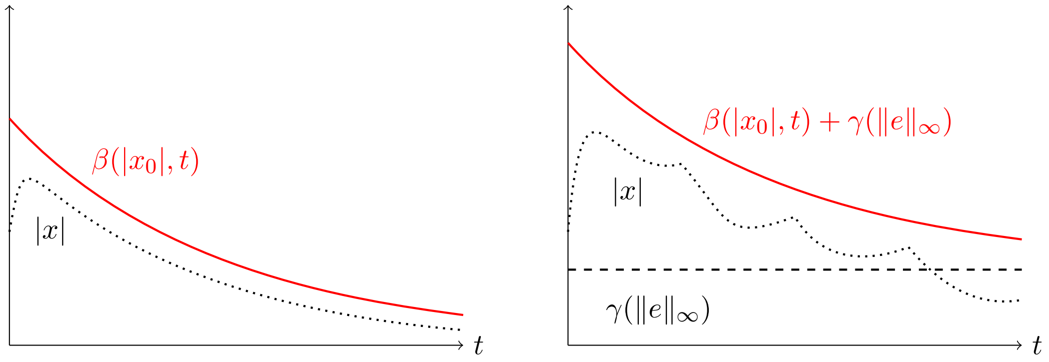
Firstly, the assumption is made that the closed-loop system, equation (2-7) is Input-to-State Stable (ISS) with respect to the sampling error $\boldsymbol{e}$. The concept of ISS is formally defined as:

Definition 2.1 (ISS (Sontag, 1989, 2008)):
The closed-loop system in equation (2-7) *is said to be Input-to-State Stable (ISS) with respect to the error* $\boldsymbol{e}$ *if a function* β *of class* KL *and a function* γ *of class* K *exist such that the following is satisfied for all initial states* $\boldsymbol{x}_0 \in \mathbb{R}^{n_x}$*, all* $\boldsymbol{u} \in \mathbb{R}^{n_u}$ *and all* $t \geq 0$

$$|\boldsymbol{x}(t)| \leq \beta(|\boldsymbol{x}_0|, t) + \gamma(\|\boldsymbol{e}\|_\infty) \tag{2-8}$$

Moreover, the system is said to be Globally Exponentially Input-to-State Stable (GEISS) if there exists $c > 0$ *and* $a > 0$ *such that the condition holds with* $\beta(|\boldsymbol{x}_0|, t) := c\mathrm{e}^{-at}|\boldsymbol{x}_0|$.

The function β requires that the transient behaviour of the system is bounded for any initial value $\boldsymbol{x}_0$ and that it dies out over time as it is a function of class KL. In other words, when there are no measurement errors, the closed-loop system is Globally Asymptotically Stable (GAS). The function γ ensures that the effect of the measurement error $\boldsymbol{e}$ on the states $\boldsymbol{x}$ of the system is also bounded. Figure 2-3 illustrates both aspects of the ISS property.



(a) Continuous applied control, $e = 0$

(b) Event triggered control

Figure 2-3 ISS condition for different control methods. The dotted line represents a possible path of the system.

If the closed-loop system is ISS with respect to the sampling error $\boldsymbol{e}$, an ISS Lyapunov function is guaranteed to exist. The following definition characterises this Lyapunov function:

Definition 2.2 (ISS Lyapunov function (Sontag and Wang, 1995)):
A smooth function $V : \mathbb{R}^{n_x} \to \mathbb{R}_0^+$ *is an ISS Lyapunov function for the closed-loop system in equation* (2-7) *if and only if functions* $\underline{\alpha}$*,* $\overline{\alpha}$*,* α *and* γ *of class* K_∞ *exist such that the following is satisfied for all* $\boldsymbol{x} \in R^{n_x}$ *and* $\boldsymbol{e} \in R^{n_x}$

$$\underline{\alpha}(|\boldsymbol{x}|) \leq V(\boldsymbol{x}) \leq \overline{\alpha}(|\boldsymbol{x}|) \tag{2-9}$$

$$\frac{d}{dt}V(\boldsymbol{x}(t)) = \frac{\partial V}{\partial \boldsymbol{x}} f(\boldsymbol{x}, g(\boldsymbol{x} + \boldsymbol{e})) \leq -\alpha(|\boldsymbol{x}|) + \gamma(|\boldsymbol{e}|) \tag{2-10}$$

The stability of the closed-loop is guaranteed as long as the Lyapunov function is monotonically decreasing over time. It follows from the definition of the ISS Lyapunov function that this is the case if the right-hand side of equation (2-10) remains non-positive.

By triggering the system to apply new inputs whenever the sampling error gets to the threshold,

$$\gamma(|\boldsymbol{e}|) = \rho\alpha(|\boldsymbol{x}|) \tag{2-11}$$

where $0 < \rho < 1$ is a positive constant which sets how much slower V is allowed to decrease as a result of using an event-triggered approach instead of an ideal closed-loop.

As a result of this triggering condition, it is automatically enforced that the function of the sampling error will remain below this threshold at all times.

$$\gamma(|\boldsymbol{e}|) \leq \rho\alpha(|\boldsymbol{x}|) \tag{2-12}$$

When the system triggers, the measurement error resets $\boldsymbol{e}(t_i) = \hat{\boldsymbol{x}}(t_i) - \boldsymbol{x}(t_i) = 0$ and therefore $\gamma(|\boldsymbol{e}(t_i)|) = 0$ as γ is a function of class K_∞. The threshold is non-negative $\rho\alpha(|\boldsymbol{x}|) \geq 0$ as α is also a function of K_∞. Hence, $\gamma(|\boldsymbol{e}|)$ is 0 when the system triggers and increases as the measurement error gets larger but can never exceed the triggering condition.

Combining this upper bound on $\gamma(|\boldsymbol{e}|)$ with the definition of the ISS Lyapunov function equation (2-10) results in the following dynamics of V:

$$\frac{d}{dt}V(\boldsymbol{x}(t)) = \frac{\partial V}{\partial \boldsymbol{x}} f(\boldsymbol{x}, g(\boldsymbol{x}+\boldsymbol{e})) \leq -(1-\rho)\alpha(|\boldsymbol{x}|) \tag{2-13}$$

As α is a function of class K_∞, it is positive for all inputs except at 0, which shows that V is guaranteed to decrease on all trajectories to the equilibrium point for $\rho < 1$.

$$\frac{d}{dt}V(\boldsymbol{x}(t)) \leq -(1-\rho)\alpha(|x|) < 0, \quad \boldsymbol{x} \neq 0 \tag{2-14}$$

This shows that using the triggering condition in equation (2-11) with $0 < \rho < 1$, V is also a Lyapunov function in the traditional sense and proves the asymptotic stability of the closed-loop event-triggered system.

At the cost of introducing a bit more conservativeness, the triggering condition can be further simplified such that its form no longer directly depends on the specific system. This does require some additional mild assumptions. By assuming that α^{-1} and γ are Lipschitz continuous on the compact set $S \subseteq \mathbb{R}^n$ in which we are trying to control the system, the bound of equation (2-12) can be rewritten as:

$$\gamma(|\boldsymbol{e}|) \leq \rho\alpha(|\boldsymbol{x}|) \iff \alpha^{-1}\left(\frac{\gamma(|\boldsymbol{e}|)}{\rho}\right) \leq |\boldsymbol{x}| \tag{2-15}$$

Because α^{-1} and γ are Lipschitz continuous, $\alpha^{-1}\left(\gamma(|\boldsymbol{e}|)/\rho\right)$ is also Lipschitz continuous on the same set. Combining the definition of Lipschitz continuity with the fact that $\alpha^{-1}\left(\gamma(|0|)/\rho\right) = 0$, as both α and γ are functions of class K. Results in the following inequality, where L is the smallest possible Lipschitz constant.

$$\left|\alpha^{-1}\left(\frac{\gamma(|\boldsymbol{e}|)}{\rho}\right) - \alpha^{-1}\left(\frac{\gamma(|0|)}{\rho}\right)\right| \leq L|\boldsymbol{e}-0| \quad \Longrightarrow \quad \alpha^{-1}\left(\frac{\gamma(|\boldsymbol{e}|)}{\rho}\right) \leq L|\boldsymbol{e}| \tag{2-16}$$

If the system is triggered at $L|\boldsymbol{e}| = |\boldsymbol{x}|$, it is automatically enforced that $L|\boldsymbol{e}| \leq |\boldsymbol{x}|$. Together with the inequality derived from the Lipschitz assumption, one finds that this new bound is more conservative and guarantees the original in equation (2-12).

$$L|\boldsymbol{e}| \leq |\boldsymbol{x}| \implies \alpha^{-1}\left(\frac{\gamma(|\boldsymbol{e}|)}{\rho}\right) \leq |\boldsymbol{x}| \iff \gamma(|\boldsymbol{e}|) \leq \rho\alpha(|\boldsymbol{x}|) \tag{2-17}$$

Therefore, triggering the system at $L|\boldsymbol{e}| = |\boldsymbol{x}|$ also ensures the asymptotic stability of the closed-loop event-triggered system. For ease of calculation, this can be further simplified by taking the square of the entire triggering condition and defining the triggering coefficient $\mu = \frac{1}{L^2}$.

$$L|\boldsymbol{e}| \leq |\boldsymbol{x}| \iff |\boldsymbol{e}|^2 \leq \mu|\boldsymbol{x}|^2 \tag{2-18}$$

The Inter-Event Time (IET)[1] $\tau_k := t_{i+1} - t_i$ is the time between two triggering instances. The Minimal Inter-Event Time (MIET) $\underline{\tau} := \inf_{i \in \mathbb{N}} \tau_i$ is the lowest possible time between two triggering events that can occur. It is important to establish a lower bound on the MIET, to ensure that it is possible to implement the controller. Because the triggering times are implicitly defined through the triggering condition, it is possible for the system to trigger infinitely many times in a finite time period. This is referred to as Zeno behaviour (see Lygeros et al. (2003)) and results in the time between events becoming arbitrarily small. However, there is a physical limit to how quickly the controller can trigger as it takes time to measure the state of the plant, calculate the new control input and apply this new input to the system. If the controller triggers quicker than it is physically possible to apply control, it is no longer possible to implement such a controller.

The triggering coefficient μ controls the trade-off between how often the system is triggered and how well the system is controlled. A higher μ results in a more lenient triggering condition and, therefore, less triggering of the system and a higher lower bound of the inter-event time. However, it also means that the convergence rate is lower and that the size of the ball around the steady state to which the system converges increases.

2-2-2 Periodic Event-Triggered Control

When implementing a CETC strategy, specialised dedicated hardware is required to monitor the triggering conditions truly continuously. The easiest method to get around this constraint is to approximate the continuous trigger monitoring by periodically checking the triggering conditions with a sufficiently short sampling time. This is much easier to implement and does not require special dedicated hardware on a digital platform. However, there are no guarantees that such a change does not significantly degrade the system performance, as the sampling time cannot be made infinitely small due to hardware limitations.

Instead of simply setting the sampling rate sufficiently high to minimise the error with a continuous implementation, PETC introduced in Heemels et al. (2011) directly models how the frequency of evaluating the triggering conditions impacts the system stability and performance. The triggering condition is checked after a fixed sampling interval $h > 0$, which allows PETC to take the best of two worlds, periodic sampling and ETC. This also opens the door to designing a system which is optimised to have a longer sampling period for the triggering condition, which can further reduce the use of resources.

[1] This is also commonly referred to as the Inter-Sampling Time, IET is adopted in this work to more clearly distinguish between the sampling interval of the control system and the sampling of the triggering mechanism.

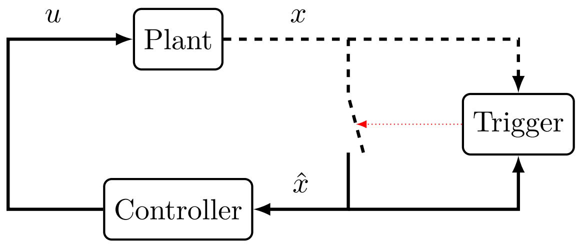


Figure 2-4 PETC control loop.

Another advantage of the PETC is that the issue of potential Zeno behaviour discussed for a CETC implementation is already ruled out by design since the time between events (IET) is automatically lower bounded by the sampling rate of the triggering condition. The MIET of a PETC can never be lower than the set fixed sampling interval h. Furthermore, scheduling the communications between the sensor nodes and controller is made simpler as the transfer of information only occurs at set times, which are known in advance. A maximum inter-event time $\overline{k}$ can arise naturally but is also often enforced to reduce the set of possible triggering times to a finite set (Gleizer and Mazo Jr., 2020a). Note that a maximum inter-event time does not naturally occur for systems with a mixed triggering condition.

Since the system can no longer trigger instantly when it hits the triggering condition, it becomes possible for the triggering condition to become positive temporarily. Consequently, the equality in the triggering condition has to be replaced with an inequality. Together with the periodic checking of the triggering condition, the recursive definition of the triggering times changes to

$$t_0 = 0, \quad t_{i+1} = \inf\{t = kh > t_i,\ k \in \mathbb{N} \mid \mathcal{C}(\boldsymbol{x}(t), \boldsymbol{x}(t_i)) > 0\} \tag{2-19}$$

The triggering condition also has to be adapted to take into account that there might be a period of time between the moment the triggering condition becomes positive and when the system actually triggers. This delay in triggering can never be larger than the sampling interval h. However, the trigger condition must be constructed such that if the triggering threshold is not exceeded at t_i, it ensures stability and performance of the system until the triggering condition is rechecked at t_{i+1}. As a result of this adaptation, the triggering condition becomes more conservative. Increasing the sampling interval will thus lead to the more frequent triggering of the system.

Different methods exist to design such triggering conditions, either by adapting an existing CETC triggering condition or by constructing the conditions directly from a continuous time controller. See, for instance, Heemels et al. (2013), Heemels and Donkers (2013), Postoyan et al. (2013) for concrete examples of formulating PETC triggering conditions that guarantee the stability of the resulting closed-loop system. Adapting the relative triggering condition introduced in the previous section 2-2-1 would result in having to set a lower μ than for a CETC to guarantee a stable closed-loop system.

For Linear Time-Invariant (LTI) systems Heemels et al. (2013) provides a general framework for the stability and performance analysis of PETC systems with a quadratic triggering condition without an absolute threshold ($\beta = 0$). See section 2-3 for details on such triggering conditions. Different Linear Matrix Inequality (LMI)'s are constructed that can be used to derive stability conditions for the system, where a piece-wise linear method results in the least conservative bounds. In the absence of noise, this results in guarantees that the close-loop PETC system is Globally Exponentially Stable (GES).

Definition 2.3 (Global Exponential Stability):
The closed loop system in equation (2-7) *is said to be Globally Exponentially Stable (GES) with a lower bound on the decay rate of* c *if there exists* $c > 0$ *and* $a > 0$ *such that the following is satisfied for all initial states* $\boldsymbol{x}_0 \in \mathbb{R}^{n_x}$ *and all* $t \geq 0$

$$|\boldsymbol{x}(t)| \leq c\mathrm{e}^{-at}|\boldsymbol{x}_0| \tag{2-20}$$

The stability of systems with noise and disturbances is discussed in section 2-5.

2-2-3 Self-Triggered Control

Event-triggered methods rely on continuously or regularly monitoring the state of the system to decide when to update the control inputs to the system. However, it might not always be possible or desirable to constantly use resources or even dedicated hardware to keep an eye on the state of the system. STC, introduced in Velasco et al. (2003), provides a solution for these situations by eliminating the need to monitor the triggering conditions while maintaining the advantages of aperiodic sampling. Instead of monitoring when a triggering condition is met, the controller simply predicts the earliest time instance at which the system could trigger and samples the system at this predicted time.

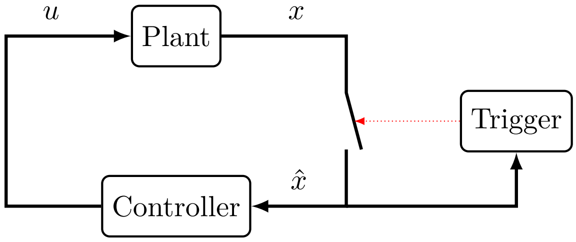


Figure 2-5 STC control loop.

When the controller updates the control input, it also uses the state of the system together with its knowledge of the plant dynamics to calculate how long it can wait before the control inputs need to be updated again to ensure that the system will remain stable and continues to meet the performance requirements. A triggering function $\mathcal{F} : \mathbb{R}^{n_x} \to \mathbb{R}^+$ is constructed that gives the minimum amount of time in which the system could hit the triggering condition $\mathcal{C}$ in the absolute worst-case scenario. The system first triggers when it is initialised at $t_0 = 0$. After that, the next triggering time is computed using the triggering function $t_{i+1} = t_i + \mathcal{F}(\boldsymbol{x}(t_i))$ whenever the system is triggered. The inter-event time is set by the triggering function and only depends on the state $\tau_i = \mathcal{F}(\boldsymbol{x}(t_i))$. Since $\mathcal{F}$ is constructed to guarantee that the system will stay within the boundaries of a triggering condition until the next triggering time, the resulting self-triggered closed-loop system will be stable. See Mazo Jr. et al. (2009); Mazo Jr. and Tabuada (2009) for a more comprehensive treatment of self-triggered control for LTI systems.

A big advantage of STC is that the scheduling of communications is far easier because the triggering times are known in advance, allowing for more efficient use of the available bandwidth. Much the same as with PETC, an STC system does not require any special hardware to constantly monitor the triggering conditions. In addition, STC demands less frequent measurements of the system, which can significantly reduce the hardware requirements, especially if a sensor requires a substantial amount of power to run or needs to make complex computations to interpret its measurement.

In the absence of disturbances and noise, it is possible to perfectly predict when the system will hit the triggering condition, in such a case, STC would be able to generate the same triggering times and therefore performance as an ETC mechanism. However, when external disturbances are present, the triggering function is forced to be more conservative, resulting in shorter triggering intervals because the predictions have to always account for the worst-case scenario in terms of the disturbances. Another downside of STC is that it is less robust than ETC. The closed-loop can quickly become unstable if disturbances turn out to be larger than modelled since the triggering times computed with $\mathcal{F}$ depend directly on the disturbance model.

2-3 Triggering conditions

A range of different triggering conditions can be used to ensure that the closed-loop event-triggered system is stable. This includes conditions based on the state error, input error or directly on a Lyapunov function. The most common method to ensure the stability of the closed-loop relies on constructing the triggering conditions such that it guarantees a Lyapunov function V to decrease monotonically at all times.

2-3-1 Absolute

An absolute triggering condition compares the sampling error to a fixed threshold and is most commonly found in earlier works, including Åarzén (1999); Åström and Bernhardsson (1999); Miskowicz (2006); Otanez et al. (2002); Lunze and Lehmann (2010). The measured states are kept within a set distance of the true states of the system at all times. Consequently, the distance between the input to the system and the ideal closed-loop input is also bounded. As a result, the system remains stable when the sampling error is kept sufficiently small. The triggering condition for this approach is given by

$$|\boldsymbol{e}(t)|^2 \geq \beta \tag{2-21}$$

where $\beta > 0$ is the threshold. The problem with an absolute threshold is that it becomes more sensitive as the state moves away from the origin. This leads to very frequent triggering when the state of the system is far from the equilibrium point. A larger threshold β leads to longer inter-event times of the system. However, the set around to equilibrium to which the system converges is also proportional to the same threshold β. Therefore, a trade-off between the tightness of the control around the origin and how frequently the system triggers has to be made.

Note that such an absolute triggering condition is essentially just reducing the resolution of the sensor output. Changes in the state smaller than a certain threshold are not detected. This is also referred to as a deadband since the threshold β introduces a band in which the state of the system can change without the control system reacting.

2-3-2 Relative

The relative triggering condition is introduced by Tabuada (2007) and compares the sampling error to the current value of the state. As such, the threshold becomes relative to the state and no longer becomes more sensitive as the state moves further away from the origin. Stability guarantees of this method are discussed in section 2-2-1. The relative triggering condition is defined as

$$|\boldsymbol{e}(t)|^2 \geq \mu|\boldsymbol{x}(t)|^2 \tag{2-22}$$

where $0 < \mu < 1$ is the triggering coefficient. Although the relative triggering condition solves the issues associated with the absolute triggering condition, it also has its downsides. The threshold becomes infinitesimal when the state of the system approaches the equilibrium point at the origin. This does not cause any problems in a perfect system without any disturbances. However, when arbitrarily small disturbances are present, the system is triggered excessively when close to the equilibrium point. This also causes Zeno behaviour for CETC and the system is no longer guaranteed to have a MIET (Borgers and Heemels, 2014).

2-3-3 Mixed

Combining both of the approaches mentioned above results in a mixed triggering condition suggested by Donkers and Heemels (2012). The triggering condition behaves as an absolute threshold close to the origin and as a relative condition as the relative part becomes dominant further out from the origin, thus avoiding the issues associated with both triggering conditions. The mixed triggering condition is given by

$$|\boldsymbol{e}(t)|^2 \geq \mu|\boldsymbol{x}(t)|^2 + \beta \tag{2-23}$$

A similar trade-off, as discussed for the absolute triggering condition between the size of the set around the equilibrium to which the system converges and the inter-event time, also appears for the mixed triggering condition. However, the trade-off is slightly simplified since the absolute threshold is only critical close to the equilibrium point. This means that the threshold β only significantly affects the inter-event time when close to the equilibrium and is entirely there to reduce excessive sampling at the equilibrium as a result of disturbances.

2-3-4 Quadratic

Quadratic triggering functions are a much more general class of triggering functions than the previously discussed options. Many different triggering conditions, including those introduced above, can be written in this quadratic form. The quadratic form is given by (Heemels et al., 2013)

$$\begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} \geq 0 \tag{2-24}$$

where $\boldsymbol{Q}$ is the triggering matrix. For the relative triggering condition introduced in the previous section, the triggering matrix $\boldsymbol{Q}$ looks like

$$\boldsymbol{Q} = \begin{bmatrix} (1-\mu)\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{bmatrix} \tag{2-25}$$

More examples of other triggering conditions represented in this quadratic form can be found in Heemels et al. (2013).

The quadratic triggering condition can be further generalised by expanding it to include a fixed threshold as in the mixed triggering condition (Trobinger et al., 2022).

$$\begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} \geq \beta \tag{2-26}$$

Where β is the absolute threshold used to control triggering close to the equilibrium point. This more general version can also be represented in the form of the mixed triggering condition:

$$\boldsymbol{e}(t)^{\mathsf{T}}\boldsymbol{M}\boldsymbol{e}(t) \geq \boldsymbol{x}(t)^{\mathsf{T}}\boldsymbol{N}\boldsymbol{x}(t) + \beta \tag{2-27}$$

where $\boldsymbol{M}$ and $\boldsymbol{N}$ are triggering matrices of appropriate size. The matrix $\boldsymbol{N} \succeq 0$ needs to be positive semi-definite to prevent the system from triggering again immediately after the measurement error $\boldsymbol{e}$ is reset to zero.

For this general version of the mixed triggering condition, the corresponding quadratic triggering matrix $\boldsymbol{Q}$ is given by:

$$\boldsymbol{Q} = \begin{bmatrix} M-N & -M \\ -M & M \end{bmatrix} \tag{2-28}$$

2-3-5 Dynamic

So far, all stability guarantees have been based on the notion that the Lyapunov function V is required to decrease monotonically at all times. However, to guarantee the stability of the system, it suffices if the Lyapunov function is negative on average (Ye et al., 1998). Dynamic triggering capitalises on this by relaxing the constraint on the Lyapunov function by means of an internal dynamic variable to allow the Lyapunov function to increase while ensuring that it decreases over time. The dynamic variable is a filtered version of the triggering condition and keeps track of how quickly the Lyapunov function has decreased compared to the prescribed decay rate. The Lyapunov function is given more slack to increase if it previously decayed faster than prescribed.

This idea was first used to create an ETC in Wang and Lemmon (2008) and has since been applied to CETC (Girard, 2015), PETC (Borgers et al., 2017) as well as STC (Mazo Jr. et al., 2010).

2-4 Linear Time-Invariant control setup

Using the knowledge and techniques introduced in the previous sections, a more specific control setup is presented: Consider an LTI system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{2-29}$$

The system is controlled using a linear state-feedback controller implemented with a sample-and-hold mechanism.

$$\boldsymbol{u}(t) = \boldsymbol{K}\hat{\boldsymbol{x}}(t) \tag{2-30}$$

The control input $\boldsymbol{u}(t)$ is computed using the last measured state $\hat{\boldsymbol{x}}(t)$, which is defined as

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_i), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N}_0 \tag{2-31}$$

Where the triggering times t_i are set by a quadratic triggering condition

$$t_0 = 0, \quad t_{i+1} = \inf\left\{ t > t_i \left| \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} > 0 \right.\right\} \tag{2-32}$$

This results in a CETC system. The inter-event time for a given state $\boldsymbol{x}$ is defined as

$$\tau(\boldsymbol{x}) \coloneqq t_{i+1} - t_i \tag{2-33}$$

For an LTI system, the state of the system in-between sampling times t_i and t_{i+1} can be computed as

$$\boldsymbol{x}(t_i + \sigma) = \boldsymbol{\Lambda}(\sigma)\hat{\boldsymbol{x}}(t_i) \tag{2-34}$$

$$\boldsymbol{\Lambda}(\sigma) = \mathbf{I} + \int_0^\sigma \mathrm{e}^{\boldsymbol{A}\tau} d\tau(\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}) \tag{2-35}$$

which permits writing the inter-event time explicitly as

$$\tau(x) = \inf\{\sigma > 0 \mid \boldsymbol{x}^{\mathsf{T}}\boldsymbol{\Phi}(\sigma)\boldsymbol{x} \geq 0\} \tag{2-36}$$

where $\boldsymbol{\Phi}(\sigma)$ is the transition matrix of the triggering condition defined as

$$\boldsymbol{\Phi}(\sigma) = \begin{bmatrix} \boldsymbol{\Lambda}(\sigma) \\ \mathbf{I} \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{\Lambda}(\sigma) \\ \mathbf{I} \end{bmatrix} \tag{2-37}$$

2-5 Stability in the Presence of Noise and Disturbances

Up until now, the stability guarantees have only been discussed for idealised systems without any noise or disturbances. As already mentioned in section 2-3 on triggering conditions, a small disruption to the system can yield significantly different sampling behaviour. Naturally, this also affects the stability guarantees of the closed-loop system.

For the LTI system introduced in the previous section, an additive disturbance would change the state equation to:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{E}\boldsymbol{w}(t) \tag{2-38}$$

where $\boldsymbol{w}(i) \in \mathbb{R}_{n_w}$ is an unknown disturbance and $\boldsymbol{E}$ the disturbance matrix. The measurement noise changes the definition of the last measured state to:

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_i) + \boldsymbol{v}(t_i), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N}_0 \tag{2-39}$$

where $\boldsymbol{w}(i) \in \mathbb{R}_{n_w}$ is the unknown measurement noise.

With these additions, a close-loop ETC system can no longer be guaranteed to be ISS with respect to the measurement noise. However, it can be guaranteed to be ISS with respect to the measurement noise, the disturbance and the noise. This notion is also referred to as Input-to-State Practically Stable (ISpS) (Borgers and Heemels, 2014). As a result the guarantee no longer stipulates that the system will converge to its equilibrium but to a ball around it.

2-5-1 Periodic Event-Triggered systems

For linear PETC systems with an additive disturbance Heemels et al. (2013) provides LMI conditions to verify the stability of the system. An impulsive system approach is used to derive conditions for a weaker stability notion that bounds the $\mathcal{L}_2$-gain of the system.

Definition 2.4 ($\mathcal{L}_p$ Stability):
Given $1 \leq p \leq \infty$, the closed-loop system in equation (2-7) is said to have an $\mathcal{L}_p$-gain from the exogenous input $\boldsymbol{d}(t)$ to output $\boldsymbol{y}(t)$ smaller than or equal to γ if there exists a function $\delta : \mathbb{R}^{n_x} \to \mathbb{R}_0^+$ such that the following is satisfied for any bounded exogenous input $\|\boldsymbol{d}\|_p < \infty$ and, all initial states $\boldsymbol{x}_0 \in \mathbb{R}^{n_x}$

$$\|\boldsymbol{y}\|_p \leq \delta\left(\boldsymbol{x}(0)\right) + \gamma\|\boldsymbol{d}\|_p \tag{2-40}$$

See the original paper for the LMI stability conditions and their derivations.

This is extended in Gleizer and Mazo Jr. (2020b) to enable the derivation stability conditions for PETC systems with respect to additive disturbances, measurement noise and a (positive) absolute threshold in a quadratic triggering condition. This is done by demonstrating that the given PETC system with any quadratic triggering condition meets the conditions of a theorem by (Nešić et al., 2013) that establishes the equivalence of different stability notions for homogeneous hybrid dynamical systems.

Theorem 2.1 (Equivalence of Stability Notions (Gleizer and Mazo Jr., 2020b)): *For the PETC system with a quadratic triggering condition, the following statements are equivalent:*

- *The origin of the system is Globally Exponentially Stable (GES) if the absolute threshold $\beta = 0$, additive disturbances $\boldsymbol{w} = 0$ and measurement noise $\boldsymbol{v} = 0$*
- *The system is Globally Exponentially Input-to-State Stable (GEISS)*
- *The system is $\mathcal{L}_p$ stable with respect to additive disturbances $\boldsymbol{w}$, measurement noise $\boldsymbol{v}$ and the absolute threshold β*

As a result, the LMI stability conditions presented in Heemels et al. (2013) can also be used to establish stability guarantees for PETC systems with a (mixed) quadratic triggering condition if the system without additive disturbances $\boldsymbol{w} = 0$ or measurement noise $\boldsymbol{v} = 0$ and without an absolute threshold $\beta = 0$ is GES. Then, the system with disturbances, noise and a (positive) absolute threshold in the triggering condition is still guaranteed to be GEISS and be $\mathcal{L}_p$ stable with respect to these exogenous inputs.

This does not mean that these exogenous inputs do not influence the stability guarantees, the γ function in the GEISS and γ gain of the $\mathcal{L}_p$ stability do depend on the additive disturbance, measurement noise and the value of the absolute threshold. The area around the origin to which the system is guaranteed to converge just becomes larger when the magnitude of the disturbance or noise increases and also grows as the absolute threshold β is increased.

2-6 State-dependent sampling

State-dependent sampling is a method to shift the burden of computing the next triggering time in an STC from online to offline. Instead of evaluating the triggering function $\mathcal{F}$ when the system is triggered, state-dependent sampling designs a mapping from state-space to a set of inter-event times τ ahead of time. This approach is introduced in Fiter et al. (2012). The state-space is partitioned into a finite number of regions, where each region $\mathcal{R}_s$ is labelled with the largest possible inter-event time $\underline{\tau}_s$ that guarantees the stability of the closed-loop system for every state in the region. This minimises the number of computations that need to be performed online and the delays associated with these computations while the system is running.

Finding the inter-event time for a given state $\tau(\boldsymbol{x})$ is reduced to computing in which region $\mathcal{R}_s$ the current state $\boldsymbol{x}$ is located. Instead of evaluating the triggering function $\mathcal{F}$, the inter-event time can be computed as:

$$\tau(\boldsymbol{x}) = \max_{s \in \{1,\dots,q\} \; s.t. \; \boldsymbol{x} \in \mathcal{R}_s} \underline{\tau}_s \tag{2-41}$$

However, additional conservatism is introduced in two ways to achieve the reduction from infinitely many possible states and triggering times to a finite number of stability conditions. The partitioning of the state-space introduces conservatism, as the shortest inter-event time that is required to guarantee stability for some states in the region is used for all states in the region. Secondly, since it is impossible to assess the triggering condition at an infinite number of points in time, finding the largest possible inter-event time that guarantees stability for all states in the region is impossible and thus has to be approximated with a lower bound.

The ideas of state-dependent sampling can also be used to construct abstractions of ETC systems that model the triggering behaviour of the system which can then be used to schedule the communications (Mazo Jr. et al., 2018). The largest possible inter-event time $\underline{\tau}_s$ that guarantees stability for every state in a region $\mathcal{R}_s$ of an STC system is also the shortest inter-event time after which the system could trigger if it were to be controlled using a CETC based on the same triggering condition. However, for traffic abstractions, it is also necessary to compute an upper bound of the inter-event times $\overline{\tau}_s$ that can be observed in a certain region $\mathcal{R}_s$.

Choosing the regions is crucial as this affects how much conservatism is introduced by the shift from online to offline calculations. A naive method would be to partition the state-space into orthotopes (see figure 2-6a). However, this does not take into account how the inter-event time changes with respect to the state-space. Approaches to take this into account lead to tighter intervals of the inter-event time within regions of the same volume and, therefore, less conservatism (Delimpaltadakis and Mazo Jr., 2021a).

By looking at scaling laws, it is possible to partition the state-space in regions that align with how the inter-event time changes throughout the state-space. It is impossible to derive such scaling laws for general non-linear systems. However, any smooth non-linear system can be embedded in a homogeneous system with one additional dimension (Anta and Tabuada, 2012, Lemma IV.4). The definitions of a homogeneous function and system are given by:

Definition 2.5 (Homogeneous function (Anta and Tabuada, 2012)):
A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is called homogeneous of degree $\psi \in \mathbb{R}$ if there exist $r_i > 0, i = 1, \ldots, m$ such that:

$$f_i(\lambda^{r_1}x_1, \ldots, \lambda^{r_n}x_n) = \lambda^{\psi+r_i} f_i(x_1, \ldots, x_n), \quad \forall\lambda > 0, \quad \forall x \in \mathbb{R}^n$$

where f_i is the i^{th} component of f and $\psi > -\min_i r_i$.

Definition 2.6 (Homogeneous system (Delimpaltadakis and Mazo Jr., 2021b)):
A system $\dot{x} = f(x(t), u(t))$ is called homogeneous of degree $\psi \in \mathbb{R}$ if $\tilde{f}(x(t)) = f(x(t), g(x(t)))$ is a homogeneous function of the same degree.

Homogeneous systems possess multiplicative scaling behaviour that can be used to derive a scaling law for the inter-event time that occurs as a result of the triggering condition. The scaling law is defined along lines going through the origin but excluding the origin itself. These lines are called homogeneous rays. For homogeneous systems, the change of the inter-event time along these homogeneous rays can be described as:

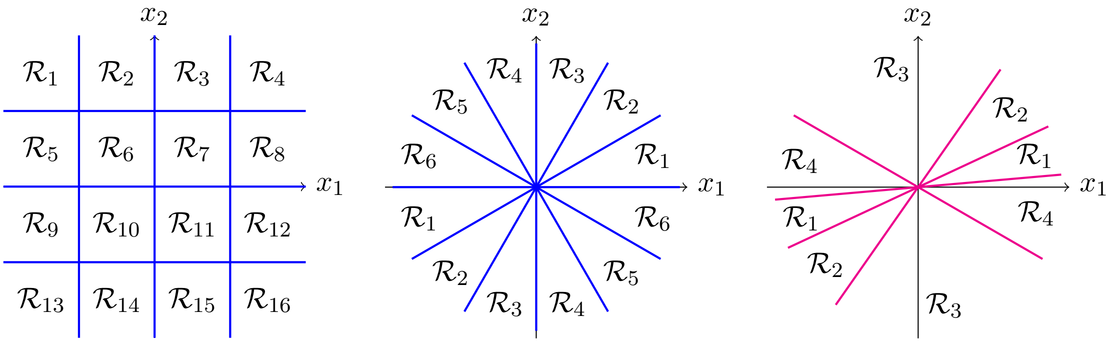
Theorem 2.2 (Scaling law for inter-event time (Anta and Tabuada, 2008)):
Consider a homogeneous system of degree ψ and a triggering function $\mathcal{C}(\cdot)$ homogeneous of degree ϕ. The inter-event times $\tau : \mathbb{R}^{n_x} \to \mathbb{R}^+$ implicitly defined by the triggering function $\mathcal{C}(\cdot)$ scale according to:

$$\tau(\lambda \boldsymbol{x}) = \lambda^{-\psi}\tau(\boldsymbol{x}), \quad \lambda > 0, \quad \boldsymbol{x} \in \mathbb{R}^n$$

The degree of homogeneity of the triggering function does not influence the scaling of the inter-event time along a homogeneous ray, as the scaling of the state and the error cancel each other out in any homogeneous function. For non-linear systems, which are homogeneous of degree $\psi > 0$, the inter-event times become larger for states closer to the origin moving along a homogeneous ray. That is, for a given direction in the state-space, the system can wait longer before updating the control inputs if the state is already closer to the origin than if the state is further away. In the particular case of linear systems, which are homogeneous of degree $\psi = 0$, the scaling law reduces to $\tau(\lambda \boldsymbol{x}) = \tau(\boldsymbol{x})$. This means that all states lying on the same homogeneous ray have the same inter-event time.

2-6-1 Linear systems

The fact that all states on a homogeneous ray have the same inter-event time motivates a partitioning of the state-space where an infinite number of homogeneous rays are bundled together to form two cones pointed at the origin from opposite directions. The boundaries of the q conic regions are constructed such that all regions together cover the entire state-space $\bigcup_{s=1}^{q} \mathcal{R}_s = \mathbb{R}^{n_x}$. See figures 2-6b and 2-6c for examples of such conic partitions in $\mathbb{R}^2$.


x_2
x_1
$\mathcal{R}_1$ $\mathcal{R}_2$ $\mathcal{R}_3$ $\mathcal{R}_4$
$\mathcal{R}_5$ $\mathcal{R}_6$ $\mathcal{R}_7$ $\mathcal{R}_8$
$\mathcal{R}_9$ $\mathcal{R}_{10}$ $\mathcal{R}_{11}$ $\mathcal{R}_{12}$
$\mathcal{R}_{13}$ $\mathcal{R}_{14}$ $\mathcal{R}_{15}$ $\mathcal{R}_{16}$
x_2
x_1
$\mathcal{R}_4$ $\mathcal{R}_3$ $\mathcal{R}_5$ $\mathcal{R}_2$ $\mathcal{R}_6$ $\mathcal{R}_1$
$\mathcal{R}_1$ $\mathcal{R}_6$ $\mathcal{R}_2$ $\mathcal{R}_5$ $\mathcal{R}_3$ $\mathcal{R}_4$
x_2
x_1
$\mathcal{R}_3$ $\mathcal{R}_2$ $\mathcal{R}_4$ $\mathcal{R}_1$
$\mathcal{R}_1$ $\mathcal{R}_4$ $\mathcal{R}_2$ $\mathcal{R}_3$

(a) Naive

(b) Isotropic

(c) Discrete-time model behaviour

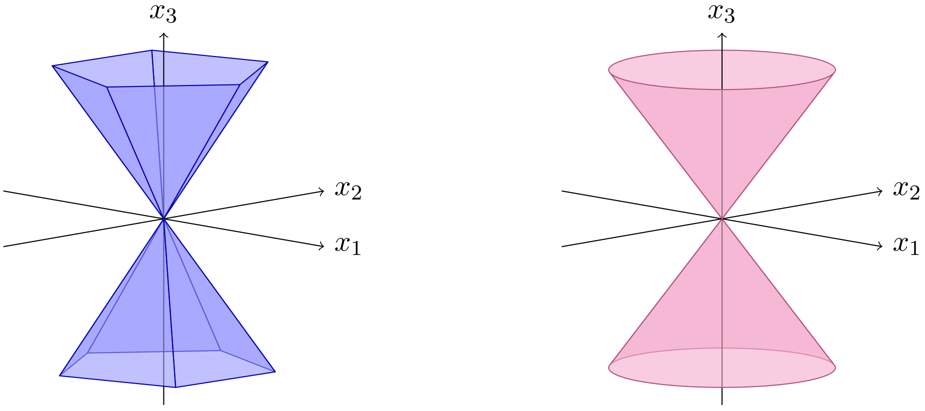
Figure 2-6 Different methods of partitioning a $\mathbb{R}^2$ state-space in regions. The blue lines represent region boundaries that are represented with linear inequalities, and the magenta boundaries are defined as intersections of quadratic inequalities. Finding the region $\mathcal{R}_s$ that a certain state $\boldsymbol{x}$ belongs to is computationally easier for regions that are described using linear inequalities.

Two different approaches of dividing the state-space into a conic covering are presented in Fiter et al. (2012). The first method evenly partitions the state-space in all dimensions, which is referred to as an isotropic covering (see figure 2-6b). The second method approximates the regions of the continuous-time model based on the dynamics of a discrete-time version of the system. It divides the state-space indirectly by evenly partitioning the possible inter-event times and finding the regions that would have an inter-event time within a specific interval in the discrete-time model (see figure 2-6c).

Creating an isotropic partitioning can easily be done in angular coordinates $\boldsymbol{x} \in \mathbb{R}^2 : (r, \omega)$. Each region $\mathcal{R}_s$ is associated with a range of the angular coordinate $\omega \in [\underline{\omega}_s, \overline{\omega}_s]$. With an isotropic covering, determining in which region $\mathcal{R}_s$ a state $\boldsymbol{x}$ is situated is computationally very simple since the resulting regions are polyhedral cones which can be represented as linear inequalities. This is important as this step has to be done online. The main downside of an isotropic covering is that the number of regions increases exponentially with the dimension n_x of the state-space. This exponential growth also extends to the precision of the partitioning. To cut the length of the inter-event interval associated with each region in half requires

2^{n_x-1} times as many regions. As a result of this exponential growth, offline computational resources become the bottleneck for the precision of the partitioning. This can be an issue as regions that contain a larger range of inter-event times introduce more conservatism. Such rapid growth of the number of regions when the dimension of the state-space increases is also referred to as the curse of dimensionality. This occurs in both the isotropic and the naive covering of the state-space.

The main downside of partitioning based on a discrete-time version of the model is that the computations to check in which region a state is situated are more involved. All states that do not trigger at a certain point in time can be represented as a quadratic cone. The region that contains all states which have not triggered at a given inter-event time can therefore be approximated as the intersection of multiple quadratic cones corresponding to different points in time prior to the given inter-event time. This can even be done exactly for PETC systems. However, to check if a state is part of such a region requires individually checking for each quadratic cone if the state lies within its boundaries. This dramatically increases the number of computations that must be performed online. The main advantage of such a partitioning is that the number of regions does not directly depend on the dimension of the state-space. Moreover, the precision of the resulting regions is (almost) directly proportional to the number of regions, as the regions are chosen such that the difference between the shortest and largest inter-event time in a region is approximately equal for all regions.



(a) Polyhedral cone **(b)** Quadratic cone

Figure 2-7 Different types of convex cones. An exact conversion between the two types is impossible since the polyhedral cone is generated by a finite number of vectors. In contrast, the quadratic cone is generated by an infinite number of vectors. However, note that in $\mathbb{R}^2$, both types of cone are equivalent (see figures 2-6b and 2-6c).

To compute the bounds on the inter-event time for all states in a region, it is essential to define the regions using quadratic inequalities. This may seem counterintuitive as quadratic cones have been described as a downside of a time-based partitioning of the state-space. The structure of the partitioning is important to be able to find to which region a state belongs quickly. However, this structure is not relevant for the offline computations where the quadratic representation is required for the LMI's are used to compute the bounds on the inter-event time. In $\mathbb{R}^2$ both polyhedral and quadratic cones are the same. However, this is not the case in higher dimensions (see figure 2-7). For state-spaces in $\mathbb{R}^2$, the conic region s can be defined using the matrix $\boldsymbol{V}_s$ with a quadratic inequality:

$$\mathcal{R}_s = \{\boldsymbol{x} \in \mathbb{R}^2 | \boldsymbol{x}^\mathsf{T} \boldsymbol{V}_s \boldsymbol{x} \geq 0\}, \quad \boldsymbol{V}_s = \boldsymbol{V}_s^\mathsf{T}, \quad s \in \{1, \dots, q\} \tag{2-42}$$

For the region consisting of all states with the angular coordinate $\omega \in [\underline{\omega}_s, \overline{\omega}_s]$ the matrix $\boldsymbol{V}_s$ is defined as:

$$\boldsymbol{V}_s = \begin{bmatrix} -2\sin(\underline{\omega}_s)\sin(\overline{\omega}_s) & \sin(\underline{\omega}_s + \overline{\omega}_s) \\ \sin(\underline{\omega}_s + \overline{\omega}_s) & -2\cos(\underline{\omega}_s)\cos(\overline{\omega}_s) \end{bmatrix} \tag{2-43}$$

The definition of such a conic covering in a higher dimensional state-space and other extensions will be discussed at the end of this section.

The next step after partitioning the state-space is to compute the lower bound of the inter-event times required to guarantee stability. Kolarijani and Mazo Jr. (2016) adapts the procedure from Fiter et al. (2012) for application to ISS-based triggering conditions. Additionally, it is also extended to include an upper bound on the inter-event time $\overline{\tau}_s$ for use in traffic abstractions. Finally, it also modifies the procedure to allow for state-spaces of more than two dimensions. The lower and upper bounds on the inter-event time have to be computed such that

$$\tau(\boldsymbol{x}) \in [\underline{\tau}_s, \overline{\tau}_s], \quad \forall \boldsymbol{x} \in \mathcal{R}_s \tag{2-44}$$

Using the explicit formula for the inter-event time of an LTI system from equation (2-36), a necessary condition for $0 < \underline{\tau} \leq \overline{\sigma}$ to be a lower bound of the inter-event time $\tau(\boldsymbol{x})$ at state $\boldsymbol{x}$ can be expressed as

$$\boldsymbol{x}^\mathsf{T} \boldsymbol{\Phi}(\sigma) \boldsymbol{x} \leq 0, \quad \forall \sigma \in [0, \underline{\tau}] \tag{2-45}$$

where $\boldsymbol{\Phi}(\sigma)$ is the transition matrix of the triggering condition and $\overline{\sigma}$ is the global upper bound on the inter-event time. This condition cannot be evaluated directly as the matrix function $\boldsymbol{\Phi}(\sigma)$ represents an infinite number of matrices. However, since the matrix function $\boldsymbol{\Phi}(\sigma)$ is continuous, it is possible to over-approximate the condition using a convex embedding approach introduced in Hetel et al. (2007).

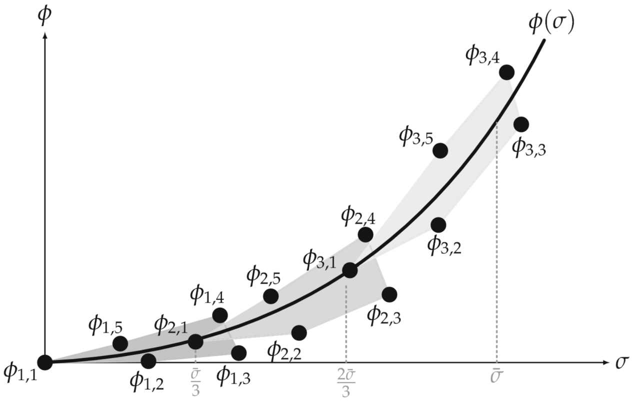


Figure 2-8 Embedding of a continuous function using three convex polytopes with five vertices each (Mazo Jr. et al., 2018). (The notation is not aligned with this work)

The method works by constructing convex polytopes around the matrix function $\boldsymbol{\Phi}(\sigma)$ using Taylor polynomials to derive a finite set of matrices $\underline{\boldsymbol{\Phi}}_\kappa$ for $\kappa \in \mathcal{K}_{\underline{\tau}}$ to replace the matrix function (see figure 2-8). The convexity of the polytopes makes it possible to over-approximate the stability condition such that it no longer directly depends on time σ but implies the necessary condition.

$$\left(\boldsymbol{x}^{\mathsf{T}}\underline{\boldsymbol{\Phi}}_{\kappa,\underline{\tau}}\boldsymbol{x} \leq 0, \quad \forall \kappa \in \mathcal{K}_{\underline{\tau}}\right) \quad \Longrightarrow \quad \left(\boldsymbol{x}^{\mathsf{T}}\boldsymbol{\Phi}(\sigma)\boldsymbol{x} \leq 0, \quad \forall \sigma \in [0, \underline{\tau}]\right) \tag{2-46}$$

This condition no longer contains a matrix function and can therefore be evaluated to compute a lower bound of the inter-event time for a given state $\boldsymbol{x}$. Details on constructing the set of matrices $\underline{\boldsymbol{\Phi}}_{\kappa,\underline{\tau}}$ can be found in (Kolarijani and Mazo Jr., 2016, Lemma 2). Using a convex embedding to represent the matrix function results in an over-approximation of the function, thus introducing more conservatism in the eventual result.

The final step is to extend the necessary condition for a lower bound on the inter-event time from a single state $\boldsymbol{x}$ to a condition valid for an entire region $\mathcal{R}_s$. The lower bound for region $\mathcal{R}_s$ is denoted as $\underline{\tau}_s$. This extension can be achieved using the S-procedure, which results in a set of LMIs presented in the following theorem:

Theorem 2.3 (Regional lower bound approximation (Fiter et al., 2012)): *Consider inter-event time $0 < \underline{\tau}_s \leq \overline{\sigma}$ and matrices $\underline{\boldsymbol{\Phi}}_{\kappa,\underline{\tau}_s}, \kappa \in \mathcal{K}_{\underline{\tau}_s}$ as constructed in (Kolarijani and Mazo Jr., 2016, Lemma 2). If there exist scalars $\underline{\epsilon}_\kappa \geq 0$ such that the following LMIs hold:*

$$\underline{\boldsymbol{\Phi}}_{\kappa,\underline{\tau}_s} + \underline{\epsilon}_\kappa V_s \preceq 0, \quad \forall \kappa \in \mathcal{K}_{\underline{\tau}_s}$$

then the inter-event time for every state $\boldsymbol{x}$ in region $\mathcal{R}_s$ is bounded from below by $\underline{\tau}_s$.

The computation of an upper bound on the inter-event time $\overline{\tau}_s$ follows a similar reasoning. See Kolarijani and Mazo Jr. (2016) for the computation of the upper bound and a more detailed treatment of the entire method.

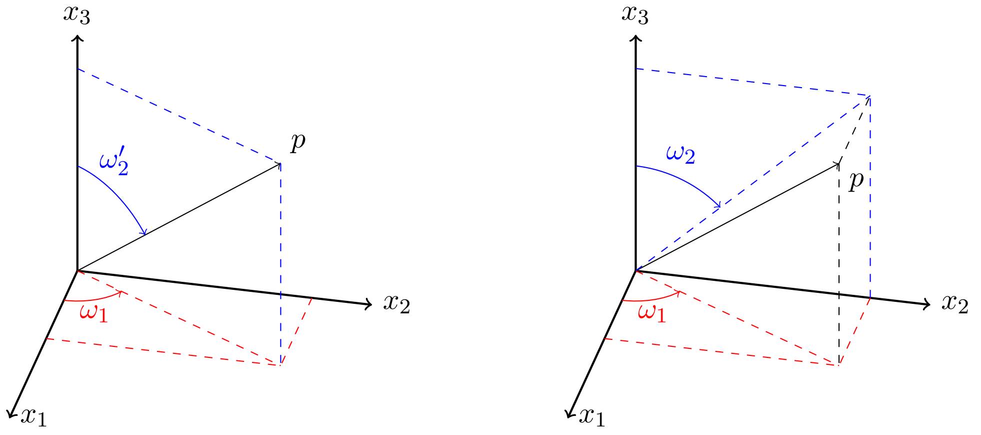
These LMIs provide necessary conditions to check whether a time σ is a valid lower (or upper) bound on the inter-event time for a region $\mathcal{R}_s$. It does not provide an explicit formula to compute the tightest possible bounds. Finding tight bounds $\underline{\tau}_s$ (and $\overline{\tau}_s$) entails running an optimisation using these conditions as constraints.

Conic coverings in N-dimensional space

Extending the conic covering to a higher dimensional space may seem trivial, but the need for a representation consisting of quadratic inequalities complicates the construction. The angular coordinates can easily be extended to more than two dimensions $\boldsymbol{x} \in \mathbb{R}^{n_x} : (r, \omega_1, \ldots, \omega_{n_x-1})$, where each region $\mathcal{R}_s$ is then associated with a range for each angular coordinate $\forall i \in \{1, \ldots, n_x - 1\}$, $\omega_i \in [\underline{\omega}_{s,i}, \overline{\omega}_{s,i}]$. This ensures that the finding in which region $\mathcal{R}_s$ and state $\boldsymbol{x}$ is situated is computationally simple.

However, there are multiple ways of defining such angular coordinates. The most conventional approach would be to use a hyperspherical coordinate system (see Blumenson (1960)) where each angular coordinate is defined in a plane which is in turn determined by the previous angular coordinates (see figure 2-9a). However, this does not result in a neat quadratic representation of two-sided cones.

Mazo Jr. et al. (2018) suggest a different approach using projections onto the coordinate planes as shown in figure 2-9b. This results in different regions, which can easily be represented as the intersection of multiple quadratic inequalities. Additionally, unlike with a hyperspherical coordinate system, the resulting regions are convex cones.



(a) Spherical coordinate system

(b) Polar projection coordinate system

Figure 2-9 3-dimensional angular coordinate systems.

The state $\boldsymbol{x}$ is projected onto $n_x - 1$ different coordinate planes, the angles in these planes are used to designate the location of the state. The $i - i + 1$ coordinate plane for $i \in \{1, \ldots, n_x - 1\}$ are used. However, different choices of the coordinate planes on which the states are projected are possible and will produce slightly different regions. Every region $\mathcal{R}_s$ can then be defined as the intersection of $n_x - 1$ quadratic inequalities:

$$\mathcal{R}_s = \left\{ \boldsymbol{x} \in \mathbb{R}^{n_x} \;\middle|\; \boldsymbol{x}^\mathsf{T} \boldsymbol{V}_{s,i} \boldsymbol{x} \geq 0, \quad \forall i \in \{1, \ldots, n_x - 1\} \right\} \tag{2-47}$$

Where the quadratic matrices contain zeros everywhere except for the four entries on the intersections of the rows and columns that are associated with the $i - i + 1$ coordinate plane:

$$\boldsymbol{V}_{s,i} = \begin{array}{c} \\ \text{row } i \\ \text{row } i{+}1 \end{array} \begin{array}{c} \begin{array}{cc} \text{column } i & \text{column } i{+}1 \end{array} \\ \left[\begin{array}{cc} -2\sin(\underline{\omega}_{s,i})\sin(\overline{\omega}_{s,i}) & \sin(\underline{\omega}_{s,i} + \overline{\omega}_{s,i}) \\ \sin(\underline{\omega}_{s,i} + \overline{\omega}_{s,i}) & -2\cos(\underline{\omega}_{s,i})\cos(\overline{\omega}_{s,i}) \end{array} \right] \end{array}, \quad \forall i \in \{1, \ldots, n_x - 1\} \tag{2-48}$$

Adapting regions for disturbances

Fu and Mazo Jr. (2018b) introduces another extension to the conic regions necessary to deal with additive disturbances. Such perturbations are not relative to the norm of the state. Therefore, the same disturbance can have a substantial effect close to the origin while being insignificant when the system is in a state far from the equilibrium. As a result, using an isotropic covering would result in a very conservative mapping, especially when the state is not close to the origin. By adding homocentric N-spheres as shown in figure 2-10, the effect of the disturbance on the system is more similar within regions, which reduces the conservatism.

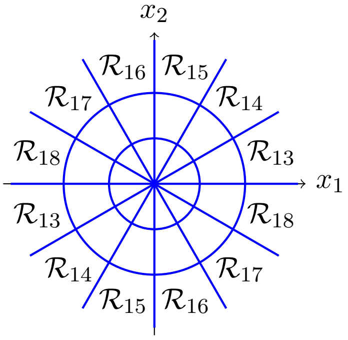


Figure 2-10 Partitioning of a $\mathbb{R}^2$ state-space with the combination of Isotropic cones and homocentric circles.

These homocentric N-spheres are formed by adding a lower and upper bound on the norm of the state in each region. The lower bound radius of the regions in the first N-sphere is set to $\underline{W}_s = 0$. The upper bound radius of the largest homocentric N-sphere can be set to ∞ to produce a partitioning that spans the entire state-space. The resulting regions can be defined as:

$$\mathcal{R}_s = \left\{ \boldsymbol{x} \in \mathbb{R}^{n_x} \;\middle|\; \boldsymbol{x}^{\mathsf{T}} \boldsymbol{V}_{s,i} \boldsymbol{x} \geq 0, \quad \forall i \in \{1, \dots, n_x - 1\}, \quad \underline{W}_s \leq |\boldsymbol{x}| < \overline{W}_s \right\} \tag{2-49}$$

This can also be rewritten as the intersection of only quadratic inequalities by adding dimension. This extension will also be used later on to take the absolute threshold of the mixed triggering condition into account as this has a similar effect and can, for this purpose, be regarded as a disturbance on the system (Gleizer and Mazo Jr., 2020b).

2-6-2 Homogeneous systems

Different methods are required for partitioning the state-space of more general homogeneous systems in a smart fashion. The objective remains the same: minimising the range of inter-event times within each region while dividing the space into as few regions as possible. To achieve this, the state-space is not partitioned a priori, where the space is divided based on some pattern after which the range of corresponding inter-event time is computed, but a posteriori, where the range of inter-event times is partitioned and the regions that correspond to these times are derived afterwards. The boundaries of these regions that correspond to a range of inter-sapling times are formed by isochronous manifolds, which are defined as:

Definition 2.7 (Isochronous Manifolds (Anta and Tabuada, 2012)):
The set $M_{\tau_\star}$ of states that result in the same inter-event time $\tau_\star$ is called an isochronous manifold of time $\tau_\star$

$$M_{\tau_\star} = \{\boldsymbol{x} \in \mathbb{R}^n : \tau(\boldsymbol{x}) = \tau_\star\}$$

Using the region in between isochronous manifolds as a partitioning of the state-space provides an optimal way to achieve the objective. The problem is that it is generally not possible to derive the true isochronous manifolds since the computation would require an analytical solution of the system (Delimpaltadakis and Mazo Jr., 2021b). Getting around this issue involves approximating the isochronous manifolds in such a manner that the guarantees are preserved. See Anta and Tabuada (2012); Delimpaltadakis and Mazo Jr. (2021b) for the development of different techniques.

2-6-3 Periodic Event-Triggered systems

For ETC systems, state-dependent sampling techniques are not useful to pre-compute the next sampling time. However, as mentioned before, the idea can also be applied to construct an abstraction of the triggering behaviour of the system. This abstraction can then be used to schedule the communications needed when the system triggers. For PETC, such abstractions are developed in Fu and Mazo Jr. (2018b) based on partitioning the state-space a priori with an isotropic covering.

As discussed earlier, partitioning the state-space a posteriori based on time instead of space scales much better as the dimension of the state-space increases. Using a partitioning over time is all the more beneficial for a PETC as the set of possible inter-event times is finite. For a linear PETC system, using a partitioning based on time allows for the construction of an exact abstraction without any added uncertainty. Gleizer and Mazo Jr. (2020a) introduces such PETC traffic abstractions that partition the state-space based on the resulting inter-event time.

Consider the LTI system with a quadratic triggering condition introduced in section 2-4, however instead of implementing it as a CETC system, it is implemented as a PETC[2]. The triggering condition is thus no longer monitored continuously but only periodically with sampling interval h. The resulting triggering times t_i are defined by:

$$t_0 = 0, \quad t_{i+1} = \inf \left\{ t = kh > t_i,\, k \in \mathbb{N} \;\middle|\; \begin{array}{l} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} > 0 \\ \vee\; t - t_i = \overline{k}h \end{array} \right\} \tag{2-50}$$

where $\overline{k}$ is the maximum number of periods after which the system is forced to trigger. As a result of the periodic monitoring of the triggering function in a PETC system, the set of possible inter-event times is no longer infinite but consists of a finite number of possible times at which the system can trigger. This makes it possible to construct a finite number of regions that only correspond to a single inter-event time.

Using the PETC triggering condition and the explicit formula for the inter-event time of a LTI system from equation (2-36) a necessary condition for $1 \leq \underline{k} \leq \overline{k}$ to be a lower bound of the inter-event time $\tau(\boldsymbol{x})$ at state $\boldsymbol{x}$ for a PETC system can be expressed as

$$\boldsymbol{x}^{\mathsf{T}} \boldsymbol{\Phi}(kh) \boldsymbol{x} \leq 0, \quad \forall k \in \{1, 2, \ldots, \underline{k}\} \tag{2-51}$$

In contrast with the case of a CETC system, this condition does not contain a matrix function since the triggering condition no longer has to be verified at an infinite number of possible moments in time, but only at the finite number of moments the system can trigger. This means that it is no longer necessary to compute the bound on the maximum allowable inter-event time using a convex embedding but can be computed exactly.

These two effects of switching from a CETC to a PETC make it possible to build abstractions that exactly reproduce the triggering behaviour of the system.

[2] The same triggering condition does not necessarily guarantee the stability of the resulting closed-loop system.

2-7 Decentralised triggering conditions

Consider the case of a system with a central controller and multiple sensor nodes which are not situated in the same location but are physically distributed. It is often more practical for all parts of the system to communicate with each other over a wireless network.

Checking a centralised triggering condition instead of using periodic control will reduce the usage of resources at the central controller. However, it will do nothing to decrease the burden on the sensor nodes and communication link, as monitoring a centralised trigger condition requires the measurements of all sensors to be constantly transmitted to the controller. This leads to high usage of the communications links, which is unnecessary to meet the performance requirements.

Using decentralised triggering conditions makes it possible to reduce the use of these resources. Rather than monitoring a single triggering condition at the central controller, which uses its knowledge of the entire model output, multiple triggering conditions are monitored locally, one at each of the different sensor nodes of the system. These decentralised triggering conditions only use the measurements of the sensors located at their node, which eliminates the need to transmit the sensor measurements constantly. The sensor measurements are only transmitted when the triggering conditions are met. Since each of the decentralised triggering conditions only uses a part of the data that the centralised triggering condition exploits, decentralised triggering can never result in fewer triggering events than the centralised setup.

An issue that arises with decentralised triggering is that two sensors can trigger an update arbitrarily close after each other leading to Zeno behaviour and making it impossible to implement the triggering mechanism in real life. The MIET for the centralised triggering condition derived in Tabuada (2007) is no longer valid. The MIET for a decentralised system depends on the local triggering thresholds θ_m and is not guaranteed to exist for all possible combinations of θ's. However, the MIET for a system with a centralised triggering condition can also be interpreted as a lower bound on the inter-event time. It is never necessary to update the control inputs before the centralised MIET has elapsed to guarantee the stability of the system. As such, the easiest method to prevent Zeno behaviour with a decentralised CETC is to enforce the MIET of the centralised triggering condition. If one of the decentralised triggering conditions is exceeded earlier, the system update is performed only after the MIET of the centralised triggering condition has elapsed.

Different decentralised triggering mechanisms will be discussed in the remainder of this section. However, it is essential first to understand how wireless communication affects resource usage.

2-7-1 Wireless communications

Generally, the transceiver used for communication with the controller is responsible for the majority of the power usage at a node. This is especially true if all the sensors at a node measure passively. Therefore, it is important to have an idea of how transmitting and receiving data influences power usage. Broadly speaking, a transceiver can be in 4 modes: Sleep, Idle (Listening), Receive data (RX) or Transmit data (TX). When the transceiver is in Sleep mode, it uses almost no power. However, it also cannot communicate. Idle (Listening) means the radio is turned on and listening for signals. If it detects a signal, it switches to the RX mode, which receives and interprets the incoming communications. The power consumption in the Idle mode ranges from about half to almost the same as in the RX mode. To send signals, the TX mode is activated, which typically uses the most power, ranging from around the same level of RX to substantially higher consumption. (Khader and Willig, 2013; Zhou et al., 2011)

Although transmitting generally consumes more power than receiving, it generally takes more energy to receive a message than to send the same message, especially for shorter messages. When sending, the radio has to be turned on for the exact amount of time it takes to send the message, but when receiving the same message, the radio has to remain powered on for a longer period. First of all, the radio has to be turned on before the message actually arrives since the exact arrival time of the signal is not known due to influences of the environment and the clocks of the sender and receiver being inherently out of sync. Additionally, the radio also stays on a little bit longer as the receiver needs to detect that it has received the entire message and only then can turn off the radio.

For the case of ETC, it is crucial to consider that the transceiver uses a considerable amount of energy while listening for signals in Idle mode, even when it is not receiving anything. Consequently, idle listening time often has the most significant impact on energy consumption (Ye et al., 2002). Switching to a PETC can reduce the idle listening time as this reduces the possible times at which the system can trigger.

2-7-2 Synchronous measurements

Mazo Jr. and Tabuada (2011) builds on the relative triggering condition of Tabuada (2007) by constructing decentralised triggering conditions for each sensor node. The centralised triggering condition is split into a set of local triggering conditions that ensure the stability of the resulting system. This is achieved by constructing the decentralised triggering condition such that one of them will always trigger before or at the same time at which the centralised condition would trigger. Whenever any of the sensor nodes is triggered, all sensor nodes send their updated measurements to the central controller. Since any local triggering condition forces all nodes to send updated information simultaneously, all states of the system are measured synchronously.

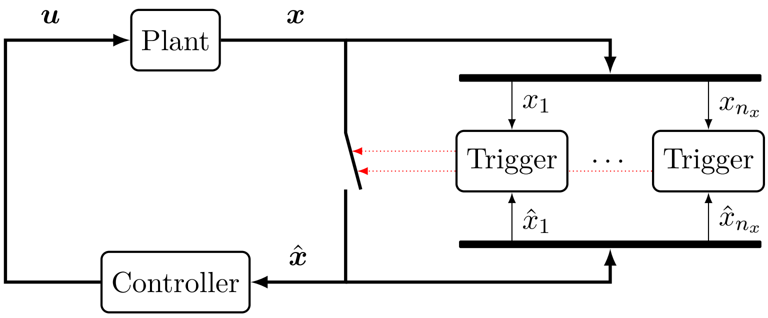


Figure 2-11 Decentralised synchronous ETC control loop.

For simplicity of notation, the case where every node only has one sensor is considered. However, the concept remains the same if certain sensors are co-located at the same node.

The starting point is the centralised relative triggering condition $|\boldsymbol{e}(t)|^2 \geq \mu|\boldsymbol{x}(t)|^2$, which can be rewritten as a sum of the parts from the n_m sensor nodes. The set of all nodes is denoted as $\mathcal{M} = \{1, 2, \ldots, n_m\}$.

$$|\boldsymbol{e}(t)|^2 \geq \mu|\boldsymbol{x}(t)|^2 \quad \Longleftrightarrow \quad \sum_{m=1}^{n_m} \left(e_m^2(t) - \mu x_m^2(t)\right) \geq 0 \tag{2-52}$$

For each node m, a local triggering threshold $\theta_m \in \mathbb{R}$ is introduced. The only constraint on these local thresholds is that they do not add up to more than zero $\sum_m^n \theta_m \leq 0$. Which results in a slightly more conservative triggering condition unless they sum up to zero exactly $\sum_m^n \theta_m = 0$. These local triggering thresholds are introduced to allow for control of how the sensitivity of the local triggering conditions is distributed among the nodes.

$$\sum_{m=1}^{n_m} \left(e_m^2(t) - \mu x_m^2(t)\right) \geq \sum_{m=1}^{n_m} \theta_m \quad \Longleftrightarrow \quad \sum_{m=1}^{n_m} \left(e_m^2(t) - \mu x_m^2(t) - \theta_m\right) \geq 0 \tag{2-53}$$

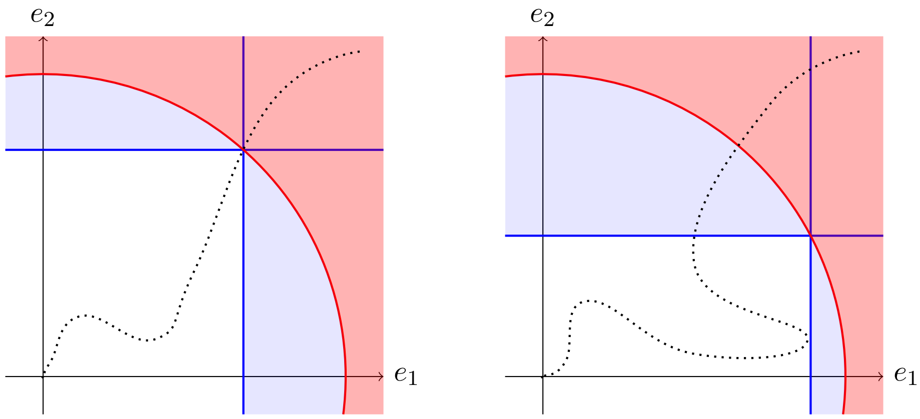
If this centralised triggering condition is met, it is implied that at least one of the parts of the sum has to be positive. These parts only rely on the information of a single node and can thus be used as the local triggering conditions for each of the sensor nodes.

$$\sum_{m=1}^{n_m} \left(e_m^2(t) - \mu x_m^2(t) - \theta_m\right) \geq 0 \quad \Longrightarrow \quad \exists\, m \in \mathcal{M}, \quad e_m^2(t) - \mu x_m^2(t) - \theta_m \geq 0 \tag{2-54}$$

The stability of the closed-loop event-triggered system is therefore ensured by setting the local triggering condition for each node m to

$$e_m^2(t) - \mu x_m^2(t) \geq \theta_m \tag{2-55}$$

Where all nodes will be requested to update their measurements if any of the local triggering conditions are met. However, using this set of triggering conditions is more conservative than the original centralised triggering condition, resulting in more frequent triggering. This conservatism arises as the local triggering conditions enforce all parts of the sum to remain non-positive instead of only the sum of all parts to be non-positive. Adjusting the local triggering thresholds θ_m to distribute the bounds between the different nodes optimally reduces the conservatism introduced by switching to decentralised triggering conditions. Depending on the system and its trajectory, it can even be possible to achieve the exact inter-event times that the centralised triggering condition produces. See figure 2-12 to get an idea of when this occurs and how the set of decentralised triggering conditions results in more conservative bounds compared to the centralised triggering conditions.



(a) Same performance as achieved with the centralised triggering condition is possible.

(b) Impossible to achieve the same performance as with the centralised triggering condition.

Figure 2-12 Consider a system with two states and an absolute central triggering condition. The red line represents the boundary of the central triggering condition, and the red shaded area shows all states where the central condition would trigger the system. The two blue lines represent the decentralised triggering condition which only depends on one of the states. The blue shaded area shows all states where the decentralised triggering condition triggers the system, but the centralised triggering condition would not. The dotted line shows how the measurement error evolves over time from the origin as the inter-event time increases. The decentralised triggering conditions have been optimised to allow for the longest possible inter-event time. The two different paths of the system illustrate that it is not always possible to achieve the same inter-event time as would be possible with a central triggering condition. The same issue arises when using a relative triggering condition.

Mazo Jr. and Tabuada (2011) suggests an algorithm to set the local triggering thresholds θ_m to be optimally distributed for a chosen triggering time in the future. Since this triggering time is not known beforehand, a guess of when the system will trigger is made using a heuristic algorithm. This heuristic approach is not optimal, which leaves room for improvement to reduce the performance gap between the decentralised and centralised triggering approaches. Furthermore, this method does not take feasibility into account and could result in θ' s that will trigger the system much earlier than the intended triggering time. This can occur since it only considers the state of the system at the intended triggering time and does not consider how the system state evolves before the intended triggering time.

Not all triggering conditions can easily be decentralised. A triggering condition can only be decentralised using the technique described above if it does not depend on any products of states measured at different nodes. A quadratic triggering condition can thus only be decentralised if all the cross terms that do not correspond to states located at the same node are zero (Trobinger et al., 2022).

Other methods to reduce the conservatism which arises when decentralising the triggering condition also exist. For instance, Behera and Bandyopadhyay (2015) only checks the central triggering condition whenever one of the local triggering conditions is met and ignores the local trigger if the central triggering condition is not met. This extends the time between control updates of the system at the cost of more frequent communication between the sensor nodes and the controller.

2-7-3 Asynchronous measurements

To further reduce the amount of communication between the sensor nodes and the controller, it can be beneficial to only update the sensors' measurements at the node which violated its local triggering condition instead of updating all measurements synchronously. This approach of asynchronously measuring the states of the system is introduced in Mazo Jr. and Cao (2011, 2012). The specific case of a linear system is also treated in Donkers and Heemels (2012). Additionally, the concept of an asynchronous decentralised triggering mechanism is applied to PETC systems in Fu and Mazo Jr. (2016, 2018a).

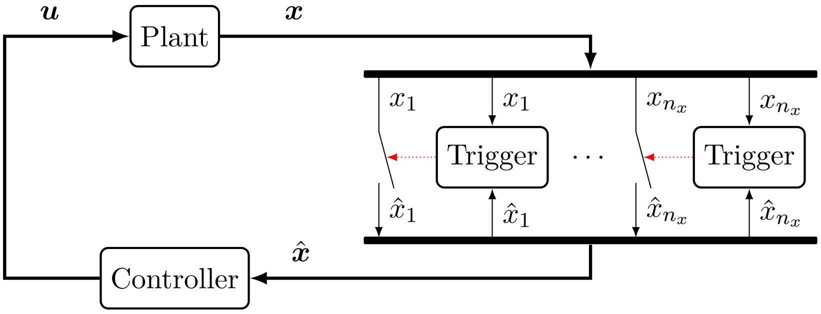


Figure 2-13 Decentralised asynchronous ETC control loop.

Although a further reduction in communication sounds very promising, the resulting decrease in energy usage might not be very productive. An asynchronous implementation will generally result in some sensors triggering considerably less than they would in a synchronised setting. However, other sensors will often trigger more frequently than they would if the system was synchronised. This occurs as the triggering conditions for a node have to be set more conservatively since they rely on older measurements of the state of the remaining nodes. The battery of a sensor node that triggers more often will deplete faster, requiring shorter intervals between services. Since the batteries of the different sensor nodes are no longer in sync, it would be necessary to service them at different times. In many cases, this is not very practical or cost-effective and is likely to result in all nodes being serviced when the first node has a depleted battery. Consequently, the batteries would be replaced earlier than in a synchronised system, even though the total energy consumption of all sensor nodes combined is lower.

Chapter 3

Optimal Decentralised PETC

The end goal is to control a system consisting of a central controller and decentralised sensor nodes, taking into account additive disturbances while using scarce energy and communication resources as efficiently as possible. In such a decentralised setting, communication with the central controller is the single biggest contributor to energy consumption at the nodes in almost all cases. Reducing communication is thus vital in attaining this goal.

Periodically sending updates of the sensor measurements from the nodes to the central controller would be a colossal waste of resources, as these updates are sent regardless of whether there is a need to update the measurements. This is incredibly wasteful as the sampling frequency needs to be chosen to cope with the worst-case scenario. Aperiodic updates can solve this issue by allowing the sampling frequency to adapt to the system's current needs. This can be achieved in either a proactive fashion with Self-Triggered Control (STC) or a reactive manner with Event-Triggered Control (ETC). STC becomes very conservative when there is noise involved, as it always has to assume the worst-case scenario and, therefore, often communicates far more than actually needed. As a result, an event-triggered scheme is more beneficial in this situation. To reduce the communication, the decision to send a new measurement has to be taken at the nodes. As a result, the triggering condition has to be decentralised to depend only on locally available information. Therefore,

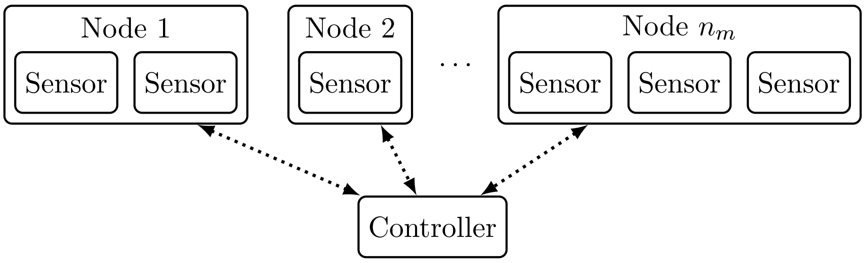


Figure 3-1 Control system with decentralised sensor nodes to measure the state of the system. The nodes communicate with the central controller over a (wireless) network.

the most obvious choice for a trigger mechanism is a decentralised Periodic Event-Triggered Control (PETC).

The periodicity of PETC has three main advantages: It allows for implementation on simple digital hardware without the need for specialised components. Moreover, it dramatically reduces the amount of time that sensor nodes have to listen for potential triggers, as these can no longer occur at any time but only at preset times. Finally, the finite number of outcomes simplifies the modelling and eliminates the need to approximate the future value of the triggering condition for an infinite number of time instances.

Shifting the decision of when to trigger from the centralised controller to the sensor nodes reduces communication. However, it also comes at a cost. The triggering conditions will have to be constructed using less information on the state of the system since only the information from the local sensors is available at the nodes. This change also introduces the question of how to distribute the bounds of the decentralised triggering condition between the different nodes since setting a more tolerant bound at one node will result in having to set tighter bounds on other nodes to preserve the stability guarantee.

The inter-event time accomplished by the centralised triggering condition can only be achieved with decentralised triggering conditions if the bounds are perfectly distributed among the nodes. However, assuming that the distribution of the bounds between nodes is only changed when the nodes communicate with the central controller at the time the system is triggered, it can still be impossible to accomplish the same inter-event time as realised by a centralised triggering system. This occurs when the bound for a node has to be set higher than the level required at the triggering time to ensure that it does not trigger between now and the desired triggering time.

Setting the bounds optimally is vital for reaching the goal since it leads to a longer period between triggering and, thus, less triggering. Moreover, doing so can also reduce the time span in which one of the local triggering conditions can trigger, which therefore reduces the number of instances at which the sensor nodes need to listen for a potential trigger.

The problem of finding this optimal distribution of the bounds between the different nodes is, however, not trivial, especially since the time between receiving the system state and sending out the new triggering thresholds is relatively short. Due to this time limitation, it might be impossible to optimise everything online as such an approach would not be quick enough. Therefore, creating a map in a similar fashion as done with state-dependant sampling is a method to deal with this issue by making it possible to pre-compute the optimisation offline.

Taking the disturbance into account only further complicates the optimisation as the path of the system becomes non-deterministic. There are different approaches to taking this uncertainty into account, either using a stochastic approach or always assuming the worst case. A stochastic approach will yield the longest inter-event time on average. However, it may occasionally also produce a very short inter-event time. Assuming the worst-case and optimising the decentralisation to trigger as late as possible will generally result in shorter average inter-event times. However, it also optimises to prevent any short inter-event times. This is important as a bound on the lowest possible inter-event time can be used to reduce listening times at the nodes since it is not needed to listen for triggers if it is known that the system will not trigger.

An asynchronous implementation would remove the need for periodic listening. However, the added conservatism needed to deal with the use of older information can lead to much more frequent triggering for some nodes, especially in the presence of noise. Additionally, this also introduces the practical issue that certain nodes need service far more frequently than others. Therefore, a synchronous implementation is studied for this use case.

This chapter builds up to the optimal decentralisation in steps. First, the optimisation for a system without disturbances is addressed. Then a region-based approach to reduce the computational complexity of the online optimisation is developed. Lastly, the methodology is expanded to include systems with disturbances.

3-1 The Decentralised Periodic Event-Triggered Control System

Consider an Linear Time-Invariant (LTI) system with state-feedback applied in a sample-and-hold fashion. The controller is centralised, whereas the sensors that measure the state of the system are spread out over the n_m sensor nodes of the system. The set containing all nodes is denoted as $\mathcal{M} = \{1, 2, \ldots, n_m\}$.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{E}\boldsymbol{w}(t) \tag{3-1}$$

$$\boldsymbol{u}(t) = \boldsymbol{K}\hat{\boldsymbol{x}}(t) \tag{3-2}$$

The state is denoted as $\boldsymbol{x}(t) \in R^{n_x}$, the most recent measurement of the state transferred to the controller is $\hat{\boldsymbol{x}}(t) \in R^{n_x}$, the control input is $\boldsymbol{u}(t) \in \mathcal{R}^{n_u}$, $\boldsymbol{w}(t) \in \mathcal{R}^{n_w}$ is an unknown disturbance. $\boldsymbol{A}$ and $\boldsymbol{B}$ are the system matrices, $\boldsymbol{E}$ is the disturbance matrix and, $\boldsymbol{K}$ is the feedback gain matrix.

New measurements of the state are only transferred to the controller whenever the system triggers and is kept constant in between triggering times. The result of this sample-and-hold mechanism given by

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_i), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N} \tag{3-3}$$

where $\{t_i\}_{i\in\mathbb{N}}$ is the set of triggering times generated by the triggering condition $\mathcal{C}(\cdot)$. The sampling error that results from using the last transferred measurement of the state instead of the true value of the state is denoted as $\boldsymbol{e}(t)$.

$$\boldsymbol{e}(t) := \hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t) \tag{3-4}$$

In a decentralised synchronised PETC, the triggering conditions at each node are verified periodically and when the triggering condition at any of the nodes is met, the entire system is triggered. The verification of these conditions occurs with a fundamental sampling period h. Since the triggering condition is only checked periodically, the system can cross the triggering condition between sampling times without triggering the system. The system is also automatically triggered after a maximum inter-event time $\overline{k}$ for modelling purposes.

In the event-triggering mechanism, a decentralised mixed triggering condition is used. This method is presented in Trobinger et al. (2022) and is a generalisation of the decentralisation approach introduced by Mazo Jr. and Tabuada (2011) combined with the adaptation for use in a PETC framework similar to the methods of Heemels and Donkers (2013).

A general version of the mixed triggering condition for a centralised system is given by

$$\boldsymbol{e}(t)^{\mathsf{T}}\boldsymbol{M}\boldsymbol{e}(t) - \boldsymbol{x}(t)^{\mathsf{T}}\boldsymbol{N}\boldsymbol{x}(t) > \beta \tag{3-5}$$

where $\boldsymbol{M}$ and $\boldsymbol{N}$ are symmetric triggering matrices of the appropriate size. The absolute threshold β prevents excessive triggering when the system is close to the origin. The system no longer converges exactly to the origin but stabilises to a set around the origin of which the size is proportional to β. This increases the inter-event time for states close to the origin when the system state is almost at the desired equilibrium and keeps the system from triggering very often to get it to the exact equilibrium. This is especially important in the presence of (non-vanishing) external disturbances, as an arbitrarily small disturbance can cause the system to trigger at almost every time-step when the state gets close enough to the equilibrium (Borgers and Heemels, 2014). The addition of the absolute threshold thus introduces some robustness to noise and reduces resource utilisation. Matrix $\boldsymbol{N} \succeq 0$ needs to be positive semi-definite to prevent repeated triggering of the system when the error $\boldsymbol{e}(t)$ resets to zero as the state measurements are updated.

This mixed triggering condition can be represented as a quadratic triggering condition for use in a PETC mechanism:

$$\begin{bmatrix}\boldsymbol{x}(t)\\ \hat{\boldsymbol{x}}(t)\end{bmatrix}^{\mathsf{T}} \boldsymbol{Q} \begin{bmatrix}\boldsymbol{x}(t)\\ \hat{\boldsymbol{x}}(t)\end{bmatrix} > \beta, \qquad \boldsymbol{Q} = \begin{bmatrix}\boldsymbol{M}-\boldsymbol{N} & -\boldsymbol{M}\\ -\boldsymbol{M} & \boldsymbol{M}\end{bmatrix} \tag{3-6}$$

Such a quadratic condition can represent many different triggering conditions. Heemels et al. (2013) provides a general framework for the stability and performance analysis of such PETC systems with a quadratic triggering condition without an absolute threshold ($\beta = 0$). As discussed in section 2-5-1, these same conditions can also be used for systems with an absolute threshold in the quadratic triggering condition but do result in different stability guarantees.

When the $\boldsymbol{Q}$ is chosen such that the PETC system without additive disturbances $\boldsymbol{w} = 0$ or measurement noise $\boldsymbol{v} = 0$ and without an absolute threshold $\beta = 0$ is Globally Exponentially Stable (GES). Then, the system with disturbances, noise and a (positive) absolute threshold in the triggering condition is still guaranteed to be Globally Exponentially Input-to-State Stable (GEISS) and be $\mathcal{L}_p$ stable with respect to these exogenous inputs (Gleizer and Mazo Jr., 2020b).

The next step is to decentralise the triggering condition of the PETC such that it can be checked at the individual nodes instead of at the central controller. A quadratic triggering condition can only be decentralised if it does not rely on cross terms between states that are not co-located at the same node. This constrains the M and N matrices to the following structure: elements $M_{ii'}$ and $N_{ii'}$ are only nonzero if state i and i' are co-located at the same node.

If these conditions are met, the triggering condition can be decentralised such that a violation of the central triggering condition implies that at least one of the decentralised triggering conditions also has to be violated. The quadratic representation of the decentralised triggering condition for each node m is given by

$$\begin{bmatrix}\boldsymbol{x}(t)\\ \hat{\boldsymbol{x}}(t)\end{bmatrix}^{\mathsf{T}} \boldsymbol{Q}(m) \begin{bmatrix}\boldsymbol{x}(t)\\ \hat{\boldsymbol{x}}(t)\end{bmatrix} > \theta_m, \qquad \boldsymbol{Q}(m) = \begin{bmatrix}\boldsymbol{M}(m)-\boldsymbol{N}(m) & -\boldsymbol{M}(m)\\ -\boldsymbol{M}(m) & \boldsymbol{M}(m)\end{bmatrix}, \quad \forall m \in \mathcal{M} \tag{3-7}$$

where θ_m is a local threshold to tune how the bounds are distributed between the different nodes. The sum of all θ' s must be less than β to ensure that a violation of the central triggering condition implies that at least one of the decentralised conditions is also violated. The matrices $\boldsymbol{M}(m) = \mathbf{J}^m \boldsymbol{M} \mathbf{J}^m$ and $\boldsymbol{N}(m) = \mathbf{J}^m \boldsymbol{N} \mathbf{J}^m$ are only non-zero in the rows and columns that correspond to the sensors located at node m.

The selection matrix $\mathbf{J}^m$ is a diagonal matrix that indicates which sensors are located at node m. It is defined as $\mathbf{J}^m := \mathrm{diag}(j_1^m, \dots, j_{n_x}^m)$ where $j_i^m = 1$ if the sensor measuring state i is located at node m and 0 otherwise. As a sensor can only be located at one node, the sum of all selection matrices should thus be the identity matrix $\sum_m^{n_m} \mathbf{J}^m = \mathbf{I}$.

For example, if there are three states and the sensors measuring the first and last state are located at node 1 while the sensor measuring the second state is situated at node 2, the selection matrices would be

$$\mathbf{J}^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{J}^2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3-8}$$

The triggering times of the resulting decentralised synchronised PETC are then given by the following recursive relation:

$$t_0 = 0, \quad t_{i+1} = \inf \left\{ t = kh > t_i,\ k \in \mathbb{N} \;\middle|\; \begin{array}{l} \exists\, m \in \mathcal{M}, \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^\mathsf{T} \boldsymbol{Q}(m) \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} > \theta_m \\ \vee\ t - t_i = \overline{k}h \end{array} \right\} \tag{3-9}$$

Extending the work of (Mazo Jr. and Tabuada, 2011, Proposition 3.2) to mixed triggering conditions provides the conditions for the decentralised system to have the same stability guarantees as the original centralised system.

Proposition 3.1 (Stability and Performance of the Decentralised System): *Consider an ETC system with a mixed triggering condition* (3-5) *that is guaranteed to satisfy some notion of stability. Suppose that the triggering condition is converted to a decentralised mixed triggering condition* (3-7)*, with θ's chosen such that:*

$$\sum_{m=1}^{n_m} \theta_m \leq \beta \tag{3-10}$$

Then, the resulting decentralised ETC system is guaranteed to satisfy the same notion of stability as the original system.

Proof. If for some $\boldsymbol{x} \in \mathbb{R}^{n_x}$ and some $\hat{\boldsymbol{x}} \in \mathbb{R}^{n_x}$ all decentralised triggering conditions are below the triggering threshold then the following implication holds:

$$\begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix}^\mathsf{T} \boldsymbol{Q}(m) \begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix} \leq \theta_m \quad \forall\, m \in \mathcal{M} \qquad \implies \qquad \begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix}^\mathsf{T} \underbrace{\sum_{m=1}^{n_m} \boldsymbol{Q}(m)}_{=\boldsymbol{Q}} \begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix} \leq \underbrace{\sum_{m=1}^{n_m} \theta_m}_{\leq \beta} \tag{3-11}$$

This shows that the central triggering condition cannot be over its threshold while all decentralised conditions are under their thresholds. That being so, the decentralised system always gets triggered at the same time or before the centralised system and is thus guaranteed to satisfy the same notions of stability as the original centralised system. □

Note that the central and decentralised triggering conditions are not equivalent, as a result, a decentralised triggering mechanism can be more conservative. The θ's determine how the bounds are distributed among the different nodes. The choice of θ's is free as long as $\sum_{m=1}^{n_m} \theta_m \leq \beta$ to guarantee that at least one of the decentralised triggering conditions is triggered at the same time or before the central triggering condition. A higher value for θ_m means the states in node m can deviate further before it will trigger, however, this requires setting tighter bounds on other nodes to maintain the stability guarantee. Tuning these θ's can thus be used to achieve the longest possible inter-event times and to make the decentralised triggering condition as close as possible to being equivalent to the central triggering condition[1].

To achieve this while keeping communications to a minimum, the θ's are only updated at the triggering time when newly updated measurements are available. The value for $\theta_m(t_i)$ is thus sent to node m at time t_i and used in the local triggering condition until the next triggering time t_{i+1} of the system.

In this setup, the communication and application of control inputs are assumed to be instantaneous. Although this is not a very realistic assumption, such imperfections, like delays and data dropouts, can be accounted for by making the triggering condition more conservative. This does come with a performance penalty restricting the possible convergence rate and has a limit. See Tabuada (2007) and Wang and Lemmon (2011) for more information.

As a result of the zero-order hold on the input, the evolution of the state in between triggering times is given by:

$$\boldsymbol{x}(t+kh) = \boldsymbol{\Lambda}(k)\hat{\boldsymbol{x}}(t) + \boldsymbol{\Theta}(k) \tag{3-12}$$

$$\boldsymbol{\Lambda}(k) := \mathrm{e}^{\boldsymbol{A}kh} + \int_0^{kh} \mathrm{e}^{\boldsymbol{A}\tau} d\tau \boldsymbol{B}\boldsymbol{K} \tag{3-13}$$

$$\boldsymbol{\Theta}(k) := \int_0^{kh} \mathrm{e}^{\boldsymbol{A}(kh-\tau)} \boldsymbol{E} w(\tau) d\tau \tag{3-14}$$

3-2 Optimal Decentralisation in the Absence of Disturbances

In the absence of any disturbances $\boldsymbol{w} = 0$ (and any noise) the future state of the system can be computed exactly using the state measured at the last triggering time. This reduces equation (3-12) to

$$\boldsymbol{x}(t+kh) = \boldsymbol{\Lambda}(k)\hat{\boldsymbol{x}}(t) \tag{3-15}$$

With the knowledge of how the state will evolve over time, it is also possible to compute the future value of the triggering conditions:

$$\hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}(k,m)\hat{\boldsymbol{x}} > \theta_m, \quad \forall m \in \mathcal{M} \tag{3-16}$$

[1] The implication in equation (3-11) can be made equivalent in the absence of noise if θ_m is updated in between triggering times.

$$\boldsymbol{\Phi}(k,m) := \begin{bmatrix}\boldsymbol{\Lambda}(k)\\ \mathbf{I}\end{bmatrix}^{\mathsf{T}} \boldsymbol{Q}(m) \begin{bmatrix}\boldsymbol{\Lambda}(k)\\ \mathbf{I}\end{bmatrix} \tag{3-17}$$

Note that using a PETC results in a finite number of possible future triggering times which makes it possible to directly evaluate[2] equation (3-16). This makes it possible to determine when each node will trigger depending on the value of θ_m. The time between two triggering instances referred to as the discrete inter-event time κ, defined as $\kappa := \frac{t_{i+1}-t_i}{h}$, can thus be explicitly computed as:

$$\kappa(\hat{\boldsymbol{x}}, \boldsymbol{\theta}) = \min \left\{ k \in \{1, 2, \dots, \bar{k}\} \;\middle|\; \begin{array}{l} \exists\, m \in \mathcal{M},\ \hat{\boldsymbol{x}}^{\mathsf{T}} \boldsymbol{\Phi}(k,m) \hat{\boldsymbol{x}} > \theta_m \\ \vee\ k = \bar{k} \end{array} \right\} \tag{3-18}$$

To lose as little performance as possible when transferring from a single centralised triggering condition to multiple decentralised triggering conditions, it is paramount to optimise the allocation of bounds by setting the θ's. Note that the relationship between θ and the discrete inter-event time κ is monotone, and a higher θ cannot result in a lower κ.

For a given state where the system is now $\hat{\boldsymbol{x}}$, the future value of the triggering condition $\hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}(k,m)\hat{\boldsymbol{x}}$ can be calculated at every possible triggering time k and for each node m. The future value of the triggering condition is the lowest value that θ_m can be set to for which node m will not trigger at time step k. However, to reach time step k, the value of θ_m also has to be high enough to prevent triggering at all preceding time steps. Therefore, the lowest possible value for θ_m which ensures that node m will not trigger in the first k time steps is given by the highest lowered bound:

$$\underline{\theta}_m(k, \boldsymbol{x}) := \max_{i \in 1,2,\dots,k} \hat{\boldsymbol{x}}^{\mathsf{T}} \boldsymbol{\Phi}(i,m) \hat{\boldsymbol{x}} \tag{3-19}$$

On the other hand, we cannot endlessly keep increasing θ's to prevent triggering as the sum of all θ's cannot be larger than β to preserve the stability guarantees of the system. As such, there will be a point in time when it is no longer possible to prevent the system from triggering while also guaranteeing stability. This can occur at an earlier instance than in the centralised case, which means that achieving the same inter-event times as the centralised triggering condition generates is not always possible (see section 2-7-2 for a more in-depth explanation). The longest time for which it is possible to prevent triggering is given by:

[2] For a Continuous Event-Triggered Control (CETC) system the expression does not simplify and the matrix functional $\boldsymbol{\Phi}$ needs to be computed using bounds (see section 2-6-1).

Proposition 3.2 (Maximum inter-event time):
For any possible choice of θ's which guarantees the stability of the closed-loop decentralised PETC system:

$$\sum_{m=1}^{n_m} \theta_m(k) \leq \beta \tag{3-20}$$

The maximum number of time-steps for which it is possible to prevent the system from triggering is given by the highest k for which it is possible to keep the sum of all lower bounds $\underline{\theta}_m$ under the absolute threshold β:

$$\kappa^*(\hat{\boldsymbol{x}}) = 1 + \max_{k \in 0,1,\ldots,\bar{k}} \quad k \tag{3-21}$$

$$s.t. \qquad \sum_{m=1}^{n_m} \underline{\theta}_m \leq \beta \tag{3-22}$$

The simplest method to achieve the maximum inter-event time is to set θ_m at the lowest value for which node m does not trigger at any of the sampling times before the maximum inter-event time is reached.

Proposition 3.3 (Optimal Decentralisation in the Absence of Noise):
For every node $m \in \mathcal{M}$ set θ_m to:

$$\theta^*_m(t_i) = \max_{k \in 1,2,\ldots,\kappa^*(\hat{\boldsymbol{x}})-1} \hat{\boldsymbol{x}}^{\mathsf{T}}(t_i)\boldsymbol{\Phi}(k,m)\hat{\boldsymbol{x}}(t_i) \tag{3-23}$$

Then, the decentralisation results in an inter-event time of κ^* time steps and is optimal in the sense that no other assignment of θ's would result in longer inter-event times while still ensuring that the decentralised triggering condition triggers before or at the same time as the original triggering condition.

Note that the bounds given by proposition 3.3 are tighter than needed to ensure that the decentralisation retains the stability guarantees, as they generally result in $\sum_{m=1}^{n_m} \theta_m < \beta$. However, this conservatism does not result in the system triggering any earlier.

For the case without the presence of any noise, a self-triggered implementation will always be superior as it would require substantially less listening time at the nodes while being able to achieve better performance. This is because the triggering times can be pre-computed precisely based on the centralised condition. Therefore this setup only makes sense in the context of the end goal and serves to further the development towards this goal.

3-3 Off-line Region-based Optimisation

The previous section focuses on optimising the decentralisation online using the current state of the system. As these computations are done in real-time, the optimum has to be computed very quickly to update the decentralised triggering conditions before the next sampling time. Depending on the system, this might not be possible or require expensive hardware, which limits the range of systems on which such a decentralisation can practically be implemented[3].

[3] The computations in the previous section can be improved by making use of the structure. However, this does nothing to resolve the underlying issue of how the solution scales.

The complexity of the online computations can be reduced by pre-computing a mapping from the state-space to the most optimal θ vector. This is done in a similar manner as the state-dependent sampling introduced in Fiter et al. (2012). The state-space is partitioned into a finite number of regions for which the best possible distribution of the triggering condition is derived. With this pre-computed map, the online optimisation is reduced to finding out in which region the current state lies.

As a result of the periodic sampling in a PETC mechanism, there is a range of $\boldsymbol{\theta}$ vectors for each state that will yield the maximum inter-event time. It is thus possible to construct a finite number of regions, each associated with their own $\boldsymbol{\theta}$ vector, that will result in an optimal decentralisation in the absence of noise. However, computing in which region a certain state lies boils down to performing the same computations as the online optimisation. This defeats the purpose of the region-based approach and fails to shift the computational burden of the optimisation from online to pre-computing it offline. It is thus important to partition the state-space into regions such that the computation to locate the region in which a certain state lies can be performed with little effort.

3-3-1 Partitioning the state-space

The goal is to construct a partitioning of the state-space into regions that are as large as possible while having similar optimal $\boldsymbol{\theta}$ vectors. Additionally, the regions need to be defined in such a way that finding out in which region a certain state is located can be done quickly. Using more regions to cover the entire state-space reduces the diversity within the regions and, thereby the conservativeness. However, a larger number of regions increases the burden of the offline computation as the optimisation has to be pre-computed for each region separately. Adapting the partitioning of the state-space to the system dynamics thus results in a less conservative mapping for the same computational effort.

To create such regions, it is useful to look at how the behaviour of the system scales throughout the state-space. The system dynamics themselves are linear. However, the mixed triggering condition is non-homogeneous, which means that the behaviour cannot be described with some type of multiplicative scaling. However, as discussed in section 2-6, the closed-loop system can be rendered homogeneous by embedding it in a system with one additional dimension. The higher dimensional state-space is denoted as $\boldsymbol{x}'$ with the first n_x states exactly the same as $\boldsymbol{x}$. The newly added state remains at the same value at all times.

With this additional dimension, the decentralised (mixed) triggering condition can be represented as a homogeneous function:

$$\begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \\ 1 \end{bmatrix}^{\mathsf{T}} \underbrace{\begin{bmatrix} \boldsymbol{Q}(m) & 0 \\ 0 & -\theta_m \end{bmatrix}}_{\widetilde{\boldsymbol{Q}}(m,\theta)} \begin{bmatrix} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}(t) \\ 1 \end{bmatrix} \geq 0, \qquad \sum_{m=1}^{n_m} \theta_m \leq \beta \tag{3-24}$$

The future values of the triggering conditions are now also homogeneous quadratic inequalities, node m does not trigger at time k when the following inequality is satisfied:

$$\begin{bmatrix}\hat{\boldsymbol{x}}(t)\\1\end{bmatrix}^{\mathsf{T}}\underbrace{\begin{bmatrix}\boldsymbol{\Phi}(k,m) & 0\\0 & -\theta_m\end{bmatrix}}_{\widetilde{\boldsymbol{\Phi}}(k,m,\theta)}\begin{bmatrix}\hat{\boldsymbol{x}}(t)\\1\end{bmatrix} \leq 0 \tag{3-25}$$

The resulting extended system scales linearly, consequently, any state $\boldsymbol{x}'$ on the same homogeneous ray going through the origin results in the same inter-event time (theorem 2.2). However, the optimal decentralisation depends not on when the system is triggered but on where in the state-space the triggering condition will be violated. Given the initial state, the state at which the system will trigger scales according to the following relationship:

Proposition 3.4 (Scaling law for triggering state of the system):
Consider a homogeneous system of degree ψ and a triggering function $\mathcal{C}(\cdot)$ homogeneous of degree ϕ. The state at which the system triggers implicitly defined by the triggering function scales according to:

$$\boldsymbol{x}'(\tau(\lambda\boldsymbol{x}_0'),(\lambda\boldsymbol{x}_0',\lambda\boldsymbol{e}_0)) = \lambda\boldsymbol{x}'(\tau(\boldsymbol{x}_0'),(\boldsymbol{x}_0',\boldsymbol{e}_0)) \tag{3-26}$$

Where $\boldsymbol{x}'(t,(\boldsymbol{x}_0',\boldsymbol{e}_0))$ is the state of the system at time t given the initial condition of the system $\boldsymbol{x}'(0)=\boldsymbol{x}_0'$, $\boldsymbol{e}(0)=\boldsymbol{e}_0=\boldsymbol{0}$.

Proof. See Appendix □

Note that both the degrees of homogeneity of the system and the triggering function do not influence the scaling. The degree of homogeneity of the triggering function has no influence as the scaling of the state and error scale cancel each other out. The degree of homogeneity of the system disappears as the scaling of the inter-event time counteracts the time scaling of the state.

Since the PETC system equation (3-1) is LTI, there is no time dilation, and the inter-event times are the same for all states of the system on the same homogeneous ray (see theorem 2.2). Combining this with the scaling behaviour of the triggering state makes it possible to derive how the maximum achievable Inter-Event Time (IET) of the decentralised system scales and how the optimal decentralisation changes.

Proposition 3.5 (Scaling law for the Optimal Decentralisation):
Consider the PETC system (3-1) *to* (3-3) *and* (3-9)*. The maximum achievable inter-event time and optimal decentralisation scale according to:*

$$\kappa^*(\lambda\boldsymbol{x}') = \kappa^*(\boldsymbol{x}'), \qquad \theta_m^*(\lambda\boldsymbol{x}_0') = \lambda^2\theta_m^*(\boldsymbol{x}_0') \tag{3-27}$$

Proof. See Appendix □

This understanding of how the system behaviour scales throughout the state-space can now be used to construct regions for which the optimum is very much alike. As all derived scaling laws are only valid along the homogeneous rays of the system, an isotropic covering in n_x+1

dimensions of the extended system is a logical starting point. This is constructed using the projection-based approach of Mazo Jr. et al. (2018) (see section 2-6-1).

However note that, as our original system is a hyperplane in the extended state-space, none of the states of the original system in a region in fact lie on the same homogeneous ray. The regions consist of a bundle of homogeneous rays that are similar because they are close to each other, the further the real states get from the origin the closer they are to lying on the same homogeneous ray. This happens as the effect of the absolute threshold in the triggering condition and the additive disturbance diminishes for states further away from the origin. A hyperplane within a partitioning that is constructed using the projection-based approach cannot capture such effects that depend on the distance from the origin evenly in all directions.

The solution to this problem is to use an isotropic covering in n_x dimensions and combine this with homocentric circles as shown in figure figure 3-2 (see section 2-6-1). The idea is introduced in Fu and Mazo Jr. (2018b) to deal with disturbances, however, the absolute threshold has a similar effect and can also be effectively mapped using such a partitioning. Although this may look very different than the starting point of an isotropic covering in n_x+1 dimensions, note that such homocentric circles are the same as using isotropic covering with spherical coordinates to partition the additional dimension.

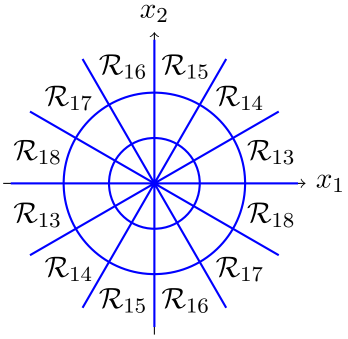


Figure 3-2 An example of the resulting partitioning for a 2-dimensional system.

With this partitioning, it is still very easy to compute which region a state belongs to as this boils down to computing the angular coordinates and distance from the origin associated with the state. The region s can then be defined as the intersection of multiple quadratic inequalities:

$$\mathcal{R}_s = \left\{ \boldsymbol{x}' \in \mathbb{R}^{n_x+1} \,\middle|\, \boldsymbol{x}'^{\mathsf{T}} \boldsymbol{V}_{s,i}\, \boldsymbol{x}' \geq 0, \quad \forall i \in \{1, \dots, n_x+1\} \right\} \tag{3-28}$$

where the first $n_x - 1$ quadratic inequalities define the conical covering as defined in equation (2-48) and the last two define the homocentric N-spheres:

$$\boldsymbol{V}_{s,n_x} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\underline{W}_s^2 \end{bmatrix}, \qquad \boldsymbol{V}_{s,n_x+1} = \begin{bmatrix} -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \overline{W}_s^2 \end{bmatrix} \tag{3-29}$$

where $\underline{W}_s$ and $\overline{W}_s$ are, respectively, the lower and upper bound of the distance from the origin for region s. The lower bound of the regions in the first N-sphere is set to $\underline{W}_s = 0$. The upper bound of the regions further from the origin needs to be chosen such that the system state always stays within the subset that the partitioning covers. This cannot be set to ∞ since the optimal decentralisation scales along the homogeneous rays of the system (see proposition 3.5)

3-3-2 Optimising over Regions

With the state-space partitioned into regions, the next step is to find the optimal local thresholds $\boldsymbol{\theta}$ that are valid for all states within a region. This is done by leveraging a lossy S-Procedure which results in the following theorem:

Theorem 3.1 (Regional lower bound for θ to prevent the unperturbed system from triggering):
Consider the PETC system (3-1) *to* (3-3) *and* (3-9) *without a disturbance* $\boldsymbol{w} = 0$*. If there exist scalars* $\epsilon_i \geq 0,\ \forall i \in \{1, \dots, n_x + 1\}$ *such that the following Linear Matrix Inequality (LMI) holds:*

$$\widetilde{\boldsymbol{\Phi}}(k, m, \underline{\theta}_m) + \sum_{i}^{n_x+1} \epsilon_i \boldsymbol{V}_s \preceq 0 \tag{3-30}$$

then $\underline{\theta}_m(k, s)$ *is a lower bound of the local threshold for which node* m *will not to triggering after* k *time-steps starting from any state in region* s

Proof. If the conditions of the theorem hold then the application of the lossy S-Procedure directly results in the following implication for all $\boldsymbol{x}'$:

$$\boldsymbol{x}'^{\mathsf{T}} \boldsymbol{V}_{s,i}\, \boldsymbol{x}' \geq 0, \quad \forall i \in \{1, \dots, n_x + 1\} \quad \Longrightarrow \quad \boldsymbol{x}'^{\mathsf{T}} \widetilde{\boldsymbol{\Phi}}(k, m, \theta)\, \boldsymbol{x}' \leq 0 \tag{3-31}$$

The second part of the proof shows that this implication is equivalent to the same implication constrained to only the hyperplane that represents the original (non-homogeneous) system in the extended space:

$$\begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}^{\mathsf{T}} \boldsymbol{V}_{s,i} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}' \geq 0, \quad \forall i \in \{1, \dots, n_x + 1\} \quad \Longrightarrow \quad \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}^{\mathsf{T}} \widetilde{\boldsymbol{\Phi}}(k, m, \theta) \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \leq 0 \tag{3-32}$$

This is essentially the same as the proof of the non-homogeneous S-Procedure (see Pólik and Terlaky (2007) for different proofs). For ease of notation, the matrix in LMI condition of the theorem is denoted as $\boldsymbol{P}$.

$$\boldsymbol{P} \coloneqq \widetilde{\boldsymbol{\Phi}}(k, m, \theta) + \sum_{i}^{n_x+1} \epsilon_i \boldsymbol{V}_s \tag{3-33}$$

To show that both implications are equivalent it is demonstrated that the following inequalities are equivalent:

$$\boldsymbol{x}'^{\mathsf{T}} \boldsymbol{P} \boldsymbol{x}' \leq 0, \quad \forall \boldsymbol{x}' \in \mathbb{R}^{n_x+1} \iff \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}^{\mathsf{T}} \boldsymbol{P} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \leq 0, \quad \forall \boldsymbol{x} \in \mathbb{R}^{n_x} \tag{3-34}$$

The fact that the first inequality implies the second is trivial as $\begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \in \mathbb{R}^{n_x+1}$. The inverse can by established by contradiction. Given that the second inequality holds, suppose there exists an $\boldsymbol{x}' = \begin{bmatrix} \boldsymbol{x} \\ y \end{bmatrix}$ such that the first inequality does not hold:

$$\boldsymbol{x}'^{\mathsf{T}} \boldsymbol{P} \boldsymbol{x}' > 0 \tag{3-35}$$

If the last coordinate of $\boldsymbol{x}'$, y is zero than there also exists a $\boldsymbol{x}'$ where y is not zero since $\boldsymbol{x}'^{\mathsf{T}} \boldsymbol{P} \boldsymbol{x}'$ is continuous. Dividing $\boldsymbol{x}'$ by y gives the following inequality:

$$\begin{bmatrix} \boldsymbol{x} \\ y \end{bmatrix}^{\mathsf{T}} \boldsymbol{P} \begin{bmatrix} \boldsymbol{x} \\ y \end{bmatrix} > 0 \iff \begin{bmatrix} \frac{\boldsymbol{x}}{y} \\ 1 \end{bmatrix}^{\mathsf{T}} \boldsymbol{P} \begin{bmatrix} \frac{\boldsymbol{x}}{y} \\ 1 \end{bmatrix} > 0 \tag{3-36}$$

Which contradicts the original assumption and thus proves the equivalence.

□

Note that the embedding of our original system in a higher dimensional system does not introduce additional conservatism. However, the use of the *lossy* S-Procedure does mean that the regional lower bound is not exact. The procedure is lossy as the regions are defined as the intersection of *multiple* quadratic inequalities. As a result, there is no guarantee that it is possible to achieve the same performance as the online optimisation by increasing the number of regions.

Theorem 3.1 can then be used to find the lowest regional lower bound for which node m will not trigger at time-step k starting from any state in region s. These can then be combined in the same manner as in the online optimisation to create a map from every region s to the optimal decentralisation of the triggering condition $\boldsymbol{\theta}^*$.

3-4 Decentralisation in the Presence of Disturbances

In the presence of a disturbance, it is no longer possible to compute the future values of the triggering conditions exactly since the future value of the triggering condition for node m at time k now also depends on the realisation of the disturbance $\boldsymbol{w}$:

$$\mathcal{C}(\hat{\boldsymbol{x}}(t), \boldsymbol{w}(t, \ldots, t+kh), k, m) = \begin{bmatrix} \boldsymbol{x}(t+kh) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix}^{\mathsf{T}} \boldsymbol{Q}(m) \begin{bmatrix} \boldsymbol{x}(t+kh) \\ \hat{\boldsymbol{x}}(t) \end{bmatrix} - \theta_m > 0 \tag{3-37}$$

One way to deal with this issue would be to compute the expectation of the future value of the triggering condition, which makes it possible to optimise the average inter-sampling time of the system. Another approach is to find an upper bound of the future value of the triggering condition. With this upper bound, the worst-case inter-sampling time can be optimised to prevent the system from triggering in rapid succession. The second option is chosen in this thesis as this leads to a guarantee that the system cannot trigger before a certain time and thus opens up the possibility of saving resources by not having to listen whether other nodes in the system have triggered.

The worst-case scenario of the perturbation using the combination of the techniques described in Fiter et al. (2015) and Fu and Mazo Jr. (2018b). Due to the disturbance, it is impossible to compute the future value of the triggering condition as $\boldsymbol{\Theta}(k)$ is non-deterministic. However, it is possible to determine an upper bound on the future value of the triggering condition $\overline{\mathcal{C}}$ which does not depend on the realisation of the disturbance $\boldsymbol{w}$. This upper bound can then be used as a substitute for the future value of the triggering condition since:

$$\overline{\mathcal{C}}(\hat{\boldsymbol{x}}(t), k, m) < 0 \implies \mathcal{C}(\hat{\boldsymbol{x}}(t), k, m, \boldsymbol{w}(t, \dots, t + kh)) < 0 \tag{3-38}$$

Some knowledge of the disturbance $\boldsymbol{w}$ is needed to construct such an upper bound. It is therefore assumed that an upper bound of the disturbance is known:

$$\exists \mathcal{W} \geq 0, \qquad \|\boldsymbol{w}\|_\infty \leq \mathcal{W} \tag{3-39}$$

This assumption follows Fu and Mazo Jr. (2018b) and unlike Fiter et al. (2015) does not assume that the disturbance vanishes as the system converges. Using this assumption, the upper bound of the future value of the triggering condition is given by:

Proposition 3.6 (Upper Bound on the future value of the triggering condition): *Consider the PETC system* (3-1) *to* (3-3) *and* (3-9)*. If there exist a scalar* $\mu \geq 0$ *and a symmetric matrix* $\boldsymbol{\Psi}$ *such that*

$$\boldsymbol{Q}_1(m) + \boldsymbol{\Psi} \preceq \mu \mathbf{I} \tag{3-40}$$

Then the future value of the triggering condition $\mathcal{C}(\hat{\boldsymbol{x}}, \boldsymbol{w}, k, m)$ *is upper bounded by*

$$\overline{\mathcal{C}}(\hat{\boldsymbol{x}}, k, m) = \hat{\boldsymbol{x}}^\mathsf{T}\Big(\boldsymbol{\Phi}_1(k, m) + \boldsymbol{\Phi}_2(k, m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^\mathsf{T}(k, m)\Big)\,\hat{\boldsymbol{x}} + \boldsymbol{\Phi}_3(k)\mathcal{W}^2 - \theta_m \tag{3-41}$$

With

$$\boldsymbol{Q}(m) = \begin{bmatrix} \boldsymbol{Q}_1(m) & \boldsymbol{Q}_2(m) \\ \boldsymbol{Q}_2^\mathsf{T}(m) & \boldsymbol{Q}_4(m) \end{bmatrix} \tag{3-42}$$

$$\boldsymbol{\Phi}_1(k, m) = \boldsymbol{\Lambda}^\mathsf{T}(k)\boldsymbol{Q}_1(m)\boldsymbol{\Lambda}(k) + \boldsymbol{\Lambda}^\mathsf{T}(k)\boldsymbol{Q}_2(m) + \boldsymbol{Q}_2(m)\boldsymbol{\Lambda}(k) + \boldsymbol{Q}_4(m) \tag{3-43}$$

$$\boldsymbol{\Phi}_2(k, m) = \boldsymbol{\Lambda}^\mathsf{T}(k)\boldsymbol{Q}_1(m) + \boldsymbol{Q}_2(m) \tag{3-44}$$

$$\boldsymbol{\Phi}_3(k) = kh\mu\lambda_{max}\left(\boldsymbol{E}^\mathsf{T}\boldsymbol{E}\right) f_{\boldsymbol{A}}(k) \tag{3-45}$$

$$f_{\boldsymbol{A}}(k) = \begin{cases} \dfrac{e^{kh\lambda_{max}(\boldsymbol{A}+\boldsymbol{A}^\mathsf{T})} - 1}{\lambda_{max}(\boldsymbol{A}+\boldsymbol{A}^\mathsf{T})}, & \textit{if } \lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^\mathsf{T}\right) \neq 0 \\ kh, & \textit{otherwise} \end{cases} \tag{3-46}$$

Proof. See Appendix □

This upper bound makes it possible to find the lower bound $\underline{\theta}_m(k, \hat{\boldsymbol{x}})$ that θ_m can be set at to prevent node m from triggering at time k. This bound can be used to optimise the decentralisation online in the same way as described in section 3-2. However, it can also be used to construct an offline map in the same manner as for the system without a disturbance. A Lossy S-Procedure can once again be used to construct conditions for an entire region $\mathcal{R}_s$ instead of just a single state $\hat{\boldsymbol{x}}$. Additionally, the upper bound on the future value of the triggering condition $\hat{\boldsymbol{x}}$ is linearised with a Schur complement. This results in the following theorem:

Theorem 3.2 (Regional lower bound for θ to prevent the system from triggering): *If the hypothesis in proposition 3.6 holds and there exist scalars $\epsilon_i \geq 0$, $\forall i \in \{1, \ldots, n_x + 1\}$ such that the following LMI holds:*

$$\begin{bmatrix} -\boldsymbol{\Psi} & \boldsymbol{\Phi}_2^\intercal(k,m) & \mathbf{0} \\ \boldsymbol{\Phi}_2(k,m) & \boldsymbol{\Phi}_1(k,m) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_3(k)\mathcal{W}^2 - \underline{\theta}_m \end{bmatrix} + \sum_i^{n_x+1} \epsilon_i \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V}_s \end{bmatrix} \preceq 0 \tag{3-47}$$

then $\underline{\theta}_m(k,s)$ is a lower bound of the local threshold for which node m will not to triggering after k time-steps starting from any state in region s for any realisation of the disturbance $\boldsymbol{w}$

Proof. See Appendix □

This condition can then be used in exactly the same way as before to construct a map from each region $\mathcal{R}_s$ to a vector of local thresholds $\boldsymbol{\theta}$ that optimise the guaranteed IET.

Chapter 4

Examples

This chapter will give some examples of the techniques in action to illustrate the theory and showcase the performance. First, a simple 2-dimensional system is shown as this allows for superior visualisations. Then more complex and frequently used example of a Linear Batch Reactor is demonstrated.

4-1 Academic 2-Dimensional system

Consider an Linear Time-Invariant (LTI) plant and state-feedback controller as defined in equations (3-1) and (3-2) with the following system matrices from Tabuada (2007):

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}, \qquad \boldsymbol{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad \boldsymbol{E} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad \boldsymbol{K} = \begin{bmatrix} 1 \\ -4 \end{bmatrix} \tag{4-1}$$

The system is split into two nodes, each measuring one state of the system. A disturbance acts on the system for which a bound is known $\mathcal{W} \leq 2$. The Periodic Event-Triggered Control (PETC) mechanism is given by equation (3-9), where the triggering condition is checked with a sampling interval of $h = 0.05$ second. As the triggering condition, a decentralised version of the mixed condition given in equation (2-23) is used. With this, the decentralised quadratic triggering matrix is given by:

$$\boldsymbol{Q}(m) = \begin{bmatrix} (1-\mu)\mathbf{J}^m & -\mathbf{J}^m \\ -\mathbf{J}^m & \mathbf{J}^m \end{bmatrix} \tag{4-2}$$

where the relative threshold is set to $\mu = 0.04$ and the absolute threshold to $\beta = 0.05$. The maximum inter-event time is set at $\overline{k} = 15$. Higher Inter-Event Times (IETs) would be possible without this limit, however only in very close proximity to the origin [1]. With this setup, the system is Globally Exponentially Input-to-State Stable (GEISS) with a decay

[1] Without the absolute threshold (β) the system would have a natural maximum of $\overline{k} = 6$

rate of at least $\rho = 0.2002$. Keep in mind that characteristics of the absolute threshold β, additive disturbance $\boldsymbol{w}$ and measurement noise $\boldsymbol{v}$ do influence the size of the region around the equilibrium that the system is guaranteed to converge to.

The triggering performance of the central triggering condition is compared to the online optimisation given in proposition 3.3 and to different versions of the offline region-based optimisations discussed in section 3-3. Both the online and region-based are optimised twice: Once assuming that there is no disturbance and taking the disturbance into account by using an upper bound.

For the region-based optimisations, the state-space is partitioned in five different ways to illustrate the effects of increasing the number of regions. Only a region with a radius of 10 around the origin is partitioned. The four smaller divisions are used to showcase how increasing the number of cones and circles separately affects the performance. The last partitioning has a much finer granularity to understand the attainable performance better. An overview of the different partitionings is given in the following table:

	Radius	**Isotropic cones**	**Homocentric circles**	**Number of Regions**
⋈ 12 ○ 10	10	12	10	120
⋈ 12 ○ 20	10	12	20	240
⋈ 24 ○ 10	10	24	10	240
⋈ 24 ○ 20	10	24	20	480
⋈ 60 ○ 100	10	60	100	6000

Table 4-1 The different ways that the state-space is partitioned into regions

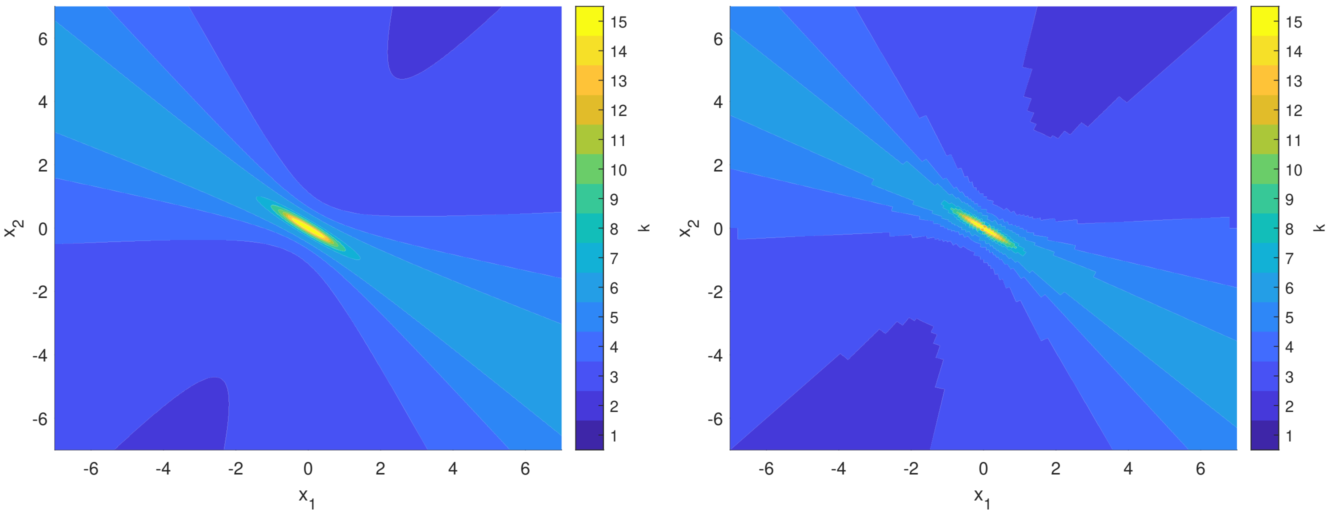
4-1-1 In the Absence of Disturbances

First, the performance is studied when there is no disturbance acting on the system. Figure 4-1a shows the optimal IET that is possible for each state of the system with the decentralised triggering function. The Minimal Inter-Event Time (MIET) that can be guaranteed for each region when using the finest partitioning is shown next to it in figure 4-1b. A simulation of the system comparing the performance of the region-based method against the central triggering condition is shown in figure 4-2.

To get an idea of how the triggering mechanism performs throughout the state-space, the system is simulated for 10000 random states uniformly distributed in a circle with a radius of 10 around the origin. For every initial state, the IET is determined for the case without any disturbances $\boldsymbol{w} = 0$. Table 4-2 provides a summary of the results.

When correctly assuming the absence of a disturbance, the decentralised triggering condition using the online optimisation is capable of achieving almost exactly the same IETs as the central triggering condition. There is just a single initial condition in the 10000 simulations for which the system triggers one time-step early due to a numerical error. The Region-based offline optimisations clearly improve as the number of regions increases. For the largest partitioning, almost 84% of the states result in the same IET, and the remaining 16% trigger only one time-step earlier. When taking a disturbance into account when optimising the decentralisation, the performance declines as the added conservatism reduces the average IET. The coarser partitionings not taking into account a disturbance trigger earlier for some states close to the origin than when using the same partitioning takes the disturbance into account. This happens as the regions are quite large, and the magnitude of the $\boldsymbol{\theta}$ vector needed for the states further makes it impossible to reach the long Inter-Event Times (IETs) possible around the origin.

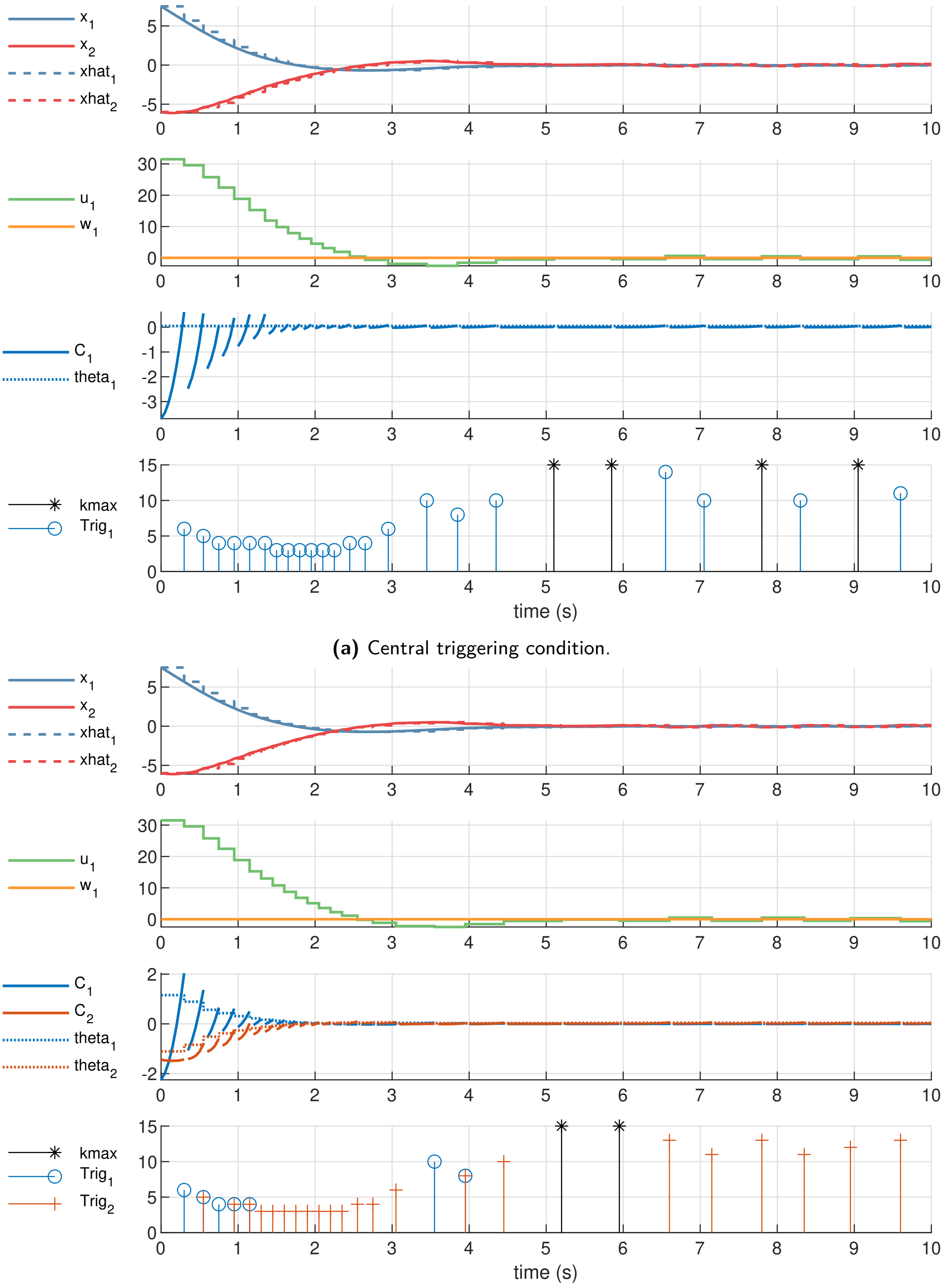
Figure 4-3 reports the IETs aggregated per region to get a better sense of how the performance changes throughout the state-space.



(a) Optimised Online

(b) Map showing the MIET in each region with a partitioning using 60 cones 100 circles

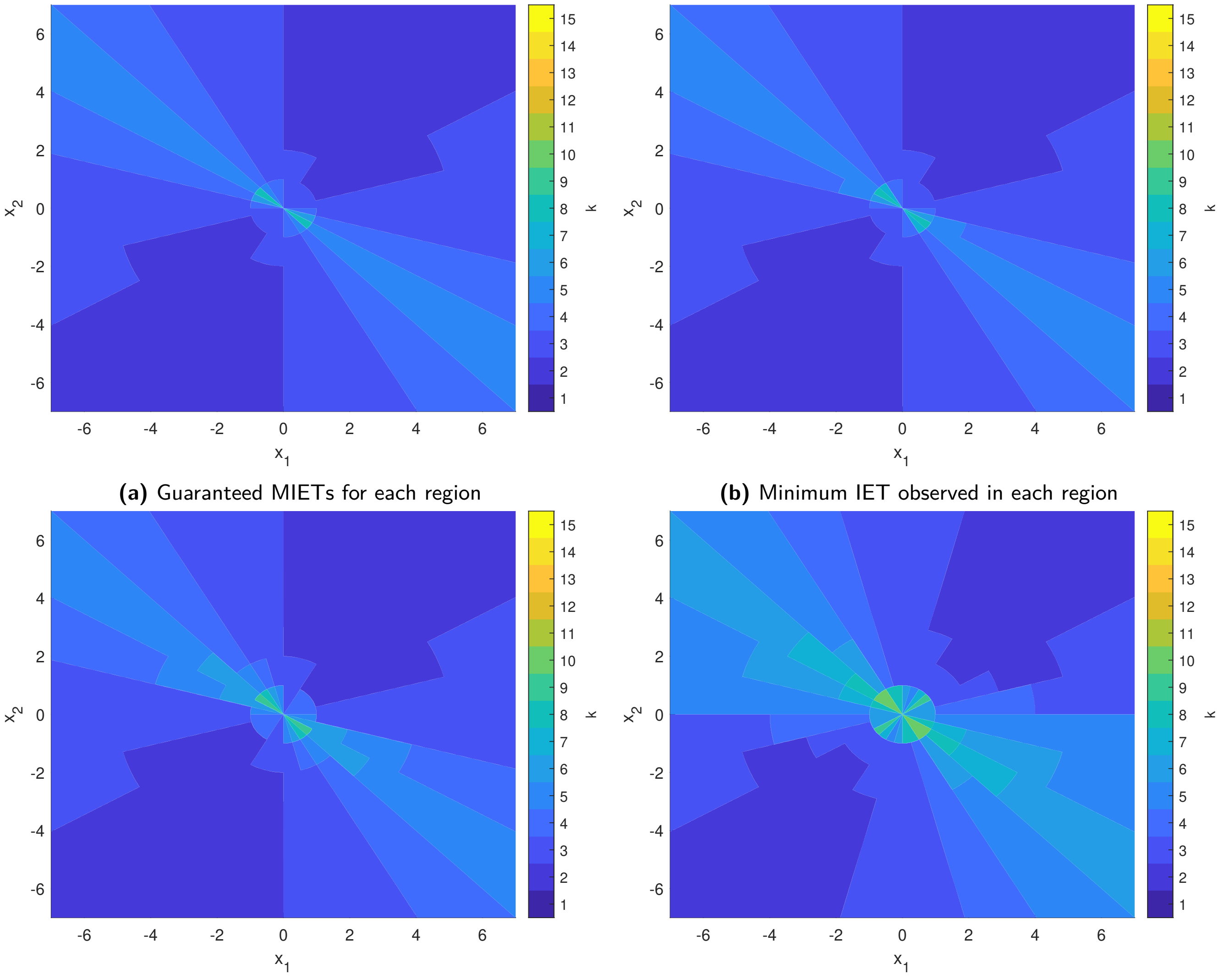
Figure 4-1 Optimised Inter-Event Times (IETs) of the system assuming no disturbances.



(a) Central triggering condition.

(b) System using a map with 60 cones and 100 circles optimised assuming no disturbance.

Figure 4-2 Simulated system response in the absence of disturbances with the initial state at $x_0 = \begin{bmatrix} 7.5 & -6 \end{bmatrix}^\mathsf{T}$. The height of the bars in the fourth plot represents the IET in time-steps.



(a) Guaranteed MIETs for each region

(b) Minimum IET observed in each region

(c) Average IET observed in each region

(d) Maximum IET observed in each region

Figure 4-3 Simulated Inter-Event Times (IETs) of the system using a map with 12 cones and 10 circles optimised assuming no disturbance. Keep in mind that all subfigures apart from **(a)** are the result of simulations and can thus be noisy.

Method				Compared to					
	IET			Central			Online ND		
	Average	Min	Max	IET same	IET one off	Max early	IET ≤	IET one off	Max early
Central	3.695	2	15	100.00 %	100.00 %	0	100.00 %	100.00 %	0
No Disturbance									
Online	3.695	2	15	99.99 %	100.00 %	1	100.00 %	100.00 %	0
Region-based									
⋈ 12 ○ 10	3.223	2	11	54.03 %	98.95 %	5	54.04 %	98.95 %	5
⋈ 12 ○ 20	3.243	2	15	55.90 %	99.00 %	5	55.91 %	99.00 %	5
⋈ 24 ○ 10	3.245	2	11	55.38 %	99.62 %	4	55.38 %	99.63 %	4
⋈ 24 ○ 20	3.355	2	15	66.20 %	99.83 %	3	66.20 %	99.84 %	3
⋈ 60 ○ 100	3.535	2	15	83.98 %	100.00 %	1	83.99 %	100.00 %	1
Upper Bound									
Online	3.263	2	15	56.95 %	99.80 %	3	56.96 %	99.80 %	3
Region-based									
⋈ 12 ○ 10	3.097	2	15	42.21 %	97.97 %	3	42.21 %	97.98 %	3
⋈ 12 ○ 20	3.118	2	15	44.03 %	98.29 %	3	44.03 %	98.30 %	3
⋈ 24 ○ 10	3.100	2	15	41.52 %	99.02 %	3	41.52 %	99.03 %	3
⋈ 24 ○ 20	3.130	2	15	44.19 %	99.28 %	3	44.19 %	99.29 %	3
⋈ 60 ○ 100	3.162	2	15	47.32 %	99.42 %	3	47.32 %	99.43 %	3

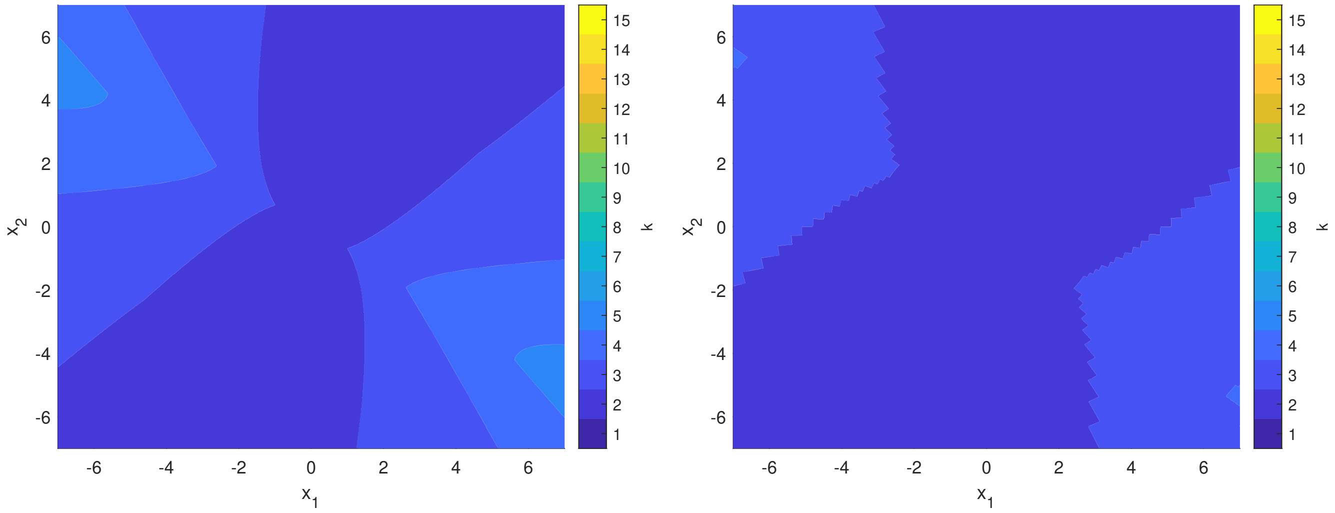
All times are given in time-steps.

Table 4-2 Simulated sampling performance in the absence of disturbances. The first column of the comparisons reports the percentage of simulations for which the same (or better) IET is reached. The next column reports the percentage of simulations in the previous column as well as those that trigger one time-step earlier. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.

4-1-2 In the Presence of Disturbances

When a disturbance is introduced, the performance guarantees deteriorate. Figure 4-4a shows the optimal MIET that is possible for each state of the system with the decentralised triggering function. The MIET that can be guaranteed for each region when using the finest partitioning is shown next to it in figure 4-4b. The effect of the disturbance is most prominent around the origin, as the magnitude of the disturbance is largest relative to the magnitude of the state and as a small shift can result in a considerable change of IET. A simulation of the system comparing the performance of the region-based method against the central triggering condition is shown in figure 4-5.

The same random states as before are also used to simulate the system with a disturbance to study how this impacts the IETs throughout the state-space. These simulations are run



(a) Optimised Online

(b) Map showing the MIET in each region with a partitioning using 60 cones 100 circles

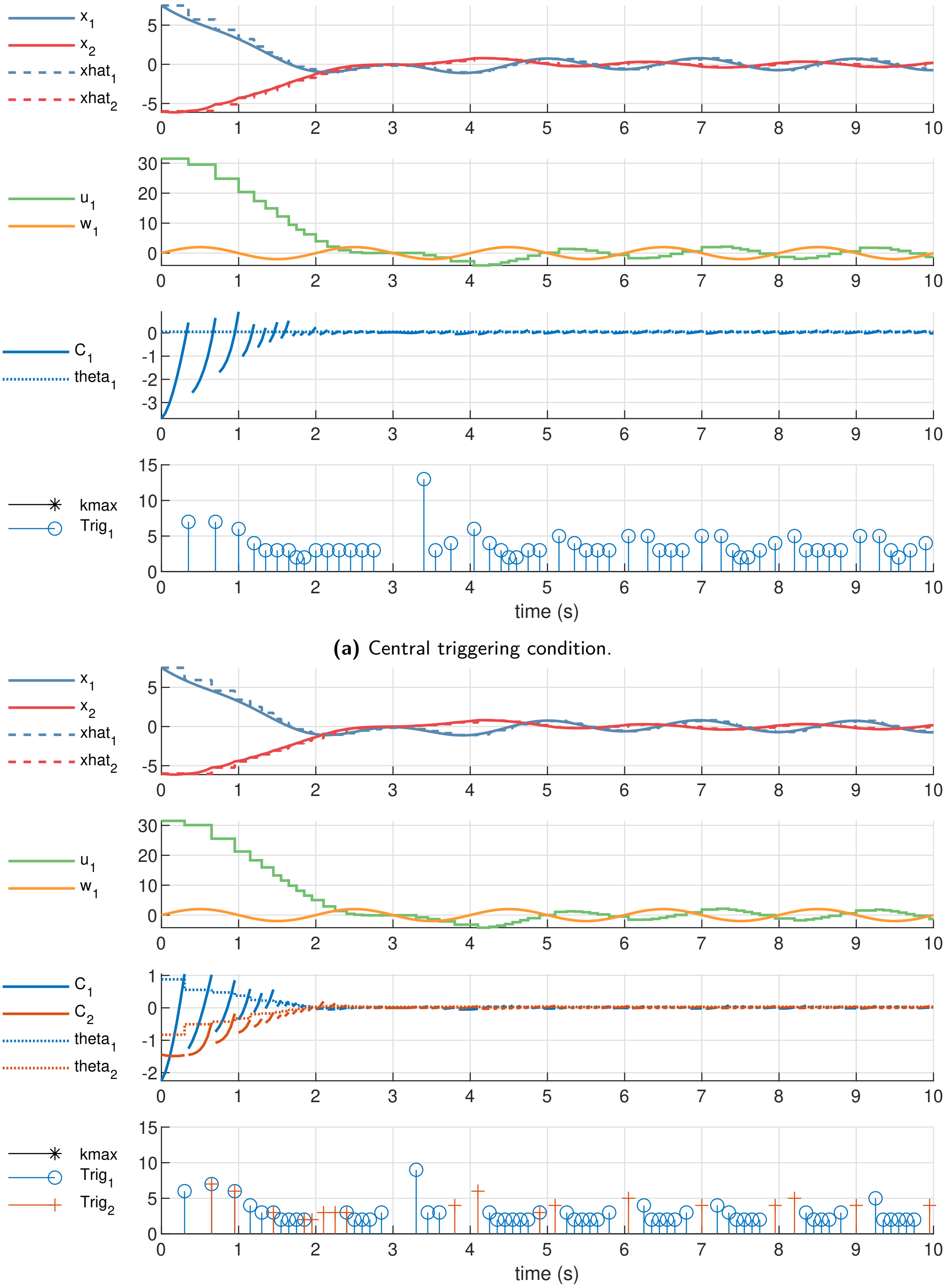
Figure 4-4 Optimised Inter-Event Times (IETs) of the system assuming the presence of a disturbance.

twice: In the first run, the disturbance $\boldsymbol{w}$ is drawn from a uniform distribution $U(-\mathcal{W}, \mathcal{W})$. In the second run, the disturbance is randomly drawn from the set of possible disturbances with the maximum magnitude $|\boldsymbol{w}| = \mathcal{W}$. Table 4-3 provides a summary of the results for the second run. The summary of the first run is included in the appendix (table B-1).

The IETs of the methods that are optimised without taking the disturbance into account are higher. However, these methods also occasionally result in very short IETs. The region-based mappings taking into account the disturbance, provide a guaranteed lower bound, which means that the nodes do not have to listen at all sampling times whether any of the other nodes has triggered. This can be used to reduce energy usage by not turning on the radio to listen to whether other nodes in the system have triggered. However, doing so reduces the robustness of the system, as any disturbance larger than assumed or an unmodeled measurement noise can now render the system unstable.

Figure 4-6 reports the IETs aggregated per region to get a better sense of how the performance changes throughout the state-space. Note the interesting location of the worst guaranteed MIET in **(a)**. They are located on $x_1 = 0$, which is the location where the disturbance, acting on x_1, has the largest effect. The effect of the disturbance is also greater relative to the state when closer to the origin, but the triggering condition also becomes more lenient close to the origin as a result of the absolute threshold. The large yellow regions in **(d)** clearly show that a disturbance can also have a positive influence on the IET.

The full distribution of the IETs is given in figure 4-7. Both methods optimised using the assumption of no disturbances result in simulations that are below the guaranteed MIET as their assumptions do not hold. However, they also result in IETs below the guaranteed MIET of the mapping that takes the disturbances into account.



(a) Central triggering condition.

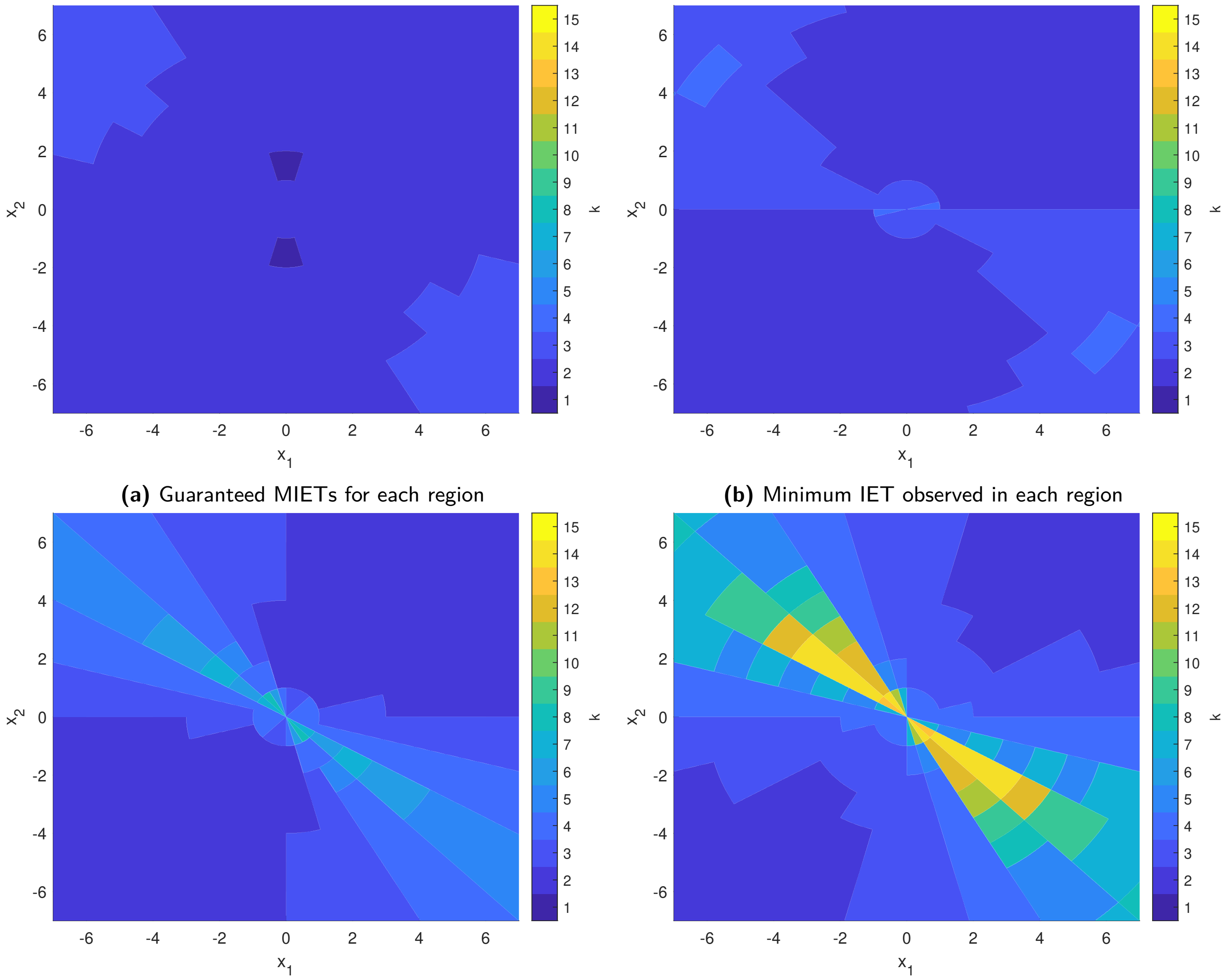
(b) System using a map with 60 cones and 100 circles optimised assuming a bounded disturbance.

Figure 4-5 Simulated system response in the presence of a disturbance $\boldsymbol{w}(t) = \mathcal{W}\sin(\pi t)$ with the initial state at $x_0 = \begin{bmatrix} 7.5 & -6 \end{bmatrix}^{\mathsf{T}}$. The height of the bars in the fourth plot represents the IET in time-steps.

				Compared to					
Method	**IET**			**Central**			**Online UB**		
	Average	Min	Max	IET same	IET one off	Max early	IET ≤	IET one off	Max early
Central	3.677	2	15	100.00 %	100.00 %	0	100.00 %	100.00 %	0
No Disturbance									
Online	3.289	1	15	67.54 %	95.11 %	7	91.78 %	99.06 %	4
Region-based									
⋈ 12 ○ 10	3.122	1	15	49.10 %	96.62 %	5	89.29 %	99.74 %	2
⋈ 12 ○ 20	3.135	1	15	50.45 %	96.47 %	4	90.57 %	99.78 %	2
⋈ 24 ○ 10	3.151	1	15	50.11 %	97.65 %	4	91.65 %	99.84 %	2
⋈ 24 ○ 20	3.214	1	15	56.63 %	97.55 %	4	93.41 %	99.91 %	2
⋈ 60 ○ 100	3.264	1	15	63.67 %	96.28 %	5	92.64 %	99.55 %	3
Upper Bound									
Online	3.175	2	15	53.91 %	96.62 %	4	100.00 %	100.00 %	0
Region-based									
⋈ 12 ○ 10	3.061	1	15	42.44 %	96.19 %	4	80.92 %	99.48 %	3
⋈ 12 ○ 20	3.081	2	15	44.23 %	96.43 %	3	84.00 %	99.53 %	3
⋈ 24 ○ 10	3.080	1	15	43.10 %	97.21 %	3	82.59 %	99.73 %	2
⋈ 24 ○ 20	3.097	2	15	44.64 %	97.40 %	3	85.41 %	99.73 %	2
⋈ 60 ○ 100	3.122	2	15	46.84 %	97.68 %	3	89.13 %	99.93 %	2

All times are given in time-steps.

Table 4-3 Simulated sampling performance in the presence of a disturbance with the maximum magnitude. The first column of the comparisons reports the percentage of simulations for which the same (or better) IET is reached. The next column reports the percentage of simulations in the previous column as well as those that trigger one time-step earlier. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.



(a) Guaranteed MIETs for each region

(b) Minimum IET observed in each region

(c) Average IET observed in each region

(d) Maximum IET observed in each region

Figure 4-6 Simulated Inter-Event Times (IETs) in the presence of a uniformly distributed disturbance with the system map consisting of 12 cones and 10 circles taking into account a bounded disturbance. Keep in mind that all subfigures apart from **(a)** are the result of simulations and can thus be noisy.

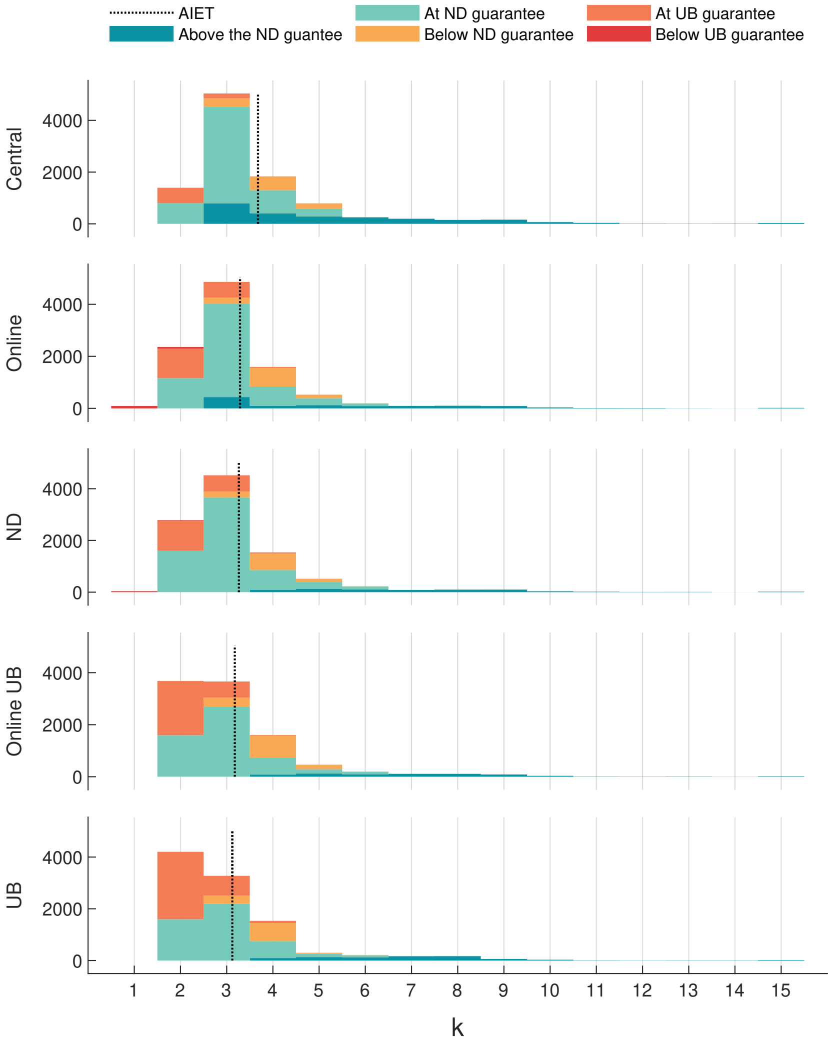


Figure 4-7 Distribution of simulated Inter-Event Times (IETs) in the presence of a disturbance with the maximum magnitude. The colours represent how each IET compares to the guaranteed IET computed when optimising the mapping.

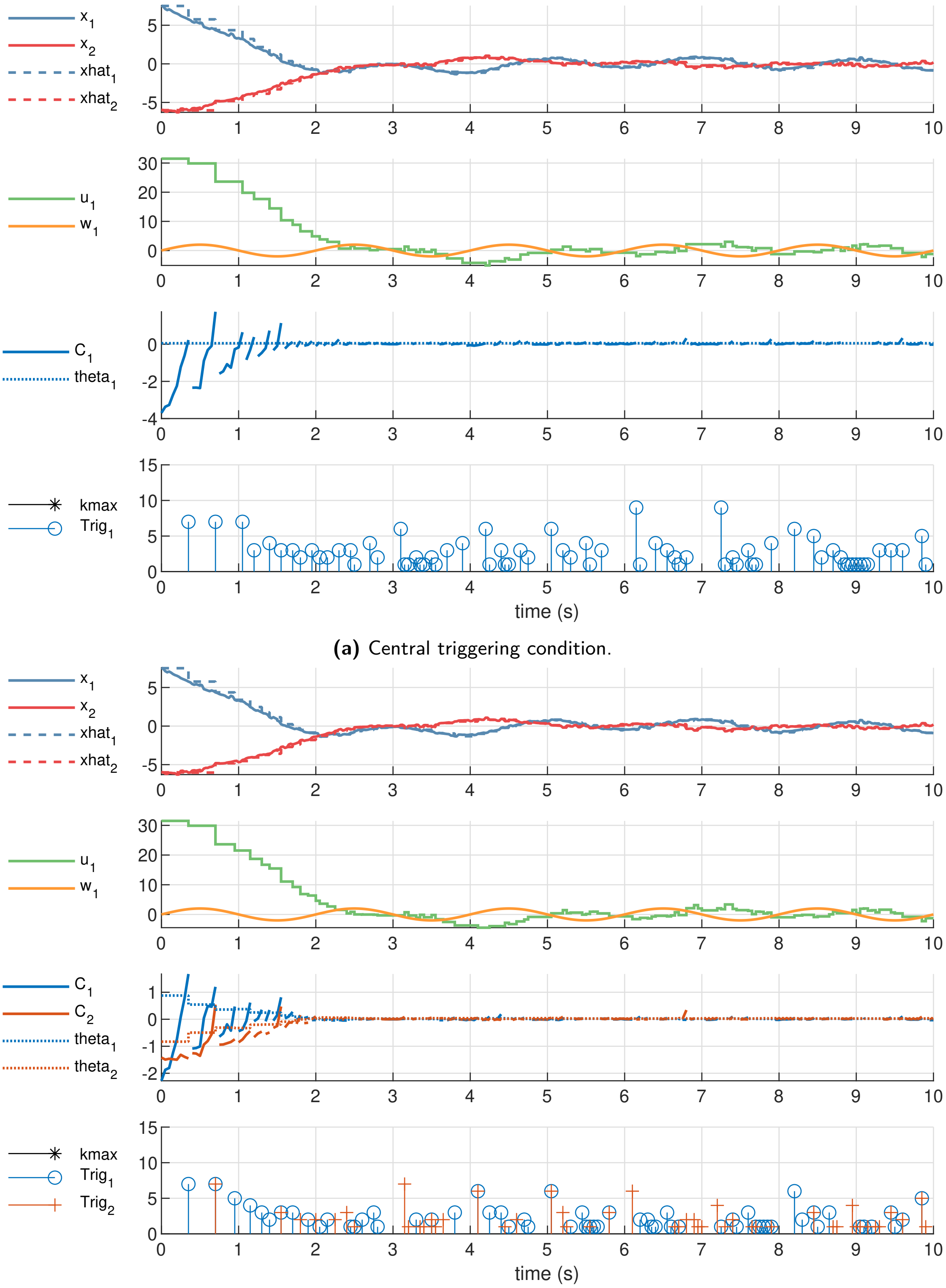
4-1-3 In the Presence of Disturbances and Noise

Finally, the performance is evaluated when measurement noise is introduced. The system is still guaranteed to be stable, although the area to which the system converges increases with a larger measurement noise. The measurement noise $\boldsymbol{v}(t) \in \mathcal{R}^{n_x}$ is introduced by changing equation (3-3) to:

$$\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t_i) + \boldsymbol{v}(t_i), \quad \forall t \in \{t_i \leq t < t_{i+1}\}, \quad \forall i \in \mathbb{N} \tag{4-3}$$

For these examples, the measurement noise is assumed to be normally distributed $\boldsymbol{v}(t) \sim \mathcal{N}(0, \sigma_v^2)$. A simulation of the system comparing the performance of the region-based method against the central triggering condition is shown in figure 4-8.

The same random states are again used to simulate the system and study the impact of measurement noise. The simulations are run for different variances of the measurement noise. A summary of the results is given in table 4-4.



(a) Central triggering condition.

(b) System using a map with 60 cones and 100 circles optimised assuming a bounded disturbance.

Figure 4-8 Simulated system response in the presence of a disturbance $\boldsymbol{w}(t) = \mathcal{W}\sin(\pi t)$ and measurement noise with a variance of $\sigma_v^2 = 0.01$ with the initial state at $x_0 = \begin{bmatrix} 7.5 & -6 \end{bmatrix}^\mathsf{T}$. The height of the bars in the fourth plot represents the IET in time-steps.

Method	IET			Compared to					
				Central			Online UB		
	Average	Min	Max	IET same	IET one off	Max early	IET $\leq$	IET one off	Max early
Central									
$\sigma_v^2 = 0$	3.677	2	15	100.00 %	100.00 %	0	100.00 %	100.00 %	0
$\sigma_v^2 = 0.001$	3.674	1	15	100.00 %	100.00 %	0	100.00 %	100.00 %	0
$\sigma_v^2 = 0.01$	3.624	1	15	100.00 %	100.00 %	0	99.00 %	99.98 %	4
$\sigma_v^2 = 0.1$	3.301	1	15	100.00 %	100.00 %	0	92.44 %	99.67 %	5
No disturbance Online									
$\sigma_v^2 = 0$	3.289	1	15	67.54 %	95.11 %	7	91.78 %	99.06 %	4
$\sigma_v^2 = 0.001$	3.223	1	15	65.20 %	92.88 %	10	91.92 %	98.76 %	7
$\sigma_v^2 = 0.01$	3.071	1	15	63.05 %	88.62 %	10	88.17 %	97.50 %	5
$\sigma_v^2 = 0.1$	2.212	1	10	43.02 %	68.57 %	13	74.70 %	91.52 %	10
Region-based No disturbance $\bowtie$ 60 $\circ$ 100									
$\sigma_v^2 = 0$	3.264	1	15	63.67 %	96.28 %	5	92.64 %	99.55 %	3
$\sigma_v^2 = 0.001$	3.204	1	15	62.19 %	93.46 %	5	92.75 %	99.16 %	3
$\sigma_v^2 = 0.01$	3.049	1	15	59.95 %	89.05 %	10	87.86 %	97.92 %	8
$\sigma_v^2 = 0.1$	2.242	1	10	44.00 %	70.34 %	13	75.64 %	91.83 %	10
Upper Bound Online									
$\sigma_v^2 = 0$	3.175	2	15	53.91 %	96.62 %	4	100.00 %	100.00 %	0
$\sigma_v^2 = 0.001$	3.119	1	15	52.70 %	94.02 %	4	100.00 %	100.00 %	0
$\sigma_v^2 = 0.01$	3.007	1	15	52.97 %	89.13 %	8	100.00 %	100.00 %	0
$\sigma_v^2 = 0.1$	2.295	1	12	33.79 %	69.89 %	12	100.00 %	100.00 %	0
Region-based Upper Bound $\bowtie$ 60 $\circ$ 100									
$\sigma_v^2 = 0$	3.122	2	15	46.84 %	97.68 %	3	89.13 %	99.93 %	2
$\sigma_v^2 = 0.001$	3.068	1	15	45.59 %	95.14 %	4	87.89 %	99.61 %	2
$\sigma_v^2 = 0.01$	2.963	1	15	48.67 %	90.07 %	6	83.46 %	98.19 %	6
$\sigma_v^2 = 0.1$	2.359	1	12	45.60 %	73.50 %	13	78.09 %	94.32 %	8

All times are given in time-steps.

Table 4-4 Simulated sampling performance in the presence of a disturbance with the maximum magnitude and measurement noise. The performance is compared to simulations with the same measurement noise variance. The first column of the comparisons reports the percentage of simulations for which the same (or better) IET is reached. The next column reports the percentage of simulations in the previous column as well as those that trigger one time-step earlier. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.

4-2 Linearised Batch Reactor

The second example is a linearised batch reactor originally described in Rosenbrock (1974) with the controller taken from Mazo Jr. et al. (2009).

$$\boldsymbol{A} = \begin{bmatrix} 1.38 & -0.20 & 6.71 & -5.67 \\ -0.58 & -4.29 & 0 & 0.67 \\ 1.06 & 4.27 & -6.65 & 5.89 \\ 0.04 & 4.27 & 1.34 & -2.10 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 5.67 & 0 \\ 1.13 & -3.14 \\ 1.13 & 0 \end{bmatrix}, \quad \boldsymbol{E} = \begin{bmatrix} 2 \\ 2 \\ 0 \\ 1 \end{bmatrix} \tag{4-4}$$

$$\boldsymbol{K} = \begin{bmatrix} 0.1006 & -0.2469 & -0.0952 & -0.2447 \\ 1.4099 & -0.1966 & 0.0139 & 0.0823 \end{bmatrix} \tag{4-5}$$

The system has three nodes: the first node measures the first state, the second node the second state and the last node the remaining two states. A disturbance acts on the system for which a bound is known $\mathcal{W} \leq 1$. The PETC mechanism is given by equation (3-9), where the triggering condition is checked with a sampling interval of $h = 0.01$ second. As the triggering condition, a decentralised version of the mixed condition given in equation (2-23) is used. With this, the decentralised quadratic triggering matrix is given by:

$$\boldsymbol{Q}(m) = \begin{bmatrix} (1-\mu)\mathbf{J}^m & -\mathbf{J}^m \\ -\mathbf{J}^m & \mathbf{J}^m \end{bmatrix} \tag{4-6}$$

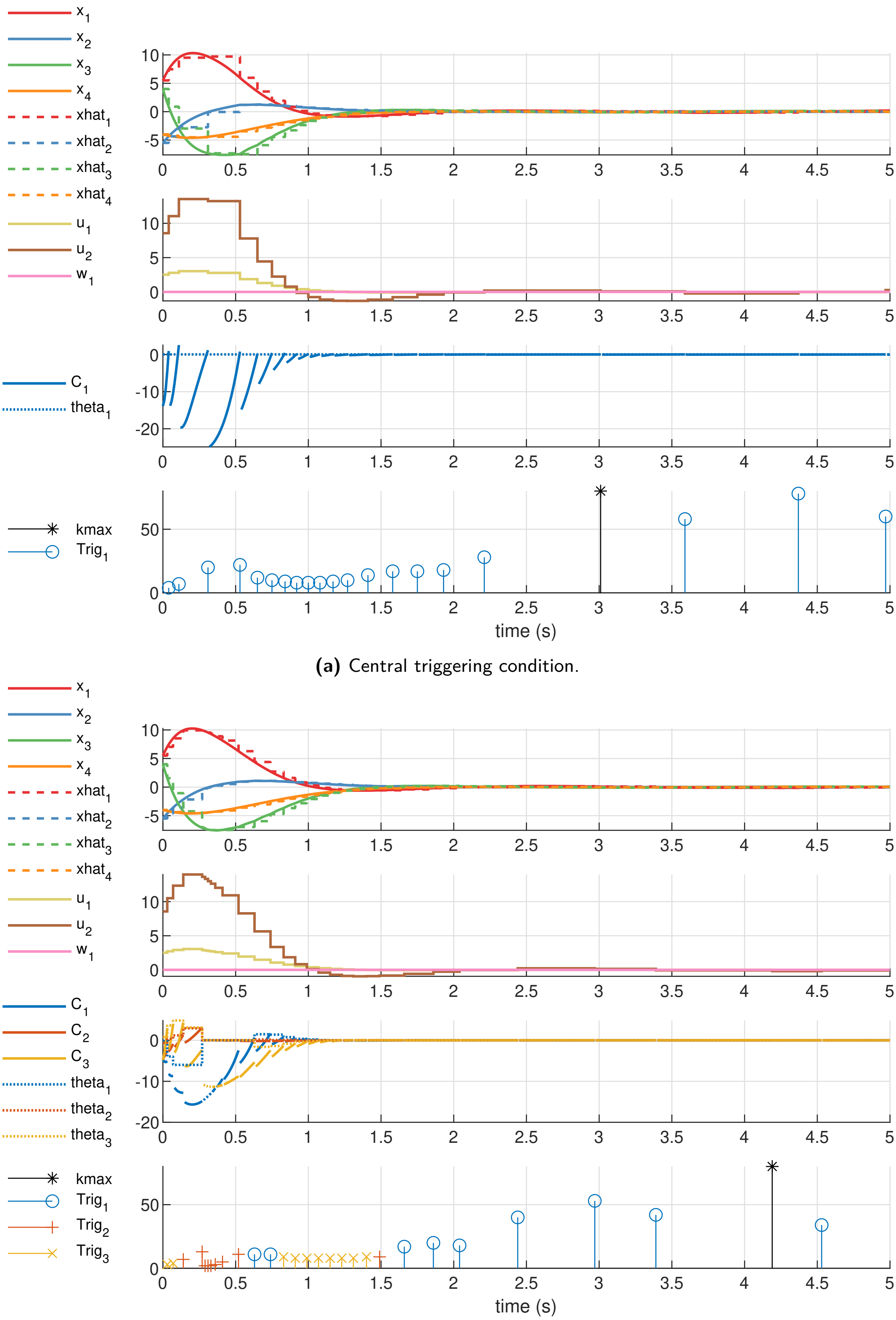
where the relative threshold is set to $\mu = 0.15$ and the absolute threshold to $\beta = 0.05$. The maximum inter-event time is set at $\overline{k} = 80$. Without the absolute threshold, the system would have a natural maximum of $\overline{k} = 39$. With this setup, the system is GEISS with a decay rate of at least $\rho = 0.0617$.

The state-space is partitioned in two different ways, both with a maximum radius of 10 around the origin. The first partitioning divides each angular coordinate into 12 parts combined with 15 homocentric N-spheres resulting in 25920 regions. The second partitioning divides the angular coordinates into 20 parts and uses 20 homocentric N-spheres resulting in 160000 regions.

4-2-1 In the Absence of Disturbances

First, the system performance is studied in the absence of a disturbance. A simulation of the system comparing the performance of the region-based method against the central triggering condition is shown in figure 4-9.

Just as with the previous system, 10000 random initial states uniformly distributed in the partitioning are simulated to determine the IET. The results are summarized in table 4-5. For the partitioning with 20 cones in angular coordinate and 20 N-spheres almost 24% of the state result in the same IET and 95% of the simulation trigger 7 or fewer time-steps earlier than the system with a centralised triggering condition. However, further increasing the number of states will yield better results. The full distribution of the IETs resulting from the different methods is given in figure 4-10.



(a) Central triggering condition.

(b) System using a map with 20 cones per angular dimension and 20 N-Spheres optimised assuming no disturbance.

Figure 4-9 Simulated system response in the absence of disturbances with the initial state at $x_0 = [5.5 \quad -5.5 \quad 4 \quad -4]]^\mathsf{T}$. The height of the bars in the fourth plot represents the IET in time-steps.

				Compared to					
Method	**IET**			**Central**			**Online ND**		
	Average	Min	Max	IET same	95% early	Max early	IET $\leq$	95% early	Max early
Central	7.995	3	39	100.00 %	0	0	100.00 %	0	0
No Disturbance									
Online	7.936	3	39	94.93 %	1	4	100.00 %	0	0
Region-based									
$\bowtie$ 12^3 $\circ$ 10	5.293	1	32	17.51 %	10	32	17.64 %	9	32
$\bowtie$ 20^3 $\circ$ 20	5.921	1	33	23.72 %	7	32	23.96 %	7	32
Upper Bound									
Online	7.277	3	35	47.88 %	2	11	50.24 %	2	8
Region-based									
$\bowtie$ 12^3 $\circ$ 10	5.144	1	31	15.60 %	10	32	15.66 %	10	32
$\bowtie$ 20^3 $\circ$ 20	5.567	1	30	18.79 %	9	32	18.91 %	8	32

All times are given in time-steps.

Table 4-5 Simulated sampling performance in the absence of disturbances. The first column of the comparisons reports the percentage of simulations for which the same (or better) IET is reached. The next column reports the number of time-steps that the system triggers earlier for the same initial state that 95% of the simulated states fall within. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.

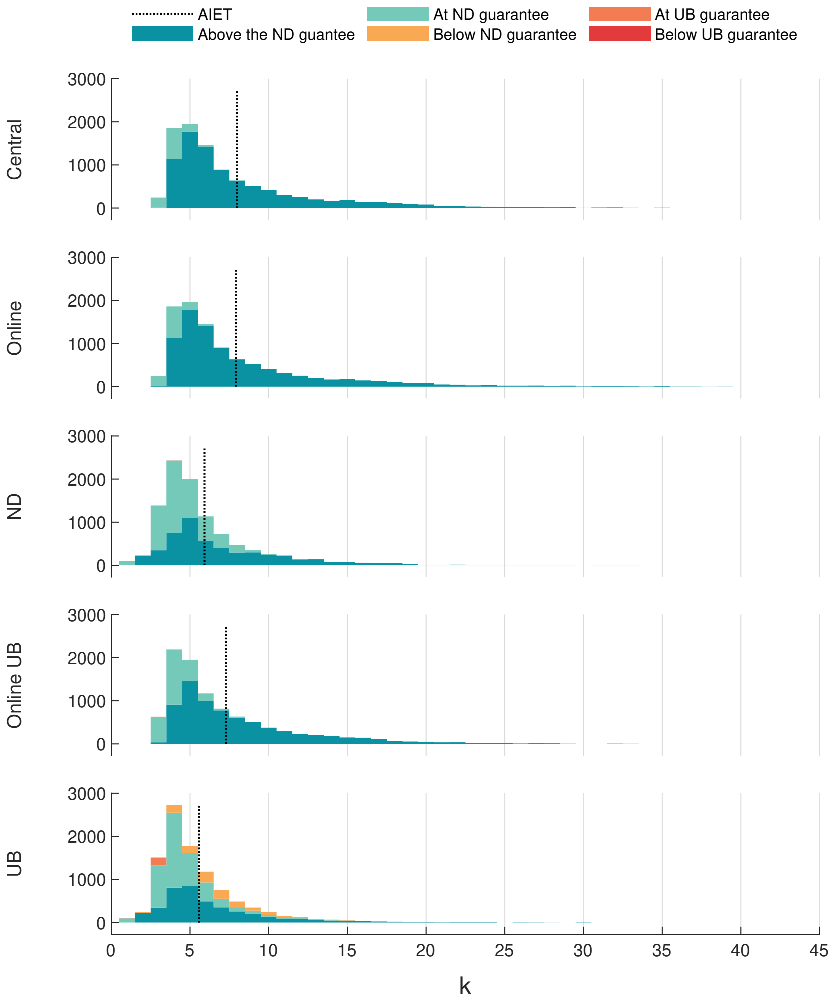


Figure 4-10 Distribution of simulated Inter-Event Times (IETs) in the absence of a disturbance. The colours represent how each IET compares to the guaranteed IET computed when optimising the mapping.

4-2-2 In the Presence of Disturbances

Adding a disturbance to the system has the same effect as before, deteriorating the system's performance. A simulation of the system comparing the performance of the region-based method against the central triggering condition is shown in figure 4-11.

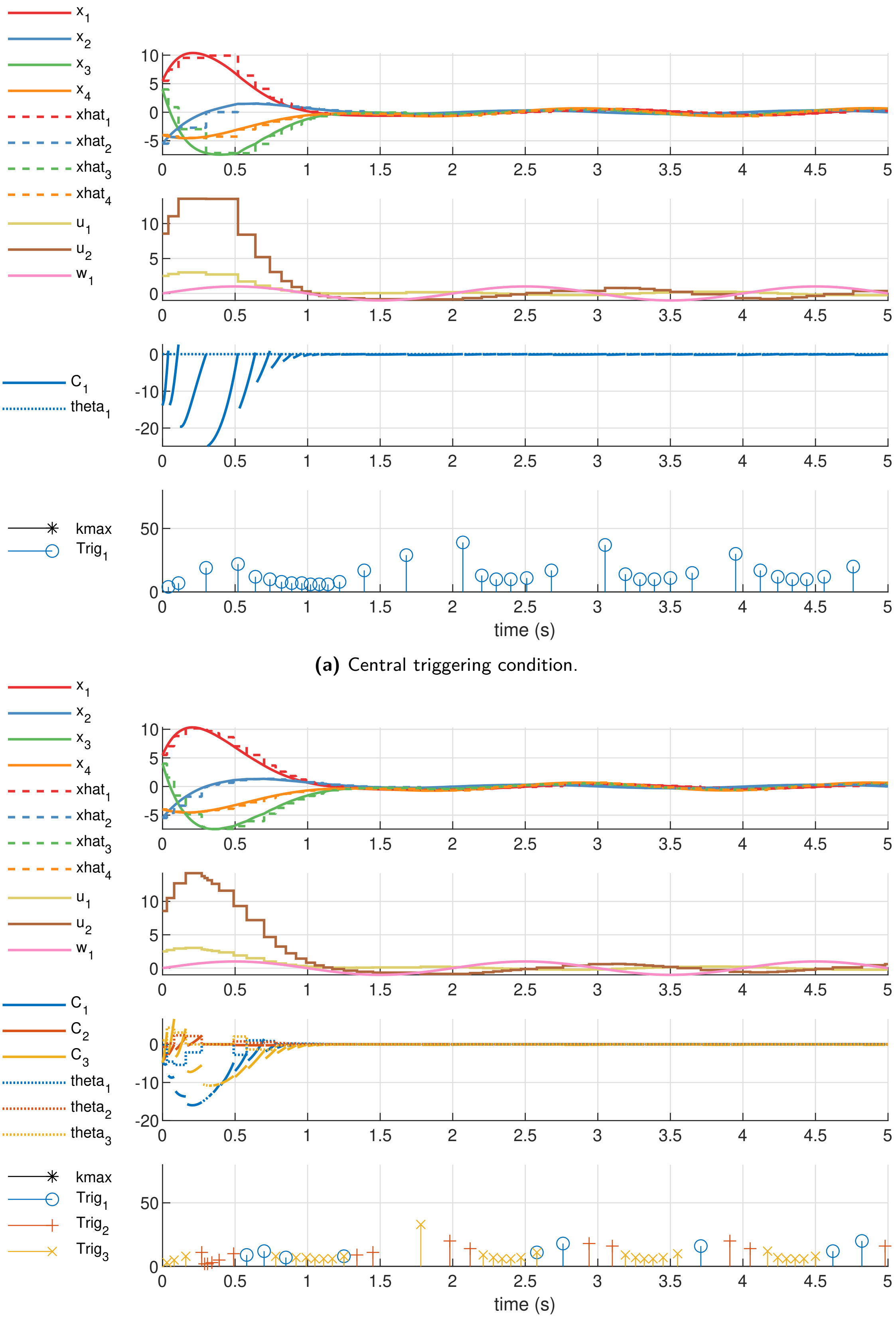
The same random states as before are also used to simulate the system with a disturbance to study the effects. These simulations are run using disturbances with the maximum magnitude $|\boldsymbol{w}| = \mathcal{W}$ to showcase the impact more clearly. The results are summarized in table 4-6. The full distribution of the IETs resulting from the different methods is given in figure 4-12.

				Compared to					
Method	**IET**			**Central**			**Online UB**		
	Average	Min	Max	IET same	95% early	Max early	IET $\leq$	95% early	Max early
Central	8.003	3	45	100.00 %	0	0	100.00 %	0	0
No Disturbance									
Online	7.278	1	37	60.70 %	3	35	92.14 %	1	31
Region-based									
$\bowtie 12^3$ $\circ$ 10	5.255	1	32	17.06 %	10	41	34.54 %	8	34
$\bowtie 20^3$ $\circ$ 20	5.923	1	34	24.50 %	8	41	49.33 %	6	34
Upper Bound									
Online	7.185	3	38	44.97 %	3	13	100.00 %	0	0
Region-based									
$\bowtie 12^3$ $\circ$ 10	5.119	1	32	15.31 %	10	40	31.44 %	8	33
$\bowtie 20^3$ $\circ$ 20	5.572	1	30	19.36 %	9	40	40.42 %	7	33

All times are given in time-steps.

Table 4-6 Simulated sampling performance in the presence of a disturbance with the maximum magnitude. The first column of the comparisons reports the percentage of simulations for which the same (or better) IET is reached. The next column reports the number of time-steps that the system triggers earlier for the same initial state that 95% of the simulated states fall within. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.

?



(a) Central triggering condition.

(b) System using a map with 20 cones per angular dimension and 20 N-Spheres optimised assuming a bounded disturbance.

Figure 4-11 Simulated system response in the presence of a disturbance $\boldsymbol{w}(t) = \mathcal{W}\sin(\pi t)$ with the initial state at $x_0 = \begin{bmatrix}5.5 & -5.5 & 4 & -4\end{bmatrix}^\mathsf{T}$. The height of the bars in the fourth plot represents the IET in time-steps.

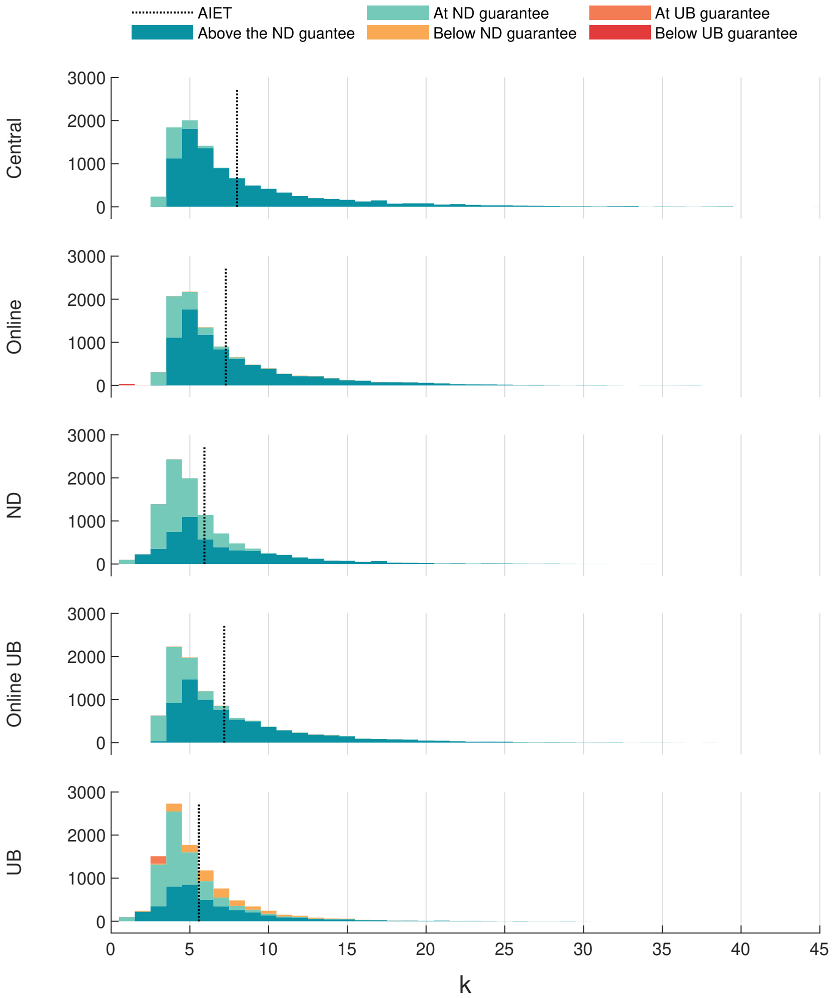


Figure 4-12 Distribution of simulated Inter-Event Times (IETs) in the presence of a disturbance with the maximum magnitude. The colours represent how each IET compares to the guaranteed IET computed when optimising the mapping.

Chapter 5

Conclusion

5-1 Conclusion

This thesis presents the methodology to optimally decentralise a quadratic triggering condition for a Periodic Event-Triggered Control (PETC) system both in the absence and presence of a bounded disturbance while maintaining its stability guarantees. Additionally, techniques to design a region-based map of the optimal decentralisation are introduced, which significantly reduces the complexity of the real-time computations needed to implement the optimised decentralisation. The state-dependent mapping opens up the way to implementing such a decentralisation on larger systems using less powerful hardware.

The proposed techniques are demonstrated on different systems showcasing the effectiveness, the influence of the granularity of the map as well as the consequence of unmodeled measurement noise.

5-2 Future Work

Several aspects of the developed techniques can be improved or further generalised to include a broader range of systems. The main inefficiency of the current approach lies in the partitioning of the state-space. Currently, the state-space is divided evenly. However, the number of regions needed to achieve a certain level of performance could be drastically reduced by adapting the partitioning to the specific system. Instead of every region covering the same angle (and radius), these can be optimised to increase the granularity in areas where the system dynamics very quickly. Alternatively, an exact partitioning based on Inter-Event Time (IET) can be constructed as done in Gleizer and Mazo Jr. (2020a), and these regions can then be approximated with a partitioning for which membership is easier to verify.

Furthermore, the effects of measurement noise can be incorporated into the mapping such that the guaranteed Minimal Inter-Event Time (MIET) remains valid for a certain level of measurement noise. Alternatively, the approach can be adapted to optimise the average IET by using the expectation instead of an upper bound on the future value of the triggering condition. These decentralisation techniques could also be extended to work with dynamic triggering conditions or dynamic controllers.

Appendix A

Proofs

A-1 Proofs for Section 3-3: Off-line Region-based Optimisation

A-1-1 Partitioning the state-space

Proposition 3.4 (Scaling law for triggering state of the system):
Consider a homogeneous system of degree ψ and a triggering function $\mathcal{C}(\cdot)$ homogeneous of degree ϕ. The state at which the system triggers implicitly defined by the triggering function scales according to:

$$\boldsymbol{x}'(\tau(\lambda\boldsymbol{x}_0'),(\lambda\boldsymbol{x}_0',\lambda\boldsymbol{e}_0))=\lambda\boldsymbol{x}'(\tau(\boldsymbol{x}_0'),(\boldsymbol{x}_0',\boldsymbol{e}_0)) \tag{3-26}$$

Where $\boldsymbol{x}'(t,(\boldsymbol{x}_0',\boldsymbol{e}_0))$ is the state of the system at time t given the initial condition of the system $\boldsymbol{x}'(0)=\boldsymbol{x}_0'$, $\boldsymbol{e}(0)=\boldsymbol{e}_0=\boldsymbol{0}$.

Proof. The closed loop system is homogeneous and thus state scales according to:

$$\boldsymbol{x}'(t,(\lambda\boldsymbol{x}_0',\lambda\boldsymbol{e}_0))=\lambda\boldsymbol{x}'(\lambda^{\psi}t,(\boldsymbol{x}_0',\boldsymbol{e}_0))$$

As derived in Anta and Tabuada (2008), inter-event times of the system scale according to:

$$\tau(\lambda\boldsymbol{x}_0')=\lambda^{-\psi}\tau(\boldsymbol{x}_0')) \tag{A-1}$$

Combining these two results in the scaling law for the state of the system at the inter-event time:

$$\boldsymbol{x}'(\tau(\lambda\boldsymbol{x}_0'),(\lambda\boldsymbol{x}_0',\lambda\boldsymbol{e}_0))=\lambda\boldsymbol{x}'(\tau(\boldsymbol{x}_0'),(\boldsymbol{x}_0',\boldsymbol{e}_0)) \tag{A-2}$$

□

Proposition 3.5 (Scaling law for the Optimal Decentralisation):
Consider the Periodic Event-Triggered Control (PETC) system (3-1) *to* (3-3) *and* (3-9)*. The maximum achievable inter-event time and optimal decentralisation scale according to:*

$$\kappa^*(\lambda\boldsymbol{x}')=\kappa^*(\boldsymbol{x}'),\qquad \theta_m^*(\lambda\boldsymbol{x}_0')=\lambda^2\theta_m^*(\boldsymbol{x}_0') \tag{3-27}$$

Proof. The lowest possible value for θ_m which ensures that node m will not trigger in the first k time steps is given by the highest lowered bound:

$$\underline{\theta}_m(k, \lambda\hat{\boldsymbol{x}}') = \lambda^2 \max_{i \in 1,2,\dots,k} \hat{\boldsymbol{x}}'^{\mathsf{T}} \widetilde{\boldsymbol{\Phi}}(i,m)\hat{\boldsymbol{x}}' = \lambda^2 \underline{\theta}_m(k, \hat{\boldsymbol{x}}') \tag{A-3}$$

The maximum number of time-steps for which it is possible to prevent the system from triggering is given by the highest k for which it is possible to keep the sum of all lower bounds $\underline{\theta}_m$ non-positive.

$$\sum_{m=1}^{n_m} \underline{\theta}_m(k, \lambda\hat{\boldsymbol{x}}') \leq 0 \iff \sum_{m=1}^{n_m} \underline{\theta}_m(k, \hat{\boldsymbol{x}}') \leq 0 \tag{A-4}$$

Which shows that the maximum inter-event time is the same for all states on the same homogeneous ray. Using this the scaling of the optimal decentralisation can simply be derived:

$$\theta^*_m(t_i, \lambda\hat{\boldsymbol{x}}') = \lambda^2 \max_{k \in 1,2,\dots,\kappa^*(\hat{\boldsymbol{x}})-1} \hat{\boldsymbol{x}}^{\mathsf{T}}(t_i)\widetilde{\boldsymbol{\Phi}}(k,m)\hat{\boldsymbol{x}}(t_i) = \lambda^2 \theta^*_m(t_i, \hat{\boldsymbol{x}}') \tag{A-5}$$

□

A-2 Proofs for Section 3-4: Decentralisation in the Presence of Disturbances

A-2-1 Upper Bound

Proposition 3.6 (Upper Bound on the future value of the triggering condition): *Consider the PETC system* (3-1) *to* (3-3) *and* (3-9)*. If there exist a scalar* $\mu \geq 0$ *and a symmetric matrix* $\boldsymbol{\Psi}$ *such that*

$$\boldsymbol{Q}_1(m) + \boldsymbol{\Psi} \preceq \mu \mathbf{I} \tag{3-40}$$

Then the future value of the triggering condition $\mathcal{C}(\hat{\boldsymbol{x}}, \boldsymbol{w}, k, m)$ *is upper bounded by*

$$\overline{\mathcal{C}}(\hat{\boldsymbol{x}}, k, m) = \hat{\boldsymbol{x}}^{\mathsf{T}}\left(\boldsymbol{\Phi}_1(k,m) + \boldsymbol{\Phi}_2(k,m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\right)\hat{\boldsymbol{x}} + \boldsymbol{\Phi}_3(k)\mathcal{W}^2 - \theta_m \tag{3-41}$$

With

$$\boldsymbol{Q}(m) = \begin{bmatrix} \boldsymbol{Q}_1(m) & \boldsymbol{Q}_2(m) \\ \boldsymbol{Q}_2^{\mathsf{T}}(m) & \boldsymbol{Q}_4(m) \end{bmatrix} \tag{3-42}$$

$$\boldsymbol{\Phi}_1(k,m) = \boldsymbol{\Lambda}^{\mathsf{T}}(k)\boldsymbol{Q}_1(m)\boldsymbol{\Lambda}(k) + \boldsymbol{\Lambda}^{\mathsf{T}}(k)\boldsymbol{Q}_2(m) + \boldsymbol{Q}_2(m)\boldsymbol{\Lambda}(k) + \boldsymbol{Q}_4(m) \tag{3-43}$$

$$\boldsymbol{\Phi}_2(k,m) = \boldsymbol{\Lambda}^{\mathsf{T}}(k)\boldsymbol{Q}_1(m) + \boldsymbol{Q}_2(m) \tag{3-44}$$

$$\boldsymbol{\Phi}_3(k) = kh\mu\lambda_{max}\left(\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\right) f_{\boldsymbol{A}}(k) \tag{3-45}$$

$$f_{\boldsymbol{A}}(k) = \begin{cases} \dfrac{e^{kh\lambda_{max}(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}})} - 1}{\lambda_{max}(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}})}, & \textit{if } \lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right) \neq 0 \\ kh, & \textit{otherwise} \end{cases} \tag{3-46}$$

Proof. Following the ideas of (Fiter et al., 2015, Proof Theorem 2) and Fu and Mazo Jr. (2018b)

The goal is to determine an upper bound on the future value of the triggering condition $\overline{\mathcal{C}}$ which does not depend on the realisation of the process noise $\boldsymbol{w}$.

$$\overline{\mathcal{C}}(\hat{\boldsymbol{x}}(t), k, m) < 0 \implies \mathcal{C}(\hat{\boldsymbol{x}}(t), k, m, \boldsymbol{w}(t, \dots, t+kh)) < 0 \tag{A-6}$$

The future value of the triggering condition can be expanded to:

$$\mathcal{C}(\cdot) = \hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}_1(k,m)\hat{\boldsymbol{x}} + \hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}_2(k,m)\boldsymbol{\Theta}(k) + \boldsymbol{\Theta}^{\mathsf{T}}(k)\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\hat{\boldsymbol{x}} + \boldsymbol{\Theta}^{\mathsf{T}}(k)Q_1\boldsymbol{\Theta}(k) - \theta_m \tag{A-7}$$

Where $\boldsymbol{\Phi}_1(k,m)$ and $\boldsymbol{\Phi}_1(k,m)$ defined as:

$$\boldsymbol{\Phi}_1(k,m) = \boldsymbol{\Lambda}^{\mathsf{T}}(k)Q_1(m)\boldsymbol{\Lambda}(k) + \boldsymbol{\Lambda}^{\mathsf{T}}(k)Q_2(m) + Q_2(m)\boldsymbol{\Lambda}(k) + Q_4(m) \tag{A-8}$$

$$\boldsymbol{\Phi}_2(k,m) = \boldsymbol{\Lambda}^{\mathsf{T}}(k)Q_1(m) + Q_2(m) \tag{A-9}$$

The cross-terms between the state and the uncertainty $\boldsymbol{\Theta}$ can be bounded using an inequality from (Gu et al., 2003, Lemma 6.2). For any positive definite $\boldsymbol{\Psi}$, the following inequality holds.

$$\hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}_2(k,m)\boldsymbol{\Theta}(k) + \boldsymbol{\Theta}^{\mathsf{T}}(k)\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\hat{\boldsymbol{x}} \leq \hat{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{\Phi}_2(k,m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\hat{\boldsymbol{x}} + \boldsymbol{\Theta}^{\mathsf{T}}(k)\boldsymbol{\Psi}\boldsymbol{\Theta}(k) \tag{A-10}$$

This partially removes the dependency on $\boldsymbol{w}$ resulting in an upper bound on the future triggering condition:

$$\mathcal{C}(\cdot) \leq \hat{\boldsymbol{x}}^{\mathsf{T}}\Big(\boldsymbol{\Phi}_1(k,m) + \boldsymbol{\Phi}_2(k,m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\Big)\,\hat{\boldsymbol{x}} + \boldsymbol{\Theta}^{\mathsf{T}}(k)\left(Q_1 + \boldsymbol{\Psi}\right)\boldsymbol{\Theta}(k) - \theta_m \tag{A-11}$$

The next step is to also find a bound for the final part of the uncertainty, this is done with Jensen Inequality (Gu et al., 2003, Proposition B.8):

$$\boldsymbol{\Theta}^{\mathsf{T}}(k)\left(Q_1 + \boldsymbol{\Psi}\right)\boldsymbol{\Theta}(k) \leq kh\int_0^{kh}\left(\mathrm{e}^{\boldsymbol{A}(kh-\tau)}\boldsymbol{E}\boldsymbol{w}(\tau)\right)^{\mathsf{T}}\left(Q_1 + \boldsymbol{\Psi}\right)\left(\mathrm{e}^{\boldsymbol{A}(kh-\tau)}\boldsymbol{E}\boldsymbol{w}(\tau)\right)d\tau \tag{A-12}$$

Using the assumption $\boldsymbol{Q}_1 + \boldsymbol{\Psi} \preceq \mu\mathbf{I}$

$$\leq kh\mu\int_0^{kh}\left(\mathrm{e}^{\boldsymbol{A}(kh-\tau)}\boldsymbol{E}\boldsymbol{w}(\tau)\right)^{\mathsf{T}}\left(\mathrm{e}^{\boldsymbol{A}(kh-\tau)}\boldsymbol{E}\boldsymbol{w}(\tau)\right)d\tau \tag{A-13}$$

$$= kh\mu\int_0^{kh}\boldsymbol{w}^{\mathsf{T}}(\tau)\boldsymbol{E}^{\mathsf{T}}\mathrm{e}^{(kh-\tau)\left(\boldsymbol{A}^{\mathsf{T}}+\boldsymbol{A}\right)}\boldsymbol{E}\boldsymbol{w}(\tau)d\tau \tag{A-14}$$

Using the inequality $\mathrm{e}^{\left(\boldsymbol{A}^{\mathsf{T}}+\boldsymbol{A}\right)} \leq \mathrm{e}^{\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}$ from Dahlquist (1958)(Van Loant, 1977, equation 2.2). Where $\lambda_{max}(\cdot)$ is the largest eigenvalue.

$$\leq kh\mu\int_0^{kh}\mathrm{e}^{(kh-\tau)\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}\boldsymbol{w}^{\mathsf{T}}(\tau)\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\boldsymbol{w}(\tau)d\tau \tag{A-15}$$

$$\leq kh\mu\lambda_{max}\left(\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\right)\int_0^{kh}\mathrm{e}^{(kh-\tau)\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}\boldsymbol{w}^{\mathsf{T}}(\tau)\boldsymbol{w}(\tau)d\tau \tag{A-16}$$

Since the disturbance $\boldsymbol{w}$ is bounded[1]

$$\leq kh\mu\lambda_{max}\left(\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\right)\int_0^{kh} \mathrm{e}^{(kh-\tau)\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}d\tau\|\boldsymbol{w}\|_\infty^2 \tag{A-17}$$

$$\leq kh\mu\lambda_{max}\left(\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\right)\int_0^{kh} \mathrm{e}^{(kh-\tau)\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}d\tau\mathcal{W}^2 \tag{A-18}$$

$$\tag{A-19}$$

Which leaves the integral to be evaluated

$$f_{\boldsymbol{A}}(k) = \begin{cases} \dfrac{e^{kh\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}-1}{\lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)}, & \text{if } \lambda_{max}\left(\boldsymbol{A}+\boldsymbol{A}^{\mathsf{T}}\right)\neq 0 \\ kh, & \text{otherwise} \end{cases} \tag{A-20}$$

Which finished the bound for the final part of the uncertainty:

$$\boldsymbol{\Theta}^{\mathsf{T}}(k)\left(Q_1+\boldsymbol{\Psi}\right)\boldsymbol{\Theta}(k) \leq \underbrace{kh\mu\lambda_{max}\left(\boldsymbol{E}^{\mathsf{T}}\boldsymbol{E}\right)f_{\boldsymbol{A}}(k)}_{\boldsymbol{\Phi}_3(k)}\mathcal{W}^2 \tag{A-21}$$

Combining everything together, the future value of the triggering condition is upper bounded by:

$$\mathcal{C}(\cdot) \leq \underbrace{\hat{\boldsymbol{x}}^{\mathsf{T}}\left(\boldsymbol{\Phi}_1(k,m)+\boldsymbol{\Phi}_2(k,m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m)\right)\hat{\boldsymbol{x}}+\boldsymbol{\Phi}_3(k)\mathcal{W}^2-\theta_m}_{\overline{\mathcal{C}}(\hat{\boldsymbol{x}}(t),k)} \tag{A-22}$$

□

Theorem 3.2 (Regional lower bound for θ to prevent the system from triggering): *If the hypothesis in proposition 3.6 holds and there exist scalars $\epsilon_i \geq 0$, $\forall i \in \{1,\ldots,n_x+1\}$ such that the following Linear Matrix Inequality (LMI) holds:*

$$\begin{bmatrix} -\boldsymbol{\Psi} & \boldsymbol{\Phi}_2^{\mathsf{T}}(k,m) & \boldsymbol{0} \\ \boldsymbol{\Phi}_2(k,m) & \boldsymbol{\Phi}_1(k,m) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{\Phi}_3(k)\mathcal{W}^2-\underline{\theta}_m \end{bmatrix} + \sum_i^{n_x+1}\epsilon_i\begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{V}_s\end{bmatrix} \preceq 0 \tag{3-47}$$

then $\underline{\theta}_m(k,s)$ is a lower bound of the local threshold for which node m will not to triggering after k time-steps starting from any state in region s for any realisation of the disturbance $\boldsymbol{w}$

Proof. The first step is to rewrite the nonlinear condition of proposition 3.6 such that it can be verified easily. To do so an additional dimension is introduced to render the future bound on the triggering condition homogeneous:

$$\begin{bmatrix}\boldsymbol{x}\\1\end{bmatrix}^{\mathsf{T}}\begin{bmatrix}\boldsymbol{\Phi}_1(k,m)+\boldsymbol{\Phi}_2(k,m)\boldsymbol{\Psi}^{-1}\boldsymbol{\Phi}_2^{\mathsf{T}}(k,m) & 0\\ 0 & \boldsymbol{\Phi}_3(k)\mathcal{W}^2-\theta_m\end{bmatrix}\begin{bmatrix}\boldsymbol{x}\\1\end{bmatrix} \leq 0 \tag{A-23}$$

[1] See Fiter et al. (2015) for using a perturbation bounded relatively to the state

A Lossy S-procedure is applied in the same manner as in theorem 3.1 to construct an LMI such that if the following LMI holds:

$$\begin{bmatrix} \mathbf{\Phi}_1(k,m) + \mathbf{\Phi}_2(k,m)\mathbf{\Psi}^{-1}\mathbf{\Phi}_2^{\intercal}(k,m) & 0 \\ 0 & \mathbf{\Phi}_3(k)\mathcal{W}^2 - \theta_m \end{bmatrix} + \sum_i^{n_x} \epsilon_i \boldsymbol{V}_s \preceq 0 \tag{A-24}$$

Then, the state lying within a region implies that the upper bound on the triggering condition holds. As a result for any state in region s node m does not trigger at time step k, irrespective of the realisation of the disturbance $\boldsymbol{w}$.

$$\begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}^{\intercal} \boldsymbol{V}_{s,i} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}' \geq 0, \quad \forall i \in \{1, \dots, n_x + 1\} \quad \implies \textit{equation (A-23)} \tag{A-25}$$

The final step is to transform this LMI to eliminate the non linear expression in order to make checking the condition straightforward. The vectors representing the regions can be split up in a matrix for the original dimensions of the system and a scalar for the added dimension. There are no cross terms between the original states and the new added state since the new dimension is only used to eliminate an absolute value from the system.

$$\boldsymbol{V}_s = \begin{bmatrix} \boldsymbol{V}_s^1 & 0 \\ 0 & V_s^4 \end{bmatrix} \tag{A-26}$$

Using this the LMI can be rewritten as:

$$\begin{bmatrix} \mathbf{\Phi}_1(k,m) + \mathbf{\Phi}_2(k,m)\mathbf{\Psi}^{-1}\mathbf{\Phi}_2^{\intercal}(k,m) + \sum_i^{n_x+1} \epsilon_i \boldsymbol{V}_s^1 & 0 \\ 0 & \mathbf{\Phi}_3(k)\mathcal{W}^2 - \theta_m + \sum_i^{n_x+1} \epsilon_i V_s^4 \end{bmatrix} \preceq 0 \tag{A-27}$$

Since $\mathbf{\Psi} \succ 0$, the Schur complement can be used to show that the following two conditions are equivalent.

$$\begin{bmatrix} \mathbf{\Phi}_1(k,m) + \sum_i^{n_x+1} \epsilon_i \boldsymbol{V}_s^1 & \mathbf{\Phi}_2(k,m) \\ \mathbf{\Phi}_2^{\intercal}(k,m) & -\mathbf{\Psi} \end{bmatrix} \preceq 0 \iff \mathbf{\Phi}_1(k,m) + \mathbf{\Phi}_2(k,m)\mathbf{\Psi}^{-1}\mathbf{\Phi}_2^{\intercal}(k,m) + \sum_i^{n_x+1} \epsilon_i \boldsymbol{V}_s^1 \preceq 0 \tag{A-28}$$

Using this the LMI can be transformed to get rid of the non linear term:

$$\begin{bmatrix} \mathbf{\Phi}_1(k,m) + \sum_i^{n_x+1} \epsilon_i \boldsymbol{V}_s^1 & \mathbf{\Phi}_2(k,m) & \mathbf{0} \\ \mathbf{\Phi}_2^{\intercal}(k,m) & -\mathbf{\Psi} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Phi}_3(k)\mathcal{W}^2 - \theta_m + \sum_i^{n_x+1} \epsilon_i \boldsymbol{V}_s^4 \end{bmatrix} \preceq 0 \tag{A-29}$$

Which can then be rewritten to get a slightly more elegant condition:

$$\begin{bmatrix} -\mathbf{\Psi} & \mathbf{\Phi}_2^{\intercal}(k,m) & \mathbf{0} \\ \mathbf{\Phi}_2(k,m) & \mathbf{\Phi}_1(k,m) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Phi}_3(k)\mathcal{W}^2 - \theta_m \end{bmatrix} + \sum_i^{n_x+1} \epsilon_i \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V}_s \end{bmatrix} \preceq 0 \tag{A-30}$$

□

Appendix B

Additional Data from Examples

				Compared to					
Method	**IET**			**Central**			**Online UB**		
	Average	Min	Max	IET same	IET one off	Max early	IET $\leq$	IET one off	Max early
Central	3.714	2	15	100.00 %	100.00 %	0	100.00 %	100.00 %	0
No Disturbance									
Online	3.469	1	15	79.22 %	97.56 %	8	95.09 %	99.23 %	5
Region-based									
$\bowtie$ 12 $\circ$ 10	3.186	2	15	49.85 %	97.92 %	6	89.36 %	99.58 %	2
$\bowtie$ 12 $\circ$ 20	3.202	2	15	51.65 %	97.84 %	6	90.77 %	99.59 %	3
$\bowtie$ 24 $\circ$ 10	3.213	2	15	51.64 %	98.59 %	4	91.60 %	99.93 %	2
$\bowtie$ 24 $\circ$ 20	3.293	1	15	59.83 %	98.54 %	6	93.59 %	99.83 %	3
$\bowtie$ 60 $\circ$ 100	3.406	2	15	72.08 %	98.02 %	5	95.17 %	99.65 %	3
Upper Bound									
Online	3.237	2	15	54.09 %	98.69 %	4	100.00 %	100.00 %	0
Region-based									
$\bowtie$ 12 $\circ$ 10	3.098	2	15	40.90 %	97.73 %	4	79.62 %	99.69 %	4
$\bowtie$ 12 $\circ$ 20	3.117	2	15	42.62 %	97.91 %	4	82.50 %	99.72 %	4
$\bowtie$ 24 $\circ$ 10	3.113	2	15	41.73 %	98.27 %	4	81.44 %	99.78 %	2
$\bowtie$ 24 $\circ$ 20	3.134	2	15	43.70 %	98.40 %	3	84.65 %	99.77 %	2
$\bowtie$ 60 $\circ$ 100	3.160	2	15	45.97 %	98.69 %	3	88.50 %	99.89 %	2

All times are given in time-steps.

Table B-1 Simulated sampling performance in the presence of a uniformly distributed disturbance. The first column of the comparisons reports the percentage of simulations for which the same (or better) Inter-Event Time (IET) is reached. The next column reports the percentage of simulations in the previous column as well as those that trigger one time-step earlier. The third column gives the maximum number of time-steps that the system will trigger earlier for the same initial state.

Bibliography

Åarzén, K.-E. (1999). A simple event-based PID controller. *IFAC Proceedings Volumes*, 32(2):8687–8692.

Anta, A. and Tabuada, P. (2008). To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE Transactions on Automatic Control*, 55(9):2030–2042.

Anta, A. and Tabuada, P. (2012). Exploiting isochrony in self-triggered control. *IEEE Transactions on Automatic Control*, 57(4):950–962.

Åström, K. J. and Bernhardsson, B. (1999). Comparison of periodic and event based sampling for first-order stochastic systems. *IFAC Proceedings Volumes*, 32(2):5006–5011.

Behera, A. K. and Bandyopadhyay, B. (2015). Decentralized event-triggered sliding mode control. *2015 10th Asian Control Conference: Emerging Control Techniques for a Sustainable World, ASCC 2015.*

Blumenson, L. E. (1960). A Derivation of n-Dimensional Spherical Coordinates. *Source: The American Mathematical Monthly*, 67(1):63–66.

Borgers, D. P., Dolk, V. S., and Heemels, W. P. (2017). Dynamic Periodic Event-Triggered Control for Linear Systems. *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control.*

Borgers, D. P. and Heemels, W. P. (2014). Event-separation properties of event-triggered control systems. *IEEE Transactions on Automatic Control*, 59(10):2644–2656.

Dahlquist, G. (1958). *Stability and error bounds in the numerical integration of ordinary differential equations.* PhD thesis, Stockholm College.

Delimpaltadakis, G. and Mazo Jr., M. (2021a). Abstracting the Traffic of Nonlinear Event-Triggered Control Systems. *preprint arXiv:2010.12341.*

Delimpaltadakis, G. and Mazo Jr., M. (2021b). Isochronous Partitions for Region-Based Self-Triggered Control. *IEEE Transactions on Automatic Control*, 66(3):1160–1173.

Donkers, M. C. and Heemels, W. P. (2012). Output-based event-triggered control with guaranteed L ∞- gain and improved and decentralized event-triggering. *IEEE Transactions on Automatic Control*, 57(6):1362–1376.

Fiter, C., Hetel, L., Perruquetti, W., and Richard, J. P. (2012). A state dependent sampling for linear state feedback. *Automatica*, 48(8):1860–1867.

Fiter, C., Hetel, L., Perruquetti, W., and Richard, J. P. (2015). A robust stability framework for LTI systems with time-varying sampling. *Automatica*, 54:56–64.

Fu, A. and Mazo Jr., M. (2016). Periodic asynchronous event-triggered control. *2016 IEEE 55th Conference on Decision and Control, CDC 2016*, pages 1370–1375.

Fu, A. and Mazo Jr., M. (2018a). Decentralized periodic event-triggered control with quantization and asynchronous communication. *Automatica*, 94:294–299.

Fu, A. and Mazo Jr., M. (2018b). Traffic Models of Periodic Event-Triggered Control Systems. *IEEE Transactions on Automatic Control*, 64(8):3453–3460.

Girard, A. (2015). Dynamic Triggering Mechanisms for Event-Triggered Control. *IEEE Transactions on Automatic Control*, 60(7):1992–1997.

Gleizer, G. de A. and Mazo Jr., M. (2020a). Scalable Traffic Models for Scheduling of Linear Periodic Event-Triggered Controllers. *IFAC-PapersOnLine*, 53:2726–2732.

Gleizer, G. de A. and Mazo Jr., M. (2020b). Self-Triggered Output-Feedback Control of LTI Systems Subject to Disturbances and Noise. *Automatica*, 120.

Gu, K., Kharitonov, V. L., and Chen, J. (2003). *Stability of Time-Delay Systems*. Birkhäuser Boston, Boston, MA.

Heemels, W. P. and Donkers, M. C. (2013). Model-based periodic event-triggered control for linear systems. *Automatica*, 49(3):698–711.

Heemels, W. P., Donkers, M. C., and Teel, A. R. (2011). Periodic event-triggered control based on state feedback. *Proceedings of the IEEE Conference on Decision and Control*, pages 2571–2576.

Heemels, W. P., Donkers, M. C., and Teel, A. R. (2013). Periodic event-triggered control for linear systems. *IEEE Transactions on Automatic Control*, 58(4):847–861.

Heemels, W. P., Gorter, R. J., Van Zijl, A., Van Den Bosch, P. P., Weiland, S., Hendrix, W. H., and Vonder, M. R. (1999). Asynchronous measurement and control: a case study on motor synchronization. *Control Engineering Practice*, 7(12):1467–1482.

Heemels, W. P., Johansson, K. H., and Tabuada, P. (2012). An introduction to event-triggered and self-triggered control. *Proceedings of the IEEE Conference on Decision and Control*, pages 3270–3285.

Hendricks, E., Jensen, M., Chevalier, A., and Vesterholm, T. (1994). Problems in event based engine control. *Proceedings of the American Control Conference*, 2:1585–1587.

Hetel, L., Daafouz, J., and Iung, C. (2007). LMI control design for a class of exponential uncertain systems with application to network controlled switched systems. *Proceedings of the American Control Conference*, pages 1401–1406.

Khader, O. and Willig, A. (2013). An energy consumption analysis of the Wireless HART TDMA protocol. *Computer Communications*, 36(7):804–816.

Khalil, H. K. (2002). *Nonlinear Systems*. Prentice Hal, 3rd edition edition.

Kolarijani, A. S. and Mazo Jr., M. (2016). Formal Traffic Characterization of LTI Event-Triggered Control Systems. *IEEE Transactions on Control of Network Systems*, 5(1):274–283.

Lunze, J. and Lehmann, D. (2010). A state-feedback approach to event-based control. *Automatica*, 46(1):211–215.

Lygeros, J., Johansson, K. H., Simić, S. N., Zhang, J., and Sastry, S. S. (2003). Dynamical properties of hybrid automata. *IEEE Transactions on Automatic Control*, 48(1):2–17.

Mazo Jr., M., Anta, A., and Tabuada, P. (2009). On self-triggered control for linear systems: Guarantees and complexity. *2009 European Control Conference, ECC 2009*, pages 3767–3772.

Mazo Jr., M., Anta, A., and Tabuada, P. (2010). An ISS self-triggered implementation of linear controllers. *Automatica*, 46(8):1310–1314.

Mazo Jr., M. and Cao, M. (2011). Decentralized event-triggered control with asynchronous updates. *Proceedings of the IEEE Conference on Decision and Control*, pages 2547–2552.

Mazo Jr., M. and Cao, M. (2012). Asynchronous Decentralized Event-triggered Control. *Automatica*, 50(12):3197–3203.

Mazo Jr., M., Sharifi-Kolarijani, A., Adzkiya, D., and Hop, C. (2018). Abstracted Models for Scheduling of Event-Triggered Control Data Traffic. *Lecture Notes in Control and Information Sciences*, 475:197–217.

Mazo Jr., M. and Tabuada, P. (2009). Input-to-state stability of self-triggered control systems. *Proceedings of the IEEE Conference on Decision and Control*, pages 928–933.

Mazo Jr., M. and Tabuada, P. (2011). Decentralized event-triggered control over wireless sensor/actuator networks. *IEEE Transactions on Automatic Control*, 56(10):2456–2461.

Miskowicz, M. (2006). Send-On-Delta Concept: An Event-Based Data Reporting Strategy. *Sensors 2006, Vol. 6, Pages 49-63*, 6(1):49–63.

Nešić, D., Teel, A. R., Valmorbida, G., and Zaccarian, L. (2013). Finite-gain Lp stability for hybrid dynamical systems. *Automatica*, 49:2384–2396.

Otanez, P. G., Moyne, J. R., and Tilbury, D. M. (2002). Using deadbands to reduce communication in networked control systems. *Proceedings of the American Control Conference*, 4:3015–3020.

Pólik, I. and Terlaky, T. (2007). A survey of the S-lemma. *SIAM Review*, 49(3):371–418.

Postoyan, R., Anta, A., Heemels, W. P., Tabuada, P., and Nešić, D. (2013). Periodic event-triggered control for nonlinear systems. *Proceedings of the IEEE Conference on Decision and Control*, pages 7397–7402.

Rosenbrock, H. H. (1974). *Computer-aided control system design.* Academic Press, London.

Sontag, E. D. (1989). Smooth Stabilization Implies Coprime Factorization. *IEEE Transactions on Automatic Control*, 34(4):435–443.

Sontag, E. D. (2008). Input to State Stability: Basic Concepts and Results. *Lecture Notes in Mathematics*, 1932:163–220.

Sontag, E. D. and Wang, Y. (1995). On characterizations of the input-to-state stability property. *Systems & Control Letters*, 24(5):351–359.

Tabuada, P. (2007). Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685.

Trobinger, M., Gleizer, G. de A., Istomin, T., Mazo Jr., M., Murphy, A. L., and Picco, G. P. (2022). The Wireless Control Bus: Enabling Efficient Multi-Hop Event-Triggered Control with Concurrent Transmissions. *ACM Transactions on Cyber-Physical Systems*, 6(1).

Van Loant, C. (1977). The Sensitivity of the Matrix Exponential. *SIAM Journal on Numerical Analysis*, 14(6):971–981.

Velasco, M., Velasco, M., Fuertes, J. M., and Marti, P. (2003). The self triggered task model for real-time control systems. *Proceedings of 24th IEEE Real-Time Systems Symposium, Work-in-Progress Session*, 384.

Wang, X. and Lemmon, M. D. (2008). Event design in event-triggered feedback control systems. *Proceedings of the IEEE Conference on Decision and Control*, pages 2105–2110.

Wang, X. and Lemmon, M. D. (2011). Event-triggering in distributed networked control systems. *IEEE Transactions on Automatic Control*, 56(3):586–601.

Ye, H., Michel, A. N., and Hou, L. (1998). Stability theory for hybrid dynamical Systems. *IEEE Transactions on Automatic Control*, 43(4):461–474.

Ye, W., Heidemann, J., and Estrin, D. (2002). An energy-efficient MAC protocol for wireless sensor networks. *Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, 3:1567–1576.

Zhou, H.-Y., Luo, D.-Y., Gao, Y., and Zuo, D.-C. (2011). Modeling of Node Energy Consumption for Wireless Sensor Networks. *Wireless Sensor Network*, 3(1):18–23.

Glossary

List of Acronyms

TX	Transmit data
RX	Receive data
ETC	Event-Triggered Control
STC	Self-Triggered Control
CETC	Continuous Event-Triggered Control
PETC	Periodic Event-Triggered Control
LTI	Linear Time-Invariant
GAS	Globally Asymptotically Stable
GES	Globally Exponentially Stable
ISS	Input-to-State Stable
GEISS	Globally Exponentially Input-to-State Stable
ISpS	Input-to-State Practically Stable
IET	Inter-Event Time
MIET	Minimal Inter-Event Time
LMI	Linear Matrix Inequality