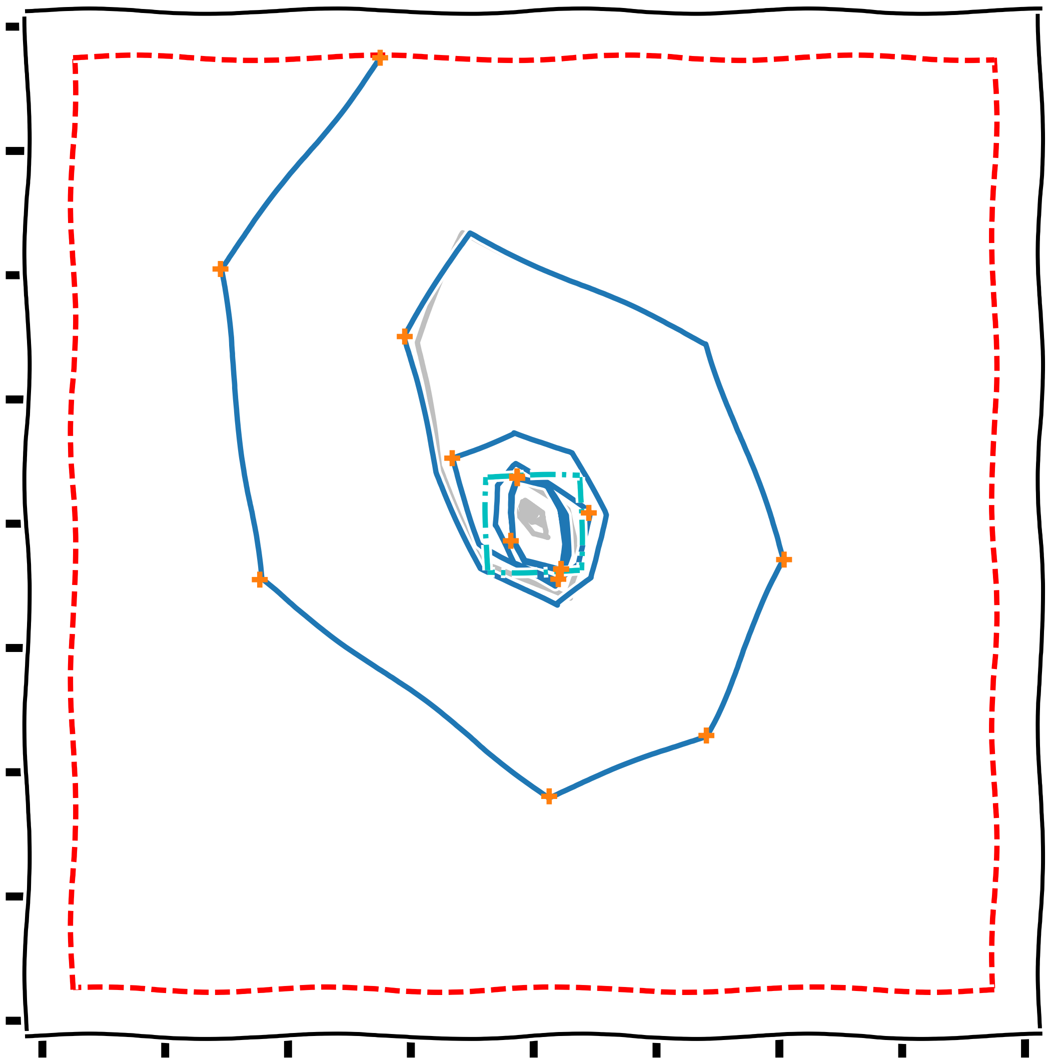
Robust Model Predictive Control with Aperiodic Actuation

Employing a Decentralized Triggering Mechanism

Sander Bregman

Master of Science Thesis



TU Delft Delft University of Technology

Delft Center for Systems and Control

Robust Model Predictive Control with Aperiodic Actuation

Employing a Decentralized Triggering Mechanism

Master of Science Thesis

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

Sander Bregman

August 14, 2017

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

The work in this thesis was supported by the Delft Center for Systems and Control (DCSC).

Delft University of Technology
Department of
Delft Center for Systems and Control (dcsc)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

Robust Model Predictive Control with Aperiodic Actuation

by

Sander Bregman

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: August 14, 2017

Supervisor(s): ______________________
dr.ir. T. Keviczky

____
A. Sharifi Kolarijani, MSc.

Reader(s): ______________________
dr.ir. M. Mazo Jr.

____
dr.ir. A. Hegyi

Abstract

In this thesis, a control design problem, in which communication between different elements of the control system takes place through a shared (possibly wireless) channel, is considered. With the implementation of the proposed approach, the use of limited resources such as network bandwidth and battery life may be reduced.

The proposal consists of a robust Model Predictive Control (MPC) approach, that is only executed at instants at which a decentralized triggering mechanism triggers. As long as no triggering occurs, inputs that have been computed at the previous MPC update are used. The triggering mechanism uses the trajectories from the MPC to calculate bounds on the error between each actual state and predicted state, for all instants up to the horizon. When all individual errors are inside their respective bounds at some instant, violation at the next instant still results in an MPC problem that is (1) guaranteed to have a feasible solution and (2) for which an upper bound for the objective function value is given that is lower than the value at the previous instant. These two properties result in stability of the closed loop system.

Simulation results are given to demonstrate the effectiveness of the proposed approach. Compared to approaches that solve similar problems that can be found in literature, the proposed approach differs in the need for weaker assumptions and/or in the maximization of the bounds on the error signal. This is made possible by letting the triggering mechanism depend on the sequences that are generated by the MPC at the last update instant, as well as the measured state.

Table of Contents

Preface and Acknowledgement

This is document is a part of my Master of Science graduation thesis. The problem that we have tried to solve during this thesis project essentially is one of efficient resource allocation under constrained circumstances, specifically applied to the communication resources that some model predictive control approach employs. Though the implementation of the techniques that are presented in this document still lie far from practical implementation, I hope to somehow have made a useful contribution to the continuously developing discipline of Systems and Control. I owe a large debt of gratitude to Arman Sharifi Kolarijani, whose patience, insights and experience have helped me to be able to finish this project. In addition, I would like to thank the people at the Delft Center for Systems and Control and all other people around me for facilitating my graduation project and my studies over the past years.

Chapter 1

Introduction

1-1 Motivation

Developments of technologies in the field of Networked Control Systems (NCSs), such as more efficient wireless communication protocols and more miniaturized and energy efficient computing units, have made Wireless Actuator and Sensor Networks (WSANs) a feasible medium for the communication between different elements of many types of control systems. WSANs can be used in control systems in various applications, for example in traffic control [1], [2], building automation [3], agricultural irrigation systems [4], [5], autonomous vehicles [6] or industrial production systems [7].

1-1-1 Wireless network characteristics

Wireless communication has several advantages compared to communication through wired connections, such as flexible connectivity and ease of deployment and maintenance, but comes with challenges as well. A survey of complications that come with the employment of (wireless) networks in control systems can be found in [8]. Some of these challenges are related to the network infrastructure, such as clock synchronization, network security and competition for network access. Other challenges may be addressed by a proper design of the control system, such as the allocation of scarce network resources: the network capacity and the energy storage of wireless elements. In this thesis the focus lies on reducing unnecessary allocation of these two resources.

From a network point of view, all other conditions being equal, these two resources need to be traded for one another. Achieving higher bandwidth communication generally requires stronger radios (that use more energy), whereas low power radios do not achieve a high data rate. For standalone sensor elements, in general, the energy spent on communication takes up the majority of the energy consumption [9]. This makes energy-aware scheduling of the communication between sensors and controller especially important.

1-1-2 Reduction in usage of scarce resources

When aiming to reduce the energy consumption related to wireless communication, more considerations than only a reduction in the amount of data sent by some element have to be made: some other factors have to be taken into account. For example, sending data, receiving data and listening (without actually receiving anything) generally consume similar amounts of power [10], [11]. Communicating with elements that are located close by takes less power than communicating to elements that are far away [10]. Furthermore, sending (a large amount of) data in bulk is more efficient than sending the same amount of data in small packages, as a result of channel acquisition overhead [12].

This last property can be exploited by predictive control approaches, combined with some mechanism that triggers an update only when this is for some reason deemed necessary. Large packages containing predictions can be sent, where the actuators keep on implementing control inputs from these predictions until new predictions are received. This thesis focuses on reducing the communication that a certain predictive control system employs. Doing so, the employment of the two scarce resources in WSANs, network capacity and battery life, is reduced simultaneously.

1-2 Related work

This section contains an introduction and pointers to literature concerning Event Based (EB) scheduling of control tasks. Later, existing approaches for Event Triggered Robust Model Predictive Control (ET-RMPC) are reviewed, with the focus on finding open problems in this field.

1-2-1 Event Based scheduling of control tasks

EB scheduling of control tasks has been under development for a long period, at least since the 1960s. Early examples of Event Triggered (ET) control are [13], [14] and [15], referring to it as being *varying sampling rate* or *adaptive sampling* techniques. More recent strategies are presented in [16], [17] (referring to it as *Lebesgue sampling*). In e.g. [18], [19] and [20] the current form of ET scheduling of control tasks was finally developed.

In general, differing from (periodic) time based scheduling, in EB scheduling approaches there is a mechanism that is responsible for determining the next sampling instant. This mechanism is called the Triggering Mechanism (TM). Based on which element of the control system is responsible for finding the next sampling instant, EB control techniques can be categorized into two main groups [21]: i) ET control strategies, where the mechanism is embedded in the sensory system and ii) Self Triggered (ST) control strategies, where the mechanism is embedded in the controller.

1-2-2 Event Triggered scheduling

Most ET approaches, such as the ones proposed in [18], [19] and [20] employ a sample-and-hold strategy, where the control input is fixed between two sampling instants. When the (norm of the) difference between the current state and the state at the last sampling instant becomes larger than some threshold, an update is triggered and this difference becomes zero. [18] is famous for deriving some threshold, for which closed-loop stability is achieved with a guaranteed lower bound on the time between two triggering instants. Alternative to [18], in which the TM has access to the full state in a centralized manner, [19] and [20] take this approach and adapt the TM such that it can be evaluated in a decentralized way, separately by each sensor.

When a measurement of all states is not available at one location, a decentralized TM offers substantial advantages over one that is centralized. The reason for this is that for the centralized TM additional communication of measurements is necessary in order to determine if measurements should be communicated.

1-2-3 Self Triggered scheduling

In approaches that apply ST scheduling of the sampling instants, the controller itself calculates the next sampling instant. This next sampling instant does not depend on the evolution of the state after an update, but is based on the state at the last update instant, the control policy and a model of the system. TMs that are purely ST in their nature are very sensitive to modeling errors and uncertainties. When combined with Model Predictive Control (MPC), the computational complexity is greatly increased compared to a time-triggered MPC implementation, such as in the approaches proposed in [22], [23] and [24]. This large burden of computational complexity is the main reason that this thesis focuses on ET scheduling approaches.

1-2-4 Event Triggered Robust Model Predictive Control (ET-RMPC)

For an overview of MPC in the literature literature, see for instance [25], [26] and [27]. Various formulations of MPC, categorized as Robust Model Predictive Control (RMPC), that are stable when the system is subjected to bounded unknown additive disturbances are introduced and developed further in [28] (*min-max optimization*), [29], [30] (*constraint tightening*) and [31], [32] (*tube based*).

The combination of MPC approaches with some EB scheduling of updates may profit more from non-periodic updates than EB approaches combined with some static feedback controllers. In the case of ET-RMPC, the control strategy is not based on the last measured state, but on some time-varying prediction of the state. Whenever a measurement of the state deviates significantly from the predicted state, an update is triggered, thereby resetting the error between measurement and prediction to zero.

1-2-5 Existing approaches for ET-RMPC

In ET-RMPC, after a controller update has been triggered and performed, the predicted trajectories up to the prediction horizon are sent to the appropriate sensors and actuators. The actuators keep using values from these predictions until some criterion is no longer fulfilled at the sensors.

Combinations of an EB sampling strategy with an MPC approach have been proposed in the literature. See for instance for ET scheduling [33], [34], [35] and [36], or for ST scheduling [22], [23] and [24].

The ET approaches share in common that the *norm of the prediction error* is compared to some threshold. The prediction error is defined as the difference between the prediction of the state made by the MPC at the last update instant and the actual measured state. When the norm of this prediction error exceeds a certain threshold, an MPC update is triggered.

In the light of this thesis, aiming to reducing the load that the communication of measurements has on the network and sensor batteries, such a (centralized) TM has limited advantages. Still, at all instants the measurements of all states need to be gathered at a centralized agent, that then decides if an update is necessary. The development of a decentralized TM to determine update instants for an RMPC approach is regarded as an open problem to the best of our knowledge.

1-3 Problem statement

The goal of this report is to, given some RMPC formulation, present a triggering strategy that determines the instants at which a finite-horizon Optimal Control Problem (OCP) needs to be solved. This strategy is intended to reduce the communication load between sensor, controller and actuator, while at the same time not increasing the computational complexity of the OCP compared to a standard time-triggered RMPC implementation. The triggering strategy, combined with the RMPC approach, is aimed to let the state and input of a discrete-time Linear Time Invariant (LTI) system converge to some target sets, while at all times satisfying constraints on the input and state.

The TM should be decentralized, such that no communication of measurements to a centralized element is necessary to check the triggering conditions. The TM should as well be computationally simple to evaluate. This last condition is to make sure that an intelligent sensor can compute its local condition with relatively simple (and low-power) hardware.

1-4 Structure

This document is structured as follows: A description of Robust MPC by constraint tightening, the approach on which the developed strategy is based, is presented in Chapter 3. This chapter is largely based on the method proposed in [30], but is given for completeness and convenient notation in the context of developing the TM and its properties. Chapter 4 introduces centralized conditions on the prediction error, such that if these are satisfied, recursive feasibility and convergence follow. Then, in Chapter 5, decentralized conditions are derived of which satisfaction implies satisfaction of the earlier centralized conditions. Simulation results and observations based on these results are given in Chapter 6. Finally, a discussion of the results and concluding remarks are given in Chapter 7.

1-5 Notation and Definitions

This section provides some definitions for the mathematical notation as used in the remainder of this report. For positive integers m and n, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote the n-dimensional Euclidean space and the $n \times m$-dimensional matrix space, respectively. The set of nonnegative integers is depicted by $\mathbb{Z}_{\geq 0}$. For a vector $x \in \mathbb{R}^n$, $||x||_\infty = \max(|x_1|, \dots, |x_n|)$. For a matrix $M \in \mathbb{R}^{n\times n}$, $M \succ 0$ indicates the positive definiteness of M. For a matrix $N \in \mathbb{R}^{n\times m}$, $N^\top$ and $N^\dagger$ represent the transpose and the Moore-Penrose pseudo-inverse of N. I_n and $0_{m\times n}$ denote the identity matrix in $\mathbb{R}^{n\times n}$ and the $m \times n$ matrix with all elements equal to zero. For a matrix $M \in \mathbb{R}^{m\times n}$ and a set $\mathcal{S} \subseteq \mathbb{R}^n$, we define the set $\mathcal{C} = M\mathcal{S} = \{Ms \mid s \in \mathcal{S}\}$. See Appendix A for the notation used to denote convex polytopes.

Definition 1.1 (Pontryagin difference)**.** *For two sets $\mathcal{A}$ and $\mathcal{B}$ in $\mathbb{R}^{n\times m}$, $\mathcal{A} \sim \mathcal{B}$ denotes the Pontryagin difference between $\mathcal{A}$ and $\mathcal{B}$. This difference is defined as:*

$$\mathcal{A} \sim \mathcal{B} = \{a | a + b \in \mathcal{A},\ \forall b \in \mathcal{B}\}. \tag{1-1}$$

Definition 1.2 (Multiplication of a Set by a Matrix)**.** *The multiplication of a set $\mathcal{S} \subset \mathbb{R}^n$ by a matrix $M \in \mathbb{R}^{m\times n}$ denotes a mapping of all its elements:*

$$M\mathcal{S} = \{c \in \mathbb{R}^m |\ \exists s \in \mathcal{S}, c = Ms\}. \tag{1-2}$$

Definition 1.3 (Point to Set Weighted Distance)**.** *For $M \succ 0$, the squared weighted distance $d(\cdot,\cdot,\cdot)$ of a point $r \in \mathbb{R}^n$ from a set $\mathcal{S} \subset \mathbb{R}^n$ is given by:*

$$d(r, \mathcal{S}, M) = \min_{s\in\mathcal{S}}\ ||r - s||_M^2 = \min_{s\in\mathcal{S}}\ (r - s)^\top M(r - s). \tag{1-3}$$

When a point s is given as the second argument, the distance $d(\cdot,\cdot,\cdot)$ is given by:

$$d(r, s, M) = ||r - s||_M^2 = (r - s)^\top M(r - s). \tag{1-4}$$

The following result will be used in our analysis:

Lemma 1.4 ([30])**.** *Let a, b be two vectors in $\mathbb{R}^n$, $\mathcal{B}, \mathcal{C}$ be two compact sets in $\mathbb{R}^n$ and M be a weighting matrix $M \succ 0$ in $\mathbb{R}^{n\times n}$. Then, using the distance function given in Definition 1.3, it follows for all $a \in \mathbb{R}^n$:*

$$d(a + c, \mathcal{B}, M) \leq d(a, \mathcal{B} \sim \mathcal{C}, M),\ \forall c \in \mathcal{C}. \tag{1-5}$$

Proof [30]. Let $d_1 = d(a, \mathcal{B} \sim \mathcal{C}, M)$. Using (1-3), we know that there exists a $b^* \in \mathcal{B} \sim \mathcal{C}$ such that $||a - b^*||_M = d_1$. From (1-1) we know that $b^* + c \in \mathcal{B}, \forall c \in \mathcal{C}$. □

Chapter 2

Communication structure

This chapter presents an overview concerning the communication structure related to implementing the different algorithms that will be introduced in Chapters 3 - 5. Figures 2-1, 2-2 and 2-3 show schematically the communication structure for the various triggering strategies.

Table 2-1 shows the number of values that have to be sent at every instant (marked by 1 in the second column) and at every Model Predictive Control (MPC) update that is triggered (marked by j in the second column).

Table 2-1: Communication structure for the various update triggering strategies. The abbreviations in the first row mean respectively: S - Sensor, TM - Triggering Mechanism, C - Controller, A - Actuator.

	comm. every .. instants	S-TM	TM-C	C-TM	C-A
MPC	1	n		-	m
	j	-		-	-
Centralized TM	1	n	-	-	-
	j	-	n	$N(n+m)$	Nm
Decentralized TM	1	-	-	-	-
	j	-	n	$2Nn$	Nm

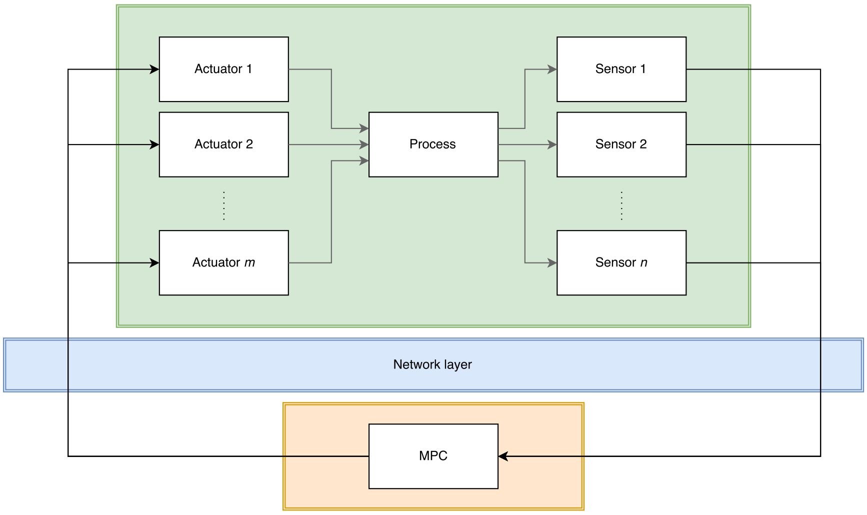


Figure 2-1: Communication structure for ordinary MPC scheduling. All communication takes place at every instant. Thin lines represent the sending of a signal (with n or m values.

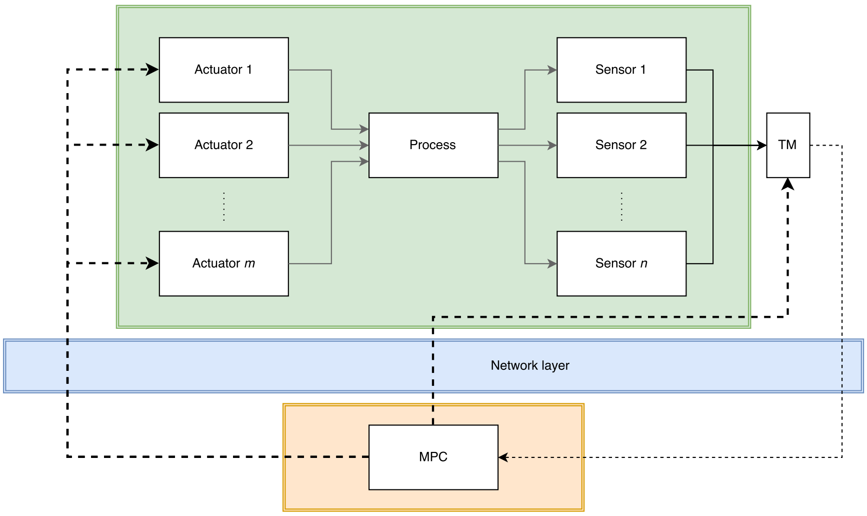


Figure 2-2: Communication structure for the centralized triggering strategy. Dashed lines show the communication that needs to take place only when an event is triggered; solid lines show communication that needs to take place without considering the triggering instants. Thin lines represent the sending of a signal (with n or m values); bold lines represent sending trajectories (with sizes of the order $N \cdot n$ or $N \cdot m$).

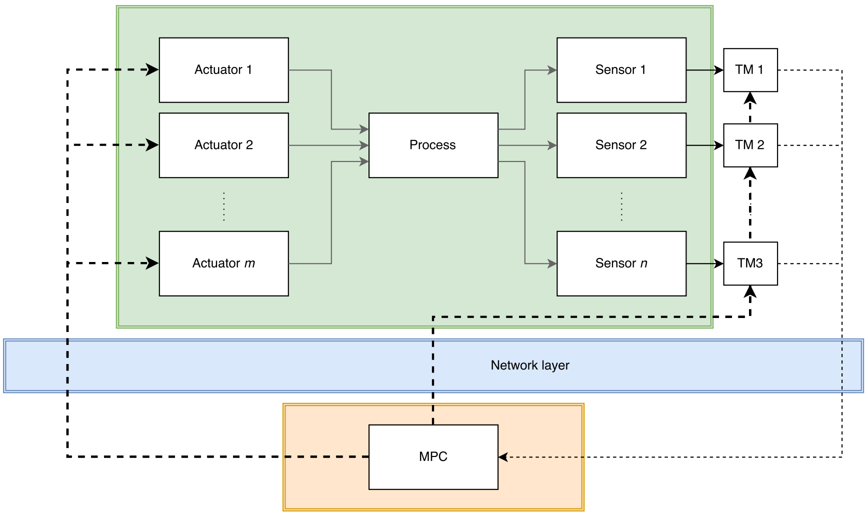


Figure 2-3: Communication structure for the decentralized triggering strategy. Dashed lines show the communication that needs to take place only when an event is triggered; solid lines show communication that needs to take place without considering the triggering instants. Thin lines represent the sending of a signal (with n or m values; bold lines represent sending trajectories (with sizes of the order $N \cdot n$ or $N \cdot m$).

Chapter 3

Robust MPC by Constraint Tightening

3-1 Introduction

This chapter introduces the Robust Model Predictive Control (RMPC) framework for which a triggering strategy will be developed. Existing RMPC approaches can be categorized into three different groups: approaches that use min-max optimization (see e.g. [28]), ones that use constraint tightening (see e.g. [29], [30]) and ones that use *tubes* (see e.g. [31], [32]).

Because RMPC with min-max optimization has a significantly higher computational complexity than ordinary Model Predictive Control (MPC), it is not regarded in this text. The remaining two approaches, constraint tightening and tube-based RMPC are very similar in their method and complexity, where RMPC with constraint tightening has the more simple formulation allowing for easier adaptations. Therefor, in the remainder of this thesis, the development will be based on RMPC with constraint tightening .

The remainder of this chapter introduces the *constraint tightening* RMPC approach from [30]. It guarantees that the state and input of some discrete-time linear time invariant system converge to some target set that contains the origin, while satisfying constraints on both the state and input. This is achieved even though a an additive disturbance that is bounded such that it lies in some convex compact polytope affects the state. The method was proposed in [29] and further developed in [30]. We reformulate the approach given in [30], abandoning the usage of tracking outputs, directly using inputs and states in the definitions of constraints and cost function. This is done in order to use the results in a notation relevant for the developments made in this thesis.

First, the formulation of the system is given, after which the Optimal Control Problem (OCP) that is at the heart of the RMPC is introduced. Subsequently, theorems are given concerning the robust recursive feasibility and robust convergence properties of the state for when this approach is implemented in a time-triggered manner.

3-2 System definition

Consider an LTI system with bounded additive perturbations given by:

$$x_{k+1} = Ax_k + Bu_k + w_k, \ \forall k \in \mathbb{Z}_{\geq 0}, \tag{3-1}$$

where the state, input and disturbance signals satisfy the following conditions:

$$x_k \in \mathbb{X} \subseteq \mathbb{R}^n, \ u_k \in \mathbb{U} \subseteq \mathbb{R}^m, \ w_k \in \mathcal{W} \subset \mathbb{R}^n. \tag{3-2}$$

The nominal system associated with (3-1), that is used to make predictions, is given by:

$$\bar{x}_{k+1} = A\bar{x}_k + Bu_k, \ \forall k \in \mathbb{Z}_{\geq 0}. \tag{3-3}$$

The goal of the controller is to let the state x_k and input u_k converge to the target sets $\mathbb{T}_x \subseteq \mathbb{R}^n$ and $\mathbb{T}_u \subseteq \mathbb{R}^m$, respectively, for $k \to \infty$, while at all instants satisfying the corresponding constraints.

Assumption 3.1. *The constraint sets $\mathbb{X}$, $\mathbb{U}$, the target sets $\mathbb{T}_x$, $\mathbb{T}_u$ and the disturbance admissible set $\mathcal{W}$ are all convex compact polytopes containing their underlying spaces' origin.*

Assumption 3.2. *The pair (A, B) is stabilizable.*

It follows from Assumption 3.2 that one is able to design an LQR controller for the nominal system (3-3) with properly chosen positive definite matrices Q and R (the pair $(A, Q^{\frac{1}{2}})$ is detectable). One could use the unique solution $P \succ 0$ of the Discrete-time Algebraic Riccati Equation (DARE) [37]:

$$P = A^\top PA - (A^\top PB)(R + B^\top PB)^{-1}(B^\top PA) + Q,$$

and find the following stabilizing state-feedback gain:

$$F = -(R + B^\top PB)^{-1}B^\top A. \tag{3-4}$$

This state-feedback gain F, applied to the terminal state, is used in the proofs of recursive feasibility and convergence. In addition to this feedback gain F, a set of disturbance feedback gains $\mathbf{K} = \{K_0, K_1, \ldots, K_{N-1}\}$ is used, which is applied to the *prediction error*[1] that is caused by the influence of the disturbances. In [30] it is suggested to use a set of feedback gains that reduce the prediction error to zero after M steps. Such a set of feedback gains can be found by employing Definition 3.3. Using such a *nilpotent* controller however is not necessary for any of the proofs of feasibility and / or convergence in this text.

Definition 3.3 (M-Step Nilpotent LQR Controller [30])**.** *Given two positive definite matrices Q and R, and two positive integers M and N such that $M < N - 1$. The following backward recursion produces as output a set of linear state feedback gains $\mathbf{K} = \{K_0, K_1, \ldots, K_{N-1}\}$ that drive the state of the nominal system (3-3) to the origin in M steps and remains there until step N:*

[1] At some point in time $k + j$, a measurement of the state, x_{k+j}, may be different from the most recent prediction made by the controller at the instant k, $x_{k+j|k}$. The difference between prediction and measurement shall be called the *prediction error*.

1. *Set* $K_j = 0_{m\times n}, \forall j \in \{M, \ldots, N-1\}$*;*

2. *Set* $P_M = I_n$*,* $S_M = I_n$*;*

3. *Compute in backward for* $j \in \{M-1, M-2, \ldots, 0\}$*:*

$$K_j = -\begin{bmatrix} I_m & 0_{m\times n} \end{bmatrix} H_{j+1}^{\dagger} \begin{bmatrix} B^\top P_{j+1} \\ S_{j+1} \end{bmatrix} A, \quad H_{j+1} = \begin{bmatrix} (R + (B^\top P_{j+1} B)) & B^\top S_{j+1} \\ S_{j+1}B & 0_{n\times n} \end{bmatrix},$$
$$S_j = (A+BK_j)^\top S_{j+1}(A+BK_j),$$
$$P_j = Q + K_j^\top R K_j + (A+BK_j)^\top P_{j+1}(A+BK_j).$$

3-3 Control strategy

A description of RMPC for a finite horizon N follows. Let $\mathbf{U}_{k|k} = \left(u_{k|k}, u_{k+1|k}, \ldots, u_{k+N-1|k}\right)$. Consider the weighted distance function $d(\cdot,\cdot,\cdot)$ as introduced in Definition 1.3, consider F to be a stabilizing feedback gain for the system with matrices (A, B), and consider $\mathbf{K} = \{K_0, K_1, \ldots, K_{N-1}\}$ a sequence of stabilizing feedback gains for the same system. Then, the optimization problem $\mathcal{P}(x_k)$ for a horizon N at the instant k reads as:

$$\mathcal{P}(x_k): \; J(x_k, \mathbf{U}^*_{k|k}) = \min_{\mathbf{U}_{k|k}} J(x_k, \mathbf{U}_k) = \min_{\mathbf{U}_{k|k}} \sum_{i=0}^{N-1} d(\bar{x}_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u_{k+i|k}, \mathcal{T}_{u,i}, R) \tag{3-5a}$$
$$\text{s.t. } \bar{x}_k = x_k \tag{3-5b}$$
$$\bar{x}_{k+i+1|k} = A\bar{x}_{k+i|k} + Bu_{k+i|k}, \; \forall\, i \in \{0,1,\ldots,N-1\} \tag{3-5c}$$
$$u_{k+i|k} \in \mathcal{U}_i, \; \forall\, i \in \{0,1,\ldots,N-1\} \tag{3-5d}$$
$$\bar{x}_{k+i|k} \in \mathcal{X}_i, \; \forall\, i \in \{0,1,\ldots,N-1\} \tag{3-5e}$$
$$\bar{x}_{k+N|k} \in \mathcal{X}_f. \tag{3-5f}$$

The constraint sets in (3-5d) and (3-5e) are defined as follows:

$$\mathcal{U}_0 = \mathbb{U}, \quad \mathcal{U}_{i+1} = \mathcal{U}_i \sim K_i L_i \mathcal{W}, \tag{3-6a}$$
$$\mathcal{X}_0 = \mathbb{X}, \quad \mathcal{X}_{i+1} = \mathcal{X}_i \sim L_i \mathcal{W}, \tag{3-6b}$$

and similarly the target sets are tightened:

$$\mathcal{T}_{u,0} = \mathbb{T}_u, \quad \mathcal{T}_{u,i+1} = \mathcal{T}_{u,i} \sim K_i L_i \mathcal{W}, \tag{3-6c}$$
$$\mathcal{T}_{x,0} = \mathbb{T}_x, \quad \mathcal{T}_{x,i+1} = \mathcal{T}_{x,i} \sim L_i \mathcal{W}, \tag{3-6d}$$

where:

$$L_0 = I_{n\times n}, \quad L_{i+1} = (A + BK_i)L_i, \; \forall i \in \{0,1,\ldots,N-2\}. \tag{3-6e}$$

The terminal state constraint set is given by

$$\mathcal{X}_f = \mathcal{R} \sim L_{N-1}\mathcal{W} \subseteq \mathbb{R}^n, \tag{3-7}$$

where $\mathcal{R}$ is a control invariant admissible set under the disturbances $L_{N-1}\mathcal{W}$ for which the following conditions must hold[2]:

$$x \in \mathcal{R} \Rightarrow \quad (A+BF)x + L_{N-1}w \in \mathcal{R}, \quad \forall w \in \mathcal{W} \tag{3-8a}$$
$$x \in \mathcal{R} \Rightarrow \quad x \in \mathcal{X}_{N-1} \tag{3-8b}$$
$$x \in \mathcal{R} \Rightarrow \quad x \in \mathcal{T}_{x,N-1} \tag{3-8c}$$
$$x \in \mathcal{R} \Rightarrow \quad Fx \in \mathcal{U}_{N-1} \tag{3-8d}$$
$$x \in \mathcal{R} \Rightarrow \quad Fx \in \mathcal{T}_{u,N-1}. \tag{3-8e}$$

Assumption 3.4. *The sets $\mathcal{U}_{N-1}$, $\mathcal{X}_{N-1}$, $\mathcal{T}_{u,N-1}$, $\mathcal{T}_{x,N-1}$ and $\mathcal{X}_f$ are all non-empty sets.*

With the control strategy defined as in this section, the following algorithm can be used to control the system, based on a regular (time triggered) receding horizon implementation of MPC:

3-4 Closed-loop system

This section presents two theorems involving properties of the closed-loop system, that together result in stability of the closed-loop. The first theorem concerns the feasibility of the optimization problems following an initial feasible solution, the second theorem provides a proof showing that when Algorithm 3.5 is used, the states and inputs converge to their target sets.

Notice that for the theorems that are given in this chapter it is possible to make use of a sequence of stabilizing disturbance feedback gains K_i that do not render the closed-loop matrix, $(A+BK_{N-1}) \cdot \ldots \cdot (A+BK_0)$, nilpotent. This property will however simplify the triggering strategies that are presented in the coming chapters.

Algorithm 3.5 Constraint tightening RMPC [30]

```
1: Calculate F according to (3-4), and K_i according to Definition 3.3, with M < N − 1.
2: Initialize by taking k = 0. Then iterate:
3: while k < ∞ do
4:     Measure x_k
5:     Solve P(x_k) (3-5)
6:     Apply u_k = u*_{k|k} from the sequence U*_{k|k} to the system
7:     k ← k + 1,
8: end while
```

3-4-1 Recursive feasibility

Theorem 3.6 (Robust Recursive Feasibility)**.** *Suppose $\mathcal{P}(x_0)$ has a feasible solution, for the dynamics in (3-1), using Algorithm 3.5 to find u_k. Then, (i) the following optimization problems $\mathcal{P}(x_k)$ have a feasible solution for all $k \in \mathbb{Z}_{\geq 0}$. Additionally, (ii) the trajectories of the system (3-1) satisfy the constraints in (3-2).*

[2] Notice that when the disturbance feedback gains render the system *nilpotent* in less than $N-1$ steps, it follows that $L_{N-1} = 0$, and the conditions on the set $\mathcal{X}_f$ become more simple.

Proof. Proof by induction. It is assumed that there is a feasible solution for $\mathcal{P}(x_0)$. The proof below states that when given a feasible solution for $\mathcal{P}(x_k)$, along with the implementation of the first input from the sequence $\mathbf{U}^*_{k|k}$, there exits a feasible candidate solution for the subsequent problem $\mathcal{P}(x_{k+1})$, for all $w_k \in \mathcal{W}$. The proof for claim (ii) results from this feasibility, as feasible solutions for $\mathcal{P}(x_k)$ imply, through (3-5d) and (3-5e) for $i = 0$, satisfaction of the constraints on x_k and u_k in (3-2) .

Consider the case such that at the instant k Problem $\mathcal{P}(x_k)$ has been solved, with its corresponding optimal input and state trajectories $\mathbf{U}^*_{k|k} = \left(u^*_{k|k}, u^*_{k+1|k}, \ldots, u^*_{k+N-1|k}\right)$ and $\mathbf{X}^*_{k|k} = \left(x^*_{k|k}, x^*_{k+1|k}, \ldots, x^*_{k+N|k}\right)$, respectively, satisfying constraints (3-5c)-(3-5f). Define $A_{\text{cl}} := (A + BF)$. At instant $k+1$, it is trivial to show that the disturbance can be derived from the most recent measurement, taking $w_k = x_{k+1} - x^*_{k+1|k}$. Then, a candidate control sequence $\hat{\mathbf{U}}_{k+1|k+1}$ is:

$$\begin{bmatrix} \hat{u}_{k+1|k+1} \\ \hat{u}_{k+2|k+1} \\ \vdots \\ \hat{u}_{k+N-1|k+1} \\ \hat{u}_{k+N|k+1} \end{bmatrix} = \begin{bmatrix} u^*_{k+1|k} \\ u^*_{k+2|k} \\ \vdots \\ u^*_{k+N-1|k} \\ Fx^*_{k+N|k} \end{bmatrix} + \begin{bmatrix} K_0 L_0 \\ K_1 L_1 \\ \vdots \\ K_{N-2} L_{N-2} \\ F L_{N-1} \end{bmatrix} w_k, \tag{3-9a}$$

which results into the following state trajectory $\hat{\mathbf{X}}_{k+1|k+1}$:

$$\begin{bmatrix} \hat{x}_{k+1|k+1} \\ \hat{x}_{k+2|k+1} \\ \vdots \\ \hat{x}_{k+N|k+1} \\ \hat{x}_{k+N+1|k+1} \end{bmatrix} = \begin{bmatrix} x^*_{k+1|k} \\ x^*_{k+2|k} \\ \vdots \\ x^*_{k+N|k} \\ A_{\text{cl}}\hat{x}_{k+N|k+1} \end{bmatrix} + \begin{bmatrix} L_0 \\ L_1 \\ \vdots \\ L_{N-1} \\ 0 \end{bmatrix} w_k. \tag{3-9b}$$

In the following text it is established that the candidate trajectories in (3-9) satisfy the constraints (3-5b)-(3-5f). This will then result in the establishment of $\tilde{\mathbf{U}}_{k+1|k+1}$ being a feasible solution for $\mathcal{P}(x_{k+1})$, if x_{k+1} is given by the system dynamics (3-1).

It holds that $\hat{x}_{k+1|k+1} = x^*_{k+1|k} + L_0 w_k = x^*_{k+1|k} + w_k = x_{k+1|k+1}$ (thereby satisfying the *initial state* constraint 3-5b).

For states further in the sequence, the following holds, making use of the linearity of the dynamics and (3-6e):

$$\begin{aligned} \hat{x}_{k+2|k+1} &= A\underbrace{(x^*_{k+1|k} + L_0 w_k)}_{=\hat{x}_{k+1|k+1}} + B\underbrace{(u^*_{k+1|k} + K_0 L_0 w_k)}_{=\hat{u}_{k+1|k+1}} = x^*_{k+2|k} + \underbrace{L_1}_{=(A+BK_0)L_0} w_k, \\ \hat{x}_{k+3|k+1} &= A\underbrace{(x^*_{k+2|k} + L_1 w_k)}_{=\hat{x}_{k+2|k+1}} + B\underbrace{(u^*_{k+2|k} + K_1 L_1 w_k)}_{=\hat{u}_{k+2|k+1}} = x^*_{k+3|k} + \underbrace{L_2}_{=(A+BK_1)L_1} w_k, \\ &\ldots. \end{aligned}$$

Lastly, to show that also the terminal state of the trajectory $\hat{\mathbf{X}}_{k+1|k+1}$ satisfies the nominal system dynamics, one can write:

$$\hat{x}_{k+N+1|k+1} = A\underbrace{\left(x^*_{k+N|k} + L_{N-1}w_k\right)}_{=\hat{x}_{k+N|k+1}} + BF\underbrace{\left(x^*_{k+N|k} + L_{N-1}w_k\right)}_{=\hat{x}_{k+N|k+1}} = A_{cl}\hat{x}_{k+N|k+1}.$$

Concluding, the *dynamics* constraint (3-5c) is satisfied by the sequences $\hat{\mathbf{U}}_{k+1|k+1}$ and $\hat{\mathbf{X}}_{k+1|k+1}$.

For the constraints on the inputs, one can observe that by the assumption that $\mathbf{U}^*_{k|k}$ is a feasible solution for $\mathcal{P}(x_k)$, $u^*_{k+i+1|k} \in \mathcal{U}_{i+1}, \forall i \in \{0,\ldots,N-2\}$. Additionally, by (3-6a), $\mathcal{U}_{i+1} = \mathcal{U}_i \sim K_i L_i \mathcal{W}$. It therefor follows[3] that $\hat{u}_{k+i+1|k+1} = u^*_{k+i+1|k} + K_i L_i w_k \in \mathcal{U}_i, \forall i \in \{0,\ldots,N-2\}$.

For the last element of the input sequence, it follows from (3-7) and (3-8d) that $\hat{u}_{k+N|k+1} \in \mathcal{U}_{N-1}$:

$$x^*_{k+N|k} \in \mathcal{X}_f \underset{(3-7)}{\Rightarrow} \hat{x}_{k+N|k+1} = x^*_{k+N|k} + L_{N-1}w_k \in \mathcal{R} \underset{(3-8d)}{\Rightarrow} \hat{u}_{k+N|k+1} = F\hat{x}_{k+N|k+1} \in \mathcal{U}_{N-1}.$$

As a result, $\hat{u}_{k+i+1|k+1} \in \mathcal{U}_i$, for all $i \in \{0,\ldots,N-1\}$ and thereby the candidate solution satisfies the constraint (3-5d).

With a similar argument, one can show that the state predictions resulting from the candidate solution are inside their respective sets: from $x^*_{k+i+1|k} \in \mathcal{X}_{i+1}, \forall i \in \{0,\ldots,N-2\}$, and the definition of the Pontryagin difference, certainly $\hat{x}_{k+i+1|k+1} = x^*_{k+i+1|k} + L_i w_k \in \mathcal{X}_i, \forall i \in \{0,\ldots,N-2\}$.

Combining $x^*_{k+N|k} \in \mathcal{X}_f$ with (3-7) gives $\hat{x}_{k+N|k+1} \in \mathcal{R}$. Then (3-8b) implies that $\hat{x}_{k+N|k+1} \in \mathcal{X}_{N-1}$. It is thus shown that $\hat{x}_{k+i+1|k+1} \in \mathcal{X}_i, \forall i \in \{0,\ldots,N-1\}$, so constraint (3-5e) is satisfied by the candidate solution.

Lastly, it needs to be shown that the terminal state $\hat{x}_{k+N+1|k+1} \in \mathcal{X}_f$ (3-5f). This simply follows from previous observations and (3-8a):

$$\hat{x}_{k+N|k+1} \in \mathcal{R} \underset{(3-8a)}{\Rightarrow} A_{cl}\hat{x}_{k+N|k+1} + L_{N-1}w_k \in \mathcal{R}, \forall w_k \in \mathcal{W} \underset{(3-7)}{\Rightarrow} \hat{x}_{k+N+1|k+1} \in \mathcal{X}_f.$$

Having shown that the candidate input and state sequences satisfy all constraints of Problem 3-5, this last observation concludes the proof for claim (i). □

3-4-2 Convergence to target sets

Theorem 3.7 (Robust convergence)**.** *Suppose $\mathcal{P}(x_0)$ has a feasible solution, then the state and input trajectories of the system (3-1), controlled by Algorithm 1, are such that $x_k \to \mathbb{T}_x$ and $u_k \to \mathbb{T}_u$, as $k \to \infty$.*

Proof. Consider the sequences $\hat{\mathbf{U}}_{k+1|k+1}$ (3-9a) and $\hat{\mathbf{X}}_{k+1|k+1}$ (3-9b) as candidate solutions for $\mathcal{P}(x_{k+1})$. In a nutshell, this proof consists of deriving a positive upper bound for $J(x_k, \mathbf{U}^*_{k|k}) - J(x_{k+1}, \hat{\mathbf{U}}_{k+1|k+1})$ that decreases as $k \to \infty$. Notice that $J(x_{k+1}, \mathbf{U}^*_{k+1|k+1}) \leq J(x_{k+1}, \hat{\mathbf{U}}_{k+1|k+1})$,

[3] Making use of Definition 1.1 (Pontryagin difference), and the assumption $w_k \in \mathcal{W}$.

i.e. the optimal solution for $\mathcal{P}(x_{k+1})$ has a cost that is lower than or equal to the cost of the candidate solution.

One can observe that, for all $i \in \{0, 1, \ldots, N-2\}$ and for all $w_k \in \mathcal{W}$:[4]

$$\begin{aligned} d(\hat{u}_{k+i+1|k+1}, \mathcal{T}_{u,i}, R) &\leq d(u^*_{k+i+1|k}, \mathcal{T}_{u,i} \sim K_i L_i \mathcal{W}, R) \\ &= d(u^*_{k+i+1|k}, \mathcal{T}_{u,i+1}, R), \end{aligned} \tag{3-10a}$$

$$\begin{aligned} d(\hat{x}_{k+i+1|k+1}, \mathcal{T}_{x,i}, Q) &\leq d(x^*_{k+i+1|k}, \mathcal{T}_{x,i} \sim L_i \mathcal{W}, Q) \\ &= d(x^*_{k+i+1|k}, \mathcal{T}_{x,i+1}, Q). \end{aligned} \tag{3-10b}$$

Notice that $x^*_{k+N|k} \in \mathcal{X}_f$ from (3-5f). Resulting from (3-8c), it follows that $\hat{x}_{k+N|k+1} \in \mathcal{T}_{x,N-1}$ and from (3-8e) it follows that $\hat{u}_{k+N|k+1} \in \mathcal{T}_{u,N-1}$. Thus, the costs associated to $\hat{x}_{k+N|k+1}$ and $\hat{u}_{k+N|k+1}$ become zero for the candidate sequences. This last observation enables us to determine an upper bound for the cost related to a candidate solution:

$$\begin{aligned} &J(x_{k+1}, \hat{\mathbf{U}}_{k+1|k+1}) = \\ &\sum_{i=0}^{N-1} d(\hat{x}_{k+i+1|k+1}, \mathcal{T}_{x,i}, Q) + d(\hat{u}_{k+i+1|k+1}, \mathcal{T}_{u,i}, R) \\ &\quad \leq \sum_{i=0}^{N-2} d(x^*_{k+i+1|k}, \mathcal{T}_{x,i+1}, Q) + d(u_{k+i+1|k}, \mathcal{T}_{u,i+1}, R) \\ &\quad = J(x_k, \mathbf{U}^*_{k|k}) - d(x^*_{k|k}, \mathcal{T}_{x,0}, Q) - d(u^*_{k|k}, \mathcal{T}_{u,0}, R), \end{aligned} \tag{3-11}$$

as well as an upper bound on the optimal cost for the optimization performed at $k+1$:

$$\begin{aligned} &J(x_{k+1}, \mathbf{U}^*_{k+1|k+1}) \leq J(x_{k+1}, \hat{\mathbf{U}}_{k+1|k+1}) \\ &\leq J(x_k, \mathbf{U}^*_{k|k}) - d(x^*_{k|k}, \mathcal{T}_{x,0}, Q) - d(u^*_{k|k}, \mathcal{T}_{u,0}, R). \end{aligned} \tag{3-12}$$

Since the distance function $d(\cdot, \cdot, \cdot)$ is nonnegative, it holds that $J(x_k, \mathbf{U}_{k|k}) \geq 0$. It follows from (3-12) that $J(x_k, \mathbf{U}^*_k)$ decreases with increasing k and converges to a steady value. This implies $d(x^*_{k|k}, \mathcal{T}_{x,0}, Q) + d(u^*_{k|k}, \mathcal{T}_{u,0}, R) \to 0$, as $k \to \infty$, from which we can conclude that $x_k \to \mathbb{T}_x$ and $u_k \to \mathbb{T}_u$ as $k \to \infty$. The claim follows. □

[4]Making use of Lemma 1.4.

Chapter 4

Centralized Triggering Mechanism

4-1 Introduction

This chapter introduces a triggering strategy that, combined with the Robust Model Predictive Control (RMPC) formulation from Chapter 3, robustly stabilizes a discrete-time linear time-invariant system with a potential for reducing the communication (and possibly computation) load, compared to a time-triggered implementation. The conditions that form the basis of this triggering strategy are inspired by the proofs of stability and convergence, in the sense that a candidate control sequence is constructed and some of its properties are used. If this candidate control sequence and resulting state sequence satisfy some conditions, guarantees can be given concerning the Optimal Control Problem (OCP) that follows from the implementation of an input calculated by the Model Predictive Control (MPC) at the previous update instant. If the criteria are not satisfied, a MPC update is triggered directly. From satisfaction of the conditions at the the previous instant we have some guarantees related to the current OCP (i.e. that it has a feasible solution, and has a lower optimal cost than the optimal cost related to the previously solved OCP). Recursive feasibility and convergence follow these guarantees.

The structure of this chapter is as follows: First, the construction of the candidate control policy is presented, after which sufficient conditions for recursive feasibility and convergence are given. Two theorems concerning the stability of the resulting closed-loop system are derived, after which a discussion concerning the tuning parameters and practical implementation considerations follow.

4-2 Candidate input and resulting state sequence

In this section an alternative algorithm is presented that applies at time step $k + j$ an input, $u^*_{k+j|k}$, that has been calculated when solving the OCP $\mathcal{P}(x_k)$, as long as the candidate sequences satisfy the conditions in the Triggering Mechanism (TM). When one of the conditions in the TM is violated, an MPC update is triggered. Here, j is the number of time steps in the past at which an MPC update was performed for the last time. The main idea is that the TM takes advantage of the measured states x_{k+j} to construct candidate input and state trajectories. These candidates are based on the prediction error and the sequences $\mathbf{X}^*_{k|k}$ and $\mathbf{U}^*_{k|k}$.[1] In short, the TM evaluates if applying the element $u^*_{k+j|k}$ from the input sequence $\mathbf{U}^*_{k|k}$ guarantees: i) feasibility of the OCP, $\mathcal{P}(x_{k+j+1})$, for any $w_{k+j} \in \mathcal{W}$, and ii) convergence of the state x_k and input u_k to their respective target sets $\mathbb{T}_x$ and $\mathbb{T}_u$.

Evaluating the TM thus requires a measurement of the state and the trajectories that have been calculated at the last triggering instant. Communication between the different elements of the control system takes place as follows: (i) At every instant k, the sensors report a measurement to the TM, (ii) at every instant at which an event is triggered, the TM sends the measurements to the controller; the controller sends (after solving $\mathcal{P}(x_k)$) the input trajectory to the actuators and both state and input trajectories to the TM. See for the communication structure Figure 2-2.

4-2-1 Construction

Consider that the instant k is the last instant at which the OCP (3-5), i.e., $\mathcal{P}(x_k)$, has been solved. Based on the optimal input sequence of $\mathcal{P}(x_k)$, a candidate input sequence $\tilde{\mathbf{U}}_{k+j|k+j}$ (4-3a) is created. This candidate sequence has as its first entry $u^*_{k+j|k}$, the entries further towards the horizon are adapted by the feedback gains K_i and the prediction error of the state. The prediction error is given by:

$$e_{k+j} = x_{k+j} - x^*_{k+j|k}. \tag{4-1}$$

The goal of the added disturbance feedback-term is to drive the predicted states in the sequence $\tilde{\mathbf{X}}_{k+j|k+j}$, which results from applying the adapted input sequence, back to the predictions previously made by the MPC. But since the first element of the input sequence is the unadapted input $u^*_{k+j|k}$, the prediction error is allowed to evolve in an open-loop sense for one step before it is rejected using the disturbance feedback gains. The adapted feedback gains $\tilde{K}_i$ and state-transition matrices $\tilde{L}_i$ are found by:[2]

$$\tilde{K}_0 = 0_{m\times n}, \qquad \tilde{K}_{i+1} = K_i, \qquad \forall i \in \{0,\dots,N-2\}, \tag{4-2a}$$
$$\tilde{L}_0 = I_{n\times n}, \qquad \tilde{L}_{i+1} = (A + B\tilde{K}_i)\tilde{L}_i, \qquad \forall i \in \{0,\dots,N-1\}, \tag{4-2b}$$

The candidate input and state sequences $\tilde{\mathbf{U}}_{k+j|k+j}$ and $\tilde{\mathbf{X}}_{k+j|k+j}$ are given by, respectively:

[1] These trajectories are extended by applying state-feedback to the terminal state, in order to find complete sequences.

[2] Notice that $\tilde{L}_1 = A$: the error evolves in open-loop for one time step.

$$\tilde{\mathbf{U}}_{k+j|k+j} = \begin{bmatrix} \tilde{u}_{k+j|k+j} \\ \tilde{u}_{k+j+1|k+j} \\ \tilde{u}_{k+j+2|k+j} \\ \cdots \\ \tilde{u}_{k+N-1|k+j} \\ \tilde{u}_{k+N|k+j} \\ \cdots \\ \tilde{u}_{k+j+N-2|k+j} \\ \tilde{u}_{k+j+N-1|k+j} \end{bmatrix} = \begin{bmatrix} u^*_{k+j|k} \\ u^*_{k+j+1|k} \\ u^*_{k+j+2|k} \\ \vdots \\ u^*_{k+N-1|k} \\ Fx^*_{k+N|k} \\ \vdots \\ FA_{\text{cl}}^{j-2}x^*_{k+N|k} \\ FA_{\text{cl}}^{j-1}x^*_{k+N|k} \end{bmatrix} + \begin{bmatrix} \tilde{K}_0\tilde{L}_0 \\ \tilde{K}_1\tilde{L}_1 \\ \tilde{K}_2\tilde{L}_2 \\ \vdots \\ \tilde{K}_{N-j-1}\tilde{L}_{N-j-1} \\ \tilde{K}_{N-j}\tilde{L}_{N-j} \\ \vdots \\ \tilde{K}_{N-2}\tilde{L}_{N-2} \\ \tilde{K}_{N-1}\tilde{L}_{N-1} \end{bmatrix} e_{k+j}, \tag{4-3a}$$

$$\tilde{\mathbf{X}}_{k+j|k+j} = \begin{bmatrix} \tilde{x}_{k+j|k+j} \\ \tilde{x}_{k+j+1|k+j} \\ \tilde{x}_{k+j+2|k+j} \\ \cdots \\ \tilde{x}_{k+N|k+j} \\ \tilde{x}_{k+N+1|k+j} \\ \cdots \\ \tilde{x}_{k+j+N-1|k+j} \\ \tilde{x}_{k+j+N|k+j} \end{bmatrix} = \begin{bmatrix} x^*_{k+j|k} \\ x^*_{k+j+1|k} \\ x^*_{k+j+2|k} \\ \vdots \\ x^*_{k+N|k} \\ A_{\text{cl}}x^*_{k+N|k} \\ \vdots \\ A_{\text{cl}}^{j-1}x^*_{k+N|k} \\ A_{\text{cl}}^{j}x^*_{k+N|k} \end{bmatrix} + \begin{bmatrix} \tilde{L}_0 \\ \tilde{L}_1 \\ \tilde{L}_2 \\ \vdots \\ \tilde{L}_{N-j} \\ \tilde{L}_{N-j+1} \\ \vdots \\ \tilde{L}_{N-1} \\ \tilde{L}_N \end{bmatrix} e_{k+j}. \tag{4-3b}$$

4-2-2 Triggering strategy

The candidate sequences from (4-3) are employed by Algorithm 4.1. For given state and input sequences, a function that calculates the corresponding cost-function value is:

$$J(\mathbf{X}_{k|k}, \mathbf{U}_{k|k}) = \sum_{i=0}^{N-1} d(x_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u_{k+i|k}, \mathcal{T}_{u,i}, R). \tag{4-4}$$

The function $\alpha(j)$ that is employed by Algorithm 4.1 provides a tuning parameter that can be used to specify some desired rate of convergence, for which $0 \leq \alpha(j) < 1, \forall j \in \{1, \ldots, N-1\}$ needs to hold. It specifies how much the cost function value should decrease as j increases, such that no update is triggered.

4-3 Closed-loop system

In this section two properties are derived for the closed-loop system that is formed when Algorithm 4.1 is used to find inputs for system (3-1): *robust recursive feasibility* of all OCPs at the triggering instants following a feasible state for the first OCP and *robust convergence* of the state and input to their target sets. The properties are derived for the case when L_{N-1}, i.e. $(A + BK_{N-1}) \cdot \ldots \cdot (A + BK_0) = 0_{n\times n}$. When the disturbance feedback gains K_i do not have this property, in addition to the conditions already present in the algorithm, the inclusion $\tilde{x}_{k+j+N|k+j} \in \mathcal{X}_f$ has to be checked as well, as this is not guaranteed to be true.

Algorithm 4.1 Event-triggered constraint tightening RMPC

1: Calculate F according to (3-4), and K_i according to Definition 3.3, with $M < N - 1$.
2: Initialize by setting $k = 0$.
3: Measure x_0
4: **loop**
5:     Solve $\mathcal{P}(x_k)$. Send $\mathbf{U}^*_{k|k}$ to the actuators, implement $u^*_{k|k}$.
6:     $j \leftarrow j + 1$,
7:     Measure x_{k+j} and calculate e_{k+j} using 4-1.
8:
9:     **if** $j > N - 1$ **then**
10:         $k \leftarrow k + j$, $j \leftarrow 0$ ▷ Update triggered
11:         **go to** 5
12:     **end if**
13:
14:     Construct sequences $\tilde{\mathbf{U}}_{k+j|k+j}$ and $\tilde{\mathbf{X}}_{k+j|k+j}$ using (4-3).
15:     **if** $\exists i, \in \{1, ..., N-1\} \mid \tilde{u}_{k+i+j|k} \notin \mathcal{U}_i,$ or $\exists i, \in \{0, ..., N-1\} \mid \tilde{x}_{k+i+j|k} \notin \mathcal{X}_i$ **then**
16:         $k \leftarrow k + j$, $j \leftarrow 0$ ▷ Update triggered
17:         **go to** 5
18:     **else if** $J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j}) > \alpha(j) J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k})$ **then**
19:         $k \leftarrow k + j$, $j \leftarrow 0$ ▷ Update triggered
20:         **go to** 5
21:     **else**
22:         Apply $u_k = u^*_{k+j|k}$ from the solution of $\mathcal{P}(x_k)$. ▷ No update triggered
23:         **go to** 6
24:     **end if**
25: **end loop**

4-3-1 Recursive feasibility for triggered updates

Theorem 4.2 (Aperiodic Robust Recursive Feasibility)**.** *Suppose that for some x_0, $\mathcal{P}(x_0)$ has a feasible solution. Then the trajectories of the system (3-1), controlled by Algorithm 4.1 and subjected to constraints and disturbances as given in (3-2), satisfy the constraints (3-2) and the optimization problems $\mathcal{P}(x_k)$ are feasible for all $k \in \mathbb{Z}_{\geq 0}$.*

Proof. It suffices to prove that for each instant at which no new MPC update is triggered, the MPC problem has a feasible candidate solution at the next instant. For each instant directly following an MPC update, we can directly use the proof from Theorem 3.6.

The adapted candidate solution satisfies constraints (3-5b, 3-5c) by construction, for $i \in \{0, 1, \ldots, N-1\}$ and initial state x_{k+j}. For the initial state (3-5b):

$$\tilde{x}_{k+j|k+j} = x^*_{k+j|k} + e_{k+j} = x_{k+j} = x^*_{k+j|k} + \tilde{L}_0 e_{k+j},$$

and for the dynamics constraints (3-5c):

$$\begin{aligned}
\tilde{x}_{k+j+1|k+j} &= A\underbrace{(x^*_{k+j|k} + \tilde{L}_0 e_{k+j})}_{=\tilde{x}_{k+j|k+j}} + B\underbrace{(u^*_{k+j|k} + \tilde{K}_0 \tilde{L}_0 e_{k+j})}_{=\tilde{u}_{k+j|k+j}} = \\
&= Ax^*_{k+j|k} + Bu^*_{k+j|k} + (A + B\tilde{K}_0)\tilde{L}_0 e_{k+j} = x^*_{k+j+1|k} + \tilde{L}_1 e_{k+j} \\
\tilde{x}_{k+j+2|k+j} &= A\underbrace{(x^*_{k+j+1|k} + \tilde{L}_1 e_{k+j})}_{=\tilde{x}_{k+j+1|k+j}} + B\underbrace{(u^*_{k+j+1|k} + \tilde{K}_1 \tilde{L}_1 e_{k+j})}_{=\tilde{u}_{k+j+1|k+j}} = \\
&= Ax^*_{k+j+1|k} + Bu^*_{k+j+1|k} + (A + B\tilde{K}_1)\tilde{L}_1 e_{k+j} = x^*_{k+j+2|k} + \tilde{L}_2 e_{k+j} \\
&\cdots\cdots \\
\tilde{x}_{k+N|k+j} &= A\underbrace{(x^*_{k+N-1|k} + \tilde{L}_{N-j-1} e_{k+j})}_{=\tilde{x}_{k+N-1|k+j}} + B\underbrace{(u^*_{k+N-1|k} + \tilde{K}_{N-j-1} \tilde{L}_{N-j-1} e_{k+j})}_{=\tilde{u}_{k+N-1|k+j}} = \\
&= Ax^*_{k+N-1|k} + Bu^*_{k+N-1|k} + (A + B\tilde{K}_{N-j-1})\tilde{L}_{N-j-1} = x^*_{k+N|k} + \tilde{L}_{N-j} e_{k+j} \\
\tilde{x}_{k+N+1|k+j} &= A\underbrace{(x^*_{k+N|k} + \tilde{L}_{N-j} e_{k+j})}_{=\tilde{x}_{k+N|k+j}} + B\underbrace{(Fx^*_{k+N|k} + \tilde{K}_{N-j} \tilde{L}_{N-j} e_{k+j})}_{=\tilde{u}_{k+N|k+j}} \\
&= (A + BF)x^*_{k+N|k} + (A + B\tilde{K}_{N-j})\tilde{L}_{N-j} e_{k+j} = A_{cl} x^*_{k+N|k} + \tilde{L}_{N-j+1} e_{k+j} \\
&\cdots\cdots \\
\tilde{x}_{k+j+N-1|k+j} &= A\underbrace{(A_{cl}^{j-2} x^*_{k+N|k} + \tilde{L}_{N-2} e_{k+j})}_{=\tilde{x}_{k+j+N-2|k+j}} + B\underbrace{(FA_{cl}^{j-2} x^*_{k+N|k} + \tilde{K}_{N-2} \tilde{L}_{N-2} e_{k+j})}_{=\tilde{u}_{k+j+N-2|k+j}} \\
&= (A + BF)A^{j-2} x_{k+N|k} + (A + B\tilde{K}_{N-2})\tilde{L}_{N-2} e_{k+j} = A_{cl}^{j-1} x^*_{k+N|k} + \tilde{L}_{N-1} e_{k+j} \\
\tilde{x}_{k+j+N|k+j} &= A\underbrace{(A_{cl}^{j-1} x^*_{k+N|k} + \tilde{L}_{N-1} e_{k+j})}_{=\tilde{x}_{k+j+N-1|k+j}} + B\underbrace{(FA_{cl}^{j-1} x^*_{k+N|k} + \tilde{K}_{N-1} \tilde{L}_{N-1} e_{k+j})}_{=\tilde{u}_{k+j+N-1|k+j}} \\
&= (A + BF)A^{j-1} x^*_{k+N|k} + (A + B\tilde{K}_{N-1})\tilde{L}_{N-1} e_{k+j} = A_{cl}^{j} x^*_{k+N|k} + \tilde{L}_N e_{k+j}
\end{aligned}$$

If the disturbance feedback gains K_i are designed to render the system nilpotent in $M < N-1$ steps, i.e. $L_{N-1} = 0_{n\times n}$, it follows that $\tilde{L}_N = L_{N-1} \cdot A = 0_{n\times n}$. Now, using (3-8a) it can be shown that in such case

$$\tilde{x}_{k+j+N|k+j} = (A+BF)^j x^*_{k+N|k} + \tilde{L}_N e_{k+j} \in \mathcal{X}_f,$$

because $(A+BF)^j x^*_{k+N|k} \in \mathcal{X}_f$. This means that using a nilpotent disturbance feedback controller guarantees that constraint (3-5f) is satisfied for the candidate sequences for Problem $\mathcal{P}(x_{k+j})$.

The only constraints that are not automatically satisfied by the adapted candidate sequences are (3-5d) and (3-5e), depending on the size and direction of e_{k+j} and on how much margin there is left between the optimal trajectories $x^*_{k+j+i|k}$ and $u^*_{k+j+i|k}$ and the constraint set boundaries. Therefore, in order to guarantee feasibility when no update is triggered, the TM needs to check if:

$$\tilde{u}_{k+i+j|k+j} \in \mathcal{U}_i, \forall i, \in \{1, ..., N-1\}, \tag{4-5a}$$
$$\tilde{x}_{k+i+j|k+j} \in \mathcal{X}_i, \forall i, \in \{0, ..., N-1\}. \tag{4-5b}$$

(Notice that $\tilde{u}_{k+j|k+j} = u^*_{k+j|k} \in \mathcal{U}_j \subseteq \mathcal{U}_0$ is automatically satisfied).

An MPC update, i.e. taking $k \leftarrow k+j$, $j \leftarrow 0$ and solving $\mathcal{P}(x_k)$, is triggered directly if one of these conditions is not satisfied. From satisfaction of the TM at the previous instant, or in case of consecutive triggering instants from the recursive feasibility property itself, we know that a feasible solution for this problem exists.

On the other hand, satisfaction of (4-5) means that the candidate sequences satisfy all constraints in the MPC (3-5c)-(3-5f), we conclude that $\tilde{u}_{k+j|k+j} = u^*_{k+j|k}$ can be implemented by the actuators, while feasibility of $\mathcal{P}(x_{k+j+1})$ is guaranteed for $x_{k+j+1} = A\tilde{x}_{k+j|k+j} + Bu^*_{k+j|k} + w_{k+j}$, as long as $w_{k+j} \in \mathcal{W}$. □

4-3-2 Convergence to target sets for triggered updates

Theorem 4.3 (Aperiodic Robust Convergence)**.** *Suppose $\mathcal{P}(x_0)$ has a feasible solution, then the state and input trajectories of system (3-1), controlled by Algorithm 2, are such that $x_k \to \mathbb{T}_x$ and $u_k \to \mathbb{T}_u$, as $k \to \infty$.*

Proof. For each instant directly following an MPC update instant k, we have already shown that $J(x_{k+1}, \mathbf{U}_{k+1|k+1}) \leq J(x_k, \mathbf{U}_{k|k})$, with equality only occurring when $x_k \in \mathbb{T}_x$ and $u_k \in \mathbb{T}_u$.

For $j > 1$ we notice, using the framework introduced in Theorem 3.7, that $J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j})$ is an upper bound the optimal cost related to Problem $\mathcal{P}(x_{k+j})$.

Satisfaction of $J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j}) > \alpha(j) J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k})$ thus guarantees that the inequality $J(x_{k+j}, \mathbf{U}^*_{k+j|k+j}) \leq J(x_k, \mathbf{U}^*_{k|k})$ holds and leads to the convergence of the states and inputs to their corresponding target sets, as long as $\alpha(j) < 1$ for all $j \geq 1$.

□

4-4 Discussion

4-4-1 Convergence rate

The rate of convergence when using Algorithm 4.1 may be slower than when Algorithm 3.5 is employed, as feedback to reduce the effect of disturbances is not applied at every step. The function $\alpha(j)$ is used to trade the number of update instants with the rate of convergence. To have a good balance between the a low number of MPC updates and fast convergence, one can take $\alpha(j) \approx 1$ for $j = 1$, and decrease its value as $j \to N$.

4-4-2 Calculation of TM

Determining if an update is necessary, i.e. evaluating the TM, requires the full state measurement and predicted trajectories to be available at one location. Therefor, the triggering strategy is categorized as a *centralized* approach. Additionally, the computation of the objective function value requires the calculation of some distances from points to the target sets. This means that either a number of small Quadratic Programming (QP) problem has to be solved, or using geometry, a number of projections need to be calculated. These might be an operations that are too complex for a sensory system with limited computational capacity to perform. For these reasons, a *decentralized* approach that uses results derived in this chapter is presented in the following chapter.

4-4-3 Communication structure

An overview of the communication structure for the algorithm that is presented in this chapter is given in Chapter 2.

4-4-4 Delay

A well known complication of the use of Networked Control Systems (NCSs) is the delay that is introduced in the communication between different elements of the control system. Such a delay might be integrated in the design of the TM by introducing additional instants of open-loop dynamics in the state state transition matrices L_i. In such a case, one has to be careful to as well limit M, such that the sequence of disturbance feedback gains $\tilde{K}_i$ still function as a nilpotent feedback controller.

Chapter 5

Decentralized Triggering Mechanism

5-1 Motivation

In the following chapter, we develop an alternative algorithm that does not suffer from the complications that arise when using the triggering strategy introduced in Chapter 4:

- First, in order to evaluate the Triggering Mechanism (TM), the candidate sequences $\tilde{\mathbf{X}}_{k+j|k+j}$ and $\tilde{\mathbf{U}}_{k+j|k+j}$ need to be constructed. For this purpose both the trajectories as calculated by the Model Predictive Control (MPC) and a measurement of the whole state needs to be available by the agent that makes the decision of triggering an update or not. In other words: the TM is *centralized*. Since the aim of the development of the triggering strategy is to reduce the communicational load on the sensory system, such a centralized mechanism reduces drastically the impact of possible benefits resulting from the use of Event Triggered (ET) scheduling of updates.

- A second complication is that the TM requires the evaluation of the MPC objective function for the sequences $\tilde{\mathbf{X}}_{k+j|k+j}$ and $\tilde{\mathbf{U}}_{k+j|k+j}$. This objective function contains distances from points to sets, making the evaluation of the TM computationally expensive. Performing complex computations might not be feasible for an agent co-located with the sensory system, and when it is, takes the allocation of costly energy resources.

For these reasons, in this chapter an alternative triggering strategy is introduced that is 1) *decentralized*, such that each individual sensor can evaluate a part of the TM that only depends on locally available information, and 2) has *low computational complexity*, making it inexpensive for the sensory system perform the necessary calculations. This simplification of the TM comes at the cost of more conservative triggering along with the controller node needing to solve more optimization problems.

5-2 Decentralized TM

In this section a strategy to find (decentralized) bounds on the prediction error is presented. This is done such that when the prediction error is within these bounds, the conditions that need to hold for the candidate sequences from the previous chapter are satisfied.

5-2-1 Bounds on the prediction error

This section introduces a decentralized, low complexity triggering mechanism. Let each sensor, j instants after the last MPC update at instant k, measure the p-th element of the state vector, denoted by x^p_{k+j}. Then a local TM evaluates if this local measurement is between some lower and upper bound:

$$x^{*,p}_{k+j|k} - \underline{e}^p_j \leq x^p_{k+j} \leq x^{*,p}_{k+j|k} + \bar{e}^p_j,\ \forall p \in \{1, \dots, n\}. \tag{5-1}$$

When Inequality (5-1) is satisfied for all $p \in \{1, \dots, n\}$, no update is triggered at instant $k+j$. If one or more violations are detected an update is triggered. The bounds $x^{*,p}_{k+j|k} - \underline{e}^p_j$ and $x^p_{k+j} \leq x^{*,p}_{k+j|k} + \bar{e}^p_j$ are calculated just after an MPC update[1] for $j \in \{1, \dots, N-1\}$ and can be sent in bulk to the sensors.

The satisfaction of Inequality (5-1) for all $p \in \{1, \dots, n\}$ results (by subtraction of $x^{*,p}_{k+j|k}$) in the satisfaction of

$$-\underline{e}^p_j \leq e^p_{k+j} \leq \bar{e}^p_j, \forall p\ \in \{1, \dots, n\}, \tag{5-2}$$

or, using set notation, in

$$e_{k+j} \in \mathcal{E}_j, \tag{5-3}$$

where the sets $\mathcal{E}_j$ are *hyperrectangles* or *n-orthotopes*, given by:

$$\mathcal{E}_j := \{\epsilon \in \mathbb{R}^n \mid -\underline{e}^p_j \leq \epsilon^p \leq \bar{e}^p_j, \forall p\ \in \{1, \dots, n\}\}. \tag{5-4}$$

This set notation for the bounds is used in the subsequent sections.

5-2-2 Finding the bounds

The edges of the sets $\mathcal{E}_j$ are found by solving an optimization problem. The objective is to make $\mathcal{E}_j$ as large as possible to reduce as much as possible the conservatism introduced by the decentralization of the TM. A logarithm on the individual bounds is used to achieve a diminishing marginal increase in the cost function that is maximized. Doing so, a small increase in a bound that is still small gives a larger gain in the objective function value than an equally small increase in a bound that is already large.

The constraints of this optimization problem are chosen such that the inclusion $e_{k+j} \in \mathcal{E}_j$ guarantees satisfaction of the TM from Chapter 4. This result is formalized and proved in Theorem 5.2.

[1]Calculating $\underline{e}^p_j$ and $\bar{e}^p_j$ only requires information that results from the sequences that are calculated by the MPC.

The optimization problem to find the set $\mathcal{E}_j$ is defined as follows:

$$\max_{\bar{e}_j^p, \underline{e}_j^p \in \mathbb{R}^+, \forall p \in \{1,\dots,n\}} \left(\sum_{p=1}^{n} \ln \bar{e}_j^p + \ln \underline{e}_j^p \right) \tag{5-5a}$$

subject to

$$\underline{e}_j^p > 0, \bar{e}_j^p > 0, \forall p \in \{1,\dots,n\} \tag{5-5b}$$

$$x^*_{k+i+j|k} \in \mathcal{X}_i \sim \tilde{L}_i \mathcal{E}_j, \forall i \in \{0,\dots,N-1\} \tag{5-5c}$$

$$u^*_{k+i+j|k} \in \mathcal{U}_i \sim \tilde{K}_i \tilde{L}_i \mathcal{E}_j, \forall i \in \{0,\dots,N-1\} \tag{5-5d}$$

$$s^*_{x,k+i+j|k} \in \mathcal{T}_{x,i} \sim \tilde{L}_i \mathcal{E}_j, \forall i \in \{0,1,\dots,N-1\} \tag{5-5e}$$

$$s^*_{u,k+i+j|k} \in \mathcal{T}_{u,i} \sim \tilde{K}_i \tilde{L}_i \mathcal{E}_j, \forall i \in \{0,1,\dots,N-1\}. \tag{5-5f}$$

In Problem (5-5), $x^*_{k+i+j|k}$ and $u^*_{k+i+j|k}$ are taken from the solution of the MPC problem for $i+j \leq N-1$. To find $x^*_{k+i+j|k}$ and $u^*_{k+i+j|k}$ for $i+j > N-1$, state-feedback is applied to the terminal state:

$$u^*_{k+i+j|k} = F(A+BF)^{i+j-N} x^*_{k+N|k} \tag{5-6a}$$

$$x^*_{k+i+j|k} = (A+BF)^{i+j-N} x^*_{k+N|k}. \tag{5-6b}$$

Consider, $\forall i \in \{0,\dots,N-1\}$:

$$s^*_{x,k+i+j|k} = \arg \min_{s_{x,k+i+j|k} \in \mathcal{T}_{x,i+j}} ||x^*_{k+i+j|k} - s_{x,k+i+j|k}||^2_Q \tag{5-7a}$$

$$s^*_{u,k+i+j|k} = \arg \min_{s_{u,k+i+j|k} \in \mathcal{T}_{u,i+j}} ||u^*_{k+i+j|k} - s_{u,k+i+j|k}||^2_R. \tag{5-7b}$$

These points $s^*_{x,k+i+j|k}$ and $s^*_{u,k+i+j|k}$ are the points within the target sets that have the smallest distance to $x^*_{k+i+j|k}$ and $u^*_{k+i+j|k}$, respectively. They have been calculated by the MPC at the previous update instant for $i+j \in \{j,\dots,N-1\}$.

For $i+j > N-1$, $x^*_{k+i+j|k}$ and $u^*_{k+i+j|k}$ are inside their respective target sets (as these are state-feedback extensions of the terminal state), so we can take:

$$s^*_{x,k+i+j|k} = x^*_{k+i+j|k} \tag{5-8a}$$

$$s^*_{u,k+i+j|k} = u^*_{k+i+j|k}. \tag{5-8b}$$

Notice that for solving Problem (5-5) only information that is known at instant k is necessary, so it can be solved for $j \in \{1,\dots,N-1\}$ directly after the MPC update. Algorithm 5.1 gives an implementation approach for the triggering strategy that is presented in this chapter.

Algorithm 5.1 Event-triggered constraint tightening RMPC

1: Calculate F according to (3-4), and K_i according to Definition 3.3, with $M < N-1$.
2: Initialize by setting $k = 0$, $j = 0$.
3: Measure x_0.
4: **loop**
5:     Collect measurement x_{k+j} from sensors
6:     $k \leftarrow k + j$, $j \leftarrow 0$ ▷ Update
7:     Solve $\mathcal{P}(x_k)$. Send the input sequence $\mathbf{U}^*_{k|k}$ to the actuators
8:     Solve Problem (5-5) for $j \in \{1, \ldots, N-1\}$
9:     Send bounds $x^{*,p}_{k+j|k} - \underline{e}^p_j$ and $x^{*,p}_{k+j|k} + \bar{e}^p_j$ to sensors $p \in \{1, \ldots, n\}$, $\forall j \in \{1, \ldots, N-1\}$
10:     Apply $u_k = u^*_{k|k}$ from the solution of $\mathcal{P}(x_k)$
11:
12:     $j \leftarrow j + 1$,
13:     **if** $j > N - 1$ **then**
14:         **go to** 6 ▷ Update triggered
15:     **end if**
16:
17:     **for** $p \in \{1, \ldots, n\}$ **do**
18:         Measure x^p_{k+j}
19:         **if** $x^p_{k+j} < x^{*,p}_{k+j|k} - \underline{e}^p_j$ or $x^p_{k+j} > x^{*,p}_{k+j|k} + \bar{e}^p_j$ **then**
20:             **go to** 6 ▷ Update triggered
21:         **end if**
22:     **end for**
23:
24:     Apply $u_k = u^*_{k+j|k}$ from the solution of $\mathcal{P}(x_k)$
25:     **go to** 12 ▷ No update triggered
26:
27: **end loop**

5-3 Properties

Theorem 5.2. *Suppose that Problem (3-5) has been solved at instant k for some state x_k. The resulting sequences $\mathbf{U}^*_{k|k} = \{u^*_{k|k}, u^*_{k+1|k}, \ldots, u^*_{k+N-1|k}\}$, $\mathbf{X}^*_{k|k} = \{x^*_{k|k}, x^*_{k+1|k}, \ldots, x^*_{k+N|k}\}$, $\{s^*_{u,k|k}, s^*_{u,k+1|k}, \ldots, s^*_{u,k+N-1|k}\}$ and $\{s^*_{x,k|k}, s^*_{x,k+1|k}, \ldots, s^*_{x,k+N-1|k}\}$ and their corresponding state-feedback extensions, (5-6) and (5-8) are used to solve Problem (5-5) for $j \in \{1, \ldots, N-1\}$, resulting in the sets $\mathcal{E}_j$ for $j \in \{1, \ldots, N-1\}$. Then, for $j \in \{1, \ldots, N-1\}$, if at instant $k+j$ the error $e_{k+j} \in \mathcal{E}_j$, it follows that, with $\tilde{x}_{k+i+j|k+j}$ and $\tilde{u}_{k+i+j|k+j}$ calculated as in (4-3):*

$$\tilde{u}_{k+i+j|k+j} = u^*_{k+i+j|k} + \tilde{K}_i \tilde{L}_i e_{k+j} \in \mathcal{U}_i, \forall i \in \{0, \ldots, N-1\}, \tag{5-9a}$$

$$\tilde{x}_{k+i+j|k+j} = x^*_{k+i+j|k} + \tilde{L}_i e_{k+j} \in \mathcal{X}_i, \forall i \in \{0, \ldots, N-1\}, \tag{5-9b}$$

$$J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j}) \leq J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k}). \tag{5-9c}$$

Furthermore, the equality in (5-9c) only occurs if $J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k}) = 0$.

Proof. The first two results, (5-9a) and (5-9b), follow from the definition of the Pontryagin difference (see Definition 1.1), set multiplication by a matrix (see Definition 1.2) and the constraints in Problem (5-5). We have, from (5-5c):

$$x^*_{k+i+j|k} \in \mathcal{X}_i \sim \tilde{L}_i\mathcal{E}_j, \forall i \in \{0, \dots, N-1\},$$

and from (5-5d):

$$u^*_{k+i+j|k} \in \mathcal{U}_i \sim \tilde{K}_i\tilde{L}_i\mathcal{E}_j, \forall i \in \{0, \dots, N-1\}.$$

Remember the definition of the Pontryagin difference, $\mathcal{A} \sim \mathcal{B} = \{a | a + b \in \mathcal{A},\ \forall b \in \mathcal{B}\}$. Clearly, taking $\mathcal{A} = \mathcal{X}_i$, $a = x^*_{k+i+j|k}$, $\mathcal{B} = \tilde{L}_i\mathcal{E}_j$ and $b = \tilde{L}_i e_{k+j} \in \tilde{L}_i\mathcal{E}_j$, it follows that $\tilde{x}_{k+i+j|k+j} = x^*_{k+i+j|k} + \tilde{L}_i e_{k+j} \in \mathcal{X}_i, \forall i \in \{0, \dots, N-1\}$.

Similarly, taking $\mathcal{A} = \mathcal{U}_i$, $a = u^*_{k+i+j|k}$, $\mathcal{B} = \tilde{K}_i\tilde{L}_i\mathcal{E}_j$ and $b = \tilde{K}_i\tilde{L}_i e_{k+j} \in \tilde{K}_i\tilde{L}_i\mathcal{E}_j$, it follows that $\tilde{u}_{k+i+j|k+j} = u^*_{k+i+j|k} + \tilde{K}_i\tilde{L}_i e_{k+j} \in \mathcal{U}_i, \forall i \in \{0, \dots, N-1\}$.

For result (5-9c), take $\mathcal{T}_{x,i} = \mathcal{T}_{x,N-1}$ and $\mathcal{T}_{u,i} = \mathcal{T}_{u,N-1}$ for all $i > N-1$. We observe that for $i + j \in \{0, \dots, N-1\}$:

$$d(x^*_{k+i+j|k}, \mathcal{T}_{x,i+j}, Q) = d(x^*_{k+i+j|k}, s^*_{x,k+i+j|k}, Q),$$

$$d(u^*_{k+i+j|k}, \mathcal{T}_{u,i+j}, R) = d(u^*_{k+i+j|k}, s^*_{u,k+i+j|k}, R),$$

Now, using (5-5e) and (5-5f), we observe that for $i + j \in \{0, \dots, N-1\}$:

$$d(x^*_{k+i+j|k}, \mathcal{T}_{x,i} \sim \tilde{L}_i\mathcal{E}_j, Q) \le d(x^*_{k+i+j|k}, s^*_{x,k+i+j|k}, Q) = d(x^*_{k+i+j|k}, \mathcal{T}_{x,i+j}, Q),$$

$$d(u^*_{k+i+j|k}, \mathcal{T}_{u,i} \sim \tilde{K}_i\tilde{L}_i\mathcal{E}_j, R) \le d(u^*_{k+i+j|k}, s^*_{u,k+i+j|k}, R) = d(u^*_{k+i+j|k}, \mathcal{T}_{u,i+j}, R).$$

Combining this result with Lemma 1.4 and the fact that $e_{k+j} \in \mathcal{E}_j$, we can conclude that for $i + j \in \{0, \dots, N-1\}$:

$$d(x^*_{k+i+j|k} + \tilde{L}_i e_{k+j}, \mathcal{T}_{x,i}, Q) \le d(x^*_{k+i+j|k}, \mathcal{T}_{x,i+j}, Q), \tag{5-10a}$$
$$d(u^*_{k+i+j|k} + \tilde{K}_i\tilde{L}_i e_{k+j}, \mathcal{T}_{u,i}, R) \le d(u^*_{k+i+j|k}, \mathcal{T}_{u,i+j}, R). \tag{5-10b}$$

We thus have established that if $e_{k+j} \in \mathcal{E}_j$,

$$\begin{aligned} J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j}) &= \sum_{i=0}^{N-1} d(x^*_{k+j+i|k} + \tilde{L}_i e_{k+j}, \mathcal{T}_{x,i}, Q) + d(u^*_{k+j+i|k} + \tilde{K}_i\tilde{L}_i e_{k+j}, \mathcal{T}_{u,i}, R) \\ &\le \sum_{i=0}^{N-1} d(x^*_{k+i+j|k}, \mathcal{T}_{x,i+j}, Q) + d(u^*_{k+i+j|k}, \mathcal{T}_{u,i+j}, R), \end{aligned}$$

because all elements in the sum on the left part of the inequality are smaller or equal to their respective elements in the sum on the right. For the right part it holds that

$$\sum_{i=0}^{N-1} d(x^*_{k+i+j|k}, \mathcal{T}_{x,i+j}, Q) + d(u^*_{k+i+j|k}, \mathcal{T}_{u,i+j}, R) =$$

$$= \sum_{i=j}^{N-1} d(x^*_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u^*_{k+i|k}, \mathcal{T}_{u,i}, R)$$

$$+ \underbrace{\sum_{i=N}^{N-1+j} d(x^*_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u^*_{k+i|k}, \mathcal{T}_{u,i}, R)}_{=0} \leq$$

$$\leq \underbrace{\sum_{i=0}^{j-1} d(x^*_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u^*_{k+i|k}, \mathcal{T}_{u,i}, R)}_{\neq 0 \text{ if } J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k}) \neq 0}$$

$$+ \sum_{i=j}^{N-1} d(x^*_{k+i|k}, \mathcal{T}_{x,i}, Q) + d(u^*_{k+i|k}, \mathcal{T}_{u,i}, R) = J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k}).$$

Thus, it follows that $J(\tilde{\mathbf{X}}_{k+j|k+j}, \tilde{\mathbf{U}}_{k+j|k+j}) \leq J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k})$, with equality only occurring when $J(\mathbf{X}^*_{k|k}, \mathbf{U}^*_{k|k}) = 0$. This concludes the proof. □

5-4 Discussion

5-4-1 Conservativeness of *decentralized* TM

The triggering strategy that is presented in this chapter is a more conservative strategy than the centralized approach as presented in the previous chapter[2]. This means that whenever the decentralized conditions are satisfied, certainly the centralized conditions are satisfied as well, and the condition $e_k + j \in \mathcal{E}_j$ can be checked instead of the centralized conditions from the previous chapter, while having the same properties holding for the closed-loop (i.e. recursive feasibility and convergence of the trajectories to the target sets). Two sources for this conservativeness are treated one by one in the following subsections.

Square sets $\mathcal{E}_j$

The most evident source of the conservativeness of the decentralized TM is the fact that the set of allowable prediction errors is approximated by an *orthotope*. This property is necessary in order to make it possible for the conditions to be evaluated independently (in a decentralized fashion). In fact, the set of all errors that give satisfaction of the centralized conditions might be of a very different shape.

When the sensors for multiple states are located together, making it possible for coupling between the bounds on those specific states to exist. This approach introduces however more

[2] At least, when for the centralized approach no extra emphasis on convergence is given; i.e. take $\alpha(j) = 1-\epsilon$, with ϵ a very small number.

complexity in the optimization problem that is solved to find the error bounds, and is therefor not explored in this thesis.

Objective function

A second source of conservativeness is the way in which convergence of the objective function value is achieved. In the centralized triggering strategy, the actual value of the objective function is calculated for the candidate sequences $\tilde{\mathbf{U}}_{k+j|k+j}$ and $\tilde{\mathbf{X}}_{k+j|k+j}$, and this value is compared to (scaled by $\alpha(j)$) the value of the objective function at the latest MPC update. In the decentralized strategy, for each distance that is part of the objective function, it is made sure that its value is smaller or equal to its equivalent distance in the MPC objective function. The conservativeness lies in the fact that the distances in the objective function for the entries from $i = 0$ to $i = j - 1$ do not play a role in the decentralized TM, whereas in the centralized case they can be used to compensate for larger distances for the entries further in the horizon.

5-4-2 Analysis of optimization problem

The optimization problems that need to be solved in order to find the bounds on the prediction error have a convex objective function, as it is simply the sum of logarithms of the decision variables, and linear constraints. $N - 1$ problems with $2n$ free variables need to be solved after every MPC update, where it is important to note that these do not all need to be solved before the first input as calculated by the MPC is implemented[3]. Appendix A-3 shows how to transform constraints of the form $x \in \mathcal{X} \sim M\mathcal{E}$ into a system of linear inequalities with as variable the bounds in $\mathcal{E}$.

5-4-3 Communication structure

An overview of the communication structure for the algorithm that is presented in this chapter is given in Chapter 2.

5-4-4 Weights in optimization

When it is known that the disturbance affects some states more than other ones, or that deviations in some state affect the disturbance feedback more than other states, one might prefer that the error bounds for these specific states are larger than other ones. The objective function that finds the error bounds (5-5) may be adapted such that these preferences are reflected by the cost related to these specific bounds, for instance by introducing weights on the bounds for the various states. As long as the solution satisfies the constraints in (5-5), the results as derived in Theorem 5.2 hold. To keep the presented approach simple however, this possibility is not explored further here.

[3] It is though preferable from a communication point of view that before the first measurement is taken, all bounds are already sent to the sensors, making use of sending information in bulk.

5-4-5 Non-nilpotent K_i

Similarly as for the centralized triggering strategy, disturbance feedback gains that do not drive the prediction error to zero in less than $N-1$ steps can be used. For the decentralized conditions that are derived in this chapter, this means that an extra constraint needs to be introduced in Problem (5-5).

Chapter 6

Simulation Experiments

6-1 Introduction

6-1-1 Motivation

This chapter presents simulation results for different scenarios, with the centralized and decentralized triggering mechanisms as proposed in the previous chapters responsible for scheduling the Model Predictive Control (MPC) updates. The goal of these simulations is to validate the theorems that have been derived in the previous chapters and to indicate qualitatively the benefit that can be achieved by using such an Event Triggered (ET) scheduling approach over a periodic time triggered approach.

6-1-2 Simulation scenarios

The system that is used to perform simulations on is a second order system with both eigenvalues outside the unit circle. Simulation experiments are performed using two scenarios. The first scenario considers the transient-behavior of the dynamics under the proposed triggering strategies. In the second scenario the strategies are evaluated for their performance when disturbances have to be rejected.

In the first scenario, the initial state is picked to be non-zero, and close to the system constraints. At each instant, the disturbance that acts on the system is chosen such that it has the worst-case effect on the state, i.e. it maximizes the norm of the state at the next instant[1]. The goal of this experiment is to validate that the theorems concerning recursive feasibility and convergence (Theorems 4.2, 4.3) and concerning the decentralized Triggering Mechanism (TM) (Theorem 5.2) hold.

The second scenario is aimed at getting an estimate on how much benefit the proposed triggering strategy can achieve when the system is subjected to additive disturbances. Two

[1]This *worst-case* disturbance is defined as follows: $w_k = \arg\max_{w_k \in \mathcal{W}} ||Ax_k + Bu_k + w_k||_2 = \arg\max_{w_k \in \mathcal{W}} ||x_{k+1}||_2$

types of disturbances are considered: one in which the disturbance is again selected as the worst-case within the allowable set, in the second experiment the disturbance is sampled from a uniform probability function, that has the size of the set $\mathcal{W}$. In all cases, a comparison is made between the centralized and decentralized TMs using the same disturbance sequence for both simulations.

6-2 Second order system

6-2-1 System description

The dynamics of the second order system are given by:

$$x_{k+1} = \begin{bmatrix} 0.8 & -0.8 \\ 0.8 & 0.8 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k + w_k. \tag{6-1}$$

The constraints on the state, $x_k \in \mathbb{X}$, the input $u_k \in \mathbb{U}$, the bounds on the disturbance $w_k \in \mathcal{W}$, and the target sets are given by:

$$\mathbb{X} = \{x \in \mathbb{R}^2 \mid ||x||_\infty \leq 7.5\}, \qquad \mathbb{U} = \{u \in \mathbb{R} \mid -1 \leq u \leq 3\}, \tag{6-2a}$$

$$\mathbb{T}_x = \{x \in \mathbb{R}^2 \mid ||x||_\infty \leq 0.75\}, \qquad \mathbb{T}_u = \{u \in \mathbb{R} \mid |u| \leq 0.8\}, \tag{6-2b}$$

$$\mathcal{W} = \{w \in \mathbb{R}^2 \mid ||w||_\infty \leq 0.1\}, \qquad \mathcal{X}_f = \{x \in \mathbb{R}^2 \mid ||x||_\infty \leq 0.2\}. \tag{6-2c}$$

The following parameters are used in the simulation:

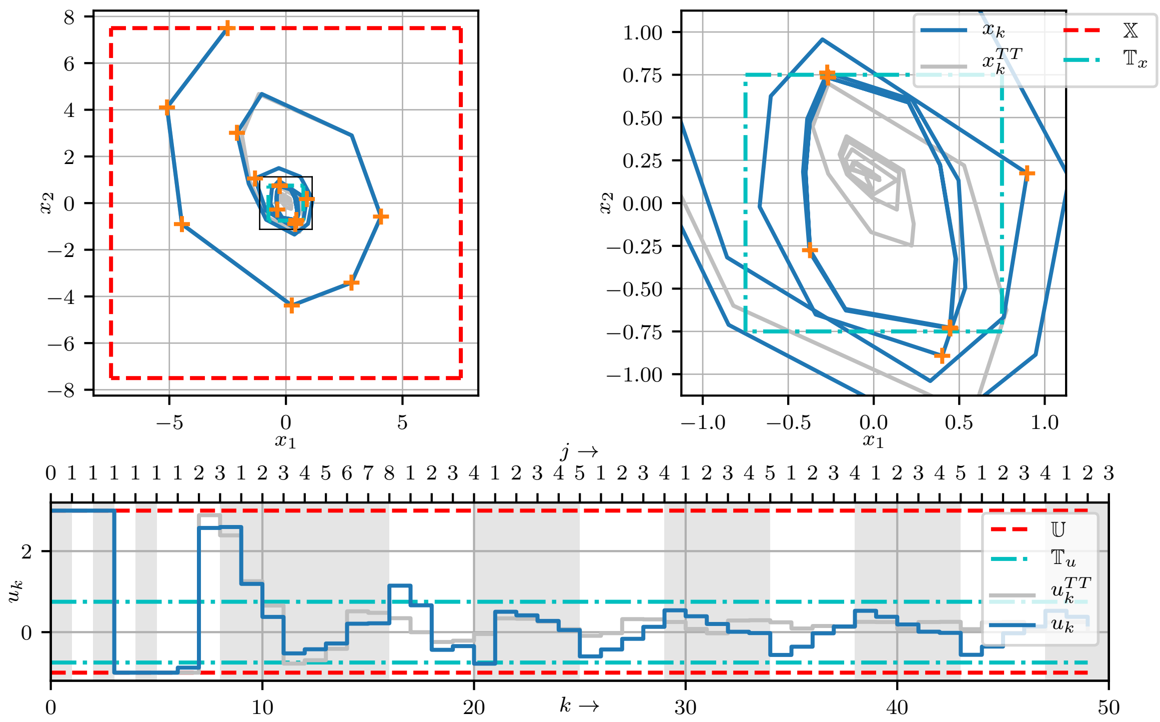
$$Q = I_2,\ R = 1,\ N = 15,\ M = 10,\ \alpha(j) = e^{-\frac{j}{5}}. \tag{6-3}$$

Using this information, the disturbance feedback gains K_i, are found using Definition (3.3) and $\tilde{K}_i$ using Equation (4-2a). The state transition matrices L_i, $\tilde{L}_i$ are found by using Equations (3-6e) and (4-2b), respectively. The tightened constraint sets and target sets are calculated according to the respective formulas in (3-6).

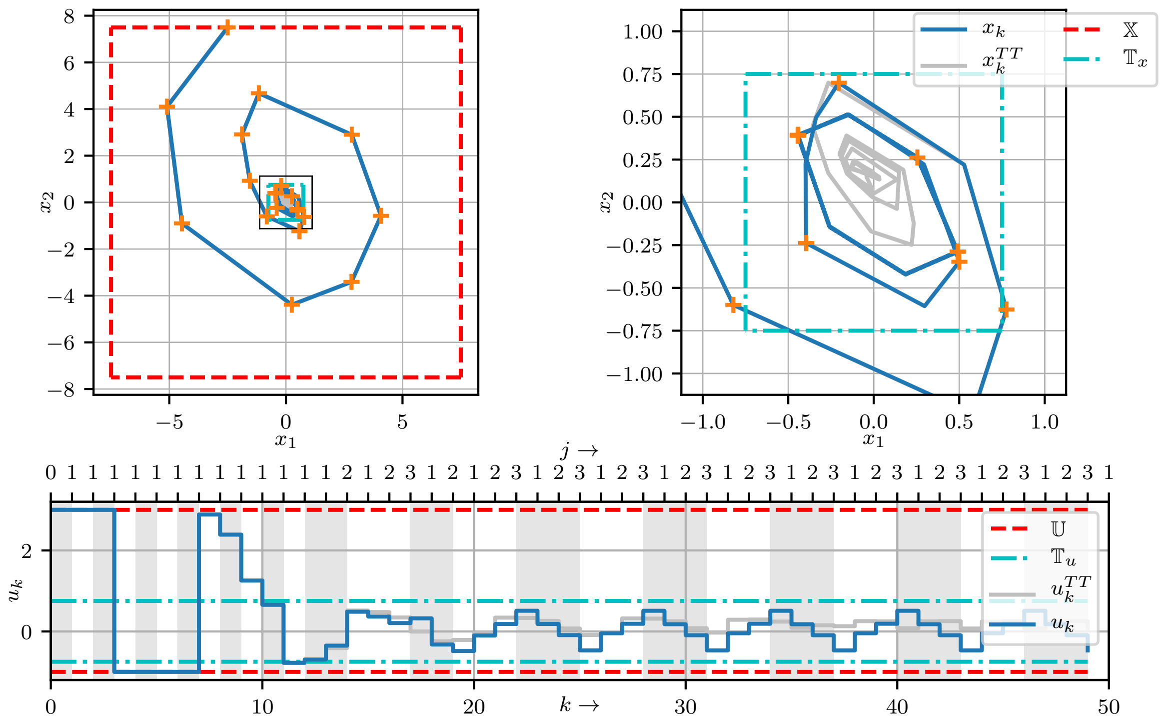
6-2-2 Transient response

The initial state for the transient response is selected to be $x_0 = \begin{bmatrix} -2.5 & 7.5 \end{bmatrix}^\top$, at an edge of the state constraint set $\mathcal{X}$, and within the region for which, with the horizons and constraints defined as in (6-3), a feasible solution for the MPC problem can be found. The simulation results for the worst-case disturbance are shown in Figure 6-1, Figure 6-2 shows the results for the case when the disturbance is drawn randomly from a uniform distribution.

Despite of the *worst-case* disturbance, both TMs achieve convergence of the state and input to the target set, while at all times satisfying the constraints. Initially, when the state and input are relatively close to their respective constraint boundaries, both TMs trigger updates at every instant. When the state and input converge, and move away from the constraint boundaries, the advantage of using the TMs becomes clear: updates are triggered more sparsely.

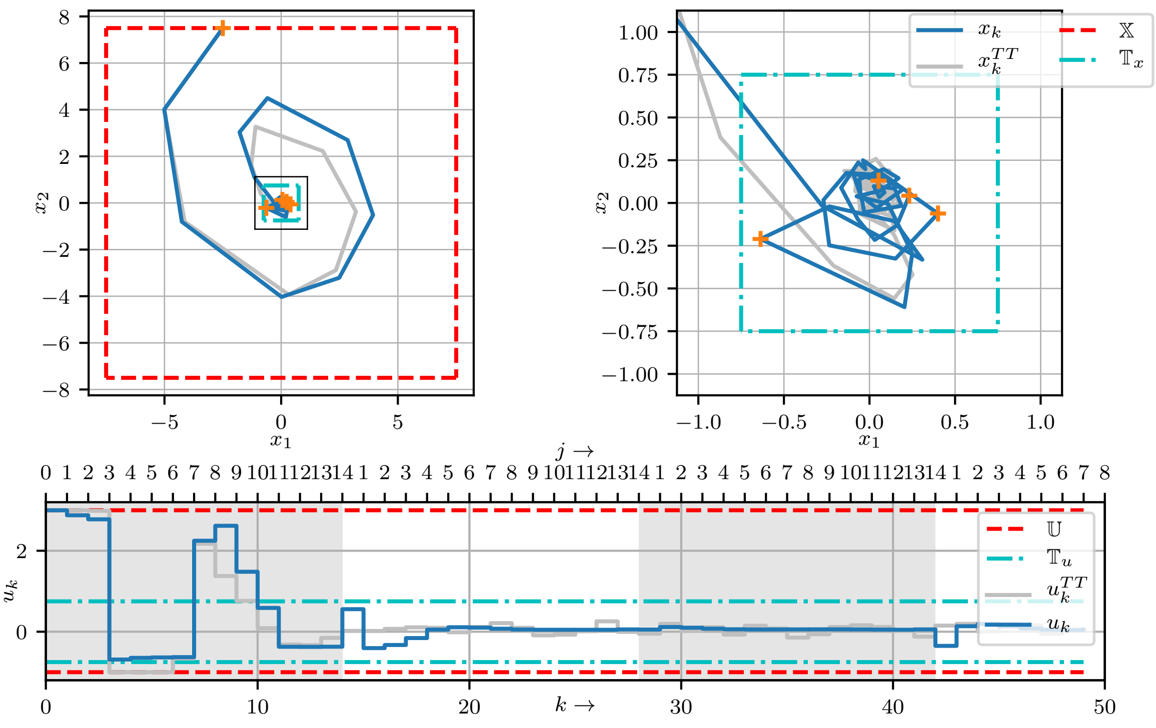


(a) Centralized TM, worst-case disturbance

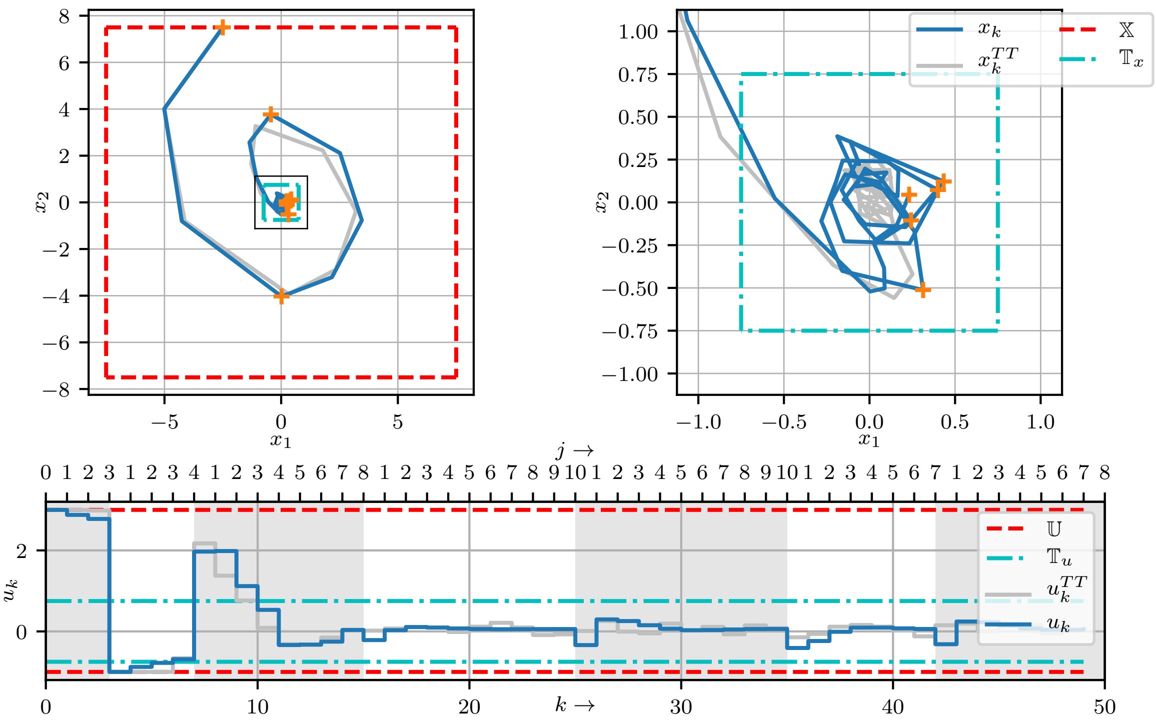


(b) Decentralized TM, worst-case disturbance

Figure 6-1: Simulation results for the transient scenario and the worst-case disturbance. Top left: Phase-plot for the state, with full view of the set $\mathbb{X}$. Top right: Close-up of the target set $\mathbb{T}_x$. Triggering instants are marked with a '+'. Bottom: input signal, with changing background color indicating a triggering instant.

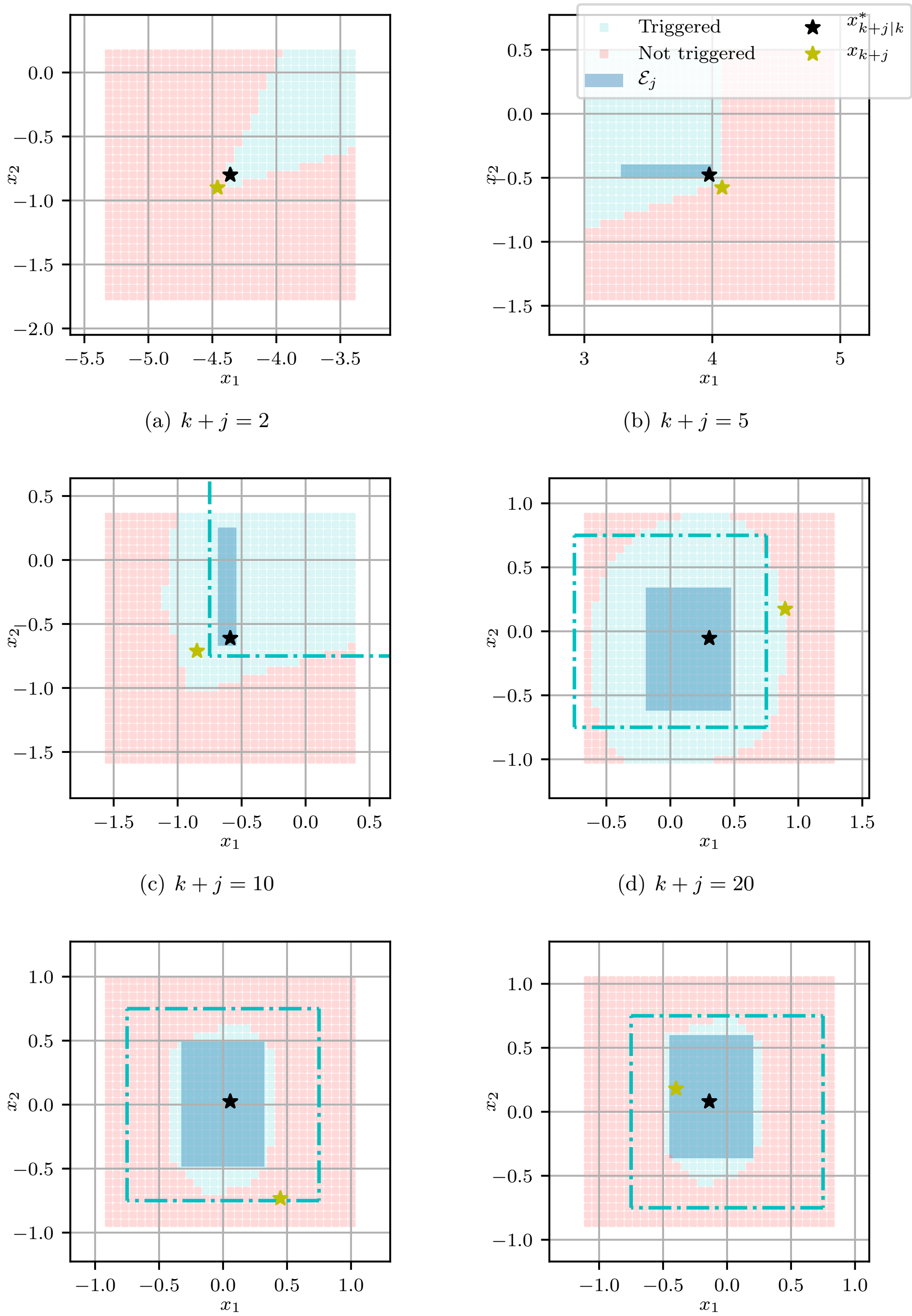


(a) Centralized TM, random disturbance



(b) Decentralized TM, random disturbance

Figure 6-2: Simulation results for the transient scenario and the uniformly distributed disturbance. Top left: Phase-plot for the state, with full view of the set $\mathbb{X}$. Top right: Close-up of the target set $\mathbb{T}_x$. Triggering instants are marked with a '+'. Bottom: input signal, with changing background color indicating a triggering instant.



(a) $k+j=2$

(b) $k+j=5$

(c) $k+j=10$

(d) $k+j=20$

(e) $k+j=34$

(f) $k+j=49$ (end of simulation)

Figure 6-3: $\mathcal{E}_j$ and centralized triggering for various $k+j$, for the simulation in Figure 6-1(a).

As expected, the centralized TM is less conservative than the decentralized TM. This conservatism appears in the form of a larger number of instants between two events, and in the state being allowed to come closer to the target set boundary before an event is triggered for the centralized TM.

For the *random* disturbance the same observations can be made, with as the most noticeable difference that triggering occurs much less often. For the centralized case, though the prediction error becomes quite large at some points, triggering only occurs because the *actuators* run out of calculated inputs. The decentralized TM triggers earlier, as it allows for smaller prediction errors.

Figure 6-3 shows, for some specific instants from the simulation of Figure 6-1(a), the set $\mathcal{E}_j$, which is known explicitly, and for a grid of points around the predicted state if the centralized TM would have triggered. This figure confirms the before-mentioned conservativeness of the decentralized TM: for all instants the set $\mathcal{E}_j$ is smaller than the set of points that would have made the centralized TM trigger an update event. This conservativeness is large when the simulation is started, but as the state converges to the target set, it diminishes gradually.

6-2-3 Disturbance rejection

Another simulation experiment is performed with the same system in order to assess the number of samples between two updates triggered by the TMs. Figures 6-4 and 6-5 show the number of updates that are triggered for disturbances that are contained in the set $\mathcal{W}$, for various sizes of this set. In case of a random disturbance, for each size of the disturbance set a simulation of 2000 samples is executed, in case the disturbance is picked as the worst-case simulations of only 100 samples are performed. The initial state is set as $x_0 = [-0.75\ 0.75]^\top$, which is one of the vertices of the target set $\mathbb{T}_x$. The horizon N is increased to 25 in order to have most updates being triggered by the TM itself, instead of by the actuators running out of inputs.

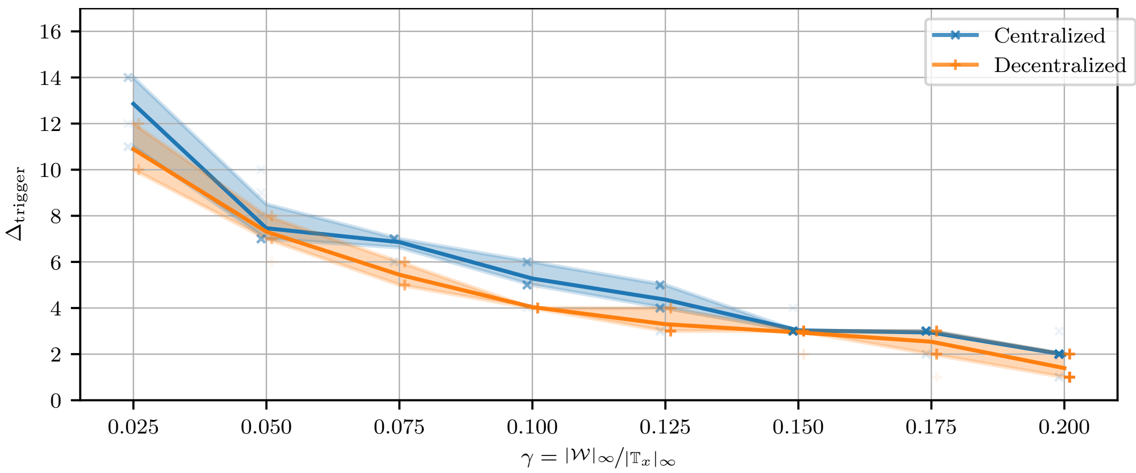


Figure 6-4: Number of time steps between MPC updates for worst-case disturbances, with $w_k \in \mathcal{W} = \gamma \cdot \mathbb{T}_x$. The solid lines show the mean values, the shaded areas mark the region in which 75% of the triggering takes place.

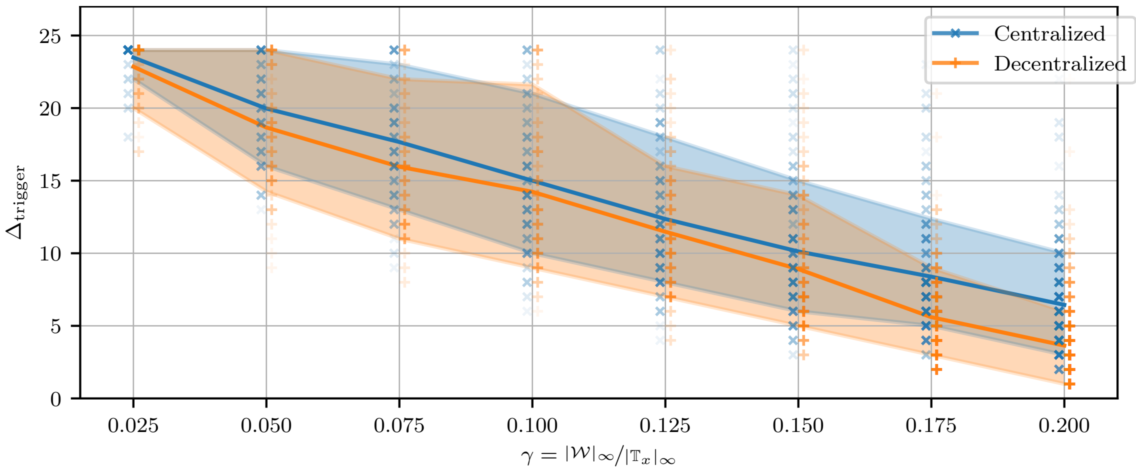
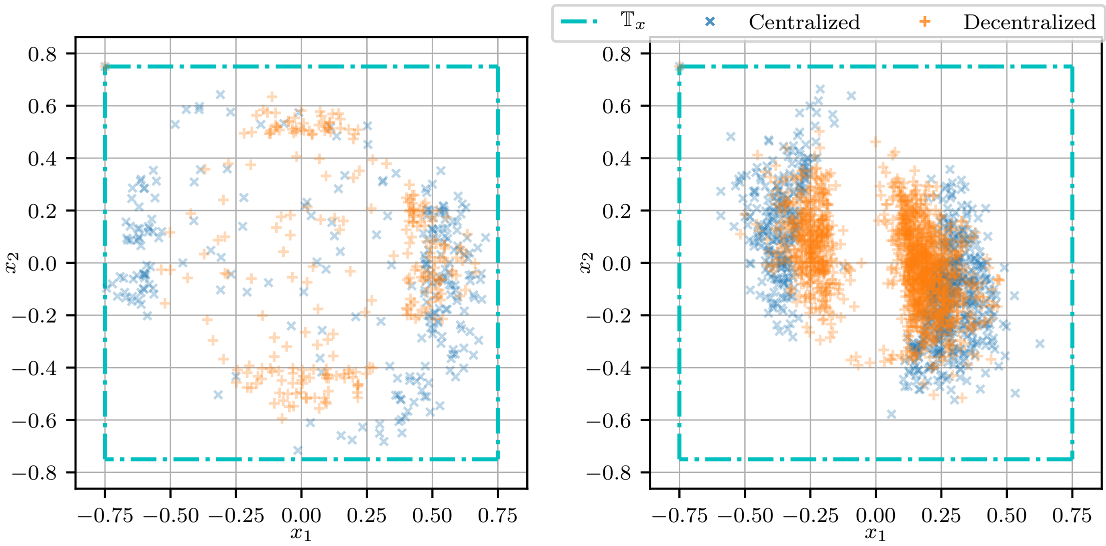


Figure 6-5: Number of time steps between MPC updates for random disturbances, with $w_k \in \mathcal{W} = \gamma \cdot \mathbb{T}_x$. The solid lines show the mean values, the shaded areas mark the region in which 75% of the triggering takes place.

Even for a relatively large set $\mathcal{W}$ the average number of consecutive instants without an update being triggered is considerably larger than 1. As has been observed previously, the decentralized TM triggers more conservatively. Figure 6-6 shows the states at which an update is triggered for two different sizes of $\mathcal{W}$, and for a simulation duration of 5000 steps. The conservativeness of the decentralized TM is exposed by the fact that the decentralized TM triggers whenever the state is further away from the edge of the target set compared to the centralized TM. Obviously, both TMs trigger updates with a larger margin from the edge of the target set whenever the set $\mathcal{W}$ is larger.



(a) $\mathcal{W} = 0.05 \cdot \mathbb{T}_x$

(b) $\mathcal{W} = 0.20 \cdot \mathbb{T}_x$

Figure 6-6: State x_k at the instants when updates are triggered, for two sizes of $\mathcal{W}$.

Chapter 7

Discussion and Conclusions

In this section several aspects of the Triggering Mechanisms (TMs) that have been presented in this thesis are discussed. First, some technical considerations are given concerning the classification of the presented TMs with regard to existing approaches. Subsequently, a discussion on some practical aspects of the given approach is given, with emphasis on the way the bounds are determined, the computational complexity and other limitations that the approach suffers from. This section is concluded with some recommendations for future directions that could follow the development that has been made in this thesis, with emphasis on the application on practical systems.

7-1 Technical considerations

7-1-1 Classification

The triggering strategies that have been presented in this thesis can be classified as Event Triggered (ET) approaches, as updates are triggered whenever the sensory system records some state measurement that violates some conditions. The approaches differ however in the way in which these conditions are determined. Where for the existing Event Triggered Robust Model Predictive Control (ET-RMPC) approaches (see for instance [33], [34], [35] and [36]) the bounds are determined / designed beforehand, in the presented approaches the bounds are variable and determined *on the fly* based on the state[1] measurement. The TMs that are proposed in this thesis can therefore be regarded as *closed-loop* in their nature, where the existing approaches have a TM that is *open-loop* in its nature.

7-1-2 Bounds

This closed-loop character of the triggering conditions may have a beneficial effect on the amount of triggering that takes place, making the error bounds small when necessary and

[1]Or more precisely, the bounds depend on the solution sequences for the state and input that are determined by the Model Predictive Control (MPC) at the previous update.

large whenever the solution sequences of the MPC allow this. Quantifying this benefit is however not a straightforward task. With the size of the bounds depending on the whole sequences $\mathbf{X}^*$ and $\mathbf{U}^*$ (the optimal state and input trajectories), many different parameters influence them. These parameters include the tuning of the disturbance feedback gains K_i and the state-feedback gain F, the horizon length N and the tuning matrices Q and R.

7-1-3 MPC related limitations

The presented control/triggering approach suffers from some limitations. It shares some limitations with a time triggered implementation of MPC, such as a high sensitivity for modeling errors, the need to solve an optimization problem for finding the control signal and limitations that originate from guaranteeing convergence (such as limitations on the reachable set, a minimum size for the target sets, minimum control/prediction horizon, etc). In addition to these MPC-related limitations, the working principles behind the developed triggering strategies depend heavily on the linearity of the system dynamics, specifically because the evolution of the state is taken as the superposition of the predicted dynamics and prediction error dynamics. This last limitation is one that other ET-RMPC approaches (such as [33], [34], [35] and [36]) suffer from as well.

7-1-4 Computational complexity

The presented approach does not alter the optimization problem that is solved to find the control inputs, so the computational complexity of the MPC is equal to that of the ordinary *constraint tightening* form of MPC ($\mathcal{O}((m \times N)^3)$). The complexity of the evaluation of the centralized TM might seem quite high, as distances from fixed points to variable points within some polytope exist in the cost-function value that follows from the candidate sequences. However, using geometrical relations this problem can be solved by only evaluating linear inequalities and calculating dot products and scalar divisions. It depends on the hardware located at the intelligent sensory system if performing these tasks is feasible. For the decentralized approach, evaluating the triggering conditions itself is a task of very low complexity, though calculating the actual bounds involves solving an optimization problem for every step in the future for which bounds need to be calculated. These optimization problems have $2 \cdot n$ free variables, considerably less than the MPC problem with $(m \times N)$ free variables. The solution of these problems only depends on the solution of the MPC problem itself, and therefore they can be solved in parallel. Compared to for instance the Self Triggered (ST)-MPC approaches from ([22], [23] and [24]), where a considerable effort is spend in maximizing the number of instants between updates, the computational complexity of the presented algorithms is much lower.

7-2 Recommendations

7-2-1 Delay

When there is a (small) delay between the time at which an update is triggered and the time at which the control input can be updated, this could be incorporated in the TMs by taking into account more instants at which the prediction error evolves in open-loop, without adding the disturbance feedback term. Finding proofs for the stability of the resulting closed-loop system is left as an open problem.

7-2-2 Simplification

The optimization problem that need to be solved for finding the error bounds may be simplified by predetermining some regions in the state-space, and to check which is the largest of these regions that satisfies the constraints of this optimization problem. Communication between the controller and the TMs can then be simplified as well; i.e. only an identifier of the region in which the state should remain at which instant has to be communicated instead of the numerical values of the bounds of some variable regions. Another way of reducing the load of computing the regions would be to stop computing the bounds whenever they become so small that it is likely that they are violated regardless, and instead always triggering an update when the system reaches that instant.

7-2-3 Asynchronous communication

The presented algorithms share the property that once an update is triggered, all the sensors are to send their measurements to the controller. Altering the algorithms such that only a sensor that violates its current triggering condition needs to send its measurement could dramatically decrease the communication load related to the algorithms.

7-2-4 Distributed Robust MPC

Following the decentralized and distributed formulations of Robust MPC by constraint tightening as described in [38] and [39], the triggering of communication and controller updates could be made suitable for systems with states that have decoupled dynamics, but for which coupling trough common inequality constraints exists.

7-2-5 Practical validation

The results presented in the theorems that are derived in this text need to be validated to work in practice. Challenges that arise when attempting this could be for instance capturing the various uncertainties in the set $\mathcal{W}$, or in the robustness of the algorithm with respect to unknown and / or variable delays.

Appendix A

Convex Polytopes - Pontryagin difference

This appendix summarizes some results found in literature regarding operations on convex polytopes. In the literature, [40] covers some basic properties of polytopes, including an introduction to the $\mathcal{H}$ (half-space) and $\mathcal{V}$ (vertices) representations. An algorithm to find a minimal $\mathcal{H}$-representation of a polytope is described in [41]. Finally, [42] gives a more detailed analysis on the Pontryagin difference, partly for its application in (constrained) control systems.

A-1 Descriptions of polytopes

A-1-1 $\mathcal{V}$-polytopes

A $\mathcal{V}$-polytope of dimension d is the convex hull of some finite set of points in $\mathbb{R}^d$. The set of points $\mathcal{X} = \{x_1, \ldots, x_N\}, x_i \in \mathbb{R}^d, \forall i \in \{1, \ldots, N\}$ defines the polytope $\mathcal{P}$ as:

$$P = \text{conv}(\mathcal{X}) := \{\sum_{i=1}^{N} \lambda_i x_i \mid \lambda_i, \ldots, \lambda_N \geq 0, \sum_{i=1}^{N} \lambda_i = 1\}. \tag{A-1}$$

A-1-2 $\mathcal{H}$-polytopes

Alternatively, one can describe a polytope of dimension d as the solution set of some system of a finite number of linear inequalities:

$$\mathcal{P} = \mathcal{P}(A, b) := \{x \in \mathbb{R}^d \mid Ax \leq b\}, \tag{A-2}$$

where $A \in \mathbb{R}^{n \times d}$, $b \in \mathbb{R}^n$, and where the inequalities in $Ax \leq b$ are taken row-wise.

A-1-3 Equivalence of descriptions

The $\mathcal{V}$- and $\mathcal{H}$-representations of polyhedrons are equivalent, see for instance [40] and [43] (this result is also called the *Main Theorem of Polytope Theory*). This means that every $\mathcal{V}$-polytope has a $\mathcal{H}$-representation (such that it is the intersection of a finite number of closed half-spaces), and that every $\mathcal{H}$-polytope can be described as the convex hull of some finite number of points. In this text polytopes are used to describe constraints and bounds on disturbances / error signals, which naturally take the form of inequalities. Therefor in most cases only the $\mathcal{H}$-representation of polytopes is used.

A-2 Pontryagin Difference for Polytopes in $\mathcal{H}$-description

The Pontryagin difference between two polytopes is especially relevant for this text. For general sets $\mathcal{A}$ and $\mathcal{B}$, it is defined as:

$$\mathcal{A} \sim \mathcal{B} = \{a | a + b \in \mathcal{A},\ \forall b \in \mathcal{B}\}.$$

We assume that $\mathcal{A}$ is a polytope, i.e.

$$\mathcal{A} = \{x \in \mathbb{R}^d \mid a_i x \leq b_i, \forall i \in \{1, \dots, N\}\},$$

where $a_i \in \mathbb{R}^{1\times d}$ are the rows of a matrix $A \in \mathbb{R}^{N\times d}$ and b_i is the i-th element of the vector $b \in \mathbb{R}^N$. The the Pontryagin difference can then be calculated by [42]:

$$\mathcal{A} \sim \mathcal{B} = \{x \in \mathbb{R}^d \mid a_i x \leq b_i - h_{\mathcal{B}}(a_i^\top), \forall i \in \{1, \dots, N\}\}. \tag{A-3}$$

If $\mathcal{B}$ is a polytope with $\mathcal{B}(A_{\mathcal{B}}, b_{\mathcal{B}})$, the support $h_{\mathcal{B}}(a_i^\top)$ can be found by solving the following Linear Programming (LP) problem:

$$\begin{aligned} h_{\mathcal{B}}(a_i^\top) = & \max_{\eta} a_i \eta \\ & s.t.\ A_{\mathcal{B}} \eta \leq b_{\mathcal{B}}. \end{aligned}$$

A-3 Pontryagin difference in constraint

In the optimization problem that is introduced in Chapter 5, we encounter an optimization problem that has the following form:

$$\min_{\bar{e}_i, \underline{e}_i} \sum_{i=1}^{n} -\log(\bar{e}_i) - \log(\underline{e}_i) \tag{A-4a}$$

$$s.t.\ x \in \mathcal{X} \sim M\mathcal{E} \tag{A-4b}$$

Here $x \in \mathbb{R}^p$ is some given point, $\mathcal{X}$ a given polytope such that $x \in \mathcal{X}$, $M \in \mathbb{R}^{p\times n}$ is a matrix and the set $\mathcal{E}$ is an *n-orthotope* that satisfies $\mathcal{E} = \{\epsilon \in \mathbb{R}^n \mid A_{\mathcal{E}} \epsilon \leq b_{\mathcal{E}}\}$, with $A_{\mathcal{E}} = \begin{bmatrix} I_n \\ -I_n \end{bmatrix}$

and $b_{\mathcal{E}} = \begin{bmatrix} \bar{e}_1 \\ \dots \\ \bar{e}_n \\ \underline{e}_1 \\ \dots \\ \underline{e}_n \end{bmatrix}$, where $\bar{e}_i > 0$ and $\underline{e}_i > 0$.

This section aims to find a matrix A_{simple} and vector b_{simple} such that the following relation holds:

$$A_{simple} b_{\mathcal{E}} \leq b_{simple} \Leftrightarrow x \in M\mathcal{E}.$$

When such a form is found, the constraint in Problem (A-4a) can be represented by a system of linear inequalities with as variable the bounds $\bar{e}_i, \underline{e}_i$, and the problem can be solved by using an existing convex optimization solvers.

A-3-1 Method

Given is the constraint

$$x \in \mathcal{X} \sim M\mathcal{E}. \tag{A-5}$$

With $\mathcal{X}$ a given polytope in $\mathcal{H}$-representation by $A_{\mathcal{X}}$ and $b_{\mathcal{X}}$. We assume that $x \in \mathcal{X}$. $\mathcal{E}$ is a polytope in $\mathcal{H}$-representation determined by $A_{\mathcal{E}}$ and $b_{\mathcal{E}}$. We can write (A-5) as

$$a_{\mathcal{X},i} x \leq b_{\mathcal{X},i} - h_{M\mathcal{E}}(a_{\mathcal{X},i}), \forall i \in \{1, \dots, p\}.$$

Now transporting all constant elements to the right and the variable elements to the left, we find:

$$h_{\mathcal{ME}}(a_{\mathcal{X},i}) \leq b_{\mathcal{X},i} - a_{\mathcal{X},i} x, \forall i \in \{1, \dots, p\}.$$

Clearly, on the right side we can take $b_{simple} = b_{\mathcal{X}} - A_{\mathcal{X}} x$. Note that b_{simple} does not contain any variables that depend on $\mathcal{E}$.

For the left side, we make the substitution

$$h_{\mathcal{ME}}(a_{\mathcal{X},i}) = \max_{\eta \in M\mathcal{E}} a_{\mathcal{X},i} \eta = [\max_{\bar{\mu}_1 \in M\bar{1}_1} a_{\mathcal{X},i} \bar{\mu}_1, \dots, \max_{\bar{\mu}_n \in M\bar{1}_n} a_{\mathcal{X},i} \bar{\mu}_n, \max_{\underline{\mu}_1 \in M\underline{1}_1} a_{\mathcal{X},i} \underline{\mu}_1, \dots, \max_{\underline{\mu}_n \in M\underline{1}_n} a_{\mathcal{X},i} \underline{\mu}_n] \, b_{\mathcal{E}}.$$

In this formula, the sets $\bar{1}_i$ and $\underline{1}_i$ are defined as follows:

$$\bar{1}_i := \{\bar{\mu} \mid \bar{\mu}_j = 0 \; \forall \; j \neq i, 0 \leq \bar{\mu}_i \leq 1\}$$
$$\underline{1}_i := \{\underline{\mu} \mid \underline{\mu}_j = 0 \; \forall \; j \neq i, 0 \leq -\underline{\mu}_i \leq 1\}.$$

Bibliography

[1] K. M. Yousef, M. N. Al-Karaki, and A. M. Shatnawi, "Intelligent traffic light flow control system using wireless sensors networks.," *Journal of Information Science and Engineering*, vol. 26, no. 3, pp. 753–768, 2010.

[2] R. Sundar, S. Hebbar, and V. Golla, "Implementing intelligent traffic control system for congestion control, ambulance clearance, and stolen vehicle detection," *IEEE Sensors Journal*, vol. 15, pp. 1109–1113, Feb 2015.

[3] J. A. Gutiérrez, "On the use of ieee std. 802.15.4 to enable wireless sensor networks in building automation," *International Journal of Wireless Information Networks*, vol. 14, no. 4, pp. 295–301, 2007.

[4] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry - recent development and future perspective," *Computers and Electronics in Agriculture*, vol. 50, no. 1, pp. 1 – 14, 2006.

[5] D. Chaudhary, S. Nayse, and L. Waghmare, "Application of wireless sensor networks for greenhouse parameter control in precision agriculture," *International Journal of Wireless & Mobile Networks*, vol. 3, no. 1, pp. 140–149, 2011.

[6] R. Nagel, S. Eichler, and J. Eberspacher, "Intelligent wireless communication for future autonomous and cognitive automobiles," in *2007 IEEE Intelligent Vehicles Symposium*, pp. 716–721, June 2007.

[7] R. A. Gupta and M. Y. Chow, "Networked control system: Overview and research trends," *IEEE Transactions on Industrial Electronics*, vol. 57, pp. 2527–2535, July 2010.

[8] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems - a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, 2013.

[9] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *IEEE Signal Processing Magazine*, vol. 19, pp. 40–50, Mar 2002.

[10] O. Khader, A. Willig, and A. Wolisz, "Wireless hart tdma protocol performance evaluation using response surface methodology," in *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2011 International Conference on*, pp. 197–206, IEEE, 2011.

[11] J. Araújo, M. Mazo, A. Anta, P. Tabuada, and K. H. Johansson, "System architectures, protocols and algorithms for aperiodic wireless control systems," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 175–184, 2014.

[12] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1548–1557, IEEE, 2001.

[13] S. Gupta, "Increasing the sampling efficiency for a control system," *IEEE Transactions on Automatic Control*, vol. 8, pp. 263–264, Jul 1963.

[14] M. Smith, "An evaluation of adaptive sampling," *IEEE Transactions on Automatic Control*, vol. 16, pp. 282–284, Jun 1971.

[15] J. Mitchell and W. McDaniel, "Adaptive sampling technique," *IEEE Transactions on Automatic Control*, vol. 14, pp. 200–201, Apr 1969.

[16] H. Kopetz, "Should responsive systems be event-triggered or time-triggered?," *IEICE Transactions on Information and systems*, vol. 76, no. 11, pp. 1325–1332, 1993.

[17] K. J. Astrom and B. M. Bernhardsson, "Comparison of riemann and lebesgue sampling for first order stochastic systems," in *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 2, pp. 2011–2016, IEEE, 2002.

[18] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, 2007.

[19] M. Mazo, Jr. and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, 2011.

[20] M. Mazo Jr. and M. Cao, "Asynchronous decentralized event-triggered control," *Automatica*, vol. 50, pp. 3197–3203, 12 2014.

[21] W. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 3270–3285, IEEE, 2012.

[22] J. B. Berglind, T. Gommans, and W. Heemels, "Self-triggered mpc for constrained linear systems and quadratic costs," *IFAC Proceedings Volumes*, vol. 45, no. 17, pp. 342–348, 2012.

[23] T. Gommans and W. Heemels, "Resource-aware mpc for constrained nonlinear systems: A self-triggered control approach," *Systems & Control Letters*, vol. 79, pp. 59–67, 2015.

[24] F. D. Brunner, W. P. M. H. Heemels, and F. Allgöwer, "Robust self-triggered mpc for constrained linear systems: A tube-based approach," *Automatica*, vol. 72, pp. 73 – 83, 2016.

[25] S. a. Keerthi and E. G. Gilbert, "Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations," *Journal of optimization theory and applications*, vol. 57, no. 2, pp. 265–293, 1988.

[26] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[27] J. M. Maciejowski, *Predictive control: with constraints.* Pearson education, 2002.

[28] P. O. M. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *Automatic Control, IEEE Transactions on*, vol. 43, pp. 1136–1142, Aug. 1998.

[29] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints," *Automatica*, vol. 37, no. 7, pp. 1019–1028, 2001.

[30] A. Richards and J. How, "Robust stable model predictive control with constraint tightening," in *2006 American Control Conference*, pp. 1557–1562, IEEE, 2006.

[31] W. Langson, I. Chryssochoos, S. V. Raković, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, pp. 125–133, Jan. 2004.

[32] D. Limon, I. Alvarado, T. Alamo, and E. Camacho, "Robust tube-based mpc for tracking of constrained linear systems with additive disturbances," *Journal of Process Control*, vol. 20, no. 3, pp. 248–260, 2010.

[33] D. Bernardini and A. Bemporad, "Energy-aware robust model predictive control based on noisy wireless sensors," *Automatica*, vol. 48, no. 1, pp. 36 – 44, 2012.

[34] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Event-triggered control for discrete-time systems," in *American Control Conference (ACC), 2010*, pp. 4719–4724, IEEE, 2010.

[35] A. Ferrara, A. N. Oleari, S. Sacone, and S. Siri, "An event-triggered model predictive control scheme for freeway systems," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 6975–6982, IEEE, 2012.

[36] D. Lehmann, E. Henriksson, and K. H. Johansson, "Event-triggered model predictive control of discrete-time linear systems subject to disturbances," in *2013 European Control Conference (ECC)*, pp. 1156–1161, July 2013.

[37] H. Payne and L. Silverman, "On the discrete time algebraic riccati equation," *IEEE Transactions on Automatic Control*, vol. 18, pp. 226–234, Jun 1973.

[38] A. Richards and J. How, "Robust distributed model predictive control," *International Journal of Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[39] A. Richards and J. How, "A decentralized algorithm for robust constrained model predictive control," in *Proceedings of the 2004 American Control Conference*, vol. 5, pp. 4261–4266 vol.5, June 2004.

[40] M. Henk, J. Richter-Gebert, and G. M. Ziegler, "16 basic properties of convex polytopes," *Handbook of discrete and computational geometry*, pp. 255–382, 2004.

[41] M. Baotić, "Polytopic computations in constrained optimal control.," *Automatika: Journal for Control, Measurement, Electronics, Computing & Communications*, vol. 50, 2009.

[42] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical problems in engineering*, vol. 4, no. 4, pp. 317–367, 1998.

[43] G. M. Ziegler and P. Kleinschmidt, "Lectures on polytopes," *Jahresbericht der Deutschen Mathematiker Vereinigung*, vol. 99, no. 3, pp. 42–43, 1997.

Glossary

List of Acronyms

DARE Discrete-time Algebraic Riccati Equation

DCSC Delft Center for Systems and Control

EB Event Based

ET Event Triggered

ET-RMPC Event Triggered Robust Model Predictive Control

LP Linear Programming

LTI Linear Time Invariant

MPC Model Predictive Control

NCS Networked Control System

OCP Optimal Control Problem

QP Quadratic Programming

RMPC Robust Model Predictive Control

ST Self Triggered

TM Triggering Mechanism

WSAN Wireless Actuator and Sensor Network