DEPARTMENT OF ENGINEERING CYBERNETICS

# TUBE-BASED METHODS FOR ROBUST LINEAR MPC

BY

HEIDI-IRENE HARVEY SOLLIE

PROJECT THESIS

SUPERVISOR: PROF. LARS IMSLAND

JUNE 19, 2019

NTNU
Norwegian University of
Science and Technology

NORWEGIAN UNVERSITY OF SCIENCE AND TECHNOLOGY

# Preface

This report was chosen as a project thesis in Cybernetics and Robotics, and has been carried out at the Department of Engineering Cybernetics under the supervision of Professor Lars Struen Imsland. It was chosen because I have a theoretical mathematical interest, and the main focus of my work in this paper is of a mathematical nature. The inspiration for the implementation of the control method, including the use the MPT Toolbox, came from Saša V. Rakoviĉ, who held a week of lectures at NTNU that I attended in the beginning of september.

I would like to thank everybody who has helped me with this report, either through inspiration, enthusiasm, guidance or proof-reading.

# Abstract

Model predictive control is a repetitive control method that solves an optimal control problem for a finite horizon at each time step, in order to find the current control action and apply it to the system.

For constrained linear systems with bounded additive disturbance, one can design a control law combining traditional feedback with the control action from model predictive control to ensure all realizations of the state trajectory stay within a range of the nominal state. When this is the case, it is simpler to consider the control of the uncertain system as controlling all possible trajectories. Further, applying tightened constraints to the nominal system ensures that the bundle of trajectories adheres to the original constraints. Thereby one has constrained the states of the actual system to a tube centered around the nominal states. This is termed *tube-based robust model predictive control*.

The main part of this paper is focused on deriving the mathematical formulation behind this specific method to demonstrate the aforementioned aspects of the control law.

In this paper, the control method is implemented in MATLAB and applied to a two-dimensional linear system modeled with bounded additive disturbance. Demonstrating how each step of this method works can aid in an intuitive understanding

of how to apply it to more complex systems. The performance of the controller is discussed along with the effect certain approximations have on the controller.

# Table of Contents

**Preface** **i**

**Abstract** **ii**

**1 Introduction** **2**

**2 Theory** **6**

2.1 Type of systems . . . . . . . . . . . . . . . . . . . . . . . . . 6

2.2 Tube-based methods . . . . . . . . . . . . . . . . . . . . . . . 8

2.3 Mathematical basis for tube-based method . . . . . . . . . . . . . 10

2.3.1 Existence of disturbance invariant set . . . . . . . . . . . 12

2.3.2 Approximation of disturbance invariant set . . . . . . . . 15

2.3.3 Explicit characterization of certain constraints . . . . . . . 17

2.4 Stability of tube-based MPC . . . . . . . . . . . . . . . . . . . . 19

2.4.1 Stability of nominal MPC . . . . . . . . . . . . . . . . . 20

**3 Methodology** **23**

3.1 System . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

3.2 Offline calculations . . . . . . . . . . . . . . . . . . . . . . . . 25

3.2.1 Feedback matrix $K$ . . . . . . . . . . . . . . . . . . . . . 25

3.2.2 Outer approximation of $S_K$ . . . . . . . . . . . . . . . . . 25

3.2.3 Tightened constraint set for state and control input $\bar{\mathbb{Y}}$ . . . 26
3.2.4 Terminal constraint set $\mathbb{X}_f$ . . . . . . . . . . . . . . . . . 27
3.2.5 $N$-step controllable tube . . . . . . . . . . . . . . . . . . 28
3.3 Online calculations . . . . . . . . . . . . . . . . . . . . . . . . 30
3.3.1 Objective function . . . . . . . . . . . . . . . . . . . . . 30
3.3.2 Equality constraints . . . . . . . . . . . . . . . . . . . . . 31
3.3.3 Inequality constraints . . . . . . . . . . . . . . . . . . . . 32
3.3.4 Execution . . . . . . . . . . . . . . . . . . . . . . . . . . 33
3.4 Controller . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37

**4 Results 39**

**5 Discussion 48**

**6 Conclusion 50**

**Bibliography 52**

**Appendices**

**A MPT Toolbox A.0-1**

**B Necessary set theory B.0-2**

**C Controllability and backward reachability C.0-4**

**D D.1-6**

D.1 Proof of explicit expression for $S_K$ when $A_K^N E = \alpha E$ . . . . . . . D.1-6
D.2 Approximating the minimal disturbance invariant set $S_K$ . . . . . D.2-7

# List of Figures


1.1 Visualization of tube-based control of an uncertain system . . . . 4

4.1 Simulation with $x_0 = x_0^1$ . . . . . . . . . . . . . . . . . . . . . . 40

4.2 Simulation with $x_0 = x_0^2$ . . . . . . . . . . . . . . . . . . . . . . 40

4.3 Simulation with $x_0 = x_0^3$ . . . . . . . . . . . . . . . . . . . . . . 41

4.4 Simulation with $x_0 = x_0^4$ . . . . . . . . . . . . . . . . . . . . . . 41

4.5 Controllable sets for $\alpha = 0.8$ and $\alpha = 0.25$ . . . . . . . . . . . . 43

4.6 Simulation with $x_0 = \tilde{x}_0^2$ . . . . . . . . . . . . . . . . . . . . . . 44

4.7 Simulation with $x_0 = \tilde{x}_0^3$ . . . . . . . . . . . . . . . . . . . . . . 45

4.8 Plot of the sequence $S_{Ki}$ and $S_{KN}(\alpha)$ . . . . . . . . . . . . . . . . 46

4.9 Minimum $N$ for varying $\alpha$ . . . . . . . . . . . . . . . . . . . . . . 47

# Listings


3.1 Calculating the feedback matrix $K$ and terminal cost matrix $P$ . . 25

3.2 Calculating the initial constraint set $S_K$ . . . . . . . . . . . . . . 26

3.3 Calculating the tightened state and control input constraint set $\bar{\mathbb{Y}}$ . 27

3.4 Calculating the tightened terminal constraint set $\mathbb{X}_f$ . . . . . . . . 28

3.5 Function for computing the $N$-step controllable tube . . . . . . . 29

3.6 Generating the matrices for the objective function . . . . . . . . . 31

3.7 Generating the matrices for the equality constraints . . . . . . . . 31

3.8 Generating the matrices for the inequality constraints . . . . . . . 33

3.9 Solving the finite-horizon optimal control problem . . . . . . . . 35

3.10 Execution of robust tube MPC . . . . . . . . . . . . . . . . . . . . 36

# Chapter 1

# Introduction

Model predictive control (MPC) was invented in industrial process control, and from there brought into academia where numerous variations of the method were developed. As a result of this unconventional evolution, there is a gap between the theoretical research and the practical applications of MPC, in the sense that for many theoretical methods the practical applications are rarely applied. [13]

Model predictive control is a method of control where at each time step, a finite horizon optimal control problem is solved for an optimal control sequence where the first element is applied to the system. The general version of this is the infinite horizon linear quadratic regulator (LQR), which is optimal in the unconstrained case. When a system is constrained, this same optimization is performed for a finite horizon under the given constraints; a trade-off between optimality and feasibility.

The feedback MPC, which includes the state of the system at each time step in the online calculation, is much more complex than the deterministic variety, and therefore a lot of research has focused on reducing complexity, by sacrificing optimality [10]. Feedback in any control system is inherently important when there is uncertainty present. The research in this paper focuses on a specific type of uncertainty, where it is assumed that the disturbance is bounded. There are numerous methods for handling this robust uncertainty in MPC, which will briefly be discussed in this chapter. One such category of methods, called tube-based methods, is the focus of this paper.

Tube-based methods, first proposed in 2005 in [8], arise from the following observation: if one introduces a control law to the system that ensures all the possible trajectories of the uncertain system, given the assumptions around the uncertainty, stay within a certain radius of the nominal system, then one can apply stricter constraints to the nominal system which in turn ensures the uncertain system adheres to the original constraints. The nominal system is the model of the system without uncertainty, and is much easier to control. Thus controlling the uncertain system by controlling this bundle of trajectories, which simplifies the problem.

The tube-based method uses a control law of the form $u = \bar{u} + K(x - \bar{x})$, where $K$ is a feedback matrix chosen by the designer, and $\bar{u}$ comes from solving a feedback MPC problem online for the nominal system with tightened constraints, calculated offline. This law ensures the aforementioned property.

In figure 1.1 this relatively simple concept is visualized for a two-dimensional system. The nominal system trajectory is in red and a few realizations of the actual trajectory are shown in black. The control law used for the actual trajectories is the one stated above, and ensures that the trajectories always stay within a certain distance of the nominal system. This specific distance is visualized by the set plotted for every time step. This set is called a disturbance invariant set, and its calculation and significance is discussed in detail in chapter two.

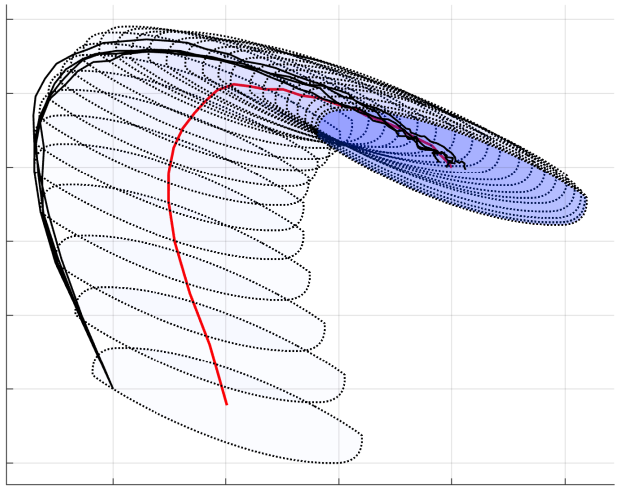

**Figure 1.1:** Visualization of tube-based control of an uncertain system

Since the control law uses feedback from the actual system, and only one of the visualized trajectories can be used as the feedback, the other realizations are not contained in this set. This does not speak to the performance of the controller.

The controller is robustly stable, provided that the uncertainty adheres to the assumptions regarding its boundedness, which will be shown in chapter two.

The paper is organized as follows. The main focus is devoted to deriving the mathematical formulation behind the tube-based method which is done in chapter two. At the end of this chapter, stability for the method is shown. The system used in the simulation is presented in chapter three, along with the structure of the implementation and relevant code snippets. Chapter four shows the results of the simulation. Chapter five is a discussion of the results, and the conclusion is presented last, in chapter six.

Note that this paper assumes the reader is well known with the model predictive control method and its formulation as it does not go into the specifics of the method, only the variations involved in the tube-based method.

# Chapter 2

# Theory

## 2.1 Type of systems

Uncertainty in a system is a realistic inevitability, and denotes any factor that contributes to the actual behavior of the system not being equivalent to the predicted behavior. This can be external factors affecting the system, model inaccuracies and much more. Depending on what is known regarding the specifics of uncertainty, designing a controller that ensures stability and constraint satisfaction can be anything from trivial to impossible.

There are different ways of modeling uncertainty in linear systems depending on how these factors interact with the system dynamics. Two main categories of modeling are parametric uncertainty and unstructured uncertainty. The former denotes uncertainty due to parametric differences between the model and the physical system, while the latter refers to structural differences.

If the system parameters are not precisely known and vary over a certain range, for example $x^+ = (A + \delta_A)x + (B + \delta_B)u$, where $\delta_{A,B}$ denotes the variation, it can be modeled as *parametric uncertainty*.

*Unstructured uncertainty*, however, denotes uncertainty in the underlying dynamics of the system. For example, an additive disturbance for a linear system would be modeled as $x^+ = Ax + Bu + Ew$ and multiplicative disturbance as $x^+ = (Ax + Bu) \cdot Ew$, where $w$ denotes the disturbance factor and $E$ denotes the disturbance matrix.

Combining these models to form *mixed uncertainty* is also common.

To have any chance of designing stable controllers for systems with uncertainty, certain assumptions have to be made. One can, for instance, consider the uncertainty a stochastic variable belonging to a specific probability distribution. This is called *stochastic uncertainty*. Another, perhaps more strict, way which will be the focusing on in this paper, is assuming the uncertainty is bounded and enclosed in a compact set $\mathbb{W}$. This is referred to as *robust uncertainty* and leads the way to *robust model predictive control*. [1]

There are numerous approaches to robust MPC, and only a few are mentioned here. One method minimizes the objective function for the worst case of the disturbance sequence. This is called *min-max MPC*. [4] Another method is tightening the constraints used in the online calculations to prevent infeasibility from a disturbance. This is *constraint tightening MPC* [11]. A third method is separating the state dynamics into the nominal state $\bar{x}$ and error state $e = x - \bar{x}$, and introducing a control law that ensures the state trajectories stay relatively close to the nominal trajectories for all realizations of the disturbance sequence. This is the method presented in the introduction, called *tube-based MPC*.

This section will focus on the mathematics behind applying the robust tube-based model predictive control method to a linear system with additive disturbance.

## 2.2 Tube-based methods

The feedback control law,

$$u = \bar{u} + K(x - \bar{x}) \ , \tag{2.1}$$

where $K$ is a feedback matrix chosen by the designer, and $\bar{u} = K\bar{x}$, is the optimal feedback control law in the unconstrained linear system.

This can be extended to the constrained case in (2.2), if $\bar{u} = K\bar{x}$ is replaced with $\bar{u} = \kappa_N(\bar{x})$ which comes from solving, online, a finite-horizon optimization problem

$$\begin{aligned} x^+ &= Ax + Bu + Ew \quad , \\ x &\in \mathbb{X}, u \in \mathbb{U}, w \in \mathbb{W} \ . \end{aligned} \tag{2.2}$$

Here, $A$ is the state matrix, $B$ is the control matrix and $E$ is the disturbance matrix. The sets $\mathbb{X}, \mathbb{U}, \mathbb{W}$ are the constraints for their respective variables.

Using this feedback control law, one can show that all realizations of the state trajectory will lie in a bounded neighborhood of the nominal trajectory. Further, subjecting the nominal system to a set of tightened constraints in the online calculations, one can control this bundle of trajectories around the nominal trajectory to adhere to the original constraints. This is the principle behind *tube based control*.

The calculation of the tightened constraints, as well as the calculation of the feedback matrix $K$, can be done offline, and this method is therefore not as computationally expensive as one might assume. The online calculations in this case are identical to nominal MPC, with the only difference being using the tightened constraints in the problem.

For linear systems with convex constraints such as the one in equation (2.2), a tube may be designed to bound all realizations of the state trajectory: $X_i = \{\bar{x}_i\} \times S$, where $S$ is a disturbance invariant set, and $\bar{x}_i$ is the state of the nominal system at time $i$.

Even though it is assumed that the reader is quite familiar with MPC, for future reference, the finite-horizon optimal control problem used in the online calculations is,

$$\min_{\bar{\mathbf{u}}} \sum_{j=0}^{N-1} \left( x_j^\top Q x_j + u_j^\top R u_j \right) + x_N^\top P x_N \ , \tag{2.3}$$

$$\begin{aligned} &\text{subject to} \\ x_0 &= x(0) \text{ given } , \\ x_{j+1} &= A x_j + B u_j, \ \ j \in \mathbb{Z}_{0:N-1} \ , \\ x_j, u_j &\in (\mathbb{X} \times \mathbb{U}), \qquad j \in \mathbb{Z}_{0:N-1} \ , \qquad (2.4) \\ x_N &\in \mathbb{X}_f \subseteq \mathbb{X} \ . \qquad (2.5) \end{aligned}$$

The cost matrices are $Q$ and $R$ for $x$ and $u$ respectively, and $P$ represents the terminal cost matrix. The set $\mathbb{Z}_{a:b}$, where $a < b$, denotes the set of integers $\{a, a+1, \ldots, b-1, b\}$ and this notation will be used throughout the text. Also note that $\mathbb{Z}_+ = \{z \in \mathbb{Z}, z \geq 1\}$.

## 2.3 Mathematical basis for tube-based method

Given the system and nominal system

$$
\begin{aligned}
x^+ &= Ax + Bu + Ew \ , &\quad (2.6)\\
\bar{x}^+ &= A\bar{x} + B\bar{u} \ , &\quad (2.7)
\end{aligned}
$$

where

$$
\begin{aligned}
x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, w \in \mathbb{R}^{n_w} \ , \\
(x, u) \in \mathbb{Y} = \{\mathbb{X} \times \mathbb{U}\} \ , \\
w \in \mathbb{W} \ .
\end{aligned}
\quad (2.8)
$$

The constraint sets are polytopic and can be described as

$$
\begin{aligned}
\mathbb{Y} = \{(x, u) \mid Cx + Du \leq e\} \ , \\
\mathbb{W} = \{w \mid E_w w \leq g_w\} \ .
\end{aligned}
\quad (2.9)
$$

One must also assume that the disturbance set $\mathbb{W}$ is a compact set that contains the origin in its interior, which ensures the existence of a disturbance invariant set that will become clear later in this chapter.

$$
\{0\} \in int(\mathbb{W}), \text{ where } \mathbb{W} \text{ is a compact set.} \quad (2.10)
$$

Allowing the transformation from $(x, \bar{x})$ to $(e, \bar{x})$ as follows,

$$
\begin{bmatrix} e \\ \bar{x} \end{bmatrix} = \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ \bar{x} \end{bmatrix} \quad , \quad (2.11)
$$

and introducing the control law in equation (2.1), yields the system equations

$$e^+ = A_K e + Ew \ , \tag{2.12}$$

$$\bar{x}^+ = A\bar{x} + B\bar{u} \quad . \tag{2.13}$$

with $A_K = A + BK$. Consequently, this is an asymptotically stable system by design, since $K$ is chosen by the designer. Since this transformation is invertible, the original states can be acquired at any time.

It is important to note that the transformation in (2.11) can be considered separating the state into $x = \bar{x} + e$, with the following separate dynamics

$$\begin{aligned} x^+ &= \bar{x}^+ + e^+ \ , \\ e^+ &= (A + BK_e)x + Ew \ , \\ \bar{x}^+ &= A\bar{x} + B\bar{u} \ . \end{aligned} \tag{2.14}$$

In this formulation there are in theory two separate feedback matrices, $K_e$ and $K_{\bar{x}}$. $K_e$ is the robust feedback matrix, used in the control law $u$ and for the calculation of the disturbance invariant set $S_K$. This is selected by the designer such that the error dynamics are stable.

The feedback matrix $K_{\bar{x}}$ used in the nominal system to calculate the terminal constraint set and terminal cost is the solution to the discrete time algebraic Riccati equation, and is used as the feedback control law in the infinite horizon optimal control problem. In this paper, $K_e = K_{\bar{x}} = K$, for simplicity and because it was not necessary as the closed-loop error dynamics was asymptotically stable with the feedback matrix $K_{\bar{x}}$.

The following notation, from [10], is used throughout the text: $\bar{u} = \kappa_N(\bar{x})$. For each state of the system, the finite-horizon optimal control problem is solved to acquire the optimal control sequence, and the first element, $\bar{u}(0; x)$, is applied to the system. This is only calculated for the actual states of the system. However, if the optimal control sequence was calculated for every possible feasible state of the problem, one can think of this as a feedback control law $\kappa_N(\bar{x}) = \bar{u}(0; x)$, $x \in \bar{X}_N$, without needing to actually calculate the exact expression.

Therefore the composite closed-loop system can be written as

$$\begin{aligned} e^+ &= A_K e + E w \ , \\ \bar{x}^+ &= A\bar{x} + B\kappa_N(\bar{x}) \ . \end{aligned} \tag{2.15}$$

### 2.3.1 Existence of disturbance invariant set

To ensure that the state trajectories stay within a certain compass of the nominal state, the existence of a disturbance invariant set must be shown.

A disturbance invariant set $X$ (also called robustly positively invariant) for $x^+ = f(x, w)$ satisfies the following equation,

$$f(x, w) \in X \ \forall \ x \in X, w \in \mathbb{W} \ . \tag{2.16}$$

Equivalently, $X$ is a disturbance invariant set if and only if

$$AX \oplus \mathbb{W} \subseteq X \ . \tag{2.17}$$

Further the *minimal disturbance invariant set* is the set with the above properties that is contained in every other closed disturbance invariant set for the same system.

One can observe the following evolution of the error dynamics, from chapter 3.5 in [10],

$$
\begin{aligned}
e_1 &= A_K e_0 + E w_0 \ , \\
e_2 &= A_K e_1 + E w_1 = A_K(A_K e_0 + E w_0) + E w_1 \\
&= A_K^2 e_0 + A_K E w_0 + E w_1 \ , \\
e_3 &= A_K e_2 + E w_2 = A_K(A_K^2 e_0 + A_K E w_0 + E w_1) + E w_2 \\
&= A_K^3 e_0 + A_K^2 E w_0 + A_K E w_1 + E w_2 \ , \\
&\vdots \\
e_i &= A_K^i e_0 + \sum_{j=0}^{i-1} A_K^j E w_{i-1-j}, \quad w \in \mathbb{W} \ .
\end{aligned}
\tag{2.18}
$$

This results in the following definition of a set sequence, denoting the set of all states reachable at $i$ starting from $e_0 = 0$,

$$
\begin{aligned}
S_{K0} &= \{0\} \ , \\
S_{Ki} &= \sum_{j=0}^{i-1} A_K^j E \, \mathbb{W}, \quad i \in \mathbb{Z}_+ \ .
\end{aligned}
\tag{2.19}
$$

Here $\sum$ denotes the set addition $\oplus$ defined in appendix B.

It can be shown that this sequence converges to a disturbance invariant set $S_K$, which is the minimal disturbance invariant set for (2.12),

$$S_K = \sum_{j=0}^{\infty} A_K^j E \, \mathbb{W} \ . \tag{2.20}$$

Since $A_K$ is stable by design, any $e_0 \in S_K$ implies $e_i \in S_K \; \forall \, i \in \mathbb{Z}_+$. This is significant because it means the possibility that for a certain set of initial states of the system, the system evolution of the error dynamics will be contained in the set, for any realization of the disturbance sequence. In other words, the system will stay in an $S_K$-neighborhood of the nominal system for any realization of the disturbance sequence.

Theorem 4.1 in [5] states that under the assumptions in (2.10), and the fact that $A_K$ is asymptotically stable, there exists a compact disturbance invariant set $S_K \subset \mathbb{R}^n$ such that $S_{Ki} \subset S_K \; \forall \, i \in \mathbb{Z}_+$ and the sequence $S_{Ki}, i \in \mathbb{Z}_+$ converges to $S_K$. In corollary 4.2, $S_K$ is shown to be the minimal disturbance invariant set.

It can be instructive to prove this theorem. Recalling that $A_K$ being asymptotically stable implies any power of $A_K$ is strictly decreasing and $\mathbb{W}$ being compact and containing the origin implies $\exists \, c \in \mathbb{R}$ such that $E\mathbb{W} \subset c\mathcal{B}^n$ where $\mathcal{B}^n$ is a ball with arbitrary radius $n$ and center in the origin. So, combined

$$\exists \, c > 0, \;\; \lambda \in (0,1) \text{ and } i \in \mathbb{Z}_+ \ , \tag{2.21}$$

$$\text{such that } A_K^i E \, \mathbb{W} \subset c\lambda^i \mathcal{B}^n \ . \tag{2.22}$$

Since $S_{Ki+1} = S_{Ki} + A_K^i E \, \mathbb{W}$, it follows that $d_H(S_{Ki+1}, S_{Ki}) \leq c\lambda^i$. Noting that a set sequence is Cauchy if for every $\epsilon > 0$, $\exists \, N$ such that $d_H(S_m, S_n) < \epsilon$ for all $m, n > N$, $(m, n, N \in \mathbb{Z}_+)$, it is apparent that the sequence $\{S_{Ki} : i \in \mathbb{Z}_+\}$ is Cauchy. Since $S_{Ki} \in \mathbb{R}^2$, which is a complete metric space, and all Cauchy sequences in complete metric spaces converge [12], the sequence converges to $S_K$, i.e. $\lim_{i\to\infty} S_{Ki} = S_K$.

This limit $S_K$ is also disturbance invariant because $S_K = A_K S_K + E\,\mathbb{W}$, and therefore

$$\forall x \in S_K,\; x^+ = A_K x + E\,\mathbb{W} \in S_K. \tag{2.23}$$

To prove minimality, let $X$ be any closed disturbance invariant set for the system (2.12), i.e.

$$x_0 \in X \to A_K^i x_0 + \sum_{j=0}^{i-1} A_K^j E w_{i-1-j} \in X \quad \forall\, i \in \mathbb{Z}_+. \tag{2.24}$$

Since $\{0\} \in \mathbb{W}$, this implies $\sum_{j=0}^{i-1} A_K^j E w_{i-1-j}$ can be zero and therefore

$$x_0 \in X \Rightarrow A_K^i x_0 \in X\, \forall\, i \in \mathbb{Z}_+. \tag{2.25}$$

The closed-loop system matrix $A_K$ being asymptotically stable means $\lim_{i\to\infty} A_K^i x_0 = 0$ and therefore $\{0\} \in X$. Furthermore, $x_0 = 0$ means the state trajectory $x_i$ stays in $X\, \forall\, i \in \mathbb{Z}_+,\; w \in \mathbb{W}$.

By the definition of the sequence (2.19), $S_{Ki} \subset X,\; \forall\, i \in \mathbb{Z}_+$, meaning $S_K \subset X$ and hence $S_K$ must be the minimal disturbance invariant set for (2.12). $\square$

### 2.3.2 Approximation of disturbance invariant set

Since this set is defined as an infinite sum, an explicit expression does not necessarily exist.

From remark 4.2 in [5], if there exists $N \in \mathbb{Z}_+$ and $\alpha \in [0, 1)$ such that $A_K^N E = \alpha E$, then $S_K = (1-\alpha)^{-1} S_{KN}$. See appendix D.1 for the proof of this.

Unfortunately, this is a special case and cannot be used very often. However, [9], $\exists\ N \in \mathbb{Z}_+$ and $\alpha \in [0, 1)$ such that $A_K^N E \subseteq \alpha E$. From there, a disturbance invariant outer approximation for $S_K$ exists and is defined as

$$S_{KN}(\alpha) = (1 - \alpha)^{-1} S_{KN}. \tag{2.26}$$

A discussion regarding the accuracy of this approximation, and the selection of $\alpha$, is taken briefly in appendix D.2

The existence of the minimal disturbance invariant set can be used in designing a disturbance invariant bounding tube for the system evolution. Let $\mathbf{X}(x, \bar{\mathbf{u}})$ be defined as follows

$$\begin{aligned} \mathbf{X}(x, \bar{\mathbf{u}}) &= \left(X(0; x), X(1; x, \bar{\mathbf{u}}), ..., X(N; x, \bar{\mathbf{u}})\right)\ , \\ X(i; x) &= \{\bar{x}_i \oplus S_K\}\ . \end{aligned} \tag{2.27}$$

This means that given an initial state $x$ and control sequence $\bar{\mathbf{u}} = \bar{u}_0, \bar{u}_1, \ldots$, the solution to the system will lie in this tube, for every realization of the disturbance sequence.

If the state of the system always lies within this tube, defining the constraints of the nominal system as

$$\begin{aligned} \bar{\mathbb{X}} &= \mathbb{X} \ominus S_K\ , \\ \bar{\mathbb{U}} &= \mathbb{U} \ominus K S_K\ , \end{aligned} \tag{2.28}$$

ensures that the state and control input of the actual system lie within their respective constraint sets $\mathbb{X}$ and $\mathbb{U}$.

Here $\ominus$ denotes the set subtraction defined in appendix B.

### 2.3.3 Explicit characterization of certain constraints

To implement this for a linear system, an explicit characterization of the tightened constraint set needs to be derived, expressed as a set of linear inequalities.

To compute an explicit characterization of the tightened set, define the *worst-case* minimal disturbance invariant set $S_{Kmax}$ as

$$S_{Kmax} = \max_{\mathbf{w}}\{\sum_{j=0}^{\infty} A_K^j E w_i \mid \mathbf{w} \in \mathbb{W}\} \ . \tag{2.29}$$

The definition of set subtraction from appendix B is

$$\bar{\mathbb{Y}} = \{(x,u) \in \mathbb{Y} \mid (x,u) + (s_1, s_2) \in \mathbb{Y} \, \forall \, (s_1, s_2) \in (S_K \times K S_K)\} \ . \tag{2.30}$$

Hence, the tightened constraint set will be

$$\begin{aligned} \bar{\mathbb{Y}} &= \{(x,u) \mid C(x + S_{Kmax}) + D(u + K S_{Kmax}) \leq e\} \\ &= \{(x,u) \mid Cx + Du \leq e - (C + DK) S_{Kmax}\} \ . \end{aligned} \tag{2.31}$$

Using the outer approximation for $S_K$ from (2.26), this can be rewritten as

$$\bar{\mathbb{Y}} = \{x, u \mid Cx + Du \leq e - \theta_N\} \ , \tag{2.32}$$

where

$$\theta_N = (1-\alpha)^{-1} \max_{\mathbf{w}} \{ \sum_{j=0}^{N-1} C A_K^j E w_j + D K A_K^j E w_j \mid \mathbf{w} \in \mathbb{W} \} \ . \tag{2.33}$$

Another element of the controller that requires an explicit formulation before moving on to the stability analysis and the implementation, is the formulation of the terminal constraint $\bar{x}_N \in \mathbb{X}_f$ for the MPC. The goal is to guarantee constraint satisfaction for an infinite horizon, which in turn contributes to stability.

To calculate a suitable set, one uses a method called "quasi-infinite horizon" [2], where the terminal constraint is calculated with the optimal feedback control law $\bar{u} = K\bar{x}$ for the unconstrained infinite-horizon.

Consider the *maximal positively invariant set*, defined in appendix C, for the nominal system with $\bar{u} = K\bar{x}$, under the tightened constraint set $\bar{\mathbb{Y}}$. This can be written as

$$\begin{aligned} \bar{x}^+ &= A_K \bar{x} \ , \\ x \in \bar{\mathbb{Y}}_k &= \{ x \mid C_K x \leq e - \theta_N \} \ , \end{aligned} \tag{2.34}$$

with $C_K = C + DK$.

Requiring the last state of the finite horizon, $\bar{x}_N$, to be in this set guarantees the evolution of the infinite-horizon closed loop nominal system never leaves the set. The reason to use the maximal set, is that it is favorable for the constraints in the online problem to be as slack as possible, to ensure maximal feasibility.

## 2.4 Stability of tube-based MPC

This section demonstrates the robust asymptotic stability (RAS) of tube-based MPC, from [10]. It is sufficient to show RAS for the system with states $(e, \bar{x})$ (2.15).

Robust asymptotic stability of a set $B$ for system $z^+ = f(z, w)$ in set $A$, where $B \subset A$, implies $\exists$ $\mathcal{KL}$-function $\beta(\cdot, \cdot)$ such that

$$d(\phi(i; z, \mathbf{w}), B) \leq \beta(d(z, B), i) \qquad \forall\, i \in \mathbb{Z}_+ \ , \tag{2.35}$$

where $\phi(i; z, \mathbf{w})$ is every solution of $z^+ = f(z, w)$ with $z_0 \in A$ and $\mathbf{w} \in \mathbb{W}^\infty$.

Using the definition in (2.35), one can show that the set $S_1 = S_K \times \{0\}$ is robustly asymptotically stable for the system in the set $S_2 = S_K \times \bar{X}_N$, where $\bar{X}_N$ is the set of all states for which there exists a feasible solution to the $N$-horizon optimal control problem (2.3). This is equivalent to proving RAS for tube-based MPC of linear systems.

The origin for the closed loop nominal system $\bar{x}^+ = A\bar{x} + B\kappa_N(\bar{x})$ is asymptotically stable. From the definition of asymptotic stability, $\exists$ $\mathcal{KL}$-function $\beta(\cdot, \cdot)$ such that

$$|\bar{\phi}(i; \bar{x})| \leq \beta(|\bar{x}|, i)\ \forall\, i \in \mathbb{Z}_+ \tag{2.36}$$

for every solution $\bar{\phi}(\cdot; \bar{x})$ of the system with initial state $\bar{x} \in \bar{X}_N$.

Noting that

$$\begin{aligned} d((e_i, \bar{\phi}(i; \bar{x})), (S_K \times \{0\})) &\leq d(e_i, S_K) + d(\bar{\phi}(i; \bar{x}), \{0\}) \\ &= d(e_i, S_K) + |\bar{\phi}(i; \bar{x})| \end{aligned} \tag{2.37}$$

and that $e_0 \in S_K \Rightarrow e_i \in S_K \, \forall \, i \in \mathbb{Z}_+$, i.e. $e_0 \in S_K \Rightarrow d(e_i, S_K) = 0 \, \forall \, \mathbb{Z}_+$, then

$$d((e_i, \bar{\phi}(i; \bar{x})), (S_K \times \{0\})) \leq \beta(|\bar{x}|, i) \tag{2.38}$$

which in turn shows us that the set $S_1$ is RAS for (2.15) in $S_2$, given the definition in (2.35).

This proof is based on the fact that the origin of $\bar{x}^+ = A\bar{x} + B\kappa_N(\bar{x})$ is asymptotically stable, which depends on the stability of the nominal MPC. The terminal cost in the objective function and the terminal constraint set ensures this stability. This is discussed further below.

### 2.4.1 Stability of nominal MPC

The robust asymptotic stability of the tube-based controller is based on the stability of the nominal MPC. The terminal constraint set $\mathbb{X}_f$ and the terminal cost $x_N^\top P x_N$ both ensure this stability. [6]

The terminal constraint set $\mathbb{X}_f$ in the nominal MPC is described in section 2.3. This is the set of states such that the evolution of the state trajectory stays in the set and satisfies the tightened constraints, when $\bar{u} = K\bar{x}$. Chapter 2.6 in [10] addresses the necessity of the terminal constraint.

The terminal cost $x_N^\top P x_N$, where $P$ is the solution to the discrete time algebraic Riccati equation also ensures stability. Let the infinite horizon objective function, given state feedback $u = Kx$, where $K$ is the state feedback matrix corresponding to the discrete-time algebraic Riccati equation, be

$$\sum_{k=0}^{\infty} x_k^\top Q x_k + u_k^\top R u_k = \sum_{k=0}^{\infty} x_k^\top (Q + K^\top R K) x_k \quad . \tag{2.39}$$

The discrete time algebraic Riccati equation,

$$A^\top PA - P + Q - A^\top PB(R + B^\top PB)^{-1}B^\top PA = 0 \ , \tag{2.40}$$

with the state feedback matrix

$$K = -(R + B^\top PB)^{-1}B^\top PA \ , \tag{2.41}$$

can be written as

$$A_K^\top PA_K - P + Q + K^\top RK = 0 \ . \tag{2.42}$$

with $A_K = A + BK$.

Then,

$$\begin{aligned}\sum_{k=0}^{\infty} x_k^\top Qx_k + u_k^\top Ru_k &= \sum_{k=0}^{\infty} x_k^\top (P - A_K^\top PA_K)x_K \\ &= \sum_{k=0}^{\infty} x_k^\top Px_k - \sum_{k=0}^{\infty} x_{k+1}^\top Px_{k+1} \\ &= x_0^\top Px_0 \ .\end{aligned} \tag{2.43}$$

Therefore one can split the infinite horizon objective function into a sum with the finite horizon objective function and the results from above,

$$\begin{aligned}\sum_{k=0}^{\infty} x_k^\top Qx_k + u_k^\top Ru_k &= \sum_{k=0}^{N-1} \left(x_k^\top Qx_k + u_k^\top Ru_k\right) + \sum_{k=N}^{\infty} \left(x_k^\top Qx_k + u_k^\top Ru_k\right) \\ &= \sum_{k=0}^{N-1} \left(x_k^\top Qx_k + u_k^\top Ru_k\right) + x_N^\top Px_N.\end{aligned} \tag{2.44}$$

This result shows that it is possible to optimize on the infinite horizon, with a finite sum in the objective function. The idea is that the control input for the finite horizon handles the constraints so the LQR controller is optimal for the rest of the horizon. [3]

A detailed proof of the stability of nominal MPC is outside the scope of this paper. However, the above derivations have hopefully given the reader an intuitive understanding of the contribution the terminal constraint set and terminal cost has to the stability of the MPC.

# Chapter 3

# Methodology

The theory described in the previous sections will be applied to a relatively simple system, presented in this chapter. The implementation in MATLAB shown in code snippets is also presented in this chapter.

Note: $(\bar{x}, \bar{u})$ is referred to as $(z, v)$ in this chapter because this is what is used in the MATLAB code for the simulation for simplicity.

## 3.1 System

Given a simple mechanical system, a double integrator $\ddot{y} = u + w$, where the uncertainty $w$ is deviation from the control force $u$ applied,

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} w \ . \tag{3.1}$$

Discretizing this system with step length $h$ yields the following discrete system

$$x^+ = \begin{bmatrix} 1 & h \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ h \end{bmatrix} u + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} w \ , \tag{3.2}$$

where

$$\begin{aligned} & x^+ = Ax + Bu + Ew \ , \\ & x, w \in \mathbb{R}^2, u \in \mathbb{R}^1 \ , \\ & x \in \mathbb{X}, u \in \mathbb{U}, w \in \mathbb{W} \ . \end{aligned} \tag{3.3}$$

The constraints have been chosen arbitrarily, but ensuring they fulfill the previous assumptions stated in (2.10),

$$\begin{aligned} \mathbb{X} &= \{x : x \in [-20, 20] \times [20, 20]\} \ , \\ \mathbb{U} &= \{u : u \in [-50, 50]\} \ , \\ \mathbb{W} &= \{w : w \in [-0.2, 0.2]\} \ . \end{aligned} \tag{3.4}$$

Written as linear inequalites this becomes

$$\begin{gathered} \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \end{bmatrix} u \leq \begin{bmatrix} 20 \\ 20 \\ 20 \\ 20 \\ 50 \\ 50 \end{bmatrix} \ , \\ \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} w \leq \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} \ . \end{gathered} \tag{3.5}$$

which is what is used in the formulation, with the names of the variables coming from (2.9).

In the implementation, $h = 0.1$ is used.

## 3.2 Offline calculations

### 3.2.1 Feedback matrix $K$

$K$ is calculated from solving the discrete time algebraic Riccati equation. In MATLAB `[K,P,eig] = dlqr(A,B,Q,R,N)` returns both the optimal feedback matrix $K$ to $u = -Kx$ and the infinite horizon solution $P$. Listing 3.1 shows the MATLAB implementation of this. For the notation in this paper, $u = Kx$ has been used and therefore the sign of $K$ must be changed.

**Listing 3.1:** Calculating the feedback matrix $K$ and terminal cost matrix $P$

```
[K, cost.P] = dlqr(system.A, system.B, ...
                   cost.Q, cost.R);
system.K = -K;
```

### 3.2.2 Outer approximation of $S_K$

The calculation of the outer approximation of $S_K$, denoted $S_{KN}(\alpha)$ in the previous chapter has to be done. This is used as the initial constraint set for the online calculations. Listing 3.2 shows the implementation of this.

**Listing 3.2:** Calculating the initial constraint set $S_K$

```
i=1;
W = Polyhedron(disturbance.E,disturbance.g);
S_K_seq(i) = W;
while and(not(system.A_K^i * W <= alpha * W), ...
                    i <= system.Nsim)
      i = i + 1;
      S_K_seq(i) = system.A_K * S_K_seq(i-1) ...
                    + system.E * W;
end
N = i;
S_K = (1 - alpha)^(-1) * S_K_seq(N);
constraints.S = S_K.A;
constraints.r = S_K.b;
```

### 3.2.3 Tightened constraint set for state and control input $\bar{\mathbb{Y}}$

From equation (2.28) an explicit characterization of the tightened constraint set for the state and control input $\bar{\mathbb{Y}}$ is implemented in MATLAB. Listing 3.3 shows the code snippet that handles this calculation in the implementation. `S_K` has already been calculated from listing 3.2.

**Listing 3.3:** Calculating the tightened state and control input constraint set $\bar{\mathbb{Y}}$

```
constraints.C_K = constraints.C ...
                  + constraints.D * system.K;
for i = 1:size(constraints.C_K,1)
    theta_N(i) = S_K.support(constraints.C_K(i,:)');
end
theta_N = theta_N';
constraints.e = constraints.e - theta_N;
```

### 3.2.4 Terminal constraint set $\mathbb{X}_f$

The terminal constraint set in the optimal control problem, seen in (2.5), is essential for stability in the finite-horizon case, because enforcing a certain constraint on the final state can ensure the stability of the trajectory past the current horizon. This is discussed in the last part of section 2.3.

Listing 3.4 shows us the code snippet that calculates the tightened terminal constraint set in the implementation. This comes after listing 3.3, so `constraints.e` is now `constraints.e - theta_N`. The condition `not(X_f(i-1) == X_f(i))` in the while-loop ensures the iteration terminates when the set converges, i.e. giving the maximal positively invariant set under these conditions. This is the implementation of the equation (C.1).

**Listing 3.4:** Calculating the tightened terminal constraint set $\mathbb{X}_f$

```
i=1;
X_f(i) = Polyhedron(constraints.C_K, constraints.e);
cond = true;
while cond
        i = i+1;
        current_target.G = X_f(i-1).A;
        current_target.h = X_f(i-1).b;
        A=[constraints.C_K;
            current_target.G*system.A_K];
        b=[constraints.e; current_target.h];
        X_f(i) = Polyhedron(A,b);
        cond = and(not(X_f(i-1) == X_f(i)), i<=system.Nsim);
end
constraints.G = X_f(i).A;
constraints.h = X_f(i).b;
```

### 3.2.5 $N$-step controllable tube

Another set that is generated offline, to be used in the plotting of the results, is the $N$-step controllable tube for the nominal system

$$\begin{aligned} \bar{x}^+ &= A\bar{x} + B\bar{u} \ , \\ \bar{x}, \bar{u} &\in \bar{\mathbb{X}} \times \bar{\mathbb{U}} \ , \end{aligned} \tag{3.6}$$

with target $\mathbb{X}_f$ from the previous section.

This tube is defined in appendix C. This is the set of all the initial states that lead to the terminal constraint in the $N$-step finite horizon. The controller can still be stable for an initial state outside of this set, however stability is then not guaranteed, as discussed in section 2.4.1.

Listing 3.5 shows the functions that generates this set. This is an implementation of the equation (C.2).

**Listing 3.5:** Function for computing the $N$-step controllable tube

```
function X=c_tube(system,constraints)
i=1;
X(i)=Polyhedron(constraints.G,constraints.h);
while i<= system.N
    i=i+1;
    current_target.G=X(i-1).A;
    current_target.h=X(i-1).b;
    %generate i-step controllable set
    L=[constraints.C constraints.D;
        current_target.G*system.A current_target.G*
            system.B];
    r=[constraints.e; current_target.h];
    X(i)=Polyhedron(L,r).projection(1:system.n);
end
```

## 3.3 Online calculations

The problem solved online at each timestep is MATLAB's `quadprog` function.

$$\begin{aligned} &\min_{\phi} \frac{1}{2}\phi^\top H\phi + g^\top\phi \text{ subject to} \\ &\quad A_{ineq}\cdot\phi \le b_{ineq} \ , \\ &\quad A_{eq}\cdot\phi = b_{eq} \ , \end{aligned} \tag{3.7}$$

where

$$\phi = \begin{bmatrix} z_0 & z_1 & \dots & z_N & v_1 & v_2 & \dots & v_{N-1} \end{bmatrix}^\top \ . \tag{3.8}$$

### 3.3.1 Objective function

The $N$-horizon optimal control problem,

$$\sum_{j=0}^{N-1}(z_j^\top Q z_j + v_j^\top R v_j) + z_N^\top P z_N \tag{3.9}$$

becomes

$$\frac{1}{2}\cdot\phi^\top\cdot 2\cdot\begin{bmatrix} Q & & & & & & \\ & \ddots & & & \mathbf{0} & & \\ & & Q & & & & \\ & & & P & & & \\ & & & & R & & \\ & & \mathbf{0} & & & \ddots & \\ & & & & & & R \end{bmatrix}\cdot\phi + \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}\cdot\phi \ . \tag{3.10}$$

Naturally, since there are no linear elements in the objective function, the matrix

$g$ contains all zeros.

Listing 3.6 shows generating these matrices in the implementation in MATLAB.

**Listing 3.6:** Generating the matrices for the objective function

```
H = blkdiag(kron(eye(N), cost.Q), cost.P, ...
            kron(eye(system.N), cost.R));
mpc_cost.H = 2 * H;
mpc_cost.f = zeros(size(mpc_cost.H,1),1);
```

### 3.3.2 Equality constraints

There is a single equality constraint, which is the system equation (2.7) expressed as

$$-Az_j + z_{j+1} - Bv_j = 0 \ . \tag{3.11}$$

$$A_{eq} = \left[\begin{array}{cccc|ccc} -A & I & & 0 & -B & & 0 \\ & \ddots & \ddots & & & \ddots & \\ 0 & & -A & I & 0 & & -B \end{array}\right], \quad b_{eq} = \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}^\top \ . \tag{3.12}$$

**Listing 3.7:** Generating the matrices for the equality constraints

```
a = [kron(eye(N), -system.A) zeros(n*N, n)] + ...
    [zeros(n*N, n) kron(eye(N), eye(n))];
b = kron(eye(N), -system.B);
mpc_constraints.Aeq = [a b];
mpc_constraints.beq = zeros(N*n, 1);
```

### 3.3.3 Inequality constraints

The inequality constraints are

$$
\begin{aligned}
(z_j, v_j) \in \bar{\mathbb{Y}}, \quad j \in \mathbb{Z}_{0:N-1} \quad , \\
z_N \in \mathbb{X}_f \quad , \\
e_0 = x_0 - z_0 \in S_K \quad ,
\end{aligned}
\tag{3.13}
$$

which, in the polytopic-case, is equivalent to

$$
\begin{aligned}
Cz_j + Dv_j \leq e - \theta_N = e_t, \quad j \in \mathbb{Z}_{0:N-1} \quad , \\
Gz_n \leq h \quad , \\
S(x - z_0) \leq r \quad .
\end{aligned}
\tag{3.14}
$$

Since the actual system state $x$ for each iteration appears in one of these inequalities the general inequality constraint in (3.7) must be edited to reflect that: $A_{in} \cdot \phi \leq b_{in} + c_{in} \cdot x$.

Listing 3.8 shows the code for generating these in MATLAB.

$$
A_{in} = \left[\begin{array}{cccc|ccc}
C & & 0 & 0 & D & & 0 \\
 & \ddots & & & & \ddots & \\
0 & & C & 0 & 0 & & D \\
 & & 0 & G & \dots & 0 & \dots \\
-S & 0 & & & \dots & 0 & \dots
\end{array}\right], \ b_{in} = \begin{bmatrix} e_t \\ \vdots \\ e_t \\ h \\ r \end{bmatrix}, \ c_{in} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ -S \end{bmatrix} .
\tag{3.15}
$$

**Listing 3.8:** Generating the matrices for the inequality constraints

```
c = blkdiag(kron(eye(N), constraints.C), constraints.G);
d = [kron(eye(N), constraints.D); ...
     zeros(size(constraints.G,1) , N) ];
a = [c d];
s = zeros(size(n, 1), size(a,2));
s(1:size(constraints.S,1),1:size(constraints.S,2)) ...
  = -constraints.S;
b = kron(ones(N,1),constraints.e);
cin = zeros(N*size(constraints.C,1) ...
          + size(constraints.G,1), n);

mpc_constraints.Ain = [a; s];
mpc_constraints.bin = [b; constraints.h; constraints.r];
mpc_constraints.cin = [cin; -constraints.S];
```

### 3.3.4 Execution

The function `online_calc(problem,x)` solves the finite horizon optimization problem described in the previous section, with the current state of the system and returns the optimal state and control sequence and the optimal cost, seen in listing 3.9.

This is simulated with the system equation and a disturbance sequence generated with the assumptions on the uncertainty, seen in listing 3.10. Since this function is executed for every time step online, the complexity has to be as low as possible so the calculation of the current control action does not slow down the system.

After the simulation, the disturbance invariant tube, which is the disturbance invariant set $S_{KN}\alpha$ with the nominal state $\bar{x}_i$ as the center, is generated to used in the plotting of the results.

**Listing 3.9:** Solving the finite-horizon optimal control problem

```
function optimal = online_calc(problem,x)
mpc_cost = problem.mpc_cost;
mpc_constraints = problem.mpc_constraints;
options = optimset('Display', 'off');
% get optimal decision variable and optimal value
[output, optimal.cost_V] = ...
    quadprog(mpc_cost.H, ...
            mpc_cost.f, ...
            mpc_constraints.Ain, ...
            mpc_constraints.bin + mpc_constraints.cin
               * x, ...
            mpc_constraints.Aeq, ...
            mpc_constraints.beq, ...
            [], [], [], ...
            options);

% devectorize output to obtain optimal x
% and optimal u

z = zeros(n,N);
v = zeros(m,N);
z(:,1) = output(1:n);
for i=1:N
    z(:,i+1) = output(i*n + 1: (i+1)*n);
    v(:,i) = output((N+1)*n + (i-1)*m + 1:(N+1)*n + i*
       m);
end
optimal.v=v;
optimal.z=z;
```

**Listing 3.10:** Execution of robust tube MPC

```
W = Polyhedron(disturbance.E, disturbance.g);
W_vertices = size(W.V,1);
for i=1:Nsim
    w_sequence(:,i) = (W.V)'*rand(W_vertices,1);
end

x(:,1) = system.x0;

for i = 1:Nsim
    % Returns optimal z and v for horizon
    optimal(i) = mpc(problem, x(:,i));
    z(:,i) = optimal(i).z(:,1);
    v(:,i) = optimal(i).v(:,1);
    u(:,i) = v(:,i) + problem.system.K * ( x(:,i) - z
        (:,i));
    x(:,i+1) = problem.system.A * x(:,i) + problem.
        system.B * u(:,i) ...
                + problem.system.E * problem.system.
                    w_sequence(:,i);
end

%generating disturbance invariant tube
for i=1:Nsim
    X_tube(i) = z(:,i) + problem.system.S_K;
end
```

## 3.4 Controller

The parameters used in the cost function of the controller were

$$Q = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}, \quad R = 2, \quad P = \begin{bmatrix} 51.75 & 27.05 \\ 27.05 & 43.97 \end{bmatrix}, \tag{3.16}$$

where $Q$ and $R$ were selected by the designer and $P$ came from solving the discrete-time algebraic Riccati equation. The feedback matrix used in the controller also came from the Riccati equation,

$$K = \begin{bmatrix} -1.1089 & -1.9130 \end{bmatrix} \ , \tag{3.17}$$

which in turn resulted in the asymptotically stable closed-loop system matrix,

$$A_K = \begin{bmatrix} 1 & 0.1 \\ -0.1109 & 0.8087 \end{bmatrix}, \\ \text{with eigenvalues: } \lambda_{1,2} = 0.9043 \pm 0.04401i \ . \tag{3.18}$$

The final result is the controller

$$\begin{aligned} u_k &= \bar{u}_k + K(x_k - \bar{x}_k) \ , \\ \bar{u}_k, \bar{x}_k &= z_0, v_0 \text{ where } \ , \\ \mathbf{z} &= z_0, z_1, \ldots, z_N \ , \\ \mathbf{v} &= v_0, v_1, \ldots, v_{N-1} \ , \end{aligned} \tag{3.19}$$

with

$$
\begin{aligned}
\mathbf{z}, \mathbf{v} =_{\mathbf{z},\mathbf{v}} & \Big( \sum_{j \in \mathbb{Z}_{0:N-1}} (x_j^\top Q x_j + u_j^\top R u_j) + x_N^\top P x_N \Big) \ , \\
& \text{subject to} \\
z_{j+1} &= A z_j + B v_j, \ \ j \in \mathbb{Z}_{0:N-1} \ , \\
z_j, v_j &\in \bar{\mathbb{X}} \times \bar{\mathbb{U}} = \{x \mid Cx + Du \leq e_t\} \ , \\
z_N &\in \mathbb{X}_f = \{x \mid Gx \leq h\} \ , \\
x_k - z_0 &\in S_{K\,N=36}(\alpha = 0.15) = \{x \mid Sx \leq r\} \ .
\end{aligned}
\tag{3.20}
$$

The actual values for these sets are not that interesting in the context of the controller, but can be calculated from the code presented in this chapter.

# Chapter 4

# Results

The results from the implementation described in the previous chapter are presented here.

The system described in 3.1, with the tube-based robust MPC controller described in 3.4, was simulated with four different initial conditions,

$$x_0^1 = \begin{bmatrix} -15 \\ -15 \end{bmatrix}, \; x_0^2 = \begin{bmatrix} 19.5 \\ -19.5 \end{bmatrix}, \; x_0^3 = \begin{bmatrix} 15 \\ 15 \end{bmatrix}, \; x_0^4 = \begin{bmatrix} -19.5 \\ 19.5 \end{bmatrix} \quad , \tag{4.1}$$

from the four quadrants of $\mathbb{R}^2$. These were selected primarily for feasibility and because all points lie on the edge of the $N$-step controllable tube explained below.

Figures 4.1, 4.2, 4.3 and 4.4 show the four graphs with four different initial conditions. The disturbance invariant tube is shown in blue and the $N$-step controllable tube generated in section 3.2.5 is also plotted in gray. The purple set shows the same tube with the disturbance invariant set $S_{KN}(\alpha)$ added for every step. This describes the $N$-step controllable tube for the actual system, based on the fact that the system trajectory $x \in \{\bar{x} \oplus S_{KN}(\alpha)\}$. The axis are simply $x_1$ and $x_2$.

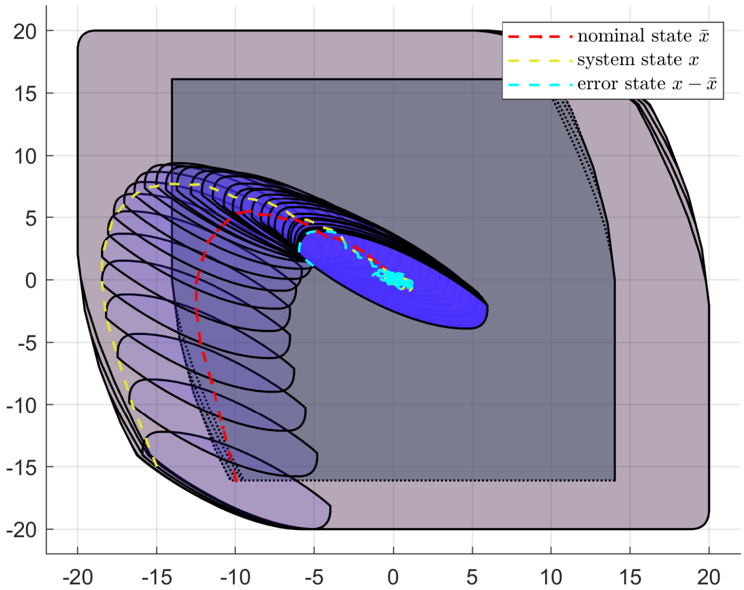


**Figure 4.1:** Simulation with $x_0 = x_0^1$

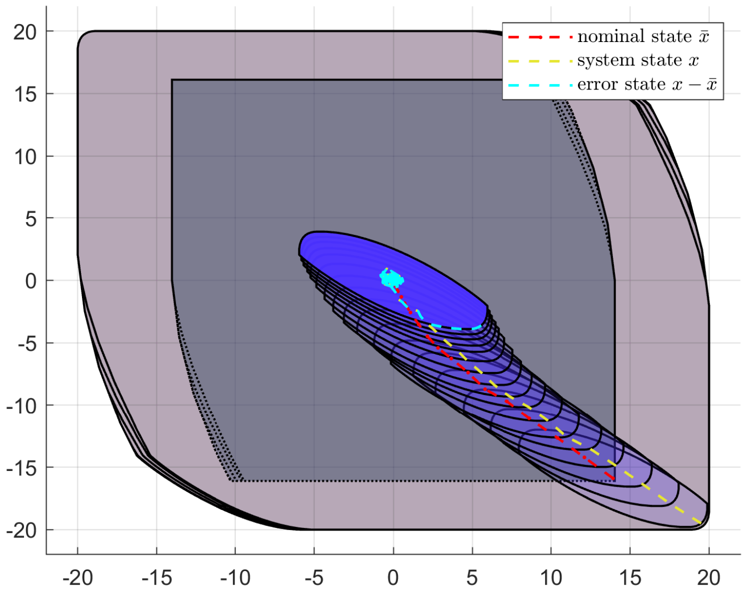


**Figure 4.2:** Simulation with $x_0 = x_0^2$

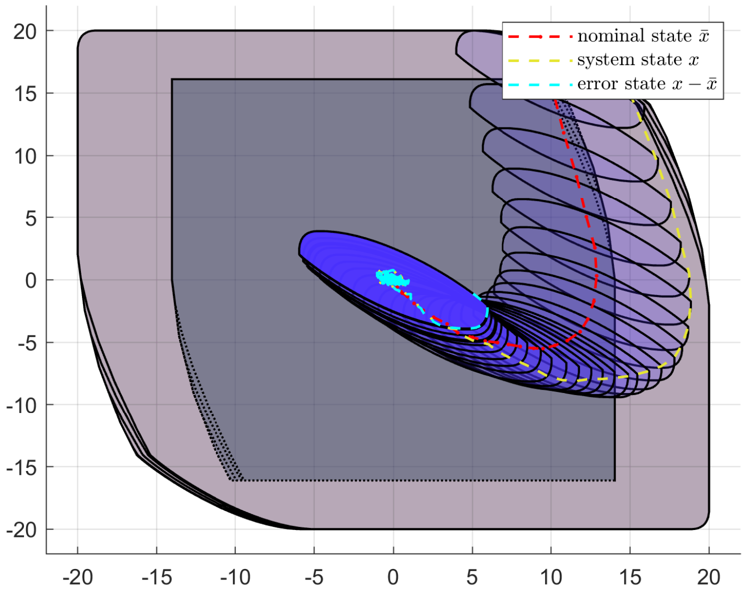


**Figure 4.3:** Simulation with $x_0 = x_0^3$

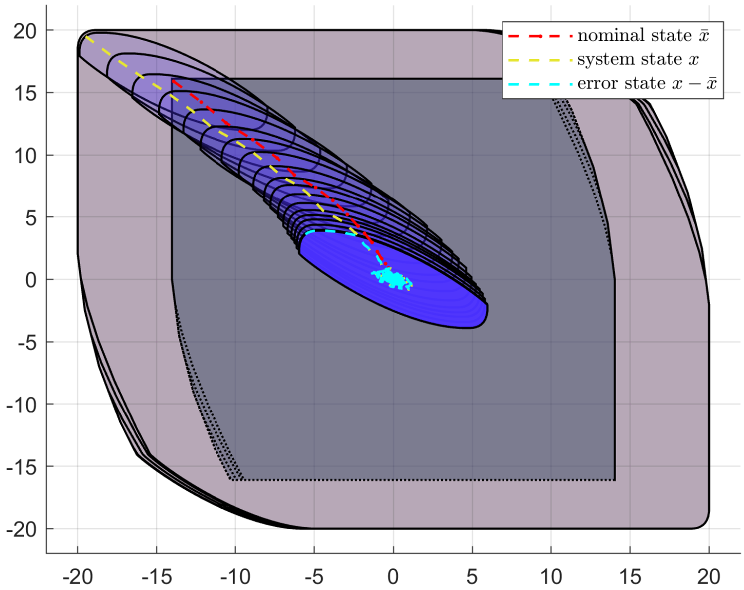


**Figure 4.4:** Simulation with $x_0 = x_0^4$

All four simulations converge towards the origin. The error dynamics behave as expected, beginning on the boundary of the set $S_K$ before eventually converging towards the origin.

The outer approximation of the minimal disturbance invariant set $S_K$ can have a large impact on the performance of the controller. Figure 4.5 displays the $N$-step controllable set for the nominal system and actual system for two different choices of $\alpha$, with minimal $N$. It is apparent that the choice of $\alpha$ has a significant impact on the set of initial states for which the controller is guaranteed stable.

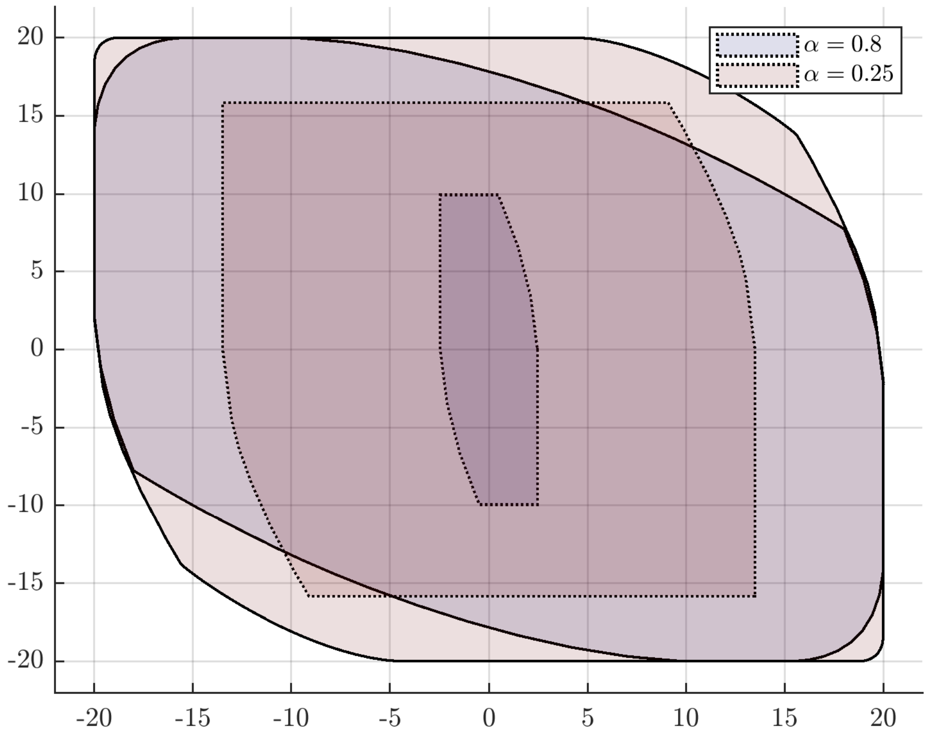


**Figure 4.5:** Controllable sets for $\alpha = 0.8$ and $\alpha = 0.25$

Figure 4.6 and 4.7 shows how the controller performs when $\alpha = 0.6$ is used to calculate the approximation, giving a less accurate and larger approximation than before. The simulation is still stable but the approximation of the disturbance invariant set in the origin, which represents the set of states for which the system can evolve to, is unnecessarily large. The initial states had to be decreased for feasibility to

$$\tilde{x}_0^2 = \begin{bmatrix} 19 \\ -19 \end{bmatrix}, \; \tilde{x}_0^3 = \begin{bmatrix} 16.5 \\ 11.5 \end{bmatrix}, \tag{4.2}$$

because the larger approximation also gives a smaller feasibility area, as seen in figure 4.5.

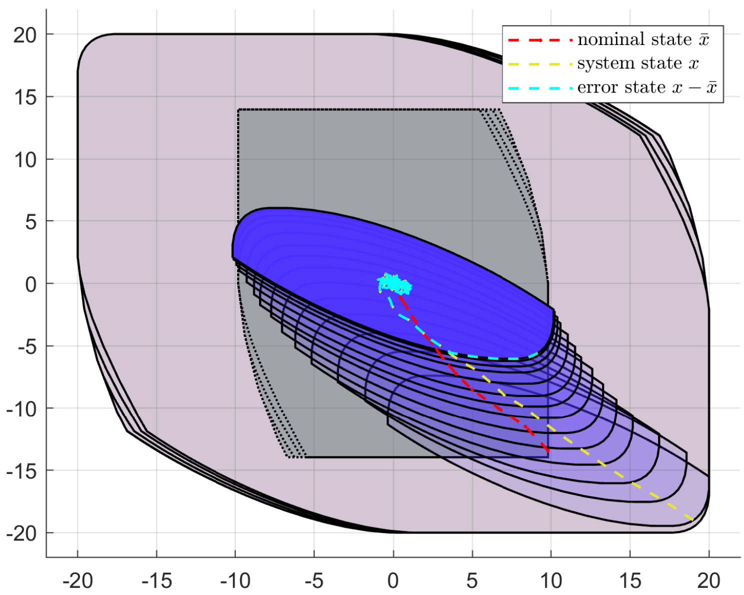


**Figure 4.6:** Simulation with $x_0 = \tilde{x}_0^2$

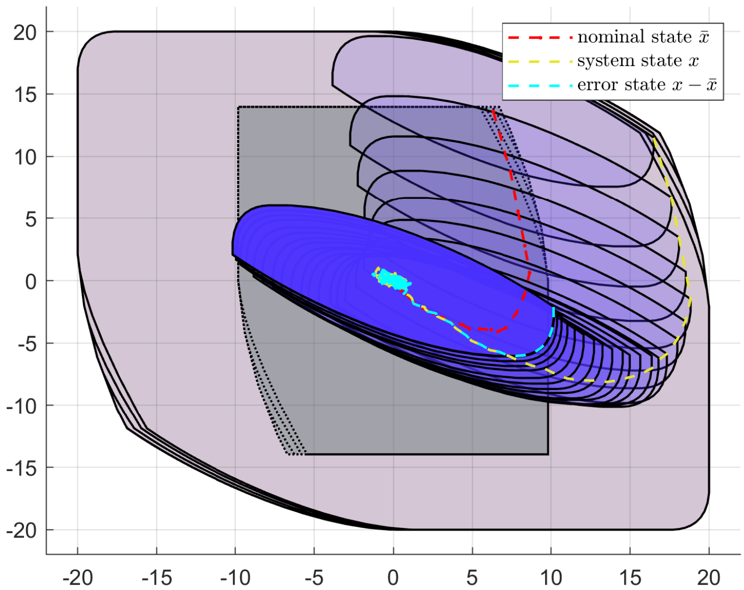


**Figure 4.7:** Simulation with $x_0 = \tilde{x}_0^3$

Figure 4.8 is included to show how accurate the outer approximation of $S_K$ is. Plotting the actual set $S_K$ isn't possible because no explicit expression exists for the system, but the sequence of sets $\{S_{Ki}\}$ for $i = 1, \ldots, 80$, is shown along with an approximation $S_{KN}(\alpha = 0.15)$.

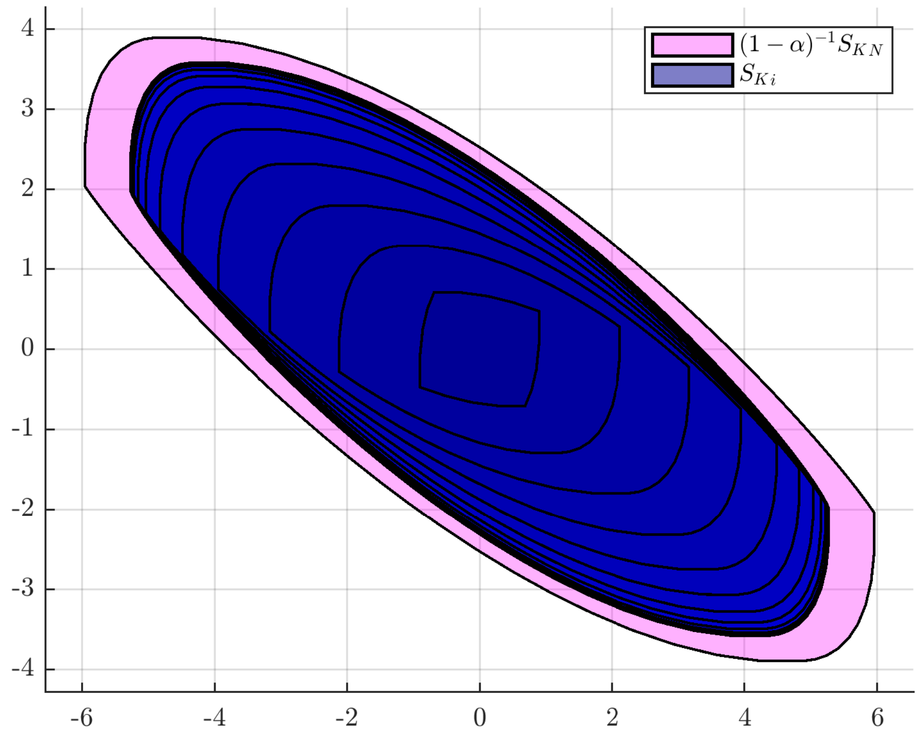


**Figure 4.8:** Plot of the sequence $S_{Ki}$ and $S_{KN}(\alpha)$

Figure 4.9 shows the minimal $N$ for varying $\alpha$, to satisfy the equation $A_K^N \mathbb{W} \subseteq \alpha\mathbb{W}$. From the figure, a minimal of $N = 13$ is needed, however this gives a high value of $\alpha$, resulting in a larger approximation than necessary. The area to choose from is above the plotted line, so there is some freedom in selecting this approximation.

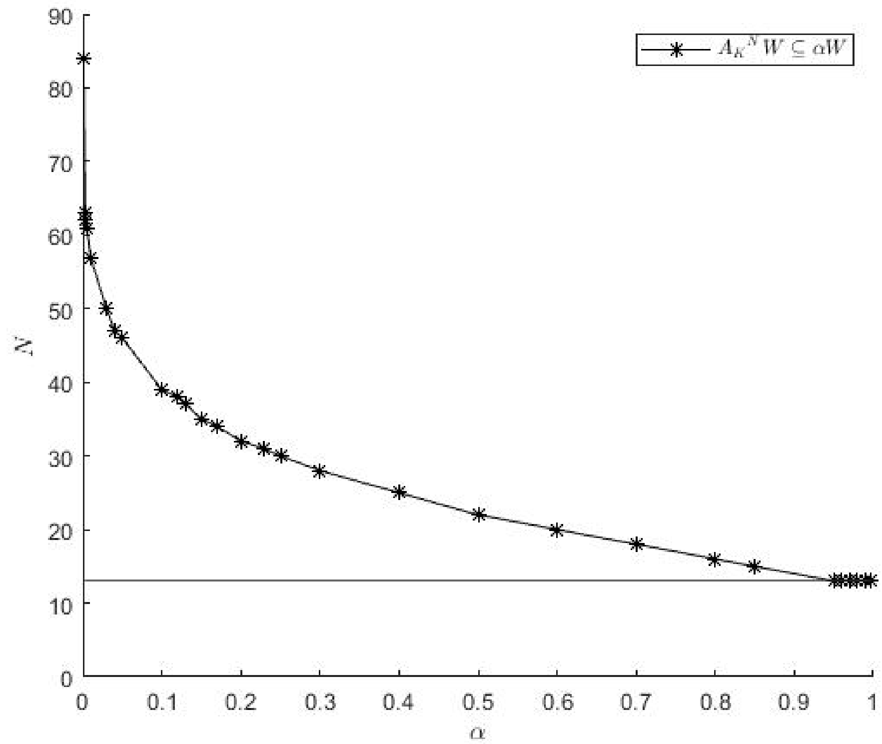


**Figure 4.9:** Minimum $N$ for varying $\alpha$

# Chapter 5

# Discussion

From the previous chapter, a stable tube-based robust model predictive control is presented for various initial states and different values of $\alpha$.

The state trajectory of the actual stays within a disturbance invariant set centered in the nominal state trajectory, which can be visualized by the error state being in the boundary of $S_K$ for the beginning of the system, then converging to the origin.

For this simulation, $\alpha = 0.15$ was chosen, after some trial and error. The accuracy of this approximation is shown in figure 4.8. The trial and error involved simulating for different values of $\alpha$ and seeing the change in the outer approximation, compared to the sequence of sets. After a certain threshold, decreasing $\alpha$ had a negligible effect on the approximation, but increased the computational cost of calculating it by increasing the minimal $N$. Selection of the ideal $\alpha$ to approximate $S_K$ prior to the simulation is not straight forward. As discussed in appendix D.2, and shown in figure 4.8 in the results, this selection can affect the performance of the controller.

Using one of the methods discussed in [9] to find the best approximation of $S_K$ is a potential improvement for the controller. This was not focused on in this paper, as the approximation of $S_K$ does not have a large impact.

The work done in [9] regarding the invariant approximations to $S_K$ is essential to the implementation of the tube-based MPC. There is however, room for improvement regarding the accuracy of the approximation. Specifically, developing an algorithm for the selection of $\alpha$ and $N$ would be a tremendous advancement.

Why is it beneficial for the disturbance invariant set to be a small as possible? Figure 4.6 shows the system with the controller using $\alpha = 0.6$ to approximate $S_K$. Remembering that the system should be as close to the origin as possible, and that the set displayed is the area for which the system state can evolve, having this set be minimal is parallel to the goal of the controller.

Another improvement to the controller is choosing two different feedback matrices, $K_e$ and $K_{\bar{x}}$. $K_{\bar{x}}$ should come from the Riccati equation to ensure stability of the MPC, but $K_e$, the feedback matrix for the error dynamics, can be selected and tuned so the system performs optimally. This can be a topic for further studies.

The use of the MPT toolbox should be reconsidered for future work. Calculations involving the structure from this toolbox has taken an unusually large amount of time given the implementation, and simulating for many initial conditions could take up to an hour.

# Chapter 6

# Conclusion

The robust control of uncertain systems is a large and important area of research because if its many applications. One such control method combines the control action from MPC with traditional feedback control to ensure the system trajectory stays close to the system modeled without uncertainty, which is easier to control. This is the tube-based method, first proposed in 2005 in [8], has increased in popularity, but most of the research is understandably focused in non-linear systems.

In this paper, the tube-based controller was applied to a two-dimensional linear system with robust additive disturbance. The intent being to visually and intuitively explain the main parts of this controller, in order to understand it better.

Chapter two discussed the mathematical basis for the controller, including the existence and approximation of a disturbance invariant set. Furthermore, the stability of the method was demonstrated. Chapter three described the implementation of the tube-based method in MATLAB and the specific linear system used in the simulation. In chapter four the results from the simulation were presented with two different outer approximations of the disturbance invariant set. Figures visually describing the accuracy of this approximation and the relationships between the parameters used to calculate the approximation were also shown. The discussion of the results in chapter five mostly focused on this outer approximation and how the potential improvement can affect the controller.

This method and the results presented in this paper can be used for all linear systems with a bounded disturbance.

The interest in the research of model predictive control has grown drastically since the beginning of its invention, and the tube-based method being only thirteen years old speaks to its potential improvement.

# Bibliography


[1] Bemporad, A., Morari, M., 1999. Robust model predictive control: A survey. In: Robustness in identification and control. Springer, pp. 207–226.

[2] Chen. H, Allgöwer, F., 1998. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. Automatica 34 (10), 1205–1217.

[3] Imsland, P. L., 2007. Introduction to model predictive control. In: Lecture Notes for TTK4135.

[4] Jia, D., Krogh, B., 2002. Min-max feedback model predictive control for distributed control with communication. Vol. 6. pp. 4507–4512.

[5] Kolmanovsky, I., G. E., 1998. Theory and computation of disturbance invariant sets for discrete-time linear systems. Mathematical Problems in Engineering 4, 317–367.

[6] Mayne, D., Rawlings, J., Rao, C., Scokaert, P., 2000. Constrained model predictive control: Stability and optimality. Automatica 36 (6), 789–814.

[7] Raković, S., 2018. Lecture slides from TK8115 numerical optimal control. In: Intesive Week with Saša Raković.

[8] Rakovic, S. V., Mayne, D. Q., 2005. A simple tube controller for efficient robust model predictive control of constrained linear discrete time systems subject to bounded disturbances .

[9] Rakovic, S., K. E.-K. K. M. D., 2005. Invariant approximations of the minimal robustly positively invariant sets. IEEE Transactions on Automatic Control 50.

[10] Rawlings, J., M. D., 2009. Model Predictive Control: Theory, Computation, and Design. Nob Hill Publishing.

[11] Richards, A., How, J., June 2006. Robust stable model predictive control with constraint tightening. In: 2006 American Control Conference.

[12] Sutherland, W. A., 1975. Introduction to Metric and Topological Spaces. Clarendon Press.

[13] Yu-Geng Xi, De-Wei Li, S. L., March, 2013. Model predictive control - status and challenges 39 (3), 222–236.

# Appendices

# Appendix A

# MPT Toolbox

The Multi-Parametric Toolbox (https://www.mpt3.org/) is an open-source Matlab-based toolbox. In this project, it is for the most part used for creating the structure `Polyhedron` in the code.

The command `S = Polyhedron(A,B)` creates the Polyhedron-structure in MATLAB equivalent to the set $S = \{x \mid Ax \leq b\}$. The Polyhedron-structure has many properties, but only a few are used in this paper.

- The property `V`, which is an array with the vertices of the set
- Addition + and subtraction − becomes set addition $\oplus$ and subtraction $\ominus$ for this structure
- The operator <= is equivalent to $\subseteq$ for this structure
- The function `s = support(S,x)` is equivalent to $s = \max_y x^\top y,\ y \in S$.

# Appendix B

# Necessary set theory

*Set addition*, also called the Minkowski sum, is defined as

$$A \oplus B = \{a + b \,|\, a \in A, b \in B\} \ . \tag{B.1}$$

*Set subtraction*, also called the Pontryagin difference, is defined as

$$A \ominus B = \{a \in A \,|\, a + b \in A \, \forall \, b \in B\} \ . \tag{B.2}$$

The Hausdorff distance $d_H(\cdot)$ is a metric for how far two subsets are from each other in any metric space (M,d), and is defined as

$$d_H(A, B) = \max\{\sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(a, b)\} \ . \tag{B.3}$$

A *close set* $S$ contains all its boundary points. A *bounded set* $S$ in some metric space $(M, d)$ means $\exists \, x \in M, \ r > 0$ such that $\forall s \in S, \ d(x, s) < r$.
A *compact set* is defined as a set that is closed and bounded.

The *interior of a set* $int(S)$, are all the points not in the boundary of $S$. The boundary of $S$ are all the points that can be approached from the outside and the inside of the set.

# Appendix C

# Controllability and backward reachability

*Backward reachability* for $x^+ = f(x), x \in \mathbb{X}$ with target set $\mathbb{X}_f$ determines the set of every $x \in \mathbb{X}$ that gives $f(x) \in \mathbb{X}_f$. The $N$-step backward reachable set for $x^+ = f(x)$ gives us the set of every $x_0 \in \mathbb{X}$ that ensures $x_1, x_2, \ldots \in \mathbb{X}$ and $x_N \in \mathbb{X}_f$.

Computing the $\infty - step$ backward reachable set with $\mathbb{X}_f = \mathbb{X}$, until the set sequence converges is called the *maximal positively invariant set*. This is often used in the nominal MPC to find the target set, or terminal constraint set, for the finite-horizon optimization problem.

The $N$-step backward reachable set for the polytopic-constraint case, where $\mathbb{X} = \{Cx \leq d\}$ and $\mathbb{X}_f = \{Gx \leq h\}$, is defined recursively as [7]

$$\begin{aligned} X_N^B &= \{Cx \leq d \,|\, G_{N-1}Ax \leq h_{N-1}\} \\ &= \{x \,|\, \leq G_N x \leq h_N\} \quad , \end{aligned} \tag{C.1}$$

where $G_{N-1}, h_{N-1}$ comes from the $(N-1)$-step backward reachable set $X^B_{N-1}$.

Much the same, *controllability* for $x^+ = f(x,u)$, $x \in \mathbb{X}$, $u \in \mathbb{U}$ with target set $\mathbb{X}_f$ determines the set of every $x \in \mathbb{X}$ for which there exists a control input $u \in \mathbb{U}$ that gives $f(x,u) \in \mathbb{X}_f$. The $N$-step controllable set gives us the set of every $x_0 \in \mathbb{X}$ for which there exists a control sequence $u_1, u_2, \ldots \in \mathbb{U}$ such that $x_1, x_2, \ldots \in \mathbb{X}$. The 1-step, 2-step, ..., $N$-step steps are together referred to as a *controllable tube*.

The $N$-step controllable set for the polytopic-constraint case, where $\mathbb{X} \times \mathbb{U} = \{Cx + Du \leq e\}$ and $\mathbb{X}_f = \{Gx \leq h\}$, is defined recursively as [7]

$$\begin{aligned} X^C_N &= Proj_{\mathbb{R}^2}\{Cx + Du \leq e \mid G_{N-1}(Ax + Bu) \leq h_{N-1}\} \\ &= \{x \mid \ \leq G_N x \leq h_N\} \quad , \end{aligned} \tag{C.2}$$

where $G_{N-1}, h_{N-1}$ comes from solving for the $(N-1)$-step controllable set $X^C_{N-1}$.

Both of these definitions can be extended to systems under uncertainty $w \in \mathbb{W}$ by requiring the exact same thing for all realizations of the disturbance sequence.

Further, a control invariant set is any controllable set where the target set $\mathbb{X}_f$ is the set itself. Meaning, the system evolution stays in the set.

# Appendix D

## D.1 Proof of explicit expression for $S_K$ when $A_K^N E = \alpha E$

We observe the following, when $A_K^N E = \alpha E,\ \alpha \in [0,1]$, given the definition for $S_K = \sum_{j=0}^{\infty} A_K^j E\,\mathbb{W}$.

$$
\begin{aligned}
S_K &= E\,\mathbb{W} + A_K E\,\mathbb{W} + \ldots A_K^{N-1} E\,\mathbb{W} \\
&\quad + A_K^N E\,\mathbb{W} + \ldots + A_K^{2N-1} E\,\mathbb{W} \\
&\quad + A_K^{2N} E\,\mathbb{W} + \ldots + A_K^{3N-1} E\,\mathbb{W} + \ldots \\
&\quad = S_{KN} + A_K^N S_{KN} + A_K^{2N} S_{KN} + \ldots \quad .
\end{aligned}
\tag{D.1}
$$

Because of this relation

$$
\begin{aligned}
A_K^N S_{KN} &= \alpha S_{KN} \ , \\
A_K^{2N} &= A_K^N A_K^N S_{KN} = \alpha A_K^N S_{KN} = \alpha^2 S_{KN} \ , \\
&\vdots \\
A_K^{tN} S_{KN} &= \alpha^t S_{KN} \ .
\end{aligned}
\tag{D.2}
$$

This results in

$$
S_K = (1 + \alpha + \alpha^2 + \ldots) \cdot S_{KN} \ . \tag{D.3}
$$

Noting that

$$
\sum_{i=0}^{\infty} r^i = \frac{1}{1-r}, \ |r| < 1. \tag{D.4}
$$

one gets the following result,

$$
S_K = (1 - \alpha)^{-1} S_{KN} \ . \tag{D.5}
$$

## D.2 Approximating the minimal disturbance invariant set $S_K$

In [9] a disturbance invariant upper approximation $S_{KN}(\alpha)$ for the minimal disturbance invariant set $S_K$ is derived.

From the assumptions regarding $A_K$ and $\mathbb{W}$, there exists $N \in \mathbb{Z}_+$ and $\alpha \in [0, 1)$ such that

$$
A_K^N \mathbb{W} \subseteq \alpha \mathbb{W} \ . \tag{D.6}
$$

Based on this, let

$$S_{KN}(\alpha) = (1-\alpha)^{-1} S_{KN} \tag{D.7}$$

be a disturbance invariant set for the system (2.12) and an outer approximation for $S_K$, i.e. $S_K \subseteq S_{KN}(\alpha)$.

To prove $S_{KN}(\alpha)$ is disturbance invariant, recall that it is equivalent to showing $A_K S_{KN}(\alpha) \oplus \mathbb{W} \subseteq S_{KN}(\alpha)$ from (2.17).

$$\begin{aligned} A_K S_{KN}(\alpha) \oplus \mathbb{W} &= A_K (1-\alpha)^{-1} \sum_{j=0}^{N-1} A_K^j E\,\mathbb{W} \oplus \mathbb{W} \\ &= (1-\alpha)^{-1} \sum_{j=1}^{N} A_K^j E\,\mathbb{W} \oplus \mathbb{W} \\ &= (1-\alpha)^{-1} A_K^N \mathbb{W} \oplus (1-\alpha)^{-1} \sum_{j=1}^{N-1} A_K^j E\,\mathbb{W} \oplus \mathbb{W} \ . \end{aligned} \tag{D.8}$$

Using the fact that $A_K^N \mathbb{W} \subseteq \alpha \mathbb{W}$ from (D.6),

$$\begin{aligned} A_K S_{KN}(\alpha) \oplus \mathbb{W} &\subseteq (1-\alpha)^{-1} \alpha \mathbb{W} \oplus (1-\alpha)^{-1} \sum_{j=1}^{N-1} A_K^j E\,\mathbb{W} \oplus \mathbb{W} \\ &= [(1-\alpha)^{-1}\alpha + 1]\mathbb{W} \oplus (1-\alpha)^{-1} \sum_{j=1}^{N-1} A_K^j E\,\mathbb{W} \\ &= (1-\alpha)^{-1} \mathbb{W} \oplus (1-\alpha)^{-1} \sum_{j=1}^{N-1} A_K^j E\,\mathbb{W} \\ &= (1-\alpha)^{-1} \sum_{j=0}^{N-1} A_K^j E\,\mathbb{W} = S_{KN}(\alpha) \ . \end{aligned} \tag{D.9}$$

Hence, $A_K S_{KN}(\alpha) \oplus \mathbb{W} \subseteq S_{KN}(\alpha)$ and therefore $S_{KN}(\alpha)$ is a disturbance invariant set. Based on the proof in section D.1, it is apparent that $S_K \subseteq S_{KN}(\alpha)$.

The same paper, [9], also proves that if one defines $N(\alpha)$ to give the smallest $N$ for any $\alpha$ such that D.6 holds then

$$S_{KN}(\alpha(N)) \to S_K \text{ as } N \to \infty \quad , \tag{D.10}$$

and vice-versa for $\alpha(N)$ when $\alpha \to 0$.

The paper suggests various methods for selecting the best approximation, one involving starting at a given $N$ and incrementing until $\alpha$ satisfying the conditions is found.