EXAMENSARBETE INOM TEKNIKOMRÅDET
TEKNISK FYSIK
OCH HUVUDOMRÅDET
ELEKTROTEKNIK,
AVANCERAD NIVÅ, 30 HP
STOCKHOLM, SVERIGE 2017

Robust Model Predictive Control for Autonomous Driving

EMMA THILÉN

KTH
SKOLAN FÖR ELEKTRO- OCH SYSTEMTEKNIK

Abstract

Autonomous driving is becoming popular nowadays. In order for autonomous cars to be fully accepted, high demands are placed on the safety side. One safety critical issue is the robustness to disturbances. In this work, a robust model predictive controller is designed for an autonomous vehicle. More specifically, robust output feedback model predictive control (ROFMPC) is used, and robustness is guaranteed through the use of robust invariant sets. The vehicle is modeled using a discretized, and linearized, version of a simple kinematic bicycle model, expressed in road-aligned coordinates. It is investigated for how large uncertainties robustness, and stability, can be guaranteed. Both external disturbances and measurement noise are considered. A steady-state Kalman filter is used to estimate the state of the vehicle. Two cases have been studied in simulation; straight line and curved line trajectory following. Results from simulations show that robustness can be ensured if the uncertainties in the system are sufficiently small. Finally, the ROFMPC algorithm is implemented on an F1/10 RC car.

Sammanfattning

Självkörande bilar blir alltmer populärt. För att självkörande bilar ska bli allmänt accepterade, har höga krav ställts på säkerheten. En viktig sak ur säkerhetssynpunkt är huruvida sytemet kan hantera störningar. I det här arbetet designas en robust modelprediktiv regulator för ett självkörande fordon. Mer specifikt används "Robust Output Feedback Model Predictive Control" (ROFMPC) och robusthet, gentemot både externa störningar och mätbrus, garanteras genom användingen av robust invarianta mängder. Fordonet modelleras med hjälp av en diskretiserad och linjäriserad enkel cykelliknande modell, uttryckt i naturliga koordinater. Två olika fall studeras genom simulering; dels då fordonet ska följa en rak bana och dels då det ska följa en krökt bana. Ett stationärt Kalman filter används till att uppskatta fordonets tillstånd. Resultaten från simuleringarna visar på att robusthet kan garanteras om störningarna är tillräckligt små. Slutligen är den robusta MPC-regulatorn implementerad på en F1/10 RC bil.

Acknowledgement

I would like to thank my supervisor Pedro F. Lima and my examiner Jonas Mårtensson for all their support throughout this thesis. Thanks also to Jay Nagdeo and Xiao Chen at the Smart Mobility Lab for their help and support with setting up the experiments.

Contents

Chapter 1

Introduction

Autonomous vehicles are no longer a far-fetched technology belonging to the distant future. In fact, autonomous vehicles are already reality. One of the main problems that has to be addressed for an autonomous vehicle is its ability to track a given path, i.e., the guidance problem. Recently, model predictive control (MPC) has become widely used for path tracking due to its ability to explicitly handle constraints.

In MPC, an optimization problem is solved at every time step in order to compute the optimal control sequence. Typically, only the first control is implemented and in the next time step when new information about the state is available a new control sequence is computed.

A system is said to be robust, with respect to a certain set of uncertainties or disturbances, if stability is guaranteed and performance requirements are met [1]. In nominal MPC it is often assumed that the feedback in the system will account for model uncertainty and external disturbances. However, a controller that provides good performance for a specific nominal model is not guaranteed to perform well when implemented on a physical system, due to model inaccuracies and the presence of uncertainties [2].

1.1 Motivation

Even though MPC already have been implemented on trucks and successfully tested [3], it is of great interest to develop controllers that are guaranteed to perform well also when there are larger uncertainties, e.g., in the position of the vehicle.

At Scania, a self-driving truck for mining applications is currently being developed [4]. Since the truck will be operating underground, GPS signals cannot be used for positioning. Therefore, it is necessary to rely on sensor measurements from IMUs (Inertial Measurement Unit), RADAR (Radio Detection and Ranging) and LIDAR (Light Detection and Ranging). The lateral position of the truck can be estimated by measuring the distances to the walls. There is, however, some amount of uncertainty attached to these measurements. This uncertainty may increase significantly when the truck enters an intersection, due to the range of the sensors. When the uncertainty increases the state estimates may become poorer. If the actual state and the estimated state differs too much

it can lead to that the controller fails to keep the truck close to its desired path. In order to mitigate the influence of the uncertainty in the position of the vehicle, it is therefore desirable to design a controller that is robust to disturbances.

1.2 Problem formulation

An overall goal of this thesis is to study how the influence of model uncertainties and external disturbances can be mitigated using robust MPC. In particular, we want to study how robust MPC can be used for autonomous vehicle trajectory tracking. For this purpose, available literature on the topic is reviewed.

The main expected outcome of this thesis is the design of a robust MPC for an autonomous vehicle that is able to follow given trajectories in the presence of model uncertainties and external disturbances. The main focus when designing the controller is on the lateral control of the vehicle. Moreover, only linear MPC is considered within this project and therefore we derive a linearized model of the vehicle dynamics, to be used as prediction model in the MPC.

In this thesis only the path following problem is considered, i.e., the path planning problem is assumed to be solved and a reference trajectory has been provided. The general flow of the control process is depicted in Figure 1.1. In the filtering part of the control process prediction and measurement data are merged to give a better estimate of the state. A common choice of filter for nonlinear systems is the Extended Kalman Filter (EKF). However, as mentioned, in this project the vehicle dynamics are modeled by a linear model and, therefore, a Kalman filter (KF) is used.

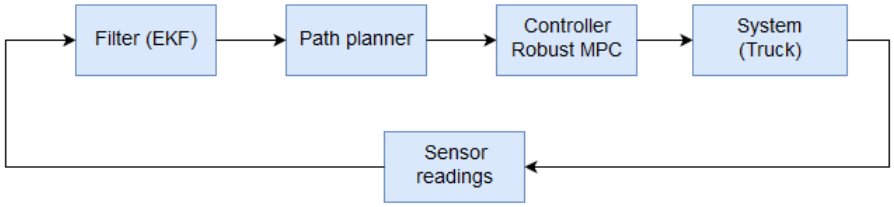

Figure 1.1: The flow of the control process for the guidance of an autonomous vehicle.

Simulations are used in order to study and evaluate the performance of the controller. For the sake of developing a realistic simulation environment the following tasks will be addressed:

- How to model sensor noise and disturbances in a simple but realistic manner.
- Generation of feasible reference trajectories.

One of the main features of the controller that we are interested to investigate is how large uncertainties of the state estimate the controller is capable of handling. Another objective is to investigate if it is possible to impose stability on the system by introducing a constraint on the final state in the MPC.

The final goal is to implement and test the controller on small scale models in the Smart Mobility Lab (SML) at KTH.

1.3 Related work

Through the years there has been a lot of research on robust MPC and several techniques for synthetizing a robust model predictive control have been proposed. One of these methods is the so-called min-max formulation and it was first proposed by Campo and Morari [1987]. With this method the idea is to minimize the worst-case deviation from some reference. In Kothare et al. [1996] a technique involving linear matrix inequailities (LMIs) is presented.

Tube-based robust MPC is a class of techniques in which all possible trajectories resulting from the realization of the disturbance are assumed to lie inside a tube. One tube-based approach was presented by Rakoviç and Mayne [2005]. In Mayne et al. [2005] a technique in which the initial state of the system is a decision variable is provided.

1.4 Thesis outline

In Chapter 2, the basics of MPC, as well as some basic definitions in set theory, are found. Moreover, different approaches for robust MPC are described briefly in this chapter. In Chapter 4, some basic theory about the Kalman filter is found. The vehicle model used in the simulations can be found in Chapter 3. The design of the robust controller is described in detail in Chapter 5 and Chapter 6. In Chapter 7, results from simulation are presented. Details about the implementation on the small scale car are given in Chapter 8. Finally, in Chapter 9, some conclusions are drawn and we also give some suggestions for future work.

Chapter 2

Background

2.1 Model Predictive Control

MPC is typically formulated in the state space [1]. A system that is often considered in the literature is the discrete linear time-invariant system

$$x(t+1) = Ax(t) + Bu(t), \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ denote the state and the input, respectively. The basic idea in MPC is to minimize some cost function, while still ensuring that a set of constraints are fulfilled. The MPC problem is typically written on the form

$$\begin{aligned} \min_{\mathbf{u}} \quad & J(x(t), \mathbf{u}) \\ \text{subject to} \quad & x_{t+k+1} = Ax_{t+k} + Bu_{t+k}, \ \forall k = 0, ..., N-1 \\ & x_{t+k} \in \mathbb{X}, \ \forall k = 0, ..., N-1 \\ & u_{t+k} \in \mathbb{U}, \ \forall k = 0, ..., N-1 \\ & x_{t+N} \in \mathbb{X}_f, \\ & x_t = x(t), \end{aligned} \tag{2.2}$$

where $\mathbf{u} = (u_t, ..., u_{t+N-1})$ is a sequence of control inputs, x_{t+k} is the state at time $t+k$ as predicted at time t, and N is the prediction horizon [5]. The sets $\mathbb{X} \in \mathbb{R}^n$ and $\mathbb{U} \in \mathbb{R}^p$ define the constraints on the state and the input, respectively. The set $\mathbb{X}_f \subseteq \mathbb{X}$ defines the terminal constraint on the state.

If a regulation problem is considered, i.e., the system (2.1) should be steered to the origin, then a common choice of the cost function $J(x(t), \mathbf{u}))$ is the quadratic cost function

$$J(x(t), \mathbf{u}) = x_{t+N}^T P_f x_{t+N} + \sum_{k=1}^{k=N} x_{t+k}^T Q x_{t+k} + u_{t+k}^T R u_{t+k}, \tag{2.3}$$

where $P_f, Q \succeq 0$ (semi-positive definite) and $R \succ 0$ (positive definite).

In the case of the servo problem, i.e., tracking of a reference signal, the

modified quadratic cost function

$$\begin{aligned} J(x(t), \mathbf{u}) =& (x_{t+N} - x_{t+N}^{ref})^T P_f (x_{t+N} - x_{t+N}^{ref}) \\ &+ \sum_{k=1}^{k=N} (x_{t+k} - x_{t+k}^{ref})^T Q (x_{t+k} - x_{t+k}^{ref}) + u_{t+k}^T R u_{t+k}, \end{aligned} \tag{2.4}$$

where $x_{t+k}^{ref}, x_{t+N}^{ref}$ describes the reference trajectory, is often used.

The standard MPC algorithm can be described by the following steps [1]:

1. Measure the current state $x(t)$.
2. Solve the optimization problem (2.2).
3. Apply the first control of the optimal control sequence.
4. Wait one sampling time and go back to 1.

There exists efficient solvers for convex optimization problems and it is therefore desirable that the MPC problem (2.2) is convex. This is ensured if the cost function is convex, e.g., quadratic, the prediction model is linear and the constraint sets $\mathbb{X}, \mathbb{U}$ are convex. Throughout this thesis, only linear prediction models will be considered.

2.2 Robustness

A system is said to be robust with respect to a certain set of uncertainties if stability can be guaranteed and the performance specifications are met [1].

A robust controller has to ensure that the constraints are never violated for any admissible disturbance realization.

The uncertainties that may arise in a system are due to several factors [6]. Typical sources of uncertainty are external disturbances, measurement noise, inaccurate values of the model parameters, unmodeled non-linearities, etc...

One of the most common types of uncertainties that is considered in the literature is additive disturbance. Usually, it is assumed that the actual state of the system can be measured, i.e., it is assumed that there is no noise in the measurements. Another common way to model the uncertainty of the system is to assume parametric uncertainty or to assume that the true system belongs to a polytopic set.

2.3 Set theory

As mentioned in Section 2.1, it is necessary that the constraint sets are convex, in order for the MPC problem (2.2) to be convex. A common choice is to let $\mathbb{X}, \mathbb{X}_f$, and $\mathbb{U}$ be polyhedral or polytopic sets. Another possibility is to require that the constraint sets are C sets. The definitions for these sets are given as follows:

Definition 1 (Polyhedron) *a polyhedron is the convex space defined by the intersection of closed halfspaces [5].*

Definition 2 (Polytope) *A polytope is a bounded polyhedron [5].*

Definition 3 (C set) *A C set is compact, convex, and contains the origin in its interior [7].*

Some useful set operations are the Minkowski set addition and the Pontryagin set difference [5].

Definition 4 (Minkowski sum) *The Minkowski sum of two sets P and Q is defined as*

$$P \oplus Q = \{x + y : x \in P,\ y \in Q\}.$$

Definition 5 (Pontryagin difference) *The Pontryagin difference of two sets P and Q is defined as*

$$P \ominus Q = \{x \in \mathbb{R}^n : x + y \in P,\ \forall\ y \in Q\}.$$

In robust MPC, invariant sets play an important role and, therefore, we state some important definitions here, and we follow the structure of [5].

Consider the autonomous system

$$x(t+1) = f_a(x(t)), \tag{2.5}$$

and the system with controllable input

$$x(t+1) = f(x(t), u(t)). \tag{2.6}$$

Both systems are subject to the constraints

$$x(t) \in \mathbb{X}, u(t) \in \mathbb{U}, \forall t \in \mathbb{N}_+, \tag{2.7}$$

where the sets $\mathbb{U}$ and $\mathbb{X}$ are polyhedra.

Definition 6 (Positive invariant set) *For the system* (2.5) *subject to the constraints* (2.7)*, a set $\Omega \subset \mathbb{X}$ is called positive invariant if*

$$x(t) \in \Omega \implies x(t+k) \in \Omega,\ \forall k \in \mathbb{N}_+.$$

Definition 7 (Control invariant set) *For the system* (2.6) *subject to the constraints* (2.7)*, a set $\Omega \subset \mathbb{X}$ is called positive control invariant if*

$$x(t) \in \Omega \implies \exists u(t) \in \mathbb{U},\ \text{s.t. } x(t+1) \in \Omega, \forall t \in \mathbb{N}_+.$$

In this thesis we are concerned about systems subject to uncertainties. Consider, therefore, the autonomous system with an unknown disturbance

$$x(t+1) = f(x(t), w(t)), \tag{2.8}$$

and the system with external controllable input, as well as an unknown disturbance

$$x(t+1) = f(x(t), u(t), w(t)). \tag{2.9}$$

Both systems are subject to the constraints

$$x(t) \in \mathbb{X}, u(t) \in \mathbb{U}, w(t) \in \mathbb{W}, \forall t \in \mathbb{N}_+, \tag{2.10}$$

where the sets $\mathbb{X}, \mathbb{U},$ and $\mathbb{W}$ are polyhedra.

Definition 8 (Robust positive invariant set) *For the system* (2.8) *subject to the constraints* (2.10)*, a set* $\Omega \subset \mathbb{X}$ *is called robust positive invariant (RPI) if*

$$x(t) \in \Omega \implies x(t+k) \in \Omega, \forall w \in \mathbb{W}, \forall k \in \mathbb{N}_+.$$

Definition 9 (Robust control invariant set) *For the system* (2.9) *subject to the constraints* (2.10)*, a set* $\Omega \subset \mathbb{X}$ *is called positive control invariant if*

$$x(t) \in \Omega \implies \exists u(t) \in \mathbb{U}, \text{ s.t. } x(t+1) \in \Omega, \forall w \in \mathbb{W}, \forall t \in \mathbb{N}_+.$$

Definition 10 (Minimal robust positive invariant set) *The minimal positive invariant (mRPI) set for the system* (2.8) *is the RPI set that is contained in every other closed RPI set of* (2.8)*. [8]*

2.3.1 Computation of minimal robust positive invariant sets

Let F_∞ denote the minimal robust positive invariant set (mRPI) for a system

$$x(t+1) = Ax(t) + w(t), \tag{2.11}$$

where $x \in \mathbb{R}^n$ and w is a disturbance belonging to the C set $\in \mathbb{W} \subset \mathbb{R}^n$. Let $\mathcal{U}_1, ..., \mathcal{U}_n$ be a sequence of sets and define $\bigoplus_{j=1}^{n} \mathcal{U}_j \triangleq \mathcal{U}_1 \oplus ... \oplus \mathcal{U}_n$. Then F_∞ is given by [8]

$$F_\infty = \bigoplus_{j=0}^{\infty} A^j \mathbb{W}. \tag{2.12}$$

2.4 Robust MPC

The most common approach to deal with uncertainties is to simply ignore them and rely on the inherent robustness of MPC. That is, to only consider the nominal system when designing the controller and then, hope that the fact that a new control is computed at every sample, mitigates any disturbances. There are, however, no guarantees that the system performs well, or even remains stable, in the presence of uncertainties [2]. Therefore, it is desirable to develop MPC schemes that directly take into account uncertainties in the system.

An uncertain system commonly considered is the LTI-system with additive disturbance and measurement noise

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + Ew(t), \\ y(t) &= Cx(t) + Dv(t), \end{aligned} \tag{2.13}$$

where w is the unknown state disturbance and v is the measurement noise. In order to control the system (2.13), using robust MPC, the following optimization

problem is solved

$$\begin{aligned} \min_{\mathbf{u}} \quad & J(x(t), \mathbf{u}) \\ \text{subject to} \quad & x_{t+k+1} = Ax_{t+k} + Bu_{t+k} + Ew_{t+k}, \ \forall k = 0, ..., N-1 \\ & x_{t+k} \in \mathbb{X}, \forall w \in \mathbb{W}, \forall v \in \mathbb{V}, \ \forall k = 0, ..., N-1 \\ & u_{t+k} \in \mathbb{U}, \forall w \in \mathbb{W}, \forall v \in \mathbb{V}, \ \forall k = 0, ..., N-1 \\ & w_{t+k} \in \mathbb{W}, \forall k = 0, ..., N-1 \\ & v_{t+k} \in \mathbb{V}, \forall k = 0, ..., N-1, \\ & x_{t+N} \in \mathbb{X}_f, \\ & x_t = x(t), \end{aligned} \tag{2.14}$$

where $J(x(t), \mathbf{u})$ is the cost function. In the sequel, if not stated otherwise, the sets $\mathbb{X}$ and $\mathbb{U}$ are assumed to be polyhedral and polytopic sets, respectively. The sets $\mathbb{W}$ and $\mathbb{V}$ are typically assumed to be bounded since, in general, robustness cannot be guaranteed if the disturbances are too big.

2.5 Feedback MPC

In conventional MPC, the optimization problem is solved in open-loop, i.e, the fact that there is a possibility to adjust the control in the next time step is not taken into account. This can, however, be a very prohibitive approach for an uncertain system and even lead to feasibility problems.

In order to overcome these problems, it is possible to use so-called feedback MPC instead. In feedback MPC, a control policy is determined, as opposed to a control sequence. The control policy is a sequence of control laws, which ensures that the closed-loop influence is explicitly taken into account. Thus, the algorithm takes into account that at the next time step there is a chance to update the control sequence and, thereby, adjust the control depending on the actual state, which could not be predicted exactly at the previous time.

The result is that feedback MPC performs better than conventional MPC in the presence of uncertainty. However, a big drawback with feedback MPC, is that the complexity of the optimization problem increases when the decision variable is a control policy [6].

2.6 Tube-based robust MPC

When there is uncertainty present in the system, there is a bundle of possible future trajectories at every time instant, each of them corresponding to a particular disturbance realization. For a constrained system, all trajectories in the bundle have to satisfy the constraints. The main idea in tube-based MPC is to construct a tube, in which all controlled state realizations are confined.

In the approach described in [6], the tube is generated in two steps. First, conventional MPC with tightened constraints is used to determine the center of the tube. Then, a local feedback controller is used in order to restrict the size of the tube.

In [9], the uncertain time invariant system (2.13), where the sets $\mathbb{W}$ and $\mathbb{V}$ are assumed to be C sets. The basic idea in the proposed control methodology

is to use an observer to estimate the system state and then, control the observer state $\hat{x}(t)$, rather than the actual state. For this to be meaningful, however, the state estimation error has to be bounded, in order to ensure that the actual state does not violate the constraints, and that it also is steered to a neighbourhood of the desired setpoint. In the paper, a Luenberger observer is used to estimate the state and the estimation error is defined as

$$\tilde{x}(t) = x(t) - \hat{x}(t). \tag{2.15}$$

Moreover, a nominal system, where the noise and disturbance in the original system are ignored, is introduced:

$$\bar{x}(t+1) = A\bar{x}(t) + B\bar{u}(t). \tag{2.16}$$

The difference between the state estimate and the nominal state,

$$e(t) = \hat{x}(t) - \bar{x}(t), \tag{2.17}$$

is called the control error. The actual state can be expressed as

$$x(t) = \bar{x}(t) + e(t) + \tilde{x}(t). \tag{2.18}$$

Thus, provided that the observer and the control error can be bounded and shown to lie in sufficiently "small" sets, it is possible to find an admissible control sequence $\bar{\mathbf{u}}$ and $\bar{x}$, such that the real state of the system satisfies the constraints. In [9], it is shown that, under the given circumstances, $\tilde{x}$ and e can be bounded using robust positive invariant sets. After computing the invariant sets $\tilde{S}$ and $\bar{S}$, for the observer error and control error, respectively, the authors proceed by establishing tighter constraints on the observer state, the nominal state, and the nominal control. The conservativeness of the control depends on the size of the sets $\tilde{S}$ and $\bar{S}$ and it is, therefore, desirable that they are small.

Next, MPC is used to control the nominal state, subject to the tightened constraints, such that the real state is ensured to satisfy the original constraints. Thus, an optimal control problem for the nominal system is formulated. The main difference, as compared to conventional MPC, is that the initial nominal state $\bar{x}_t$ is treated as a decision variable. Moreover, a terminal state constraint is introduced for the nominal system, $\bar{x}_N \in \mathbb{X}_f$. This constraint is a stabilizing constraint and if the state can be steered to $\mathbb{X}_f$ it will remain inside this set. By solving the nominal optimal control problem, an optimal nominal initial state $\bar{x}_t^*$ and an optimal nominal control sequence $\bar{\mathbf{u}}^*$ are obtained. The control u that is applied to the system is given by $u(t) = \bar{u}_t^*(\hat{x}(t)) + K(\hat{x}(t) - \bar{x}_t^*(\hat{x}(t)))$.

All the sets, $\tilde{S}, \bar{S}$, and $\mathbb{X}_f$, can be precomputed and the optimization problem that has to be solved online is only marginally more complex than a standard MPC problem [9].

In [10], by the same authors, an extension to the previously described control methodology is presented. The main difference is that, in [9], a steady state assumption is made for the observer error, while this condition is relaxed in [10]. In [10], it is assumed that the initial estimation error and control error, $\tilde{x}(0)$ and $e(0)$ lie in the invariant sets $\tilde{S}_0$ and $\bar{S}_0$, respectively. The authors then show that the estimation and control error are confined to sets that are, at least, non-increasing. The difference is, thus, that larger errors are allowed

in the beginning and by ensuring that the errors belong to smaller and smaller sets, less conservative control can be used later in the process.

Also, in [11], a similar approach for the simpler case when the there is no measurement noise, i.e., the observer state is the same as the real state, is presented.

2.7 Min-max robust MPC

The min-max formulation for robust MPC was first proposed in 1987 by Campo and Morari [2]. The main idea behind the control scheme they proposed, is the minimization of the worst-case deviation from a reference trajectory. In min-max MPC, the optimization problem is typically formulated on the form

$$\begin{aligned} \min_{\mathbf{u}} \max_{w} \quad & J(x(k), \mathbf{u}) \\ \text{subject to} \quad & x_{t+k} \in \mathbb{X}, \ \forall \ w \in \mathbb{W}, \ \forall \ k = 0, ..., N-1 \\ & u_{t+k} \in \mathbb{U}, \ \forall \ w \in \mathbb{W}, \ \forall \ k = 0, ..., N-1 \\ & w_{t+k} \in \mathbb{W}, \ \forall \ k = 0, ..., N-1. \end{aligned} \tag{2.19}$$

Thus, the idea is to minimize the worst-case performance cost. A drawback with this approach is that the optimization is done in open-loop, meaning that the control sequence has to be able to handle all admissible disturbance sequences.

2.8 Feedback min-max MPC

In the min-max formulation discussed in the previous section, the predictions are made in open-loop. However, as discussed before, this can be unnecessarily conservative and may lead to feasibility problems. A min-max feedback MPC approach is proposed in [12]. The main idea of the method is to steer the system to a robust control invariant set $\mathbb{X}_f$ using min-max MPC and then use a linear state feedback controller when the state is in the control invariant set. The objective function is assumed to be convex and zero for any state inside the control invariant set. Moreover, in the optimization problem, the set $\mathbb{X}_f$ is used as a terminal state constraint and is necessary to ensure stability [12].

The control methodology consists of the following steps: at each sampling time, measure the state $x(t)$. If $x \in \mathbb{X}_f$, use the control $u = -Kx$, otherwise solve a min-max optimization problem, and set u to the first control of the obtained optimal control sequence.

The main drawback with this approach, compared to the open-loop prediction case, is the computational load. This is due to the fact that all possible disturbance realizations are considered in the optimization, which in the general case leads to an optimization problem of infinite dimension. However, if the system is linear and the constraint sets are convex, this problem can be solved. In [12], it is shown that if the disturbance set $\mathbb{W}$ is a polytope in $\mathbb{R}^n$ then it is sufficient to only consider the worst-case realizations of the disturbance. That is, only the vertices of $\mathbb{W}$ have to be considered in the optimization problem.

Thus, the computational complexity can be significantly reduced by only considering extreme disturbance realizations. However, the number of realizations that has to be accounted for increases combinatorially as the prediction

horizon increases [12]. Therefore, it is concluded that min-max feedback MPC should be avoided if a large horizon is needed. Still, the method is assumed to be efficient for small horizons.

2.9 LMI-based robust MPC

Another approach, based on linear matrix inequalities (LMIs), for synthetizing a robust MPC was proposed in [13] by Kothare et al. In the paper, the case of polytopic uncertainty, as well as structured feedback uncertainty is considered. Moreover, it is assumed that the exact state can be measured at every sampling time. In the LMI formulation of the problem, input and state constraints are embedded into LMIs. The aim is then to minimize a worst-case cost, hence, a min-max optimization problem has to be solved. In the paper, the quadratic cost function given in (2.3) with $P_f = Q$ is considered. However, unlike the min-max methods described in previous sections, in [13], an upper bound on the worst-case cost is derived. Then, this upper bound is minimized using a constant linear feedback law $u_k = Lx_k$. As in conventional MPC, only the first control is applied to the system and when new measurements become available a new feedback gain L is computed.

Chapter 3

Vehicle model

3.1 The kinematic bicycle model

The vehicle can be modeled using the non-linear kinematic bicycle model.

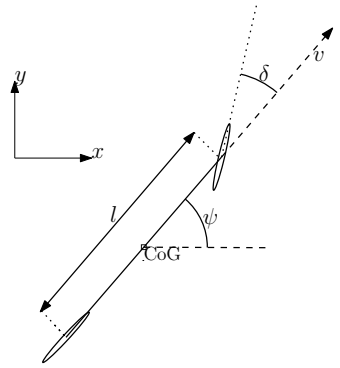

Figure 3.1: The non-linear kinematic bicycle model.

In order to simplify the model, it is assumed that only the front wheel can be steered, which also is the case in most vehicles [14]. Moreover, in this work it is assumed that the vehicle does not slip, any slippage is thus considered as an external disturbance. Under this assumption, the slip angle is zero, meaning that the velocity is directed along the heading of the vehicle, and the equations that describes the model are [14]

$$\dot{x} = v\cos(\psi), \tag{3.1}$$
$$\dot{y} = v\sin(\psi), \tag{3.2}$$
$$\dot{\psi} = \frac{v}{l}\tan(\delta), \tag{3.3}$$

where l is the length between the axes on the vehicle, ψ is the heading angle of the vehicle, and the speed v and the steering angle δ are the input signals to the system.

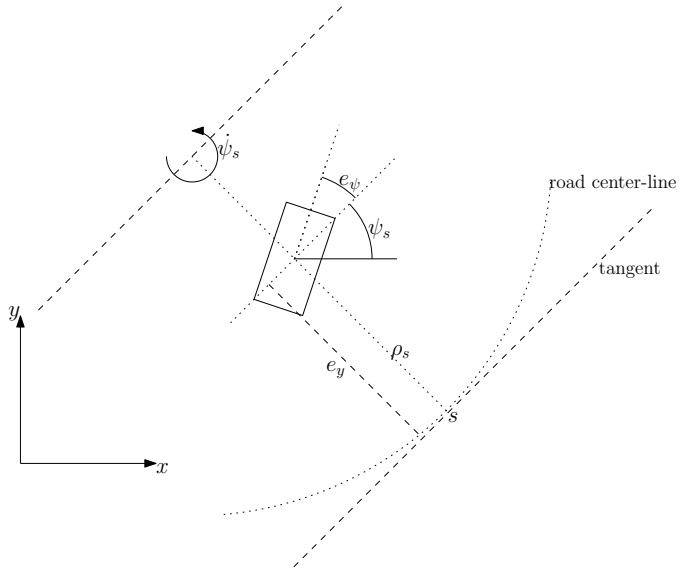

Figure 3.2: The vehicle in the road-aligned coordinate frame.

3.2 Road-aligned coordinates

In order to easier handle roads with different curvatures, it is convenient to use a road-aligned coordinate frame. By introducing a variable s, which represents, the distance along the center-line and then modeling the system dynamics in terms of s, a completely time-dependent, spatial-based, representation of the vehicle model can be obtained. The states that are of interest to control are the lateral deviation with respect to the center-line and the error in the heading angle.

From Figure 3.2 the following relations can be derived

$$\dot{e}_y = \quad v\sin(\psi) \tag{3.4}$$

$$\dot{e}_\psi = \quad \dot{\psi} - \dot{\psi}_s \tag{3.5}$$

$$\dot{s} = \quad \frac{\rho_s v\cos(e_\psi)}{\rho_s - e_y}, \tag{3.6}$$

where e_y and e_ψ are the the lateral and heading deviation with respect to the center-line, ρ_s is the road radius and ψ_s is the heading angle of the road. Using that $\frac{d(\cdot)}{ds} = \frac{d(\cdot)}{dt}\frac{dt}{ds} = \frac{d(\cdot)}{dt}\frac{1}{\dot{s}}$, the following relations can be derived

$$\begin{aligned} e_y' = \frac{de_y}{ds} \quad &= \frac{\dot{e_y}}{\dot{s}} \quad = \frac{(\rho_s - e_y)\kappa}{\rho_s\cos(e_\psi)} - \psi', \\ e_\psi' = \frac{de_\psi}{ds} \quad &= \frac{\dot{e_y}}{\dot{s}} \quad = \frac{\rho_s - e_y}{\rho_s}\tan(e_\psi), \end{aligned} \tag{3.7}$$

which is a spatial-based representation of the vehicle dynamics [15]. Linearizing the system (3.7) around $\begin{bmatrix} e_y & e_\psi \end{bmatrix}^T = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ and discretizing with sampling distance $\Delta s = vT_s$, where T_s is the sampling time, yields

$$\begin{bmatrix} e_y(t+1) \\ e_\psi(t+1) \end{bmatrix} = \begin{bmatrix} 1 & \Delta s \\ -\kappa_{ref}^2 \Delta s & 1 \end{bmatrix} \begin{bmatrix} e_y(t) \\ e_\psi(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta s \end{bmatrix} \tilde{\kappa}(t), \tag{3.8}$$

where it has been assumed that the desired vehicle curvature κ_{ref} is equal to the road curvature, i.e, $\kappa_{ref} = \kappa_s = \frac{1}{\rho_s}$, and $\tilde{\kappa} = \kappa - \kappa_{ref}$. The vehicle curvature κ is related to the steering angle as $\kappa = \frac{\tan(\delta)}{l}$ where l is the length of the vehicle.

Chapter 4

State estimation

When the states are not directly observable or when there is measurement noise present it is necessary to use an observer to get a good estimate of the state of the system. In [9], a Luenberger observer is used to estimate the state, and the dynamics of the estimation error are given by $e(t+1) = (A - LC)e(t)$. However, in this case only the measurements up to time t are used to estimate $x(t+1)$. If also the current measurement, at time $t+1$, is incorporated in the estimation, then the estimate of $x(t+1)$ is improved. In this case the error dynamics become $e(t+1) = (I - LC)Ae(t)$. Another possibility is to use a Kalman filter to estimate the state. The benefit of using a Kalman filter is that it provides an optimal estimate under certain circumstances.

4.1 The Kalman filter

Consider the system (2.13). In the case when w and v are white Gaussian noise with covariances given by Q_w and R_v, respectively, then the Kalman filter provides the optimal state estimate, in the sense that no other observer will provide a better estimate [16]. The Kalman filter consists of two main steps; the prediction step and the correction (or filtering) step. Given an initial estimate of the state $\hat{x}_{t|t}$ and the covariance $P_{t|t}$ the predicted state at time $t+1$ is

$$\hat{x}_{t+1|t} = A\hat{x}_{t|t} + Bu(t), \tag{4.1}$$

with covariance

$$P_{t+1|t} = AP_{t|t}A^T + BQ_wB^T. \tag{4.2}$$

The predicted measured state is $\hat{y}(t+1) = C\hat{x}_{t+1|t}$ and the measured state is $y(t+1) = Cx(t+1) + Dv(t+1)$. The corrected state estimate at time $t+1$ is given by

$$\hat{x}_{t+1|t+1} = A\hat{x}_{t+1|t} + L_{t+1}(\hat{y}(t+1) - y(t+1)), \tag{4.3}$$

with covariance

$$P_{t+1|t+1} = P_{t+1|t} - L_{t+1}CP_{t+1|t}, \tag{4.4}$$

where

$$L_{t+1} = P_{t+1|t}C^T(CP_{t+1|t}C^T + R_v)^{-1} \tag{4.5}$$

is the Kalman gain calculated at time $t+1$.

4.1.1 Steady-state Kalman filter

For a LTI-system, the covariances of the predicted and the corrected state will go towards stationary values and the Kalman gain will go toward an optimal stationary value. The steady-state value of the covariance can be computed by solving the discrete-time algebraic Riccati equation

$$P = APA^T + EQE^T - (APC^T)(CPC^T + R)^{-1}CPA^T, \tag{4.6}$$

and the stationary Kalman gain is given by

$$L = PC(CPC^T + R)^{-1}. \tag{4.7}$$

If we denote $\hat{x}(t+1) \coloneqq \hat{x}_{t+1|t+1}$, then, from (4.1) and (4.3) the expression

$$\hat{x}(t+1) = (I - LC)A\hat{x}(t) + (I - LC)u(t) + Ly(t+1), \tag{4.8}$$

where $y(t)$ is the measured state, can be derived. Using (4.8) and the system equations (2.13), the error dynamics of the Kalman filter are found to be

$$e(t+1) = (I - LC)Ae(t) + (I - LC)Ew(t) - LDv(t). \tag{4.9}$$

The Kalman filter is stable if all eigenvalues of $(I - LC)A$ lie inside the unit circle. In fact, the matrix $(I - LC)A$ will always be stable if the system (2.13) is controllable and observable, and the covariance matrices Q_w and R_v are positive definite [16].

Chapter 5

Robust output feedback model predictive control

The approach for robust MPC that has been considered to be the most appropriate for this project is the "Robust Output Feedback Model Predictive Control" (ROFMPC) approach described in [9]. The main reason for this is that it provides a way to explicitly deal with uncertainties at low computational cost. In addition, in [9], measurement noise is considered, as compared to the other approaches that have been investigated. The design of the robust controller in this thesis mostly follows the method outlined in [9], although some modifications have been made. The main difference is that the observer is changed so that it also uses the current time measurement.

In [9], the uncertain time invariant system (2.13), with D as the identity matrix, is considered. Here, however, we do not constrain D to be the identity. The uncertainty sets $\mathbb{W}$ and $\mathbb{V}$ are assumed to be C sets.

As described in Section 2.6 the main idea of the proposed control methodology in [9] is to use an observer to estimate the system state and then, control the observer state, rather than the actual state. The aim is to control the observer in such a way that the state constraints always are satisfied by the actual state $x = \hat{x} + \tilde{x}$, and the control constraints are satisfied by the associated control.

In order to ensure that the actual state does not violate the constraints, it is necessary that the estimation error can be bounded and shown to lie in a sufficiently small set.

In the paper, a Luenberger observer with dynamics

$$\hat{x}(t+1) = A\hat{x}(t) + Bu(t) + L(y(t) - \hat{y}(t)), \tag{5.1}$$

$$y(t) = C\hat{x}(t), \tag{5.2}$$

where L is the observer gain, is used to estimate the state. However, as discussed in Chapter 4, the estimation error can be decreased by using a observer which also incorporates the current time measurement. The dynamics of the estimation error of such an observer are given by

$$\tilde{x}(t+1) = A_L\tilde{x}(t) + (I - LC)Ew(t) - LDv(t), \quad A_L = (I - LC)A. \tag{5.3}$$

In order to bound the estimation error the observer gain has to be chosen such that $\rho(A_L) < 1$ [9].

The nominal system, where the noise and disturbance in the original system are ignored, is introduced:

$$\bar{x}(t+1) = A\bar{x}(t) + B\bar{u}(t). \tag{5.4}$$

The control error $e(t) = \hat{x}(t) - \bar{x}(t)$ satisfies

$$e(t+1) = A_K e(t) + (LCA\tilde{x}(t) + LCEw(t) + LDv(t)), \quad A_K = A + BK. \tag{5.5}$$

As in [9], K should be chosen such that $\rho(A_K) < 1$.

The actual state can be expressed as

$$x(t) = \bar{x}(t) + e(t) + \tilde{x}(t). \tag{5.6}$$

Thus, provided that the observer and the control error can be bounded and shown to lie in sufficiently "small" sets, it is possible to find an admissible control sequence $\bar{\mathbf{u}}$ and $\bar{x}$, such that the real state of the system satisfies the constraints.

5.1 Bounding the estimation and control errors

In [9], it is shown that $\tilde{x}$ and e can be bounded using robust positive invariant sets.

First, the estimation error $\tilde{x}$, equation (5.3), is considered. It may be rewritten as

$$\tilde{x}(t+1) = A_L\tilde{x}(t) + \tilde{\delta}(t), \qquad \tilde{\delta} = (I - LC)Ew(t) - LDv(t), \tag{5.7}$$

where $\tilde{\delta}$ can be considered as a disturbance that lies in the C set

$$\tilde{\Delta} = (I - LC)E\mathbb{W} \oplus (-LD\mathbb{V}). \tag{5.8}$$

Since it is assumed that L is chosen such that $\rho(A_L) < 1$, it is possible to find a robust positive invariant C set $\tilde{S}$ for the system in (5.7). It then also follows that the set $\tilde{S}$ satisfies $A_L\tilde{S} \oplus \tilde{\Delta} \subseteq \tilde{S}$. Consequently, it follows that $x(t+i) \in \hat{x}(t+i) \oplus \tilde{S}$ for all $i \in \mathbb{N}_+$ and any admissible disturbance sequence if $\tilde{x}(t) = x(t) - \hat{x}(t) \in \tilde{S}$. Thus, if the observer state can be controlled to lie in the smaller constraint set $\mathbb{X} \ominus \tilde{S}$ then the real state of the system is guaranteed to lie in the original constraint set $\mathbb{X}$.

Next, the control error e is considered. Equation (5.5) can be rewritten as

$$e(t+1) = A_K e(t) + \bar{\delta}(t), \qquad \bar{\delta}(t) = LCA\tilde{x}(t) + LCEw(t) + LDv(t), \tag{5.9}$$

where $\bar{\delta}(t)$ may be considered as a disturbance belonging to, (recall that $\tilde{x}(t)$ is bounded by the set $\tilde{S}$), the C set

$$\bar{\Delta} = LCA\tilde{S} \oplus LCE\mathbb{W} \oplus LD\mathbb{V}. \tag{5.10}$$

Since it is assumed that K is chosen such that $\rho(A_K) < 1$ it is possible to find a robust positive invariant C set $\bar{S}$ for the system (5.9). Hence, the set $\bar{S}$ satisfies $A_K\bar{S} \oplus \bar{\Delta} \subseteq \bar{S}$. As a consequence, if $e(t) = \hat{x}(t) - \bar{x}(t) \in \bar{S}$ is satisfied, then

$\hat{x}(t+i) = \bar{x}(t+i) \oplus \bar{S}$ for all $i \in \mathbb{N}_+$ and any admissible disturbance sequence. Clearly, it is then possible to ensure that $\hat{x}$ remains in the tightened constraint set $\mathbb{X} \ominus \tilde{S}$ by keeping the nominal state inside the even tighter set $\mathbb{X} \ominus (\tilde{S} \oplus \bar{S})$. Defining $S = \tilde{S} \oplus \bar{S}$, we can write the tightened constraint on the nominal state as

$$\bar{x}_{t+k} \in \mathbb{X} \ominus S \triangleq \bar{\mathbb{X}}, \quad \forall\ k = 0, ..., N. \tag{5.11}$$

An illustration of how the sets $\tilde{S}$,$\bar{S}$, and S are related is found in Figure 5.1. In Figure 5.2, the constraint sets for the nominal and the actual state are displayed, as well as the set S, which has been shifted in order to better demonstrate the fact that if the nominal state satisfies the tightened constraints, then the actual state satisfies the original constraints.

Obviously, the sets $\bar{S}$ and $\tilde{S}$ are required to satisfy $S = \tilde{S} \oplus \bar{S} \subset \mathbb{X}$ and $K\bar{S} \subset \mathbb{U}$. These requirements are fulfilled if the uncertainty sets $\mathbb{W}$ and $\mathbb{V}$ are sufficiently small, i.e., robustness cannot be guaranteed if the disturbances are too big.

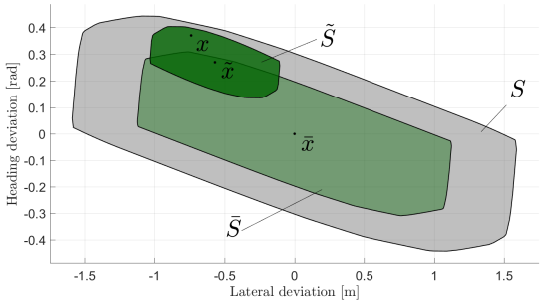

Figure 5.1: The robust invariant sets $\tilde{S}, \bar{S}$, and S, as well as one possible configuration of the actual, estimated, and nominal state. Note that the set $\tilde{S}$ is shifted by $\hat{x}$.

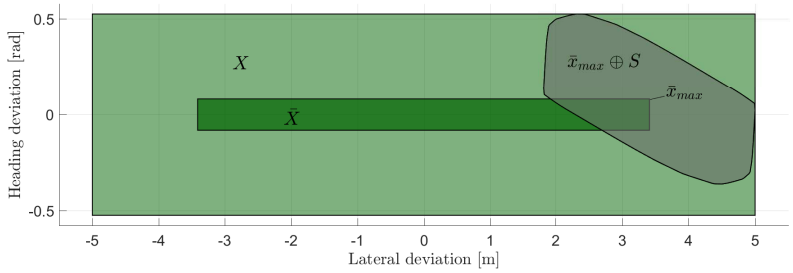

Figure 5.2: In the figure the robust invariant set S is displayed, here shifted by $\bar{x}_{max}$, being one extreme point of the nominal constraint set. From the figure it can be seen that if the nominal state satisfy the nominal constraint $\bar{x} \in \bar{\mathbb{X}}$, then also the actual state satisfies the original constraints $x \in \mathbb{X}$.

5.2 Computation of $\tilde{\mathbf{S}}$ and $\bar{\mathbf{S}}$

It is desirable to have the robust positive invariant sets $\tilde{S}$ and $\bar{S}$ as small as possible, since their sizes affect the size of nominal state and control constraint sets. To achieve this, the minimal robust positive invariant sets (mRPIs) of the systems in (5.7) and (5.9) therefore need to be computed. According to (2.12) the mRPI set $\tilde{S}_\infty$ may be computed as

$$\tilde{S}_\infty = \bigoplus_{j=0}^{\infty} A_L^j \tilde{\Delta}, \tag{5.12}$$

and, similarly, the mRPI set $\bar{S}_\infty$ may be computed as

$$\bar{S}_\infty = \bigoplus_{j=0}^{\infty} A_K^j \bar{\Delta}. \tag{5.13}$$

In general, however, it is not possible to obtain an explicit expression for the mRPI set using (2.12). Instead, an invariant outer approximation of the mRPI set can be computed, an efficient method for that is found in [8].

5.3 The nominal control

It is also necessary to ensure that the control constraint $u \in \mathbb{U}$ is satisfied. Since the control is given by $u = \bar{u} + Ke$, it is thus required that $\bar{u}$ satisfies the tightened constraint

$$\bar{u}_{t+k} \in \mathbb{U} \ominus K\bar{S} \triangleq \bar{\mathbb{U}}, \quad \forall\, k = 0, ..., N-1. \tag{5.14}$$

5.4 Terminal set constraint

A terminal state constraint is introduced for the nominal system, $\bar{x}_f \in \mathbb{X}_f$. The terminal constraint set $\mathbb{X}_f$ has to satisfy the following conditions:

1. $\mathbb{X}_f \subset \bar{\mathbb{X}}$,
2. $K_f \mathbb{X}_f \subset \bar{\mathbb{U}}$
3. $A_{K_f} \mathbb{X}_f \subset \mathbb{X}_f$,

where $A_{K_f} = A + BK_f$. It is worth noting that K_f does not have to be the same as K [9].

Clearly, the first two assumptions ensure that the tightened constraints for the nominal state and control are satisfied also in the final state. The third condition, however, is a stabilizing condition, and it guarantees that if the final nominal state x_f can be brought to the terminal set, then the nominal state will remain inside the terminal set for all future time, when subject to the state feedback control law

$$\bar{x}(t+1) = (A + BK_f)\bar{x}(t). \tag{5.15}$$

5.5 Optimal control problem

In the previous sections, tighter constraints on the nominal state and control were derived. The next step is to control the nominal state using MPC, such that the real state is ensured to satisfy the original constraints.

Thus, an optimal control problem for the nominal system is formulated. The constraints on the nominal state and control are given by (5.11) and (5.14), and the terminal constraint. The main difference, as compared to conventional MPC, is that the initial nominal state $\bar{x}_t$ is treated as a decision variable and is subject to the constraint

$$\hat{x}(t) \in \bar{x}_t \oplus \bar{S}.$$

The nominal optimal control problem that has to be solved is

$$\begin{aligned} \min_{\bar{x}_t, \mathbf{u}} \quad & J(\bar{x}_t(\hat{x}(t)), \bar{\mathbf{u}}) \\ \text{subject to} \quad & \bar{x}_{t+k+1} = A\bar{x}_{t+k} + B\bar{u}_{t+k}, \ \forall k = 0, ..., N-1 \\ & \bar{x}_{t+k} \in \bar{\mathbb{X}}, \ \forall k = 0, ..., N-1 \\ & \bar{u}_{t+k} \in \bar{\mathbb{U}}, \ \forall k = 0, ..., N-1 \\ & \hat{x}(t) \in \bar{x}_t \oplus \bar{S}, \\ & \bar{x}_{t+N} \coloneqq \bar{x}_f \in \mathbb{X}_f, \end{aligned} \tag{5.16}$$

where $\hat{x}(t)$ is the estimated state at current time t. The cost function $J(x, \bar{\mathbf{u}})$ is assumed to be quadratic as in (2.4). Since the problem is convex it can be solved efficiently using existing optimization algorithms.

5.6 The control input

By solving the nominal optimal control problem, an optimal nominal initial state $\bar{x}_t^*$ and an optimal nominal control sequence $\bar{\mathbf{u}}^*$ are obtained. Finally, the control signal to be fed to the system is computed as the sum of the first element of the optimal control sequence and the feedback control. Thus, the control u that is applied to the system is given by $u = \bar{u}_t^*(\hat{x}(t)) + K(\hat{x}(t) - \bar{x}_t^*(\hat{x}(t)))$.

5.7 Tuning parameters

There are several parameters that have to be tuned for the controller to work as good as possible. First of all the observer gain L and the feedback gain K have to be chosen such that $\rho(A_L) < 1$ and $\rho(A_K) < 1$. However, the choice of L and K will influence the size of the robust invariant sets $\tilde{S}$ and $\bar{S}$, which in turn have a direct impact on how much the original constraints need to be tightened.

Moreover, the weights of the MPC for the nominal control have to be tuned, as well as the prediction horizon N. As we will see, these parameters will affect the behaviour and the feasibility of the MPC.

Another parameter that also affects the controller is the sampling time. One thing that should be kept in mind is that, since we are considering discrete time systems, the disturbance w is to be interpreted as how much disturbances that act on the system under one sampling period. Thus, if the sampling time is decreased, then the maximal disturbance per sample that can be handled is also decreased.

Chapter 6

Autonomous vehicle path following

In order to investigate if the ROFMPC algorithm, described in Chapter 5, is suitable to use for autonomous vehicle trajectory tracking, a controller has been implemented using Matlab, together with the toolboxes Yalmip [17] and MPT3 [18]. MPT3 is mainly used for the computations of the robust invariant sets, and the nominal optimal control problem is solved using Yalmip.

The controller is designed for the road-aligned vehicle model (3.8). It is assumed that the disturbance and measurement noise are additive, as in the uncertain system (2.13). In the simulations, we have direct access to the states and, for simplicity, we therefore set C to be the the identity matrix. For simplicity we also set the matrices E and D equal to the identity. Hence, the uncertain system that we want to control is

$$\begin{aligned} \begin{bmatrix} e_y(t+1) \\ e_\psi(t+1) \end{bmatrix} &= \begin{bmatrix} 1 & \Delta s \\ -\kappa_{ref}^2 \Delta s & 1 \end{bmatrix} \begin{bmatrix} e_y(t) \\ e_\psi(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \Delta s \end{bmatrix} \tilde{\kappa}(t) + \begin{bmatrix} w_y(t) \\ w_\psi(t) \end{bmatrix}, \\ \begin{bmatrix} z_y(t) \\ z_\psi(t) \end{bmatrix} &= \begin{bmatrix} e_y(t) \\ e_\psi(t) \end{bmatrix} + \begin{bmatrix} v_y(t) \\ v_\psi(t) \end{bmatrix}, \end{aligned} \tag{6.1}$$

where z_y and z_ψ are the observed lateral and heading deviation, respectively. The disturbance is modeled to belong to the rectangular set defined by

$$\begin{bmatrix} -w_{y,max} \\ -w_{\psi,max} \end{bmatrix} \leqslant \begin{bmatrix} w_y(t) \\ w_\psi(t) \end{bmatrix} \leqslant \begin{bmatrix} w_{y,max} \\ w_{\psi,max} \end{bmatrix}, \tag{6.2}$$

where it has been assumed that the disturbances are symmetric around 0. The measurement noise is modeled to belong to a similar rectangular set, with bounds given by v_{max}, and it is assumed that also the measurement noise is symmetric around 0.

Given the uncertainty sets, the first task at hand is to compute the robust invariant sets and the tightened constraints, as described in Sections 5.1-5.2. Before that can be done, however, an observer gain L, and a feedback control gain K, have to be chosen. Details on how that is done are given in Sections 6.3.1-6.3.2. After calculating the constraint sets, $\bar{\mathbb{X}}, \bar{\mathbb{U}}$, and $\mathbb{X}_f$, it should be checked that they are non-empty. If the sets are empty, then it can be concluded

that either, L and K are badly tuned, or that the uncertainties simply are too large so that robustness cannot be guaranteed.

The performance of the controller was evaluated by simulating several different scenarios. One scenario is when the vehicle already is very close to the reference path in the initial state, and it only has to be kept there. Another scenario is when the initial position is far (> 0.5 m) from the reference path, then the controller first need to steer the vehicle towards the reference and then ensure that it remains close to the reference. In addition, it is investigated how the road curvature affects the controller.

Moreover, different sorts of uncertainties have been considered. As mentioned before, the uncertainties has to be bounded, and it is therefore not possible to ensure stability to purely Gaussian noise. Instead, in the simulation, we use "almost Gaussian noise", which is discussed in Section 6.2. Even though Gaussian noise might be considered more realistic, the disturbances and noise acting on the system will most often be quite small, as compared to the maximum admissible uncertainties. Thus, for the purpose of showing that all admissible uncertainty sequences can be handled with the ROFMPC, simulations in which the disturbance and measurement noise only are assigned extreme values are performed.

6.1 Vehicle parameters

In the simulation, the vehicle is assumed to be a medium-sized car with a turning radius of 5.5m [19], which implies that the maximal vehicle curvature is 0.18 m^{-1}. The semi-width of the road was set to 5 m in the first simulations, which arguably is a quite wide road, and therefore, the semi-width was decreased to 2.5m in some of the simulations.

6.2 Noise and disturbance modeling

As discussed before, there will always be external disturbances present in a real system, as well as measurement noise. For instance, for a road vehicle, slippage may be considered as a disturbance. There are different ways of modeling these uncertainties. One simple, but still realistic, way of modeling is to assume that the disturbance and the noise are additive. Furthermore, there will typically be unmodeled dynamics which also contributes to the overall uncertainty. In this work it will be assumed that the model uncertainties can be accounted by the additive disturbance.

As explained earlier, the noise and disturbance have to be bounded. Therefore, in order to guarantee that the uncertainties do not exceed their bounds, they cannot be modeled as purely Gaussian. Instead, the noise and disturbance are modeled as almost Gaussian, meaning that, in the simulation, the noise and disturbances is generated as zero mean Gaussian noise but if the limits are exceeded it will be set to the maximum or minimum value. In the experimental evaluation on the F1/10 platform this is of course not possible, however, if the variances of the noise and disturbance is well below the limits for what can be handled, then the event of a non-admissible disturbance or noise can be considered as most unlikely. As an example, if the standard deviation of the

measurement noise σ_v is less than one third of the maximal allowed noise, then the probability that the magnitude of the noise exceeds the maximum value is less than 0.0027.

6.3 Selection of tuning parameters

The parameters that have to be tuned are the observer and controller gains, the feedback gain K_f, the prediction horizon N, the weights Q and R, as well as the sampling distance. The tuning process is not always straight forward and some assumptions were made to make the process go faster. It is here described how the most important parameters were tuned.

6.3.1 The observer gain

Since a steady-state Kalman filter is employed the stationary Kalman gain given by (4.7) is used. The use of the stationary Kalman gain ensures that $\rho(A_K) < 1$ as required. A word of caution is needed here though. Through experience it has been noted that, depending on the relation between the disturbance and the noise, the stationary Kalman gain may sometimes render $\bar{\mathbb{U}}$ empty while other choices of the observer gain does not. It is therefore believed that in some cases, especially when the noise cannot be considered as Gaussian, it is preferable to use a standard observer.

6.3.2 The feedback control gain

In order to simplify the tuning process, only the case when $K = K_f$ is considered. Furthermore, the gain is chosen to be the optimal LQR state-feedback gain, K_∞. This is a reasonable choice since it ensures that $\rho(A_{K_\infty}) < 1$, under the conditions that the system is controllable and the weight Q is positive definite [5]. The terminal weight is also chosen to be equal to the LQR weight. That $\rho(A_K) < 1$ is however no guarantee that the problem will be feasible. The feedback gain $K = K_\infty$ can be tuned by choosing the the cost matrices Q and R in an appropriate way.

6.3.3 Sampling distance

The main requirement on the sampling distance Δs in the simulations is that it should realistic. For instance, in the simulations, a modest sampling time of 0.1 s is assumed, and the velocity is set to be 10 m/s, which gives a sampling distance of 1 m. The same sampling distance can of course be achieved for multiple combinations of sampling time and speed.

6.3.4 The prediction horizon

Since the prediction horizon does not affect the other parameters the tuning of the horizon length N is less involved. The only requirement is that the optimization problem should be feasible. The prediction horizon should therefore be chosen such that the terminal set $\mathbb{X}_f$ can be reached from any state in the nominal constraint set $\bar{\mathbb{X}}$. Thus, it becomes clear that the horizon length will be dependent on the sampling distance. If the sampling distance is short a longer

horizon will be needed, and vice versa. In short, it is desirable to keep N small for computational reasons, but on the other hand it has to be sufficiently large for the problem to be feasible.

Chapter 7

Simulation results

In all simulation results presented in this chapter, the same values on the tuning parameters, Q, R, N, have been used. The reason for this, is to facilitate comparison of the controller performance in different situations. The weights Q and R, which are used in the nominal optimal control problem, and to obtain the feedback control gain, were set to

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 20 \end{bmatrix}, R = 15,$$

meaning that deviation in the heading and control input (vehicle curvature) are penalized more than lateral deviation. Furthermore, the prediction horizon used in the simulations is set to $N = 15$, if not stated otherwise. This length of the horizon showed to render the optimization problem feasible, for all cases considered here, and is not excessively long.

In the simulations, we also decided to keep the sampling distance constant, which corresponds to driving at constant speed, assuming that the sample time is fixed. Hence, the sampling distance Δs was set to 1 m which, for instance, corresponds to a speed of 1 m/s and sampling time 0.1 s.

7.1 Straight line trajectory following

The case when the path is a straight line is a simple, yet insightful, case study. In this case, the desired vehicle curvature, κ_{ref}, is zero. The main focus was to tune the observer and feedback gains such that as large disturbances and noise as possible could be handled. The reason for this was simply that it was deemed interesting to find limits for how large uncertainties that can be handled. However, another approach, and perhaps more realistic, would be to tune the gains for optimal performance, given a specific disturbance set $\mathbb{W}$ and noise set $\mathbb{V}$.

7.1.1 Finding the maximal admissible uncertainty

The maximum disturbance and measurement noise, towards which we have been able to guarantee robustness, are found in Table 7.1.

	lateral	heading
Disturbance	0.04 m	1.1 deg
Noise	0.1 m	2.9 deg

Table 7.1: The maximum magnitude of the disturbance and measurement noise, for which we have been able to show robustness.

In Figure 7.1, the constraints on the nominal state is shown, given that the uncertainties are bounded by the maximum values in Table 7.1, and in Figure 7.2 also the constraint on the terminal state can be seen. From the figures, we can see that the set $\bar{\mathbb{X}}$ is fairly large w.r.t. the lateral deviation, which is what we would like to have, while the terminal set $\mathbb{X}_f$ is quite small. What also can be seen from the figures is that the constraint on the nominal heading angle is very tight, which indicates that the uncertainties are very close to the limit for which robustness can be guaranteed.

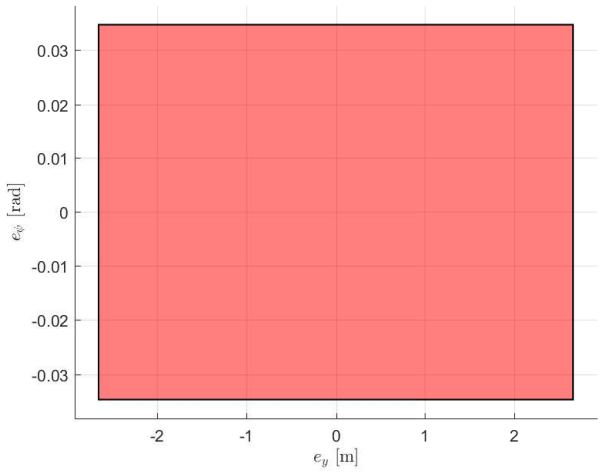

Figure 7.1: The set $\bar{\mathbb{X}}$ in which the nominal state should be contained at all times.

In Figure 7.3, the constraint set $\bar{\mathbb{U}}$ for the nominal control input is displayed. As can be seen, the constraints are much tighter ($|\bar{u}| < 0.02\text{m}^{-1}$) than the original constraint ($|\bar{u}| \leqslant 0.18\text{m}^{-1}$). This means that the contribution from the nominal control problem will be very small, even when the vehicle is far from the reference, since most of the available control input will be reserved for the feedback part. This is, however, not very surprising. Since the feedback control is used to mitigate the influence of the uncertainties in the system, it is reasonable that, when the uncertainties are large, most of the control input is used for mitigating disturbances.

In the computations of the nominal constraint sets, $\bar{\mathbb{X}}$, and $\bar{\mathbb{U}}$, it was discovered that $\bar{\mathbb{U}}$ tended to become empty before $\bar{\mathbb{X}}$, when increasing the uncertainties. It can therefore be concluded that the constraining factor, as to how big

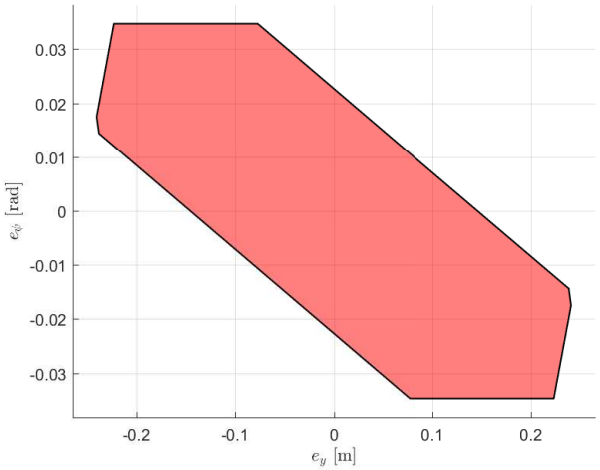

Figure 7.2: The set $\mathbb{X}_f$ in which the terminal nominal state must be contained in order to guarantee stability.

uncertainties that can be handled, are the bounds on the vehicle curvature.

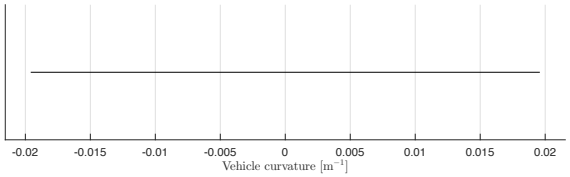

Figure 7.3: The set $\bar{\mathbb{U}}$, which the nominal control is constrained to lie in.

In Figure 7.4 the simulation result for the case when the disturbance and noise is almost Gaussian, and the vehicle is on the desired path in its initial position, is displayed. As can be seen, the fluctuates around the center-line. The maximum lateral deviation is found to be 0.15 m and the maximum deviation in the heading angle is 0.08 rad in this case. As also can be seen, the nominal state is equal to the reference through the whole simulation. The reason for this is that the first nominal state is an optimization variable. Hence, if the observed state is close enough to the reference, it is optimal to let the nominal state equal the reference. This also implies that only feedback control will be used when the deviation of the estimated state is sufficiently small.

7.1.2 Straight line trajectory tracking with smaller uncertainties

Since the previous case with uncertainties that are almost at the limit for what can be handled, a controller has also been designed for when the magnitudes of the uncertainties in the lateral position only are half as big, see Table 7.2.

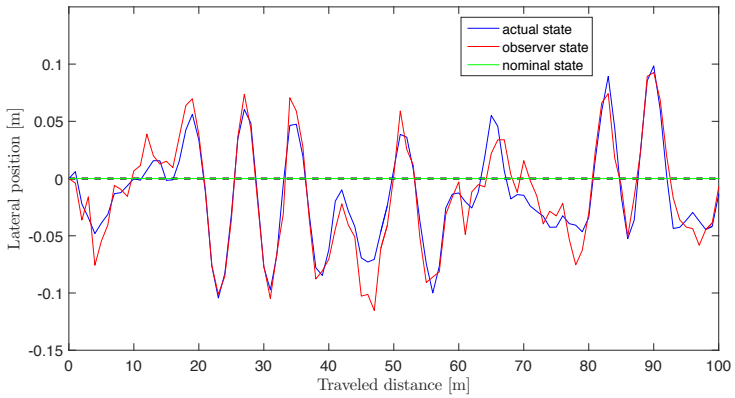

Figure 7.4: Simulation result when the uncertainties are almost Gaussian.

	lateral	heading
Disturbance	0.02 m	1.1 deg
Noise	0.05 m	2.9 deg

Table 7.2: Simulations were performed in which the maximal magnitude of the disturbance and measurement noise were given by the values stated here.

The main impact this will have on the system is that the sets $\tilde{S}$ and $\bar{S}$ shrinks, leading to less tight constraints on the nominal state and control. In Figure 7.5 the set $S = \tilde{S} \oplus \bar{S}$ for both cases are depicted.

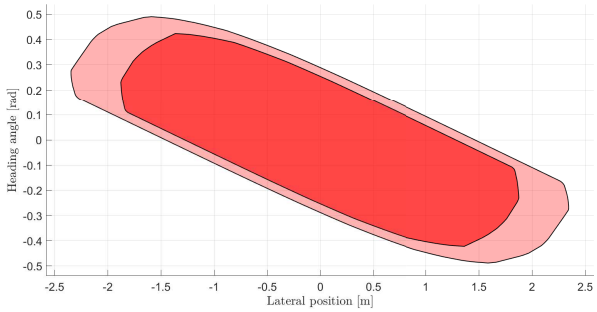

Figure 7.5: The set S, in which the difference, $x - \bar{x}$, between the actual and nominal state should be contained, for both the larger uncertainty sets (lighter red) and the smaller uncertainty sets (darker red).

Several simulations have been performed for this particular choice of uncertainty sets. First the vehicle was simulated when the lateral and heading deviation is zero in its initial position. This simulation was run with both almost Gaussian uncertainties, as well as randomly picked sequences of maximal

and minimal uncertainties. The results from these simulation are similar to the results obtained for the larger uncertainties, but the deviation from the reference path is smaller, which is what we expect to happen. When the uncertainties are Gaussian, the maximum lateral deviation and heading obtained in simulation are 0.10 m and 0.06 rad, respectively.

Simulations were also run in which the initial position of the vehicle is 3 meters away from the center-line and the heading angle is set to 0. Again, the simulation was run with both almost Gaussian uncertainties, as well as randomly picked sequences of maximal and minimal uncertainties. The results are found in Figure 7.7 and Figure 7.8. As can be seen, the controller is able to steer the vehicle close to the reference path, in approximately the same time, in both cases. As would be expected, we also find that the average deviation is considerably larger in the case when the uncertainties only takes extreme values. In Figure 7.6 both the lateral and heading deviation, for the case of almost Gaussian uncertainties, are displayed. In the figure also the nominal terminal set $\mathbb{X}_f$, as well as $\mathbb{X}_f \oplus \bar{S}$, and $\mathbb{X}_f \oplus S$ are depicted. As can be seen, the vehicle starts at $[3\ 0]^T$ and is then steered towards $[0\ 0]^T$.

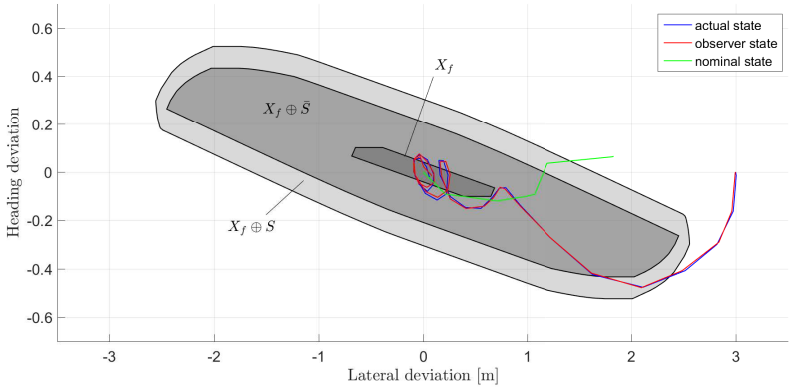

Figure 7.6: Simulation result when the uncertainties are almost Gaussian and the initial position of the vehicle is 3 m away from the center-line and the heading deviation is 0. Only the first 35 meters of the simulation are shown here.

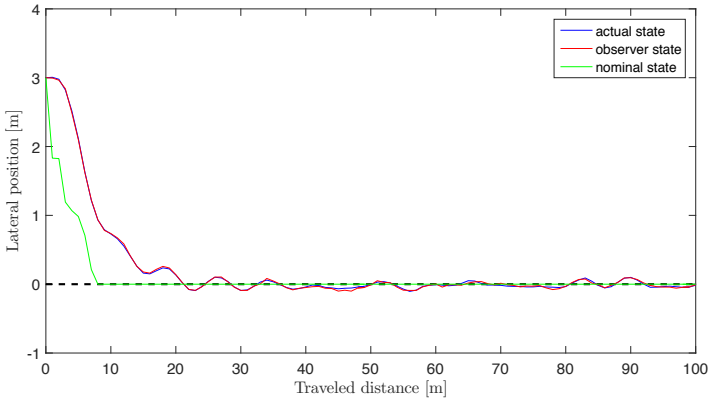

Figure 7.7: Simulation result when the vehicle start 3 meters away from the center-line and the uncertainties acting on the system are Gaussian. In the beginning of the simulation, the nominal state is different from the reference, meaning that the nominal control is non-zero.

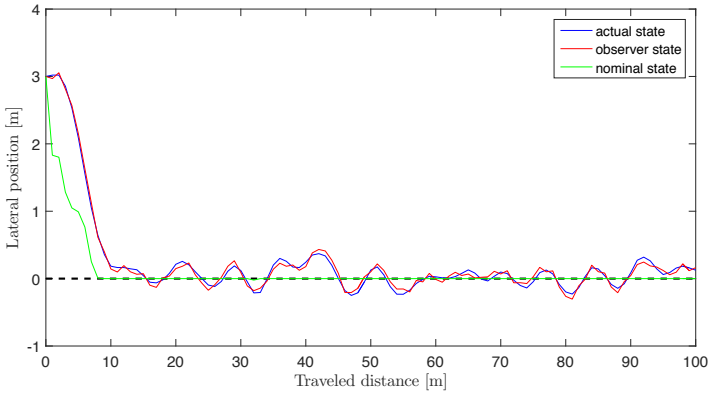

Figure 7.8: In this simulation the disturbances and noise only takes extreme values, meaning that the average uncertainty is larger, than in the case of almost Gaussian uncertainties. As a result, we see that there are larger fluctuations in the lateral position of the vehicle in this case.

7.1.3 Narrow road

The road width that has been used in the previous simulations is quite large, and in reality the vehicle might be driving on narrower roads. Therefore, we also simulate the vehicle when the semi-width of the road is 2.5 m. The only things that need to be changed in the controller in order to adapt it to the narrower road, assuming that the uncertainty bounds are as in the previous case (see Table 7.2), are the nominal state constraints and the terminal state constraint.

The new, tighter, constraints can be find in Figures 7.9. As can be seen, the nominal state constraints have been tightened so that the same safety margin is maintained. This is exactly what should be expected to happen, since the uncertainties are the same as in the previous case.

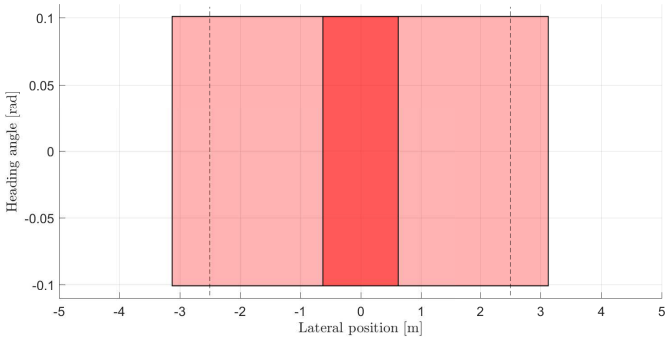

Figure 7.9: The nominal state constraints for the wide road (lighter red), and the narrower road (darker red). The width of the narrower road is indicated by the dashed lines.

7.2 Curved line trajectory following

A more general case is when the reference path is a line with a constant curvature different from zero. However, since we have modeled the system in road-aligned coordinates, the only difference in the model equations, compared to a straight line, is that the desired vehicle curvature is non-zero. The fact that the desired vehicle curvature is non-zero leads to that less control will be available for mitigating the influence of disturbances and measurement noise.

Since we require the vehicle curvature to satisfy

$$\kappa_{min} \leqslant \kappa \leqslant \kappa_{max}, \tag{7.1}$$

where κ_{min} and κ_{max}, as input to our system (6.1), we have

$$\tilde{\kappa} = \kappa - \kappa_{ref}, \tag{7.2}$$

it follows that

$$\kappa_{min} - \kappa_{ref} \leqslant \tilde{\kappa} \leqslant \kappa_{max} - \kappa_{ref}, \tag{7.3}$$

where κ_{min} and κ_{max} This is exactly the same constraint on the control input as before, but now the vehicle curvature reference is non-zero.

The constraint (7.3) on $\tilde{\kappa}$ becomes asymmetric if the uncertainty sets $\mathbb{W}$ and $\mathbb{V}$ are symmetrical. In order for the problem to make sense, we must require that

$$\kappa_{min} - \kappa_{ref} \leqslant 0 \leqslant \kappa_{max} - \kappa_{ref}. \tag{7.4}$$

Thus, for the nominal vehicle curvature, here denoted $\bar{\kappa}$, (7.3) leads to the constraint

$$(\kappa_{min} - \kappa_{ref}) \ominus K\bar{S} \leqslant \bar{\kappa} \leqslant (\kappa_{max} - \kappa_{ref}) \ominus K\bar{S}, \tag{7.5}$$

and, similarly as for the original constraint, we need to require that

$$(\kappa_{min} - \kappa_{ref}) \ominus K\bar{S} \leqslant 0 \leqslant (\kappa_{max} - \kappa_{ref}) \ominus K\bar{S}. \tag{7.6}$$

Otherwise, we cannot ensure that the nominal optimal control problem will be feasible.

Simulations have been performed in which the road curvature has been set to 0.1 m^{-1}. Using the same weights, Q and R as in the previous cases, when the road curvature was set to zero, it is found that the uncertainties have to be significantly smaller in order to guarantee robustness and stability, as should have been expected. The maximum magnitudes of the disturbance and noise for which we, in this work, have been able to show robustness, are found in Table 7.3. It may still be possible, though, to achieve robustness to uncertainties that are somewhat larger than those presented here by adjusting the weights. Nevertheless, the restriction (7.6) on the nominal control indicates that we must require the uncertainties to be smaller when driving on a curvy road, as compared to a straight road, in order to guarantee robustness. Simulation result for the case when the uncertainties are almost Gaussian can be found in Figure 7.10. In this case the vehicle starts on the center-line. The maximum lateral deviation is less than 3 cm and the maximum heading deviation is less than 1 deg.

	lateral	heading
Disturbance	0.01 m	0.6 deg
Noise	0.01 m	1.1 deg

Table 7.3: For a vehicle driving on a road with a curvature of 0.1 m^{-1}, it was found that robustness can be guaranteed towards disturbances and measurement noises with magnitudes as stated in this table.

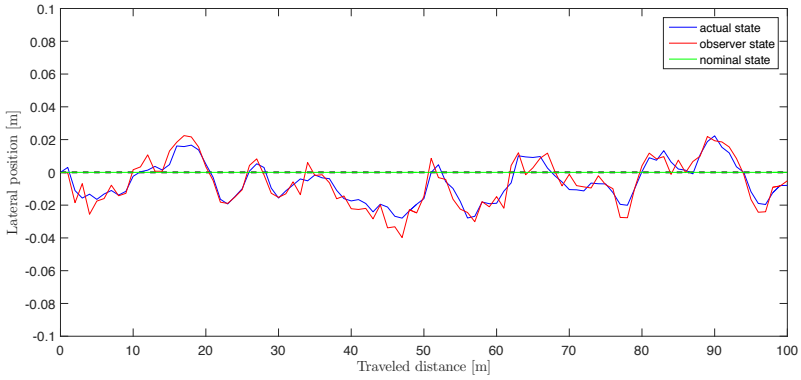

Figure 7.10: Simulation result for curved line trajectory following in the presence of almost Gaussian uncertainties. The maximum magnitude of the uncertainties are given in Table 7.3. The curvature of the road is 0.1 m^{-1}.

Chapter 8

Experimental evaluation

In order to further evaluate the performance of the controller, and to verify that it also works in practice, the ROFMPC algorithm has been implemented and tested on an F1/10 RC car.

8.1 The F1/10 RC platform

The F1/10 RC car is a small racing car in 1/10 scale. The F1/10 RC car is equipped with an inertial measurement unit (IMU) and a range-finder (LIDAR). It also has an on-board computer, Jetson [20], as well as a Teensy micro controller.

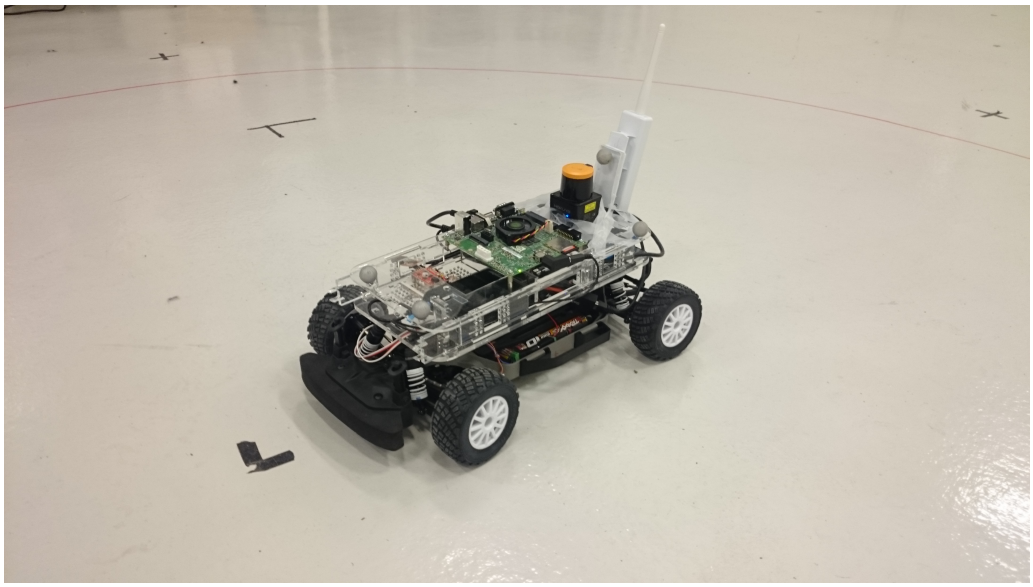

Figure 8.1: The ROFMPC algorithm was implemented and tested on the F1tenth car.

8.2 Implementation

The Robot Operating System (ROS) was used for the implementation. ROS is also supported in MATLAB through the Robotic System Toolbox, which provides an interface between MATLAB and ROS [21]. Using the Robotic System Toolbox it is possible to create ROS nodes in MATLAB and communicate over a ROS network with other nodes. Information is shared between the nodes through topics, meaning that one node can publish data to a topic and another node can subscribe to that topic.

The implementation consisted of mainly two parts, one part to be run on the car itself, and one part to be run off-board. The idea is to use data from the LIDAR to estimate the pose of the vehicle and then use the ROFMPC algorithm to obtain a steering angle to feed into the system. An overview of the ROS nodes and the flow of information are shown in Figure 8.2. The node called "LIDAR" in the figure publish data on the topic "/scan" and, therefore, a node that subscribes to the "/scan" topic needs to be created. This node uses the data from the LIDAR to calculate the lateral and heading deviation w.r.t. the center-line (see Section 8.3), which is then published on the topic "/deviation". The control algorithm is contained in the node called "ROFMPC" which is created in MATLAB. One difference from the simulations is that the nominal MPC problem is formulated as a standard QP-problem on matrix form, i.e., Yalmip is not used. The reason for this is that it makes the optimization faster. Thus, the "ROFMPC" node subscribes to the /deviation topic and uses the data from this topic to compute the steering angle, which it then publishes on the topic "/control input". On the F1tenth, there is a node that subscribes to the "/control input" topic which converts the steering angle into PWM duty cycle values and publish them to the "/drive pwm" topic. Finally, the node on the Teensy listens to this topic and generates waveforms with the specified duty cycle [22].

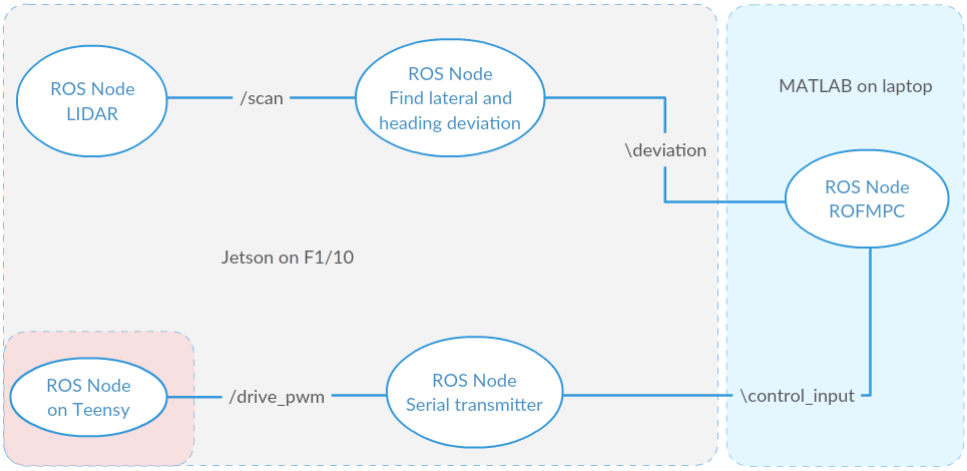

Figure 8.2: Several ROS nodes on the Jetson, and one single node in Matlab (on laptop), are created. The nodes are connected to the same local ROS network, and they can share information through topics.

8.3 Pose estimation

The lateral and heading deviation with respect to the center line were estimated using data from the LIDAR. The data from the LIDAR consists of information about how far the nearest wall, or other obstacles, are in every direction within the scanning range, see Figure 8.3.

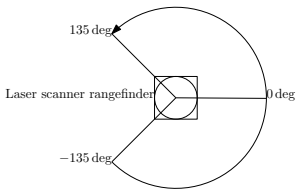

Figure 8.3: The range-finder on the F1tenth has a scanning range of 270° and the scanning rate is 40 Hz.

From Figure 8.4 the following expressions for the distance d to the wall and the heading angle α can be derived,

$$\alpha = \arctan \frac{a\cos\theta - b}{\sin\theta} - \eta, \tag{8.1}$$

$$d = b\cos(\eta - \alpha), \tag{8.2}$$

where a and b are known distances, and the angles θ and η also are known.

Since the measurements from the LIDAR are expected to be noisy, multiple pairs of beams were used to calculate the distance and the heading angle, and

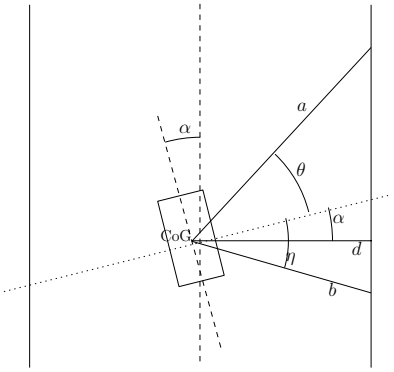

Figure 8.4: The distance d to the wall and the heading angle α w.r.t. the center-line (dashed) can be calculated using the known distances a and b and the also known angles θ and η. The dotted line is orthogonal to the yaw of the vehicle.

then the mean value is computed. Both the distance to the left and to the right wall are computed in this manner. Consequently, two values for the heading angle are obtained, and the value used for the observed heading deviation is therefore the mean of these. The width of the road is assumed to be given by the sum of the distances to the left and the right wall. The deviation from the center-line may then be computed as

$$e_{y,measured} = (d_l + d_r)/2 - d_l, \tag{8.3}$$

where d_l and d_r are the distance to the left and to the right wall, respectively.

In order to get a good estimate of the measurement noise an experiment in which the car is placed close to the middle of an approximately 3 m wide corridor has been performed. Then the lateral and heading deviation is computed using the data obtained from the LIDAR, as described previously, while the car is standing still in the same position. In Figures 8.5 and 8.6 the observed lateral and heading deviation are shown, respectively,

The standard deviation, as computed from the observation series, is found to be 0.0042 m for the lateral deviation, and 0.0087 rad for the heading deviation. From computations of the robust invariant sets it is found that uncertainties of magnitudes at least as large as stated in Table 8.1 can be handled. In Figure 8.5 and 8.6 we see that these limits occasionally are exceeded. However, since these occasions are very sparse it is believed that the impact on the system performance will be very small.

	lateral	heading
Disturbance	0.01 m	1.7 deg (0.03 rad)
Noise	0.02 m	2.9 deg (0.05 rad)

Table 8.1: Magnitudes of the uncertainties which it has been found that the controller for the F1/10 car should be able to handle.

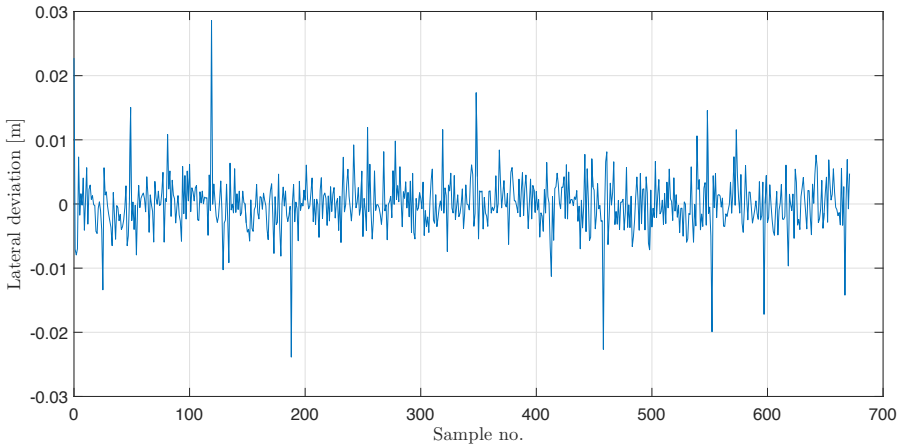

Figure 8.5: The observed lateral deviation when the F1/10 car is standing still.

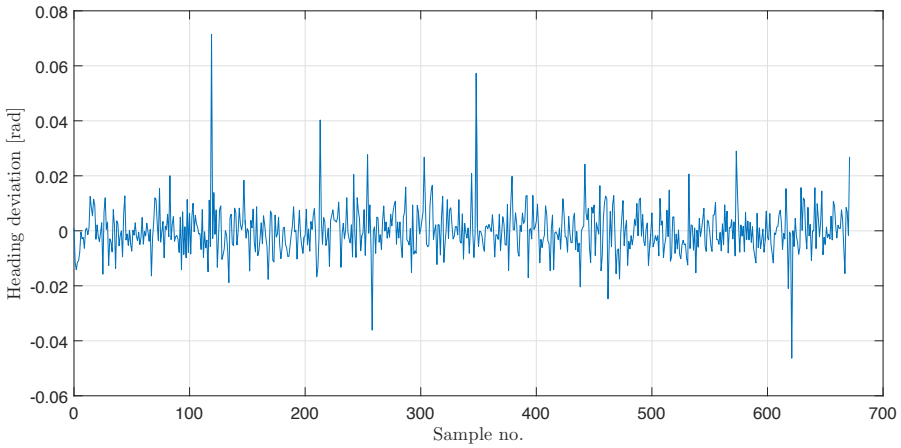

Figure 8.6: The observed heading deviation when the F1/10 car is standing still.

Chapter 9

Conclusions

The main conclusion that can be drawn from this work is that ROFMPC can be used to guarantee robustness to sufficiently small disturbances and measurement noise, of the studied system. The simulation results in Section 7.1.2 show that the controller is able to steer the vehicle close to the reference trajectory and keep it there, despite external disturbances and noise.

A drawback with ROFMPC is found to be that, if the uncertainties are allowed to be large, then the main contribution to the control signal will come from the output feedback control. This means that some of the optimality of the MPC will be sacrificed, but that is the price that has to be paid in order to ensure robustness.

In this work, we have been able to find approximate bounds for how large uncertainties it is possible to guarantee stability. Also, the experimental evaluation shows that the ROFMPC algorithm can be used in real time applications, which is important in a practical perspective.

9.1 Future work

Even though the results obtained in this work are promising and suggests that there is potential for the use of ROFMPC, it is, nevertheless, concluded that more investigation is needed, in order for ROFMPC to be considered as a reliable control methodology when it comes to autonomous vehicles. Therefore, as a first step, a more thorough study of the controller performance on curvy roads is suggested. We also believe that it would be useful to examine the impacts of using a time-varying observer or Kalman filter for the state estimation.

It would be interesting to investigate, in more detail, how the choice of tuning parameters affects the controller. In particular, it would be of great interest to learn more about how the observer and feedback control gain affects the constraints on the nominal state and control. Moreover, it is believed that it would be valuable to study if there exists a fundamental limit for how small the robust invariant sets, $\tilde{S}$ and $\bar{S}$, can be made, and how this limit can be found.If such a limit can be determined it would then be possible to analytically assess the upper bound for how large uncertainties that can handled.

Finally, a comparison between ROFMPC and standard MPC would also be of great relevance. It is especially recommended to investigate if there are

cases in which the standard MPC fails to keep the vehicle on the road while the ROFMPC succeeds.

Bibliography

[1] A. Bemporad, and M. Morari, "Robust Model Predictive Control: A Survey", In: Garulli A., Tesi A. (eds) Robustness in identification and control. Lecture Notes in Control and Information Sciences, vol 245. Springer, London, 1999. [Online]. Available: http://link.springer.com. [Accessed 28 Nov. 2016].

[2] P. J. Campo and M. Morari, "Robbust Model Predictive Control" in *Proceedings of the American Control Conference, June 10-12, 1987, Minneapolis, MN, USA* , [Online]. Available: IEEE Xplore, http://ieeexplore.ieee.org. [Accessed 10 Feb. 2017].

[3] KTH School of Electrical Engineering, "MPC allows heavy-duty vehicles to drive by themselves", 11 Feb. 2016. [Online]. Availabe: https://www.kth.se/en/ees/nyheterochpress/nyheter/mpc-allows-heavy-duty-vehicles-to-drive-by-themselves-1.625422. [Accessed 20 Feb. 2017].

[4] Scania, "Mine blowing", 19 Jan. 2016. [Online]. Available: https://www.scania.com/group/en/mine-blowing/. [Accessed 19 Feb. 2017].

[5] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems", 2015.

[6] J. B. Rawlings and D. Q. Mayne, "Model Predictive Control: Theory and design" Madison, Nob Hill Publishing, 2009. [Online]. Available: http://www.nobhillpublishing.com.

[7] F. Blanchini, "Set Invariance in Control", *Automatica*, vol. 35, pp. 1747-1767, Nov. 1999, survey paper. [Online]. Available: ScienceDirect, http://www.sciencedirect.com. [Accessed 9 Feb. 2017].

[8] S.V. Rakoviç, E.C. Kerrigan, K.I. Kouramas and D.Q. Mayne "Invariant approximations of the minimal robust positively invariant set", in *IEEE Transactions on Automatic Control* vol. 50, no. 3, pp. 406-410, March. 2005. [Online]. Available: IEEE Xplore, http://ieeexplore.ieee.org. [Accessed 1 March. 2017].

[9] D. Q. Mayne, S. V. Rakoviç, R. Findeisen and F. Allgöwer, "Robust Output Feedback Model Predictive Control of Constrained Linear Systems", in *Automatica*, vol.42, pp. 1217-1222, 2006. [Online]. Available: ScienceDirect, http://www.sciencedirect.com/. [Accessed 21 Feb. 2017].

[10] D. Q. Mayne, S. V. Rakoviç, R. Findeisen and F. Allgöwer, "Robust Output Feedback Model Predictive Control for Constrained Linear Systems Under Uncertainty Based on Feed Forward and Positive Invariant Feedback Control", in *Proceedings of the 45th IEEE Conference on Decision and Control, December 13-15, 2006, San Diego, CA, USA* [Online]. Available: IEEE Xplore, http://ieeexplore.ieee.org. [Accessed 20 Feb. 2017].

[11] D. Q. Mayne, M. M. Seron, and S. V. Rakoviç, "Robust Model Predictive Control of Constrained Linear Systems with Bounded Disturbance", *Automatica*, vol. 41, pp. 219-224, 2005. [Online]. Available: ScienceDirect, http://www.sciencedirect.com. [Accessed 27 Jan. 2017].

[12] P. O. M. Scokaert and D. Q. Mayne, Min-Max Feedback Model Predictive Control for Constrained Linear Systems *IEEE Transactions on Automatic Control,* vol. 43, no. 8, pp. 1136 - 1142, Aug. 1998. [Online]. Available: IEEE Xplore, http://ieeexplore.ieee.org. [Accessed 16 Feb. 2017]

[13] M.V. Kothare, V. Balakrishnan, and M. Morari, "Robust Constrained Model Predictive Control using Linear Matrix Inequalities", *Automatica*, vol. 32, no. 10, pp. 1361-1379, 1996. [Online]. Available: ScienceDirect, http://www.sciencedirect.com. [Accessed 27 Jan. 2017].

[14] J. Kong, M. Pfeiffer, G. Schildbach, F. Borrelli, "Kinematic and Dynamic Vehicle Models for Autonomous Driving Control Design", in *Proceedings of the IEEE Intelligent Vehicle Symposium*, pp. 1094-1099, June 2015. [Online]. Available: IEEE Xplore, http://ieeexplore.ieee.org. [Accessed 10 May 2017].

[15] Y. Gao, A. Gray, J. Frasch, T. Lin, E. Tseng, J. Hedrick, and F. Borrelli, "Spatial predictive control for agile semi-autonomous ground vehicles", in Proceedings of the International Symposium on Advanced Vehicle Control, Sep. 2012.

[16] D. Simon, "Optimal State Estimation: Kalman, H∞, and Nonlinear Approaches", John Wiley & Sons, Inc., 17 Jan. 2006. [Online]. Available: onlinelibrary.wiley.com.

[17] J. Löfberg, "YALMIP : A Toolbox for Modeling and Optimization in MATLAB", *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [https://yalmip.github.io/]

[18] M. Herceg and M. Kvasnica and C.N. Jones and M. Morari, "Multi-Parametric Toolbox 3.0", in *Proceedings of the European Control Conference*, pp. 502-510, Zürich, Switzerland, July 17-19, 2013. [http://control.ee.ethz.ch/ mpt]

[19] Trafikverket, "Vägars och gators utformning, begrepp och grundvärden" [Online]. Available: https://trafikverket.ineko.se/se/tv000237. [Accessed 11 April 2017].

[20] The F1/10 Team, "Tutorial 3: The Jetson: Introduction", [Online]. Available: http://f1tenth.org/lectures. [Accessed 21 April 2017].

[21] MathWorks, "Robot Operating System (ROS) Support from Robotics System Toolbox". [Online]. Available: https://se.mathworks.com/hardware-support/robot-operating-system.html. [Accessed 17 May 2017].

[22] The F1/10 Team, "Week 1 Lab Exercise 1: The Robot Operating System: Keyboard Control". [Online]. Available: http://f1tenth.org/lectures. [Accessed 5 May 2017].

TRITA EE 2017:090
ISSN 1653-5146