CHALMERS
UNIVERSITY OF TECHNOLOGY

Robust Control for Autonomous Truck Vehicle Combinations

Deriving a method for synthesizing controllers and providing behavioral bounds and guarantees on the resulting closed loop system

Master's thesis in Systems, Control and Mechatronics

NILS ASSARSSON
MATTIAS JUHLIN

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2020

Master's thesis

Robust Control for Autonomous Truck Vehicle Combinations

Deriving a method for synthesizing controllers and providing behavioral bounds and guarantees on the resulting closed loop system

NILS ASSARSSON
MATTIAS JUHLIN

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2020

Robust Control for Autonomous Truck Vehicle Combinations
NILS ASSARSSON
MATTIAS JUHLIN

Supervisors:
Torsten Wik, Department of Electrical Engineering, Systems and Control
Thorsten Helfrich, Volvo Group Trucks Technology
Examiner:
Torsten Wik, Department of Electrical Engineering, Systems and Control

Master's thesis in Systems, Control and Mechatronics
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone +46 (0)31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2020

Abstract

For autonomous vehicles, safety is most important. Thus, it is desired to know how the vehicle will behave given e.g. a steering input. Additionally, it is important to predict how the vehicle would behave given some perturbation. Therefore, this thesis investigates which analytical methods can be used to quantify the robustness, performance, and stability properties of the system. To ensure safety, how limits can be derived regarding disturbances or uncertainties is also considered.

To investigate these matters, linear vehicle models augmented with uncertain parameters and adapted to the behavior of a more complex nonlinear model are used as tools for analysis and controller synthesis. In both the time and the frequency domain, specifications regarding the desired performance and the system uncertainty are created. PID, LQ, and H_∞ control are then used and compared. Optimization of controllers, given system behavior objectives, is also performed.

Uncertainty analysis gives a trustworthy range of system behavior for the linear model. Adapting this model to the nonlinear model behavior results mostly in similar system responses, making the argument that analytical results can be carried over from the linear to the nonlinear model easier for these cases. Most of the optimized controllers result in the wanted system behavior. The analytical methods successfully quantified system behavior. However, some methods proved more useful than others. Uncertainty and disturbance limits are both computed through simulation, with an analytical verification of the frequency domain uncertainty.

Keywords: Robust control, Robustness analysis, Autonomous vehicle, Parametric uncertainty, Controller optimization, H-infinity control, LQI control

Acknowledgements

We want to thank our thesis supervisor and examiner Torsten Wik for knowledgeable advice and discussion regarding the project as well as experienced insights into important details.

Also, we want to thank Volvo Trucks, where our supervisor Thorsten Helfrich helped initiating the thesis work, introducing us to, and discussing, everything we needed to know throughout the progression of the thesis. He was also very meaningful in helping by consultation and with ideas of interesting topics, perspectives, and knowledge in the field.

Nils Assarsson, Mattias Juhlin
Gothenburg, June 1, 2020

Contents

List of Figures

List of Tables

List of Notations

Abbreviations

CAN	Controller Area Network
CoG	Center of Gravity
DSTM	Dynamic Single-Track Model
GoF	Gang of Four
LQI	Linear-Quadratic-Integral
LQR	Linear-Quadratic Regulator
PID	Proportional-Integral-Derivative
PSO	Particle Swarm Optimization
SLC	Single Lane Change
VTM	Volvo Truck Model

Roman Letters

S_{ab}	transfer function from signal b to signal a
v_x	longitudinal velocity [m·s^{-1}]
v_y	longitudinal velocity [m·s^{-1}]

Greek Letters

δ	steering angle [rad]
γ	controller performance level [ ]
ψ	yaw angle [rad]
Θ	road slope angle [rad]
φ	articulation angle [rad]

1
Introduction

Motion control for autonomous vehicles is concerned with controlling the driving dynamics of the whole vehicle combination, both laterally and longitudinally. It has the task to control all the available actuators in a vehicle to safely and efficiently follow the intended path along the road. One of the major challenges is ensuring and communicating a predictable behavior that other parts of the system can rely on. In short, the control system acting directly on the actuators has to be robust if the main controller acting on overall vehicle behavior is to be robust. Stability must be ensured and behavior guaranteed for all parts of the vehicle dynamics regardless of present disturbances, as safety is of the essence.

At Volvo Trucks, the control of an autonomous vehicle is divided into several levels. These are visualized in Figure 1.1 and are explained as follows: The Driver side, with information from the Cloud, makes the decision on how the Vehicle shall be driven in a specific situation by requesting an acceleration and yaw rate behavior from the truck using a positional controller acting on a reference model representing the truck. The controllers for the actuators in the truck acts on a complex dynamic system. Because of this, there is a need to translate system behavior from a complex model of the system and communicate them to the Driver, i.e. communicate limits on how the truck can be maneuvered while maintaining stability, and subsequently how well the requested behavior can be followed. Furthermore, as the real system is affected by disturbances, delays, and uncertainties, the closed-loop system has to be proven robust against these perturbations.

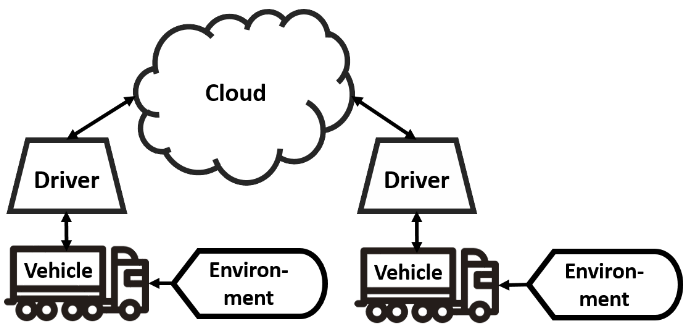


Figure 1.1: System overview

1.1 Aim

This thesis aims at investigating different system behavior metrics, comparing their qualities and developing a method for deriving bounds for parameter uncertainties and external inputs such as disturbances, noise and reference signals, to ensure robustness, stability, and performance of a truck system given a plant model, a controller and time domain specifications. Further, controllers should be synthesized with the goal of fulfilling a desired system behavior. The aim is encapsulated by defining the research questions for this thesis:

- Which analytical methods can be used to quantify robustness, performance, and stability regarding lateral and longitudinal controllers in the context of autonomous truck vehicles? How do these compare?
- How can bounds for parametric uncertainty and external disturbances be derived, to ensure safety and stability in a closed-loop consisting of a plant model and controller?

1.2 Limitations

Throughout the thesis, some things are chosen not to investigate, to allow focus on what is regarded as more important to be able to answer the research questions.

- The thesis takes into account only the vehicle and environment part of the entire system explained in the introduction and visualized in Figure 1.1, leaving out any effect the cloud and driver could have on the system behavior. Driver behavior is replaced by a general reference such as a step or a first order system response.
- Both the longitudinal and lateral behavior of the vehicle is investigated, however, these cases are simplified to a two-dimensional behavior, leaving out e.g. roll dynamics.
- The number of controller types is selectively limited to allow for more time spent on developing the analysis tools and other important parts of the thesis. Moreover, some things are assumed rather than investigated for the same reason, e.g. that the system can provide full state feedback to enable the use of LQR and LQI controllers.
- The complexity of the linear models is limited, e.g. assuming simple first-order behavior of actuators. Moreover, varying delays present in the real system are not included, dead-time is instead considered static.
- The opportunity to test any result on the real system and/or implement real system data for analysis is not possible due to time and business reasons.

1.3 Approach

Firstly a literature review is carried out, mainly in the fields of vehicle dynamics and robust control theory. This is to get an overview of earlier work and to deepen knowledge about the system at hand to find appropriate methods of controlling it. Then, linear models are derived to enable the use of linear analysis tools. These models are tuned to behave like a more complex model of the vehicle. Some candidate control theories fit for the task at hand are also chosen. From them, the most promising ones are applied to the control task by designing, implementing, and tuning them for the linear models.

With the chosen controllers implemented, their closed-loop robustness is evaluated with a focus on parametric uncertainty, disturbance, and noise. For these, metrics are defined to be able to quantify robustness, performance, and stability in time and frequency domain, such that the behavior can be communicated to the Driver, as mentioned in Section 1. Optimizing methods are used to maximize or minimize certain aforementioned measures to find what can be achieved for different controllers. To validate the method described above, optimized controllers are implemented in a more complex and supposedly more realistic nonlinear model to validate the accuracy of the linear model closed-loop behavior and the resulting metrics.

1.4 Report outline

The rest of the report is divided into the following 4 chapters,

- **Background**
  In this chapter, the background information needed to understand the origin of the problem for the thesis is presented. Furthermore, certain terminology is defined and control theory used is explained.

- **Approach**
  In this chapter, the vehicle models used in the thesis are introduced, the control problem, and how it is approached is presented. Further, the analytical methods in time and frequency domain along with the analytical metrics of interest are described. The method of considering these while optimizing a controller using different optimizing approaches is presented.

- **Results & Discussion**
  In this chapter, the methods introduced earlier are tested in several case studies. Their results are presented and analyzed. Further, the methodology of the thesis and the research questions with regards to the thesis results are discussed.

- **Conclusion & Outlook**
  The thesis is concluded by summarizing its content and drawing conclusions

from the results. Additionally, ideas for future work are proposed.

Throughout the report, note that in figures where a system response consists of two curves, they are showing the sampled step-wise minimum and maximum response range of that uncertain system. This applies in both the time and frequency domain. Note also that the nominal system refers to the system with no uncertainties, i.e. all uncertain parameters are set to their nominal value.

2

Background

Autonomous control systems rely on a controller design process that to some degree can predict the behavior of the system and synthesize a controller such that it in closed-loop is automatically regulated to a desired behavior. This design process is often based on a model of the real system. Thus, the model must exhibit similar dynamics to what is aimed to be controlled in the real system. However, this does not necessarily mean that a complex model is always required. For instance, the inherent robustness of feedback systems enables the use of simple models in control design. Sometimes, a more complex model is needed, e.g. for feedforward control design. For this purpose, a hierarchy of models is often used [1]. But for system analysis, a linear model is often preferred. It has been shown that, for vehicle applications, a linear model can capture the behavior of a more complex, non-linear model to a certain degree [2, 3], thus enabling use of linear models.

A control system is said to be robust if it is insensitive to the differences between the real-world system that it acts on and the model of that system which is used to design the controller [4]. Thus, there is a need to identify the differences and preferably, in controller synthesis, take them into account so that the control system works no matter the situation. Since no model is perfect, there will always exist a difference between its behavior and reality. To deal with that problem, one can e.g. introduce parametric uncertainty and disturbances, and then try to incorporate them with the model to bridge the gap. Control synthesis regarding parametric uncertainty is for instance considered in [5]. Since there are many aspects to take into account, a method of automatically synthesizing controllers can prove useful as was done in [6].

The control problem investigated in this thesis can be clarified by first getting an overview of the entire control system presented in Figure 2.1, previously illustrated more abstract in Figure 1.1. The *Cloud* determines the vehicle mission given data from the current vehicle, other vehicles, traffic, and other data that affects the choice of mission. A *path planning controller* then generates a path to be followed. A *path following controller* acting on a *reference model* of the vehicle generates a longitudinal acceleration and lateral yaw rate vehicle behavior. This acts as a reference behavior for the *vehicle controllers* acting on the real vehicle. Synthesizing the *vehicle controllers* to ensure that this behavior is met is the control problem considered in this thesis. More specifically, some guarantee of how well this reference can be met, given a vehicle controller, is of interest.

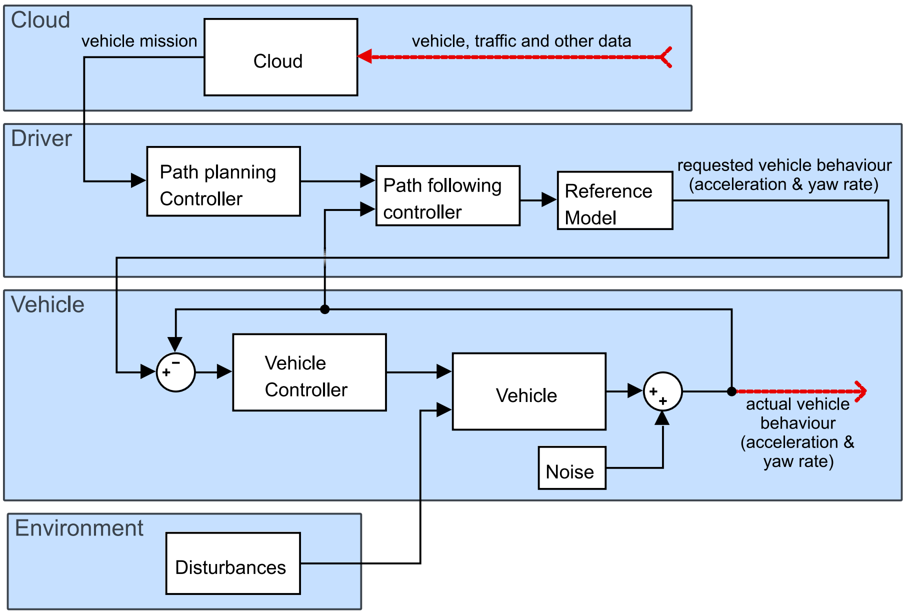


Figure 2.1: Control system overview

The problem is analyzed by considering the control system in Figure 2.2, which is the control system reduced to the *Vehicle* part only. The investigated reference is thus seen as any given reference rather than one generated by the reference model.

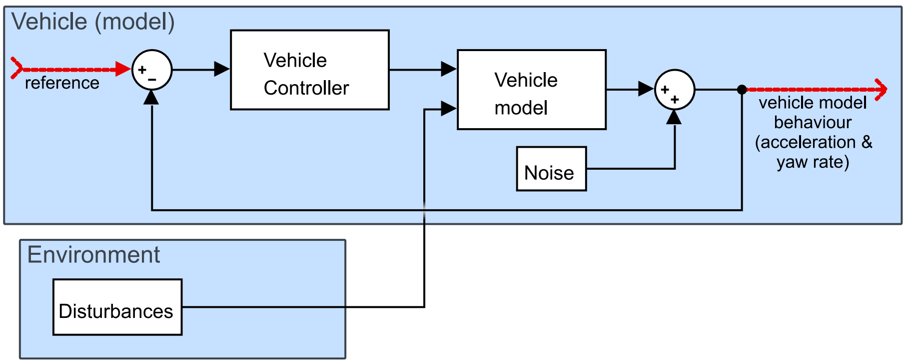


Figure 2.2: Control problem of interest in this thesis

2.1 Terminology

Part of this work covered defining exactly what type of system behavior should be guaranteed and in what sense it can be described in a coherent manner. To expand on some important terms:

Stability

The ability of a system to not diverge in its behavior. Stability needs to be guaranteed in a robust system. The dynamics of the truck are not inherently unstable since both longitudinal states (e.g. forward velocity) and lateral states (e.g. yaw-rate) converge to zero (with the exception of articulation angle, converging to any value in its range of operation). This can be achieved with states initialized at reasonable values without any external input to the system. However, if the system is equipped with a bad controller, instability (e.g. positive feedback) or marginal instability (e.g. constant oscillation in state variables) can occur.

Parametric uncertainty

Uncertainty due to that the exact value of a parameter is unknown.

Robust stability

The ability of a system to remain stable in spite of parametric uncertainty.

External input

Any type of plant input that will affect system behavior, e.g. a reference signal, disturbance, or noise.

External output

Any type of plant output, e.g. system measurements passed on to another system or an error output to be minimized.

Robust performance (robustness)

Can be defined as the system being robustly stable and all external inputs will get amplified at a gain less than unity to the external outputs. The lower the gain, the more robust the system is. Since disturbance is an external input, this term includes **disturbance rejection**.

Time domain analysis

By, for instance, simulating a sudden change (e.g. a step) in an external input and looking at how the system responds, one can judge how the system might act in that certain situation.

Frequency domain analysis

By, for instance, using a Bode plot of the system one can look at gain and phase shift at all possible frequencies of external inputs, and thus judge how the system might act in a wider range of situations.

2.2 Control theory

In this thesis, four different controllers were considered, explained briefly in this section. These are all feedback controllers, i.e. they all generate a control signal u to a system given the error e between the measured current system state y and the reference r, which is the desired system state, i.e. $e = r - y$. In general, an observer is often used to get an estimate $\hat{x}$ of the full system state x. In this thesis, however, the signal y is considered being a direct measurement of the full system state x and, thus, there is no observer.

2.2.1 PID controller

The PID controller is one of the most intuitive controller types. The control signal u is computed by the equation

$$u = Ke = (P + I\frac{1}{s} + D\frac{s}{T_d s + 1})e, \tag{2.1}$$

where P is the proportional gain, I is the integral gain, D is the derivative gain and T_d is the filter coefficient defining the low-pass filtering of e for the derivative term [1]. This filter is crucial as the derivative might otherwise be high due to high frequent measurement noise.

2.2.2 LQI controller

A Linear-Quadratic-Integral (LQI) controller is an extended form of the Linear-Quadratic-Regulator (LQR) state-feedback controller, with the latter defining the control law as

$$u = -Kx, \tag{2.2}$$

where K is the optimal gain matrix [1]. To remove steady-state errors, LQR is extended with integral action to form an LQI controller with state- and integral-feedback

$$u = -Kz, \tag{2.3}$$

where z is the state vector x extended with an integrated error state, i.e.

$$z = \begin{bmatrix} x \\ \frac{1}{s}e \end{bmatrix}. \tag{2.4}$$

The theory in the rest of this section is shared by both LQR and LQI unless otherwise stated. Given weighting matrices, Q and R, used for penalizing state and control signal offset, and the parameter γ_{lq} (introduced by using Disturbance Rejection LQ [7]), the optimal gain matrix K and the worst-case disturbance is found by solving the min-max problem

$$\min_u \max_d J(u, d) \tag{2.5}$$

where

$$J(u, d) = \int_0^\infty \Big(z(t)^T Q z(t) + u(t)^T R u(t) - \gamma_{lq}^2 d(t)^T d(t)\Big)dt \tag{2.6}$$

i.e. finding the minimizing control for the maximum disturbance d. The parameter γ_{lq} determines how much the optimal gain matrix should be modified due to the disturbance. A large value will lower the gains in K, so γ_{lq} has to be minimized if extra control signal is wanted to compensate for disturbances.

Solving the Modified Continuous time Algebraic Riccati Equation (MCARE) yields the optimal solution to the problem according to [8]. The solution fulfills the saddle point condition

$$\text{(Best case control)} \quad J(u^*, d) \leq J(u^*, d^*) \leq J(u, d^*) \quad \text{(Worst-case disturbance).} \tag{2.7}$$

The Riccati equation is

$$SA_e + A_e^T S + Q - S(B_{2e} R^{-1} B_{2e}^T - \frac{1}{\gamma_{lq}^2} B_{1e} B_{1e}^T) S = 0 \tag{2.8}$$

and from the solution, the minimizing control

$$u^*(t) = -R^{-1} B_{2e}^T S x(t) = -K x(t) \tag{2.9}$$

and the worst-case disturbance

$$d^*(t) = \frac{1}{\gamma_{lq}^2} B_{1e} S x(t) = K_d x(t) \tag{2.10}$$

is found. Note that for LQI,

$$A_e = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \quad B_{1e} = \begin{bmatrix} B_1 \\ 0 \end{bmatrix} \quad B_{2e} = \begin{bmatrix} B_2 \\ 0 \end{bmatrix} \tag{2.11}$$

where A, C, B_1, B_2 are the standard state-space matrices (B_1 - for disturbance, B_2 - for control signal). In the case of LQR, $A_e = A$, $B_{1e} = B_1$, $B_{2e} = B_2$.

To compensate for disturbances as well as possible, a minimization of γ_{lq} is performed until the Riccati equation becomes unsolvable and the lowest possible γ_{lq} can be chosen if robustness is to be prioritized. If the controller yields a too aggressive control signal, one has to limit the minimization by stopping at a higher $\gamma_{lq,\min}$.

2.2.3 H_∞ controller

The H$_\infty$ controller is synthesized in a way that minimizes the infinity-norm of the closed-loop gain from external inputs to external outputs. That means, it minimizes the maximum gain for the closed-loop over all frequencies to find the lowest overall gain γ that stabilizes the closed-loop. To do this, the plant should be included in a generalized plant, as visualized in Figure 2.3. This can be done through the method of Linear Fractional Transformation [9].

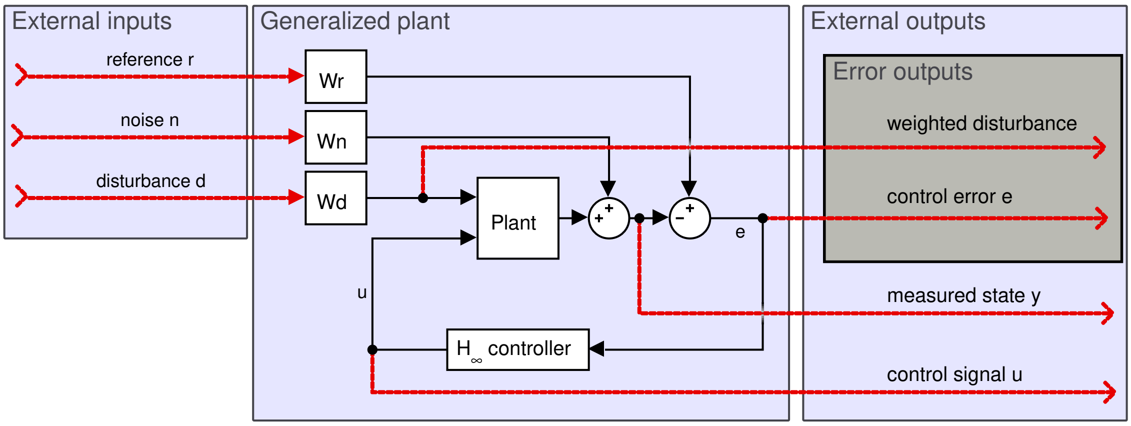


Figure 2.3: Generalized plant setup for H_∞ synthesis

The controller essentially minimizes the effect of the defined external inputs on the defined external outputs and it specifically minimizes the defined error outputs w.r.t. the infinity-norm. Without any tuning, the synthesized controller K in closed-loop will have a low γ, and can thus be compensating for the external disturbances too much, yielding an aggressive control signal. To achieve a less oscillatory behavior, one must set a target $\gamma_t > \gamma$ for the algorithm to stop at.

Since the synthesis method limits gain also on desired signals like the reference, one should introduce weighted filters on the general input signals to improve it. These filters should cover the critical frequency range for each input and raise their gain if their effect on the general outputs is more important to minimize. Thus, high gain is often used in W_d to remove disturbances quickly and low gain in W_r to still follow references.

Given the generalized plant formulation with weighting filters, one can solve Linear Matrix Inequalities to get a controller on Linear Time-Invariant (LTI) state-space form which then can be turned into a transfer function for further analysis [10].

3
Approach

In this chapter, the opening modeling section presents the linear models used and their parametric uncertainties along with how they are adapted to fit their task in this thesis. Further, the setup of the control problem is presented. The control problem section also presents details about the involved controllers and the resulting transfer functions to be analyzed.

With the models and control problem defined, the analysis method is presented. In this section, evaluation metrics for robustness, performance, and stability are defined. Further, the uncertainty and disturbance analysis methods are explained.

In the final section of this chapter, the method of automatic controller synthesis is presented. This involves optimizing the different controller types for the desired system behavior (in terms of evaluation metrics) in closed-loop with the aforementioned linear models.

3.1 Modeling

Having linear models of the truck was required to enable the use of linear analysis and controller synthesis tools. These models are presented in Section 3.1.1 and 3.1.2 for the lateral and longitudinal cases respectively. Actuator dynamics and signal delays are considered in Section 3.1.3. To emulate behavioral differences between the linear model and the more complex non-linear Volvo truck model (VTM) (and also in some sense to the real truck), the linear model was considered with some uncertainties as presented in Section 3.1.4. The vehicle parameter values used for the linear models were extracted from the VTM and are excluded due to confidentiality. To conclude, the linear models were validated in simulation against the VTM in Section 3.1.5.

3.1.1 Lateral model

Modeling the lateral behavior of the truck was done using a dynamic single track model (DSTM), also known as a bicycle model. The concept of single track vehicle modeling is common for lateral modeling and control design [1, 11]. The model used here was derived in [12] using equations of motion, linearized for constant longitudinal velocity, and assuming small articulation and tire slip angles. Two models were used, one for the tractor only and one for the tractor with a semi-trailer. These are presented below, starting with the tractor-only model.

Tractor

The DSTM considering the tractor is visualized in Figure 3.1 with its state space formulation presented in Eq. 3.1. The parameters used are presented in Table 3.1. The states of the model are the lateral velocity v_y and the yaw rate $\dot{\psi}$. The model takes as input the steering angle of the front wheels, δ.

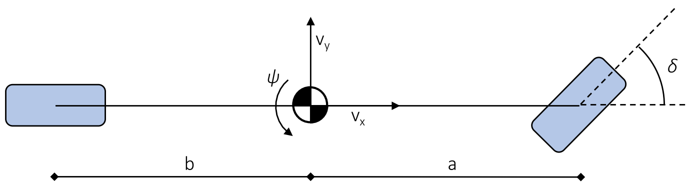


Figure 3.1: Single track tractor model representation

Table 3.1: Lateral tractor model parameters

Mass	m_1	[kg]
Wheelbase	l_1	[m]
CoG distance to front axle	a_1	[m]
CoG distance to rear axle	b_1	[m]
Yaw of inertia	$I_{zz,1}$	[kg·m^2]
Cornering stiffness front axle	$C_{\alpha,F}$	[N·rad^{-1}]
Cornering stiffness rear axle	$C_{\alpha,R}$	[N·rad^{-1}]
Rear axle to kingpin distance	h_1	[m]

$$\begin{bmatrix} \dot{v}_y \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -\frac{C_{\alpha,F}+C_{\alpha,R}}{m_1 v_x} & \frac{b_1 C_{\alpha,R}-a_1 C_{\alpha,F}}{m_1 v_x} - v_x \\ \frac{b_1 C_{\alpha,R}-a_1 C_{\alpha,F}}{I_{zz,1} v_x} & -\frac{a_1^2 C_{\alpha,F}-b_1^2 C_{\alpha,R}}{I_{zz,1} v_x} \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha,F}}{m_1} \\ \frac{a_1 C_{\alpha,F}}{I_{zz,1}} \end{bmatrix} \delta \tag{3.1}$$

Tractor with semi-trailer

The DSTM considering the tractor with an attached trailer is visualized in Figure 3.2 with its state space formulation presented in Eq. 3.3. To make the state space equations easier to read a few simplifications were made, as presented in Eq. 3.2. The parameters used are found in Table 3.2. The states of the model are the lateral velocity of the truck v_y, the yaw rate of the truck $\dot{\psi}$, the articulation angle rate $\dot{\varphi}$, and the articulation angle φ. The model takes as input the steering angle of the front tractor wheels, δ.

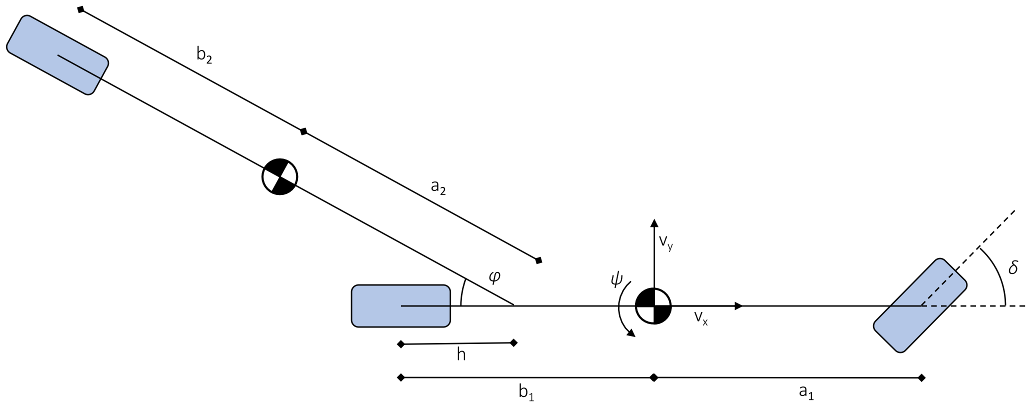


Figure 3.2: Single track tractor with trailer model representation

Table 3.2: Lateral tractor with semi-trailer model parameters

Tractor mass	m_1	[kg]
Tractor wheelbase	l_1	[m]
Tractor rear axle to kingpin distance	h_1	[m]
Tractor CoG distance to front axle	a_1	[m]
Tractor CoG distance to rear axle	b_1	[m]
Tractor yaw of inertia	$I_{zz,1}$	[kg·m^2]
Tractor cornering stiffness front axle	$C_{\alpha,F}$	[N·rad^{-1}]
Tractor cornering stiffness rear axle	$C_{\alpha,R}$	[N·rad^{-1}]
Trailer mass	m_2	[kg]
Trailer rear axle to kingpin distance	l_2	[m]
Trailer CoG distance to kingpin	a_2	[m]
Trailer CoG distance to rear axle	b_2	[m]
Trailer yaw of inertia	$I_{zz,2}$	[kg·m^2]
Trailer cornering stiffness rear axle	$C_{\alpha,T}$	[N·rad^{-1}]

$$C_1 = C_{\alpha,F} + C_{\alpha,R} \tag{3.2a}$$

$$Cs_1 = a_1 C_{\alpha,F} - b_1 C_{\alpha,R} \tag{3.2b}$$

$$Cq_1^2 = a_1^2 C_{\alpha,F} + b_1^2 C_{\alpha,R} \tag{3.2c}$$

$$\begin{bmatrix} m_1+m_2 & -m_2(h_1+a_2) & -m_2a_2 & 0 \\ -m_2h_1 & I_{1,zz}+m_2h_1(h_1+a_2) & m_2h_1a_2 & 0 \\ -m_2a_2 & I_{2,zz}+m_2a_2(h_1+a_2) & I_{2,zz}+m_2a_2^2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v}_y \\ \ddot{\psi} \\ \ddot{\varphi} \\ \dot{\varphi} \end{bmatrix} = -\frac{1}{v_x} \times$$

$$\times \begin{bmatrix} C_1+C_{\alpha,T} & C_1s_1-C_{\alpha,T}(h_1+l_2)+(m_1+m_2)v_x^2 & -C_{\alpha,T}l_2 & -C_{\alpha,T}v_x \\ C_1s_1-C_{\alpha,T}h_1 & C_1q_1^2+C_{\alpha,T}h_1(h_1+l_1)-m_2h_1v_x^2 & C_{\alpha,T}h_1l_2 & C_{\alpha,T}h_1v_x \\ -C_{\alpha,T}l_2 & C_{\alpha,T}l_2(h_1+l_2)-m_2a_2v_x^2 & C_{\alpha,T}l_2^2 & C_{\alpha,T}l_2v_x \\ 0 & 0 & -v_x & 0 \end{bmatrix} \times \tag{3.3}$$

$$\times \begin{bmatrix} v_y \\ \dot{\psi} \\ \dot{\varphi} \\ \varphi \end{bmatrix} + \begin{bmatrix} C_{\alpha,F} \\ a_1C_{\alpha,T} \\ 0 \\ 0 \end{bmatrix} \delta$$

3.1.2 Longitudinal model

Modeling the longitudinal behavior of the truck was done using the force balance equation

$$m\ddot{x} = F_T - F_B - F_{vdrag} - F_{wdrag} - F_{slope} - F_{road} \tag{3.4}$$

where F_T is the engine torque force, F_B the braking torque force, F_{vdrag} the vehicle aerodynamic drag force, F_{wdrag} the opposing wind aerodynamic drag force, F_{slope} the road slope force and F_{road} the road/rolling resistance force. These are visualized in Figure 3.3 and defined as

$$F_T = c_{pt}\frac{T_T}{r_w} \tag{3.5a}$$

$$F_B = \frac{T_B}{r_w} \tag{3.5b}$$

$$F_{drag} = \frac{1}{2}\rho c_d A v_x^2 \tag{3.5c}$$

$$F_{slope} = mg\sin\theta \tag{3.5d}$$

$$F_{road} = mgc_r\cos\theta \tag{3.5e}$$

where T_T is the traction torque, T_B is the brake torque and θ is the road slope angle.

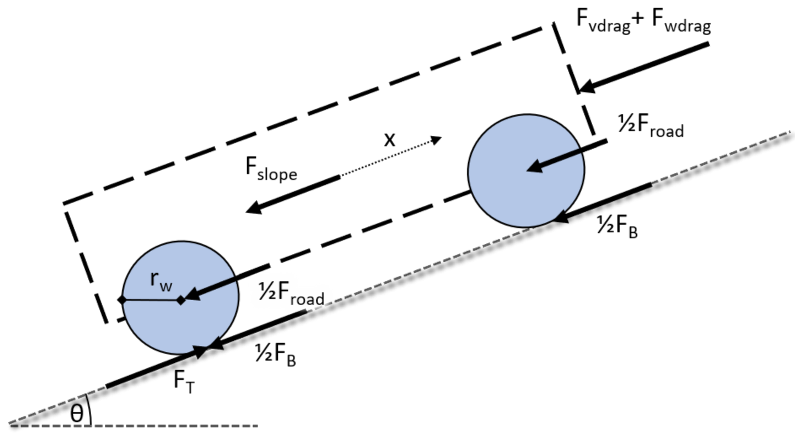


Figure 3.3: Longitudinal force balance model

using the parameters defined in Table 3.3. The powertrain coefficient was added to cover non-modeled powertrain factors like e.g. gearing and was quantified to better match the longitudinal behavior of the VTM.

Table 3.3: Longitudinal model parameters

Mass (tractor)	m	[kg]
Wheel radius	r_w	[m]
Air density	ρ	[kg/m^3]
Drag coefficient	c_d	[ ]
Tractor front area	A	[m^2]
Rolling resistance coefficient	c_r	[ ]
Powertrain coefficient	c_{pt}	[ ]

This model was linearized for small angles θ, around a constant vehicle velocity $\dot{x}_0$ and a constant headwind velocity v_{w0} such that the state-space model became

$$\begin{bmatrix}\ddot{x}\end{bmatrix} = -\frac{1}{m}F_{vdrag}\begin{bmatrix}\dot{x}\end{bmatrix} + \frac{1}{m}\begin{bmatrix}-mg & -mgc_r & -\frac{1}{2}\rho c_d A v_{w0} & c_{pt}/r_w & -1/r_w\end{bmatrix}\begin{bmatrix}\theta \\ 1-\theta \\ v_{wind} \\ T_T \\ T_B\end{bmatrix} \quad (3.6)$$

where the state $\dot{x}$ is the longitudinal velocity of the truck.
As the model is quite simple in design, adding a semi-trailer to it was just a matter of adding its mass such that $m = m_1 + m_2$ given the masses from Tables 3.1 and 3.2.

3.1.3 Actuator modeling and signal delay

Altering the steering angle of the front wheels cannot be done instantly, no matter i it is done by a human or through an automated system, due to inertia. Likewise, the engine and driveline torque is not instant. Thus, these actuator behaviors must be modeled. For this thesis, it was assumed that steering actuation behaved like a first order system with a rise time assumed to be 0.3 seconds, with the rise time defined as the time it takes for a system step response to go from 10% to 90% of its final value. Included in the steering actuator rise time was also the force build-up time in the tire contact patch occurring when changing the steering angle. Moreover, there was a maximum steering angle $\delta_{max} = 45°$ and steering angle rate $\dot{\delta}_{max} = 22.5°/\text{s}$ representing physical limitations on the vehicle in reality. These input limits should not be exceeded while simulating the lateral models.

Similarly, for the longitudinal model, the powertrain and brake torque actuation model was assumed to be first order systems rise times of 0.75 and 0.25 seconds respectively. The longitudinal models also have input limitations, in the form of maximum total wheel torques for acceleration and deceleration. These were approximated given a maximum vehicle acceleration $a_{max} = 3$ m/s^2 and deceleration $d_{max} = 8$ m/s^2 possible for the tractor at low velocities, such that

$$T_{Tmax} = a_{max} r_w m = 10440 \text{ [Nm]} \tag{3.7a}$$

$$T_{Bmax} = d_{max} r_w m = 27840 \text{ [Nm]} \tag{3.7b}$$

Running the longitudinal model with a trailer decreased the approximated acceleration such that $a_{max} = 2$ m/s^2.

Furthermore, there is a signal delay due to the controller area network (CAN) bus used for signal transmissions between the onboard micro-controllers. In reality, this introduces a varying dead time delay depending on the traffic and priority settings on the bus. For this thesis, it was assumed to be a constant 0.05 seconds.

3.1.4 Model adaption and uncertainties

The parameters set for the models include some uncertainties, i.e. intervals in which each parameter varies. This was necessary as some parameters tend to change during a run or between runs and cannot be accurately measured. As an example, the cornering stiffness coefficient of the wheels changes depending on both tire-specific dynamics such as tire pressure and environmental factors such as temperature. Likewise, the rolling resistance coefficient can change 20% depending only on how worn the tires are [13]. This coefficient depends also on additional environmental factors. Including uncertain parameters ranges in a model result in a range of model responses given a certain input.

The uncertainty range for each type of parameter regarding the lateral and longitudinal models were assumed to be the values presented in Table 3.4.

Table 3.4: Linear model parameter uncertainty ranges

Mass	± 5%
Cornering stiffness	± 20%
Yaw inertia	± 5%
CoG position	± 0.1m
Air density	± 10%
Drag coefficient	± 5%
Rolling resistance coefficient	± 30%

Uncertainties can also be included in models to create a requested model response range. In this case, it was desired that the model response should exhibit similar behavior to the VTM. As such, some uncertain parameters were added and quantified to meet this criteria. These parameters and their respective uncertainty ranges are presented in Table 3.5.

Table 3.5: Model adaptation parameters

Powertrain coefficient	c_{pt}	0.8	± 20%
Brake system coefficient	c_{bs}	0.95	± 20%
Lateral tractor model state uncertainty	$u_{dLat,1}$	1	± 10%
Lateral tractor semi-trailer model state uncertainty	$u_{dLat,2}$	1	± 30%

While these parameter additions did add some accuracy to the model behavior compared to the VTM, the model outputs were found through open-loop simulation to still be offset in comparison. To counteract this, the state space outputs were manually scaled according to

$$y_{trac} = u_{dlat,1} \begin{bmatrix} 0.65 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi} \end{bmatrix} \tag{3.8}$$

for the lateral tractor model and

$$y_{st} = u_{dlat,2} \begin{bmatrix} 1.2 & 0 & 0 & 0 \\ 0 & 1.2 & 0 & 0 \\ 0 & 0 & 1.1 & 0 \\ 0 & 0 & 0 & 1.1 \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi} \\ \dot{\varphi} \\ \varphi \end{bmatrix} \tag{3.9}$$

for the lateral semi-trailer model.

The actuator delays mentioned in 3.1.3 have an uncertainty aspect as well. The powertrain actuation varies with the current engine and turbo rpm, selected gear, etc. For these reasons, the delays were given the uncertainties presented in Table 3.6.

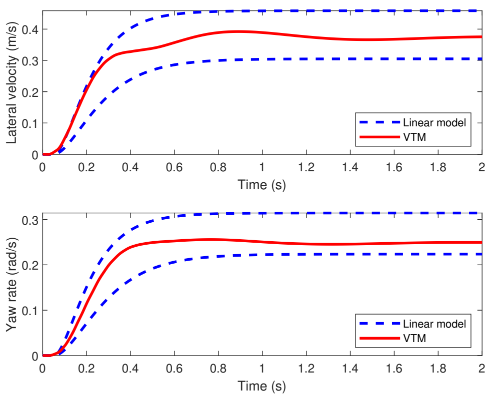
Table 3.6: Actuator rise time uncertainty

Steering actuation	0.3s	± 10%
Powertrain torque actuation	0.75s	± 50%
Brake torque actuation	0.25s	± 10%

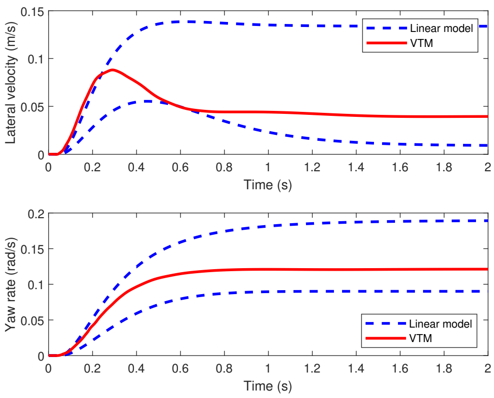
3.1.5 Model validation

The models presented in this section were to be used for controller synthesis and subsequent analysis of the resulting closed-loop system, to be able to conclude how a more complex model or the real system would behave. Thus, the models needed to have similar behavior. To validate this, the lateral and longitudinal models were compared to the VTM by simulating them in open-loop with the same constant input. The simulated steering and torque input values presented in each figure caption were chosen not too aggressively as the models were derived and linearized for small angles.

The lateral tractor model was simulated at velocities $v_x = 5$m/s and $v_x = 10$m/s, visualized in Figure 3.4. The linear model response bounds represent the maximum and minimum responses given any uncertain parameter combination. As the figure shows, the range of the responses covers the VTM behavior. At higher velocity, the uncertain model response gave a larger interval as the model dynamics, and subsequently its uncertainties, were scaled by the longitudinal velocity v_x. Moreover, the lateral velocity state of the VTM in Figure 3.4b seemed to involve more aggressive dynamics than the linear model dynamics. The assumption that the two models have similar behavior, given the uncertainty range simulation and comparison, is not completely straightforward. Hence, this is further discussed in Section 4.



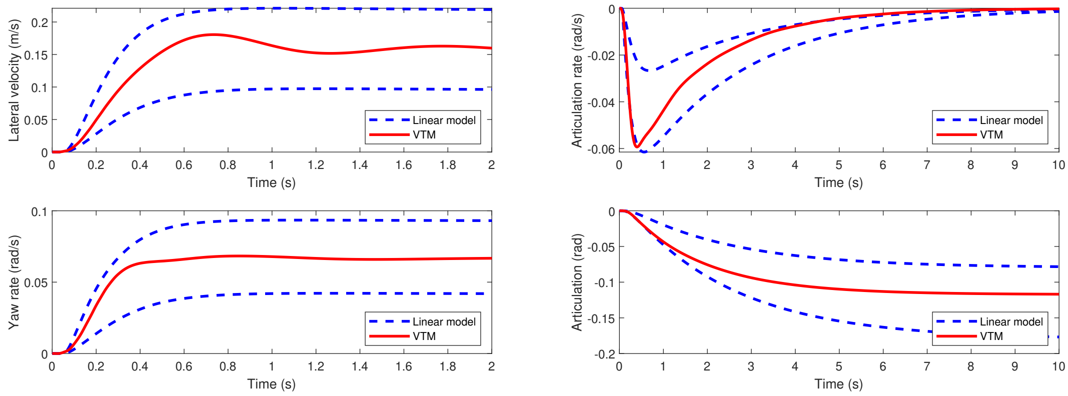
(a) Validation at $v_x = 5$m/s with a step steering input of 0.2 rad



(b) Validation at $v_x = 10$m/s with a step steering input of 0.05 rad

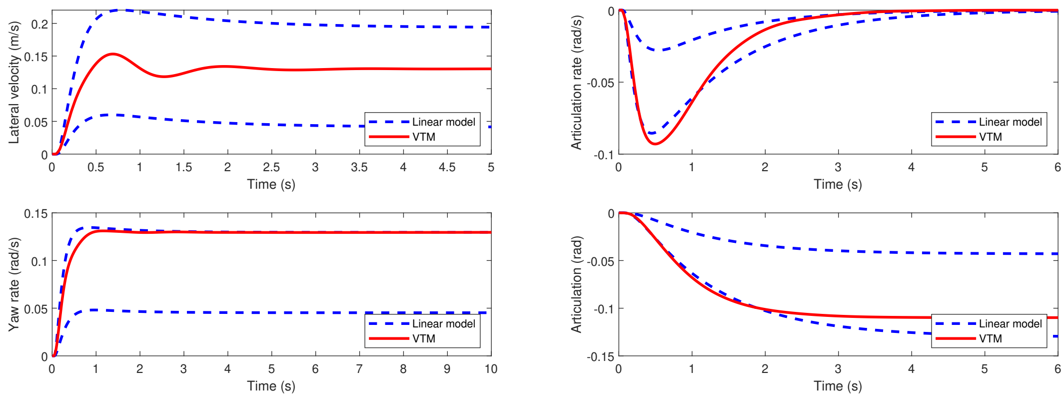
Figure 3.4: Validation of the tractor lateral model using the VTM at two different velocities

In Figures 3.5 and 3.6 the lateral tractor with semi-trailer was simulated for $v_x = 5$ m/s and $v_x = 10$ m/s respectively. In Figure 3.5 the linear model responses covers the VTM simulation quite well, although the articulation rate is slightly outside of the bounds. For the higher longitudinal velocity in Figure 3.6, it does not perform as well. The articulation angle and rate behavior are not fully covered.



(a) **(b)**

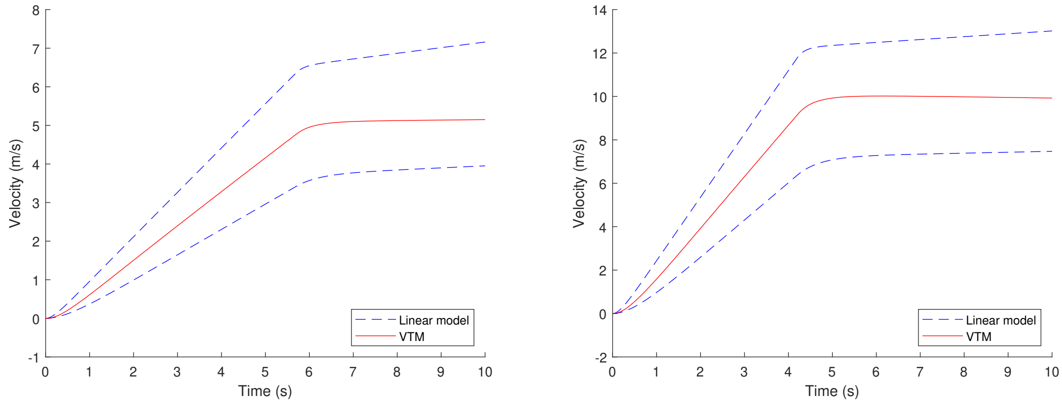
Figure 3.5: Validation of the tractor with semi-trailer lateral model using the VTM at $v_x = 5$m/s with a step steering input of 0.2 rad



(a) **(b)**

Figure 3.6: Validation of the tractor with semi-trailer lateral model using the VTM at $v_x = 10$m/s with a step steering input of 0.05 rad

The result from validating the velocity behavior of the longitudinal tractor model against the VTM is presented in Figure 3.7. This was done for two cases. For the first case, the model accelerated from stationary to 5 m/s with a low torque input. In the second case, the model accelerated from stationary to 10 m/s with a higher torque input. In both simulations, the models were subjected to an initial step input which was then decreased to end up in a steady state. Note that the linear model velocity does not keep increasing much further than the figure shows, however this was excluded to more clearly present the behavior of the first 10 seconds. It was clear from both simulations that the linear model behavior encapsulated the VTM behavior but had a slightly higher steady-state velocity given the same input torque, as the linear model velocity keeps increasing slightly while the VTM velocity did not.

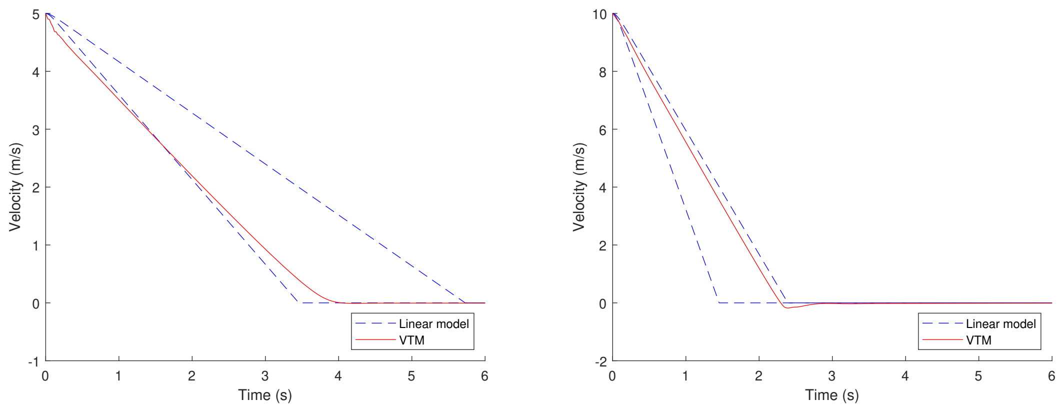


(a) Validation with an initial step torque input of 4000 Nm changing to 600 Nm at $t = 5.8$ s

(b) Validation with an initial step torque input of 10000 Nm changing to 600 Nm at $t = 4.2$

Figure 3.7: Velocity behavior validation of the tractor longitudinal model using the VTM

The braking behavior simulations for the tractor are presented in Figure 3.8. In these simulations, the tractor started at different initial states and was decelerated to stationary given different brake input torque. In the first case, Figure 3.8a, the VTM had a quicker response to the brake input than the linear model. In the second case, Figure 3.8b, the VTM response was better encapsulated within the possible DSTM response. However, the VTM displayed a momentary reversing behavior after braking to 0 m/s.



(a) Brake from 5 m/s with a step brake torque input of 4000 Nm

(b) Brake from 10 m/s with a step brake torque input of 20000 Nm

Figure 3.8: Deceleration behavior validation of the tractor longitudinal model using the VTM

Simulating the longitudinal tractor with trailer models for acceleration and braking situations resulted in very similar results to the tractor only simulation. Thus, figures displaying these were considered redundant and are not presented here.

3.2 Control problem

The control problem was to assure that certain state variables of the plant model should follow a reference under the influence of process uncertainty, disturbance, delays, and noise. The closed-loop system should also be stable. The controller types introduced in Section 2.2 are the ones chosen suitable for the task in this thesis.

The PID controller does not provide any inherent robustness guarantees but was used to enable quicker development through faster script runtime as well as being an intuitive comparison to the other controller types. The LQI has a robustness guarantee that is

$$||S_{yd}||_\infty \leq \gamma_{lq}, \tag{3.10}$$

i.e. the peak gain from disturbance to the measured state is upper bounded by γ_{lq}. The H_∞ controller has one that is

$$||N||_\infty \leq \gamma \tag{3.11}$$

where N is the generalized plant introduced in Section 2.2.3. In that way, the peak gain of all external inputs to external outputs are guaranteed to be below γ.

3.2.1 General system

To perform analysis given any kind of plant model or any of the controller types specified earlier, a general system architecture was designed which can be seen in Figure 3.9.

In this figure, the superscript $\sim$ denotes signals before being delayed. The system has three external inputs, disturbance, noise, and reference. The full state $y = \begin{bmatrix} y_{nc} & y_c \end{bmatrix}$ (where y_{nc} are the non-controlled system variables, y_c is the controlled system variable) are assumed measurable and can thus be seen as an external output of the system.

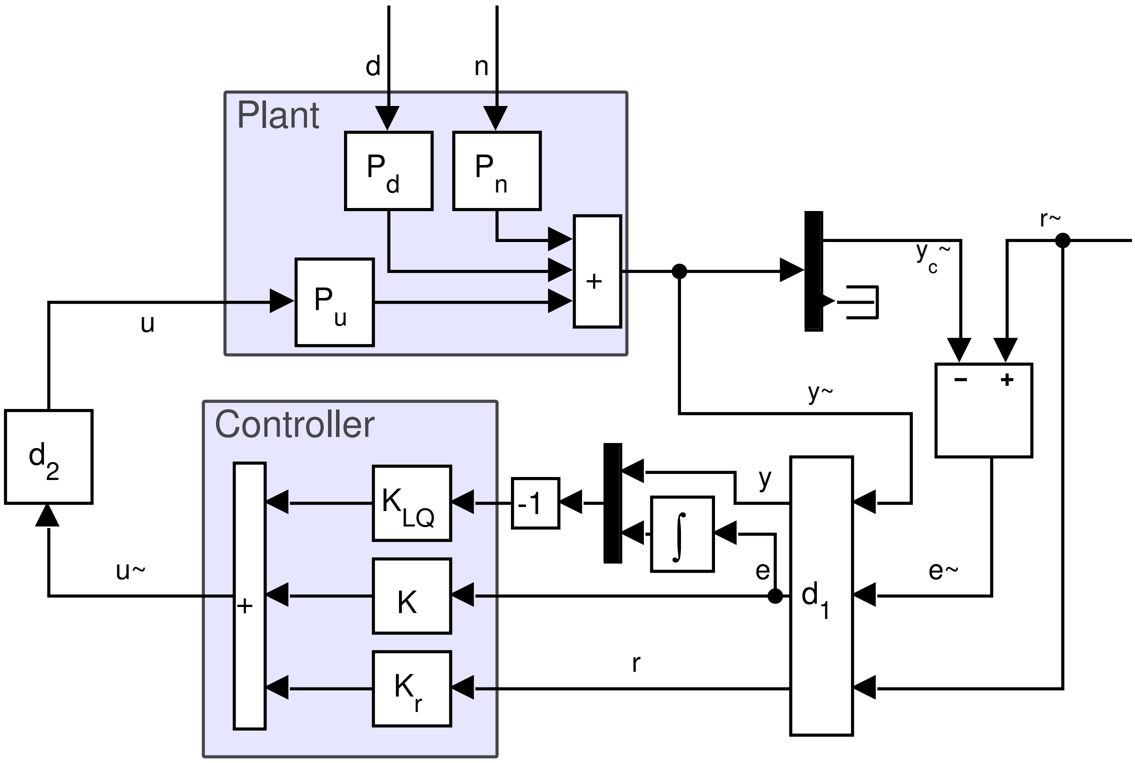


Figure 3.9: General closed-loop system

By using this architecture, one can simply change any part of the system when needed (e.g. controller K, feed-forward term K_r, delay d_1 or transfer function from $u \to y$, P_u), and if some parts are not needed, they can be set to zero or one depending on whether it is a gain (K/P), or a delay (d). Using this architecture, a PID or H$_\infty$ controller can be implemented through K and an LQI through K_{LQ}. If needed, a feed-forward term can be implemented through K_r.

For the case of PID or H$_\infty$ control, any controlled system variable y_c, its respective control signal u, and any non-controlled system variable y_{nc}, have the following closed-loop equations

$$y_c = \frac{\hat{P}_n}{I + D\hat{P}_u K} n + \frac{\hat{P}_d}{I + D\hat{P}_u K} d + \frac{D\hat{P}_u (K + K_r)}{I + D\hat{P}_u K} r \tag{3.12}$$

$$u = -\frac{D\hat{P}_n K}{I + D\hat{P}_u K} n - \frac{D\hat{P}_d K}{I + D\hat{P}_u K} d + \frac{D(K + K_r)}{I + D\hat{P}_u K} r \tag{3.13}$$

$$y_{nc} = \tilde{P}_n n + \tilde{P}_d d + D\tilde{P}_u (K + K_r) r - D\tilde{P}_u K y_c \tag{3.14}$$

$$D = d_1 d_2 \tag{3.15}$$

where d_1 is the CAN delay and d_2 is the actuator delay, D is the combined delay, $\hat{P}$ means transfer from (d/n/u) to the controlled system variable, $\tilde{P}$ to the non-controlled system variables.

If one were to use pure dead time ($e^{-\tau s}$) as CAN delay then most of the transfer functions would not be rational and thus complicate analysis so a comparison was done between Padé approximations of different orders and pure dead time. It can be seen in Figure 3.10.

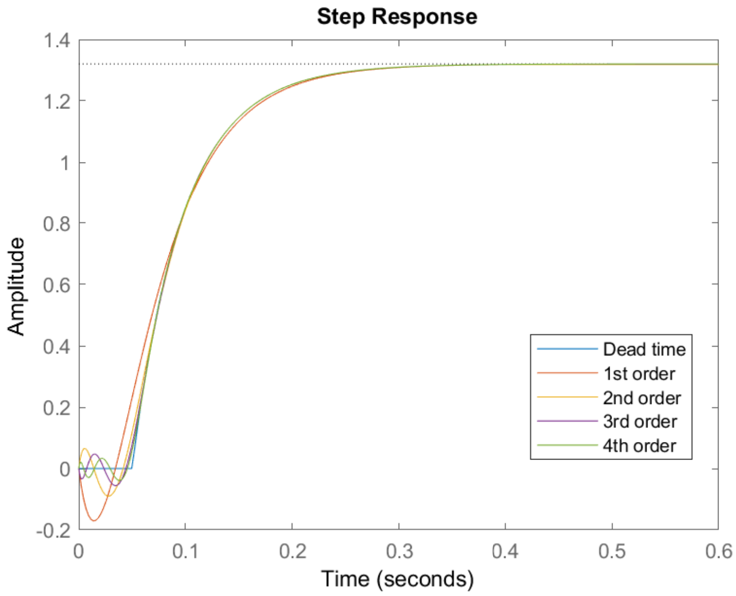


Figure 3.10: Padé approximation of different orders compared to pure dead time.

The approximation of second order was deemed good enough without introducing too many higher-order terms. Thus the CAN delay was given by

$$d_1 = e^{-\tau s} \approx \frac{\frac{\tau^2}{12}s^2 - \frac{\tau}{2}s + 1}{\frac{\tau^2}{12}s^2 + \frac{\tau}{2}s + 1}, \tag{3.16}$$

where τ is the dead time in seconds.

When using LQI control the closed-loop equations change slightly due to full state feedback. The controlled system variable equation used was

$$y_c = \frac{\hat{P}_n}{I + D\hat{P}_u(K_{LQ_y} - K_{LQ_i}\frac{1}{s})}n + \frac{\hat{P}_d}{I + D\hat{P}_u(K_{LQ_y} - K_{LQ_i}\frac{1}{s})}d \; + \tag{3.17}$$

$$+\frac{D\hat{P}_u(K_r - K_{LQ_i}\frac{1}{s})}{I + D\hat{P}_u(K_{LQ_y} - K_{LQ_i}\frac{1}{s})}r - \frac{D\tilde{P}_u K_{LQ_y}}{I + D\hat{P}_u(K_{LQ_y} - K_{LQ_i}\frac{1}{s})}y_{nc} \quad , \tag{3.18}$$

the control signal equation used was

$$u = \frac{(K_{LQ_i}\frac{1}{s} - K_{LQ_y})\hat{P}_n - K_{LQ_y}\tilde{P}_n}{I + D(K_{LQ_y}\tilde{P}_u - (K_{LQ_i}\frac{1}{s} - K_{LQ_y})\hat{P}_u)}n + \tag{3.19}$$

$$+\frac{(K_{LQ_i}\frac{1}{s} - K_{LQ_y})\hat{P}_d - K_{LQ_y}\tilde{P}_d}{I + D(K_{LQ_y}\tilde{P}_u - (K_{LQ_i}\frac{1}{s} - K_{LQ_y})\hat{P}_u)}d \; + \tag{3.20}$$

$$+\frac{D(K_r - K_{LQ_i}\frac{1}{s})}{I + D(K_{LQ_y}\tilde{P}_u - (K_{LQ_i}\frac{1}{s} - K_{LQ_y})\hat{P}_u)}r \quad , \tag{3.21}$$

and for the non-controlled system variables

$$
\begin{aligned}
y_{nc} = {} & \frac{\tilde{P}_n}{I + D\tilde{P}_u K_{LQ_y}} n + \frac{\tilde{P}_d}{I + D\tilde{P}_u K_{LQ_y}} d + \\
+ & \frac{D\tilde{P}_u (K_r - K_{LQ_i} \frac{1}{s})}{I + D\tilde{P}_u K_{LQ_y}} r + \frac{D\hat{P}_u (K_{LQ_i} \frac{1}{s} - K_{LQ_y})}{I + D\tilde{P}_u K_{LQ_y}} y_c \quad ,
\end{aligned}
\tag{3.22}
$$

where K_{LQ_y} is the LQ gain of the system state and K_{LQ_i} is the LQ gain of the integral state. In Figure 3.9, $K_{LQ} = \begin{bmatrix} K_{LQ_y} & K_{LQ_i} \end{bmatrix}$.

3.2.2 Controller parametric design

The PID controllers were designed in the way, described in Section 2.2.1, with the filter coefficient being set between 10^{-3} and 10^3, so a PID controller used four parameters in total.

For LQI control, the first parameter used was the cost for the controlled state variable offset, the second was the cost for integral error offset, the third was the cost for control signal offset and finally the fourth was used to decide whether or not to enable feedforward control. Thus, the fourth parameter was defined in the range $p_4 \in [0, 1]$, where $p_4 > 0.5$ enables feedforward control. As for LQR control, only the cost for controlled state variable offset and the cost for control signal offset was used since feedforward needs to be activated, making it two parameters in total. All costs for non controlled state variable offset were set to zero.

There is some freedom in designing an H_∞ controller. The design that was used in this work uses three weighting filters (also termed preshaping filters); a lowpass-filter for disturbances, a highpass-filter for measurement noise, and a bandstop-filter for reference signals. The use of the filters is shown in Section 2.2.3. A typical setup of filters can be seen in Figure 3.11.

Since the effect of disturbances and noise are of the highest importance to remove, their gains are higher than the reference filter. By using an LP- and HP-filter with cutoff frequencies far away from each other, the controller could more easily discern and remove each perturbation. By using a bandstop-filter for references, the frequency range of normal operation could be defined, and thus references were not dampened but instead followed. This setup led to an H_∞ controller with 7 parameters in total, consisting of gain and cutoff frequency for each filter, where the bandstop filter had two cutoff frequencies.

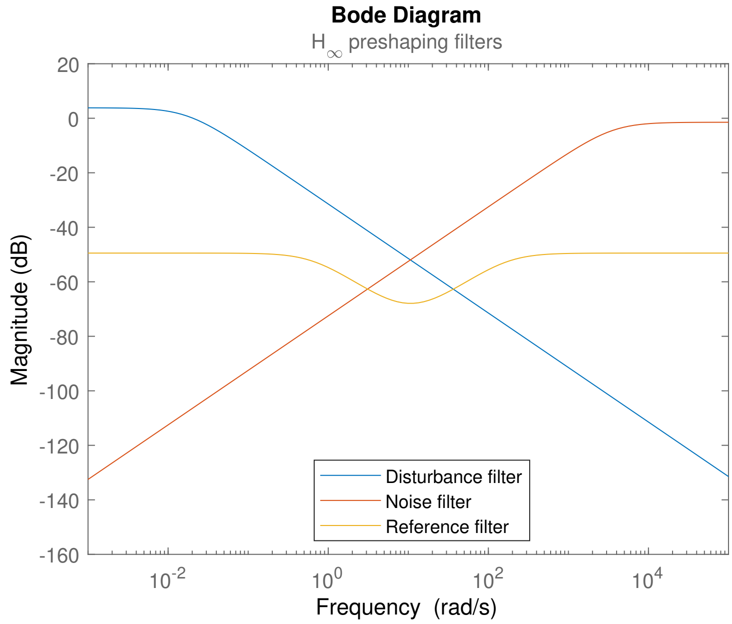


Figure 3.11: H_∞ controller: design of typical preshaping filters

3.3 Analytical methods

To perform analysis of a plant in closed-loop with a controller, analytical methods were investigated and chosen. The metrics (marked in *italic*) gained by the analysis are further on used to evaluate closed-loop systems and to optimize controllers. They are described in this section and visualized in Appendix A.

3.3.1 Stability analysis

To ensure a stable system, if the open-loop transfer function is stable, one can use the simplified Nyquist criterion by looking at the Nyquist plot of the open-loop transfer function ($L = PC$, where P is the plant and C the controller). If it crosses the negative real axis to the right of the critical point (-1, j0), the system is stable, and the greater the minimal distance to this point is, the more stable the system is [1]. This is due to that the closed-loop system will have $(1 + L)$ in its transfer function denominator and if L would be close or equal to -1, the system gain will go towards infinity. Thus, the metric *minimal Nyquist distance* was used as a measure of stability.

A generalized stability margin, used with uncertain systems in robust control is that of

$$\|H(P,C)\|_\infty = \frac{1}{\delta_v(P, -1/C)} \tag{3.23}$$

where $H(P,C)$ is the Gang of Four (GoF) matrix, consisting of the four transfer functions from the general inputs disturbance and noise to the controlled output variable and the control signal, i.e. S_{yd}, S_{yn}, S_{ud}, and S_{un}. The term $\delta_v(P, -1/C)$ is the Vinnicombe distance between P and $-1/C$ [14]. What follows from this margin is that a sample P^* from an uncertain system P is stable if

$$\delta_v(P_0, P^*) < \frac{1}{\|H(P_0, C)\|_\infty} \tag{3.24}$$

where P_0 is the nominal system. Thus, the metric for *robust stability* used in optimization was to maximize the right side of the inequality, i.e. lowering the effects of disturbance and noise onto the system output and the control signal [1].

3.3.2 Performance & Robustness analysis

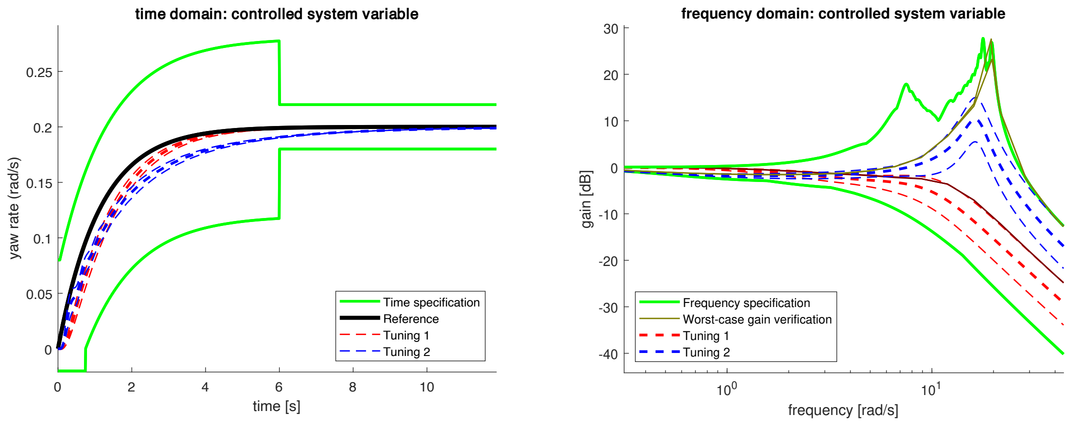
In a sense, the performance and robustness of the system are intertwined. A system with high robustness, i.e. with low system gains, will usually take a long time to reach a reference. A system that has high gains will react strongly to reference change, i.e. possibly have good performance, but might also react unwanted to perturbations such as sensor noise or external disturbances. Thus, they are described in the same section.

The performance and robustness of the system were analyzed in both time and frequency domain: in the time domain by simulating the system response to a reference, and in the frequency domain by calculating the Bode plot, which corresponds to the gain and phase curve of the transfer function. Both a time and frequency specification was created. The time specification should define acceptable or safe system behavior. The frequency specification, explained in more detail later on, is created from the time specification and thus, they are coupled. System responses going outside specification in either domain could then be regarded as having bad performance and/or robustness.

To analyze the system in the two aforementioned domains, meaningful metrics had to be chosen and defined. In the time domain, the important metrics from simulation were deemed to be the following. The *reference oscillation* of the system was calculated as the sum of the derivatives of the controlled state. A low value was often seen in robust systems. *Reference reaction time* was taken as the time from a given reference change until the reference has been achieved within 1%. The shorter the time, the better the performance. The *reference overshoot* was calculated as the highest point of the controlled system variable response divided by the reference amplitude. A low value was often preferred but if reaction time was important, then that may introduce more overshoot. Lastly, the *Non-controlled system variable response* was simulated and the point of interest was to see if it could be minimized by the controller without feedback of the signal.

In the frequency domain, *reference performance* was calculated as the mean distance between the frequency response of the system and the upper limit of the frequency specification. *Reference robustness* was calculated as the mean distance between the frequency response of the system and the lower limit of the frequency specification. Thus, these two metrics indicate high and low gain systems respectively and optimizing by them was shown to be more beneficial than using time domain metrics.

The time domain performance metric were defined as the mean distance between the system response and the reference and as robustness metric to the lower limit of the time specification. The robustness metric difference can be seen in Figure 3.12, where both solutions were tuned to find the most robust solution with Tuning 1 optimized in the frequency domain and Tuning 2 in the time domain. A slower time response did not correspond to a low gain system.



(a) Time domain **(b)** Frequency domain

Figure 3.12: Frequency domain metric compared to time domain metric

The *bandwidth* of the system is defined as the frequency where the frequency response falls below -3dB. Thus, frequencies below the bandwidth transfer most of the energy of the signal. This can be used e.g. for S_{yr}, where a high bandwidth might be wanted, to be able to also follow high frequency reference signals, improving performance. Since there are delays in the system, the *phase shift* was calculated and the integral of it over the relevant frequency range was used as a metric to minimize. Less phase shift yields a more robust system and allows for higher performance w.r.t. the specification since more time was allowed for disturbance rejection and higher gains can be used without violating the specification.

3.3.3 Disturbance analysis

To avoid disturbances affecting the system beyond acceptable behavior, one has to look into how they influence it. In the time domain, disturbances were analyzed by applying a step disturbance and evaluating the controlled system variable response. There was also a time domain disturbance specification created to define the maximum level of impact that noise and disturbance can have on the system. The following metrics for disturbance analysis were defined. *Disturbance reaction time* was defined from the time a disturbance affects the system until it has been rejected to 10% of its maximal amplitude. A low value was desired but does not take into account that the disturbance might still be of large amplitude. Thus, *disturbance robustness* was calculated as the mean distance between the system response of the controlled system variable to a disturbance and its undisturbed level. A low value means that the system was barely affected by the disturbance.

In the frequency domain, looking at the Bode plot of the system, one can look at the *critical gain* at the *critical frequency*, and thus decide at which input frequency the system was excited the most. If disturbances are known to exist within a certain frequency range, finding systems where the critical frequency was outside of this range, was then considered positive. If disturbances have large magnitude, the critical gain needs to be low so that the system remains safe.

3.3.4 Uncertainty analysis

A system with an uncertain parameter gives a range of responses. To find this range, one would have to simulate the system for a fine resolution of the uncertain parameter. This would take a large amount of time and was not convenient. The models presented in Section 3.1 involves multiple uncertain parameters. Each added uncertain parameter then increase the simulation time exponentially. To find the response range of such a system, a method which gave a good approximation in a short enough runtime had to be derived. The three different methods, presented in Figure 3.13, were investigated. For this exemplifying figure, nine samples of two parameters are done.

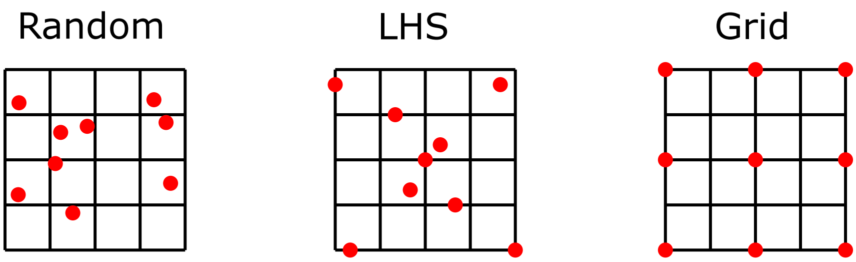
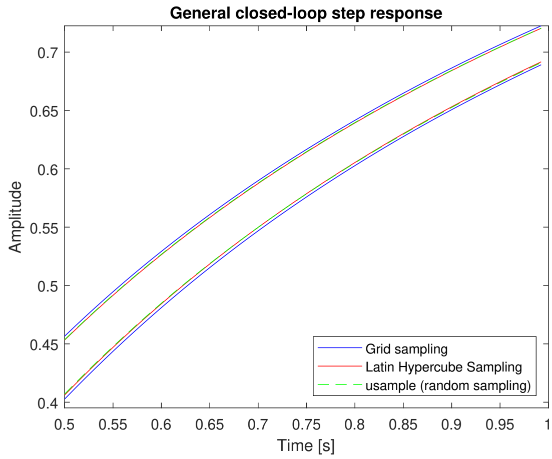


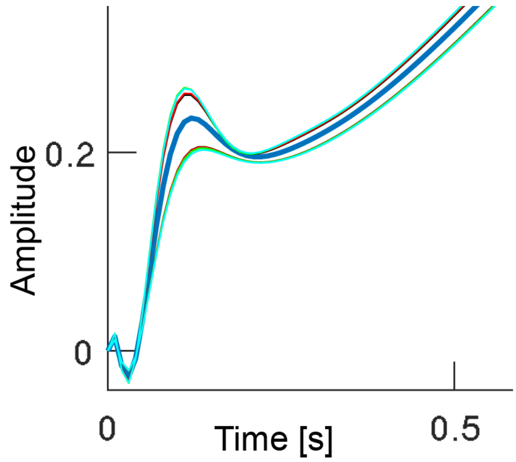
Figure 3.13: Different sampling methods tested for uncertainty sampling

Random sampling will sample fully randomly and might thus lead to an unevenly spaced distribution which will not be a close approximation of the response extremes, i.e. the actual response bounds. A semi-random approach is Latin Hypercube Sampling (LHS)[15]. It will make the distribution a bit more even by allowing only a certain number of samples in each dimension. A third approach tested was grid sampling which is done by evenly spacing the samples across the sampling space.

In Figure 3.14a, a comparison between different uncertainty sampling methods was performed for an arbitrary system. A total of 3125 samples were taken (5 different values for 5 parameters). LHS only differed around 0.01% from the random sampling, thus they yield approximately the same solution. However, the Grid sampling differed 0.5% from the random sampling. The random sampling took 28 seconds while Grid and LHS took 44 seconds. Since the time difference was acceptable, Grid sampling was thus chosen as the method for system uncertainty approximation as it yielded the highest variance of system responses.



(a) Comparison of sampling methods for uncertain parameters.



(b) Comparison of long-run random sampling

Figure 3.14: Uncertainty sampling method and runtime comparison

In Figure 3.14b, a comparison between running MATLAB function `usample` (random sampling) for a different amount of samples were performed. The runtime for each test is shown in Table 3.7, and by looking at the figure it was decided that a low number of samples (~ 100) was good as a quick approximation of uncertainties and that a longer run at around 10^4 samples does not differ much from the longest run, and was thus a good amount to use as a final sampling result.

Table 3.7: Uncertainty sampling test runtime

Color	**Nr. of samples**	**Runtime** [s]
Black	10^2	10^0
Red	10^3	10^1
Green	10^4	10^2
Cyan	10^6	10^4

A typical result of how sampled uncertain system responses can look like what is shown in Figure 3.15.

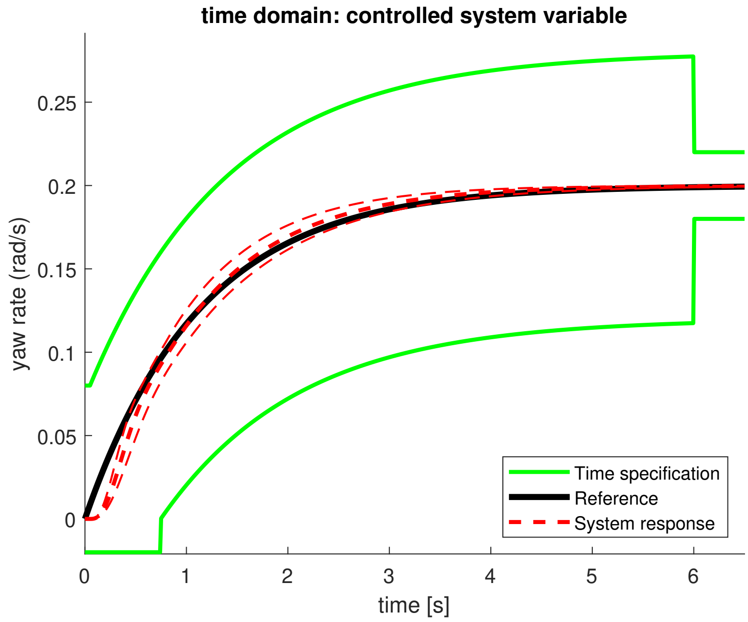


Figure 3.15: Example of uncertainty sampling

The curve in the center of the system response is the nominal system's response and the outer curves are the step-wise minimal and maximal sampled uncertain system responses.

Lastly, two metrics are measuring the sum of percentual uncertainty relative to the nominal system for reference following as well as disturbance rejection. For reference following, it is calculated in both time and frequency domain, and for disturbance rejection only in the time domain. They are denoted as *reference uncertainty* and *disturbance uncertainty* and are aimed to be minimized.

3.4 Method

A method was created that should perform the analysis of a controller-plant combination and evaluate the behavior of the general system introduced in Section 3.2.1. A flowchart of its process can be seen in Figure 3.16. There are two use cases marked by the gray boxes.

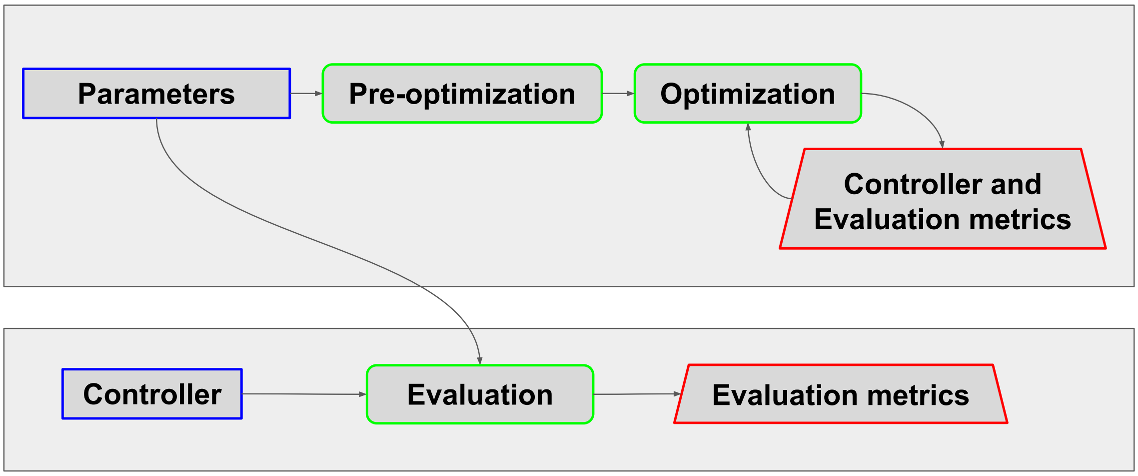


Figure 3.16: Method overview

For the bottom case, given a controller, the method can be used to evaluate its metrics in closed-loop with the plant and determine system behavior. For the top case, given the desired controller type, the method can be used to synthesize a controller according to the parametric design described in Section 3.2.2, and optimize it towards fulfilling a certain specification or wanted behavior.

The top case will be explained more in detail, as the bottom case can be considered a sub-case of it. In Section 3.4.1 the inputs to the method are defined in **Parameters**, in Section 3.4.2 the **Pre-optimization** step is explained in more detail and in Section 3.4.3 the process of the **Optimization** step is described.

3.4.1 Parameters

The inputs to the method were defined in **Parameters**. These were

- Parameters
  - Truck parameters (with uncertainties)
  - Delays, disturbance, noise
  - Control signal limitations
  - Frequency ranges of interest
  - Controller parameter range

- Plant model (linear state space)
    - Uncertain plant with disturbances, noise.
- Controller type (from the ones mentioned in Section 2.2)
- Time domain specification
- Reference signal
- Plot & optimization options

where the delays; signal dead time and actuator delay, are defined as in Section 3.1.3. The disturbance was defined as the amplitude of a disturbance step and the noise was defined by frequency and amplitude. Control signal limitations was defined by maximum and minimum input to the system and also the maximal input rate. The frequency range of interest was defined for each general input to the system (disturbance, noise, and reference) and was used to perform certain frequency analyses in that range instead of the full range.

The controller parameter range defines the lower and upper limit of each controller parameter for use in the optimizer. Time domain specification and reference signal were generated from the metrics rise time, steady-state time, transient region error, and steady-state error based on a generic first-order response but can also be chosen manually. To compensate for the Padé approximation introduced in Section 3.2.1, the specification allows for early negative responses. Typical model inputs used can be found in Section 3.1. In Figure 3.17 an example of a time domain specification is shown.

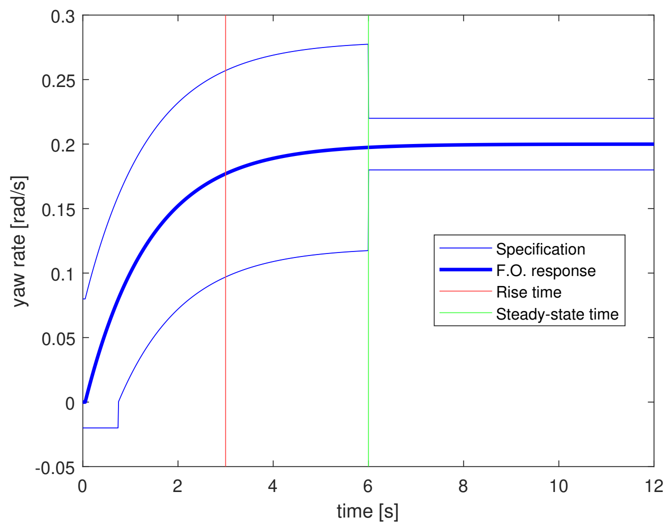


Figure 3.17: Generation of time domain specification

3.4.2 Pre-optimization

Initial parameter search & Parameter range estimation

Since some solvers require an initial guess and since it can be useful to quickly get a controller design for a quick test, this step of the method finds a controller for which closed-loop behavior stays within the time domain specification. A stochastic optimization algorithm called Particle Swarm Optimization (PSO) [16] was used for this task. The algorithm initializes particle positions and velocities randomly in the parameter space, and then each particle is evaluated according to a user-defined objective function. The best particle in the swarm is known to the other so that they can converge towards that solution, but there is also a factor of exploration.

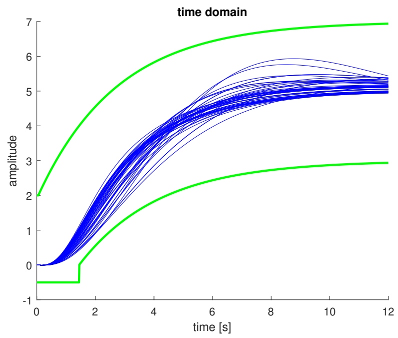
Thus, the resulting solutions are stochastic, which often causes long runtime, but the method can handle non-smooth functions quite well. Since some sets of controller parameters produce unacceptable closed-loop behavior according to a certain specification, they are penalized by functions with non-smooth elements. Thus, PSO was shown to be useful as it could find acceptable initial parameters quickly (< 1 min.), with the objective function of finding a system response in the center of the time specification.

Further, there was an attempt to narrow the controller parameter ranges defined as input to the method, which would narrow the search-space for optimization and thus maybe improve results and shorten the runtime of the method. This was implemented by a function first attempting to maximize the control signal, followed by attempting to minimize the control signal while still staying inside of the time specification, saving the minimum and maximum of each controller parameter to find a new parameter range. This function is evaluated in the next section.

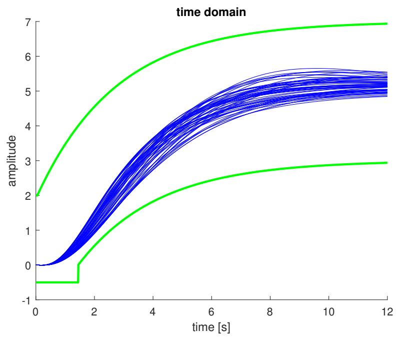
Frequency domain specification

To perform analysis also in the frequency domain a specification had to be created. This specification was not very intuitive to specify manually. Thus, a method for translating a time domain specification into the frequency domain was used [17]. Following is the implementation of this method used in this thesis; Given a time domain specification of how the closed-loop system should behave, the plant model to investigate, and a controller, the closed-loop behavior was simulated. If the behavior was inside the specification, the frequency response of the same closed-loop system was approved in the frequency domain. By iterating this process, i.e. by letting the function try different controller parameters to find a greater range of frequency responses, an approximate specification could be constructed.

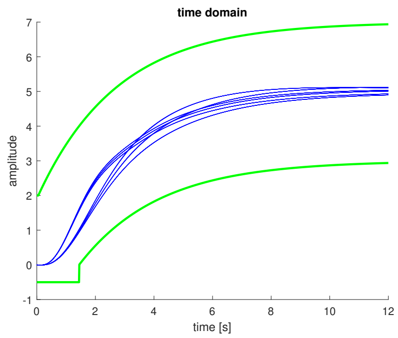
Two different ways of choosing controller parameters were investigated, grid sampling (introduced in Section 3.3.4), and a PSO solution. It was also investigated whether the parameter range estimation should be used or not. All four ways were tested for 60 seconds and in Figure 3.18 the time domain responses for each of those methods are shown. In Figure 3.19 the frequency specifications are shown. In the time domain, the green curves are the specifications and the blue curves are system responses.



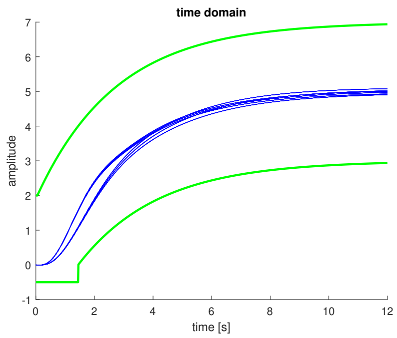
(a) PSO without parameter range estimation



(b) PSO with parameter range estimation



(c) Grid without parameter range estimation



(d) Grid with parameter range estimation

Figure 3.18: Comparison of time domain responses from creating frequency specifications using different solutions

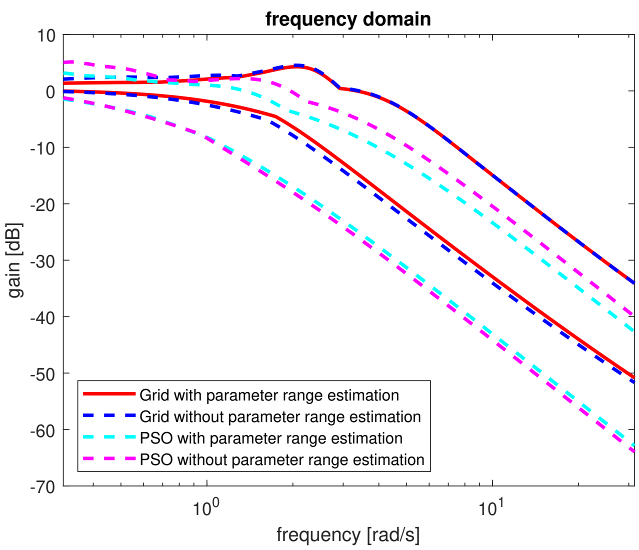
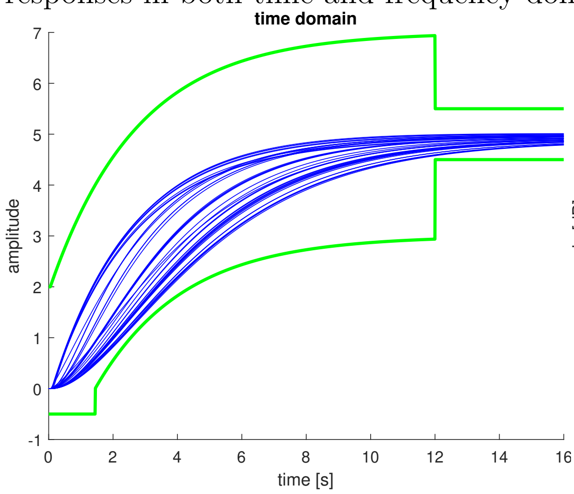


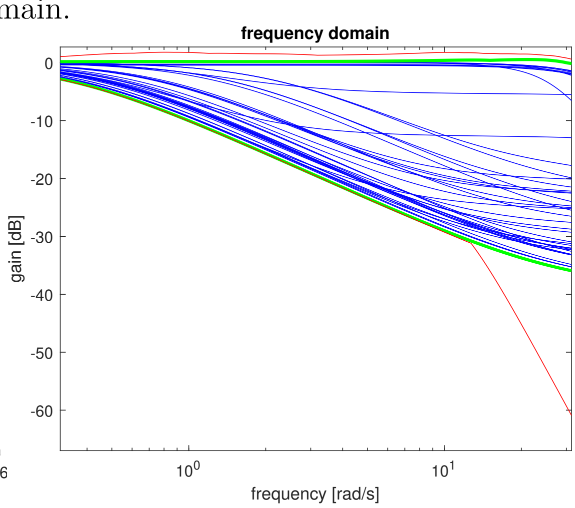
Figure 3.19: Comparison of frequency specifications created using different solutions

As the runtime was short, the resolution of the grid had to be low (exponential runtime growth), hence solutions produced by the grid solution were more narrow than the system responses of the PSO-generated solutions. Using parameter range estimation tends to skip the system responses which have the highest overshoot. The grid solution lacks most of the slower system responses, which have lower gains as can be seen in Figure 3.19 as well. Those responses are crucial if the goal is to find a controller making the system robust. Thus, the PSO solution with parameter range estimation was chosen as the method for generating the frequency specification.

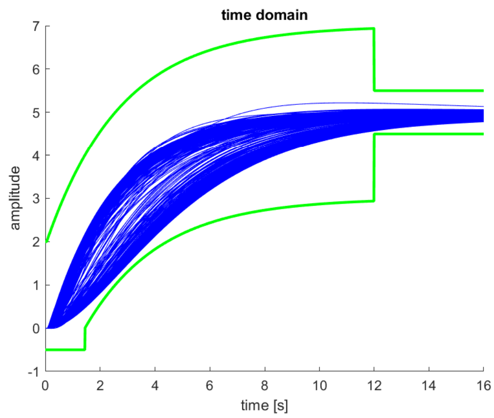
As the PSO solution is stochastic, a longer runtime should yield a better result, i.e. a wider frequency specification (larger solution-space). In Figure 3.20, a comparison was made between a 1-minute run and a 20-minute run. By looking at Figure 3.20b, where the long run specification is marked in red, it was clear that the shorter run produced a decent result since the area with most frequency content was still captured. The green curves are the specifications and blue curves are system responses in both time and frequency domain.



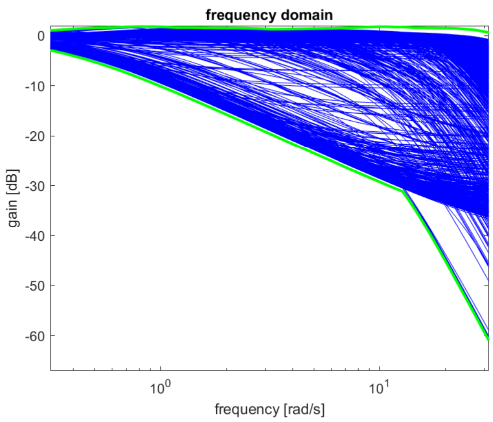
(a) Short run, time domain



(b) Short run, frequency domain (long run marked in red)



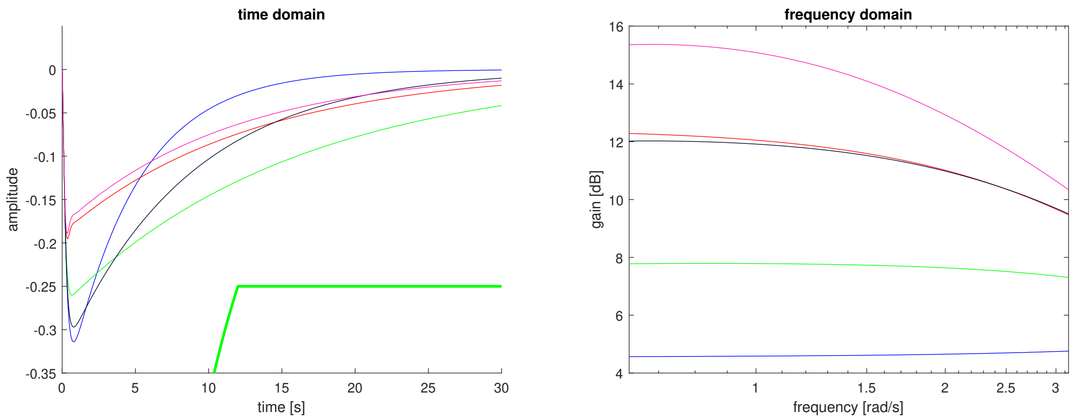
(c) Long run, time domain



(d) Long run, frequency domain

Figure 3.20: Comparison of frequency specification created by PSO with a 1 minute runtime compared to a 20 minute runtime

There was an attempt to create a frequency specification also for the disturbance rejection, but the frequency responses of the different plant, controller combinations were poorly correlated to their behavior in the time domain, resulting in that the specification could not be used. An illustration of that can be seen in Figure 3.21, where the corresponding 5 responses in time and frequency domain have the same color.



(a) Time domain **(b)** Frequency domain

Figure 3.21: Disturbance frequency specification

The pink and red solution had similar time behavior but quite different frequency responses. Red and black had a similar frequency response but very different time behavior. For disturbance rejection, one can argue that the blue and pink solutions were both good solutions in the time domain but they differed greatly in the frequency domain. No correlation was found and thus, disturbance rejection was only analyzed in the time domain.

In Figure 3.22 it is shown that each controller type produced a unique frequency specification for a certain plant model. Thus, when comparisons were made between different controller types involving frequency specifications in the results section of the thesis, the combined maximum and minimum of each specification were used when plotting results. This means that a solution might not look optimal according to the shown specification, but one should have in mind that the different solutions are restricted by their respective specifications.

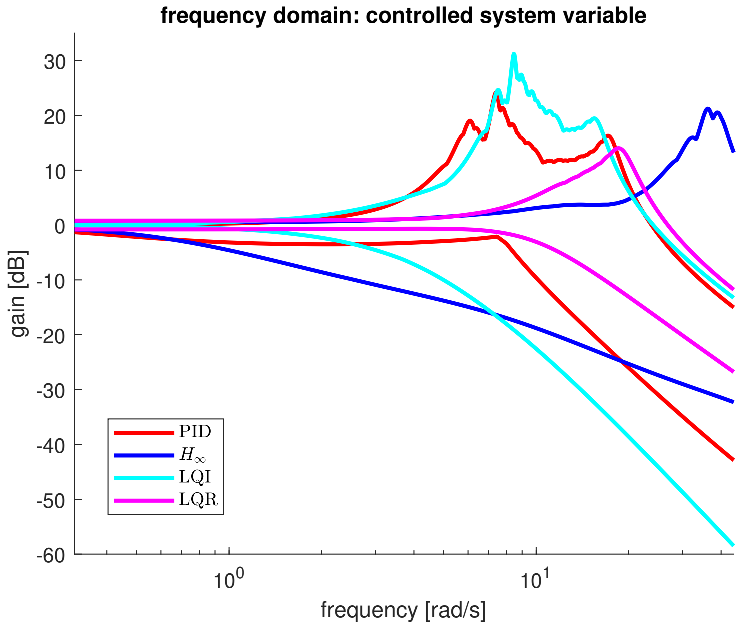


Figure 3.22: Frequency specifications for different controller types

3.4.3 Controller optimization

The method outputs a controller optimized to stay inside the given specification and minimize chosen **Evaluation metrics**, most of which are described in more detail in Section 3.3. Reference metrics are calculated from S_{yr} and disturbance metrics from S_{yd}. Metrics marked with a $*$ were inversed to be maximized. The evaluation metrics used in the time domain are

- Reference oscillation
- Reference overshoot
- Reference reaction time
- Disturbance reaction time
- Disturbance robustness
- Disturbance uncertainty
- Non-controlled system variable response

and the evaluation metrics used in the frequency domain are

- Reference performance
- Reference robustness
- Critical gain
- Critical frequency
- Bandwidth
- $*$Minimal Nyquist distance
- Phase shift
- $*$Robust stability

and in both time and frequency domain, Reference uncertainty.

The objective function of the controller optimization was defined as

$$\min_{x}\Big((p_u(\mathrm{em}(x)^2 W(p_{y_c t}+p_{y_c f}+p_{dt}+p_{y_{nc} t}))\Big) \tag{3.25}$$

where x is the controller parameters, $\mathrm{em}(\cdot)$ is the function evaluating the closed-loop metrics described above and W is a weighting vector used for choosing what to optimize for. Of most importance, the score was raised by a scale of how much the control signal exceeded its max value or max rate (p_u). Further, it was designed so that one can also choose to penalize violation of any of the following four specifications.

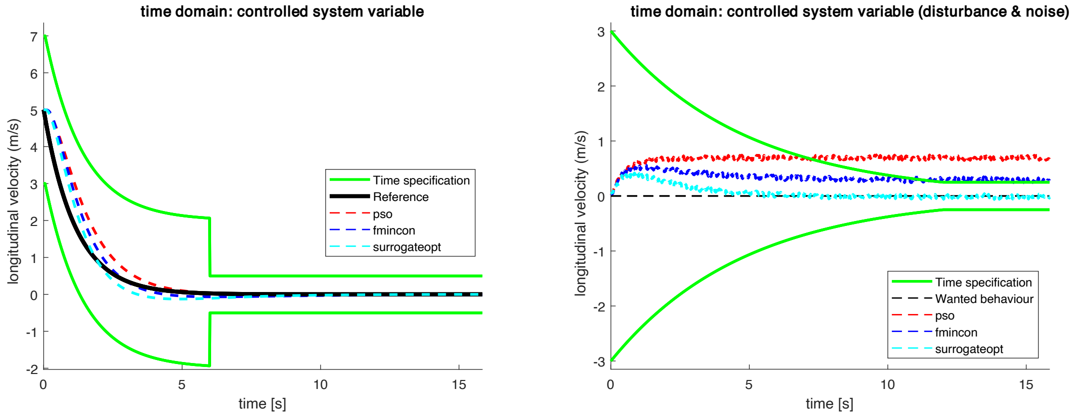
- Controlled system variable time specification - $p_{y_c t}$
- Controlled system variable frequency specification - $p_{y_c f}$
- Disturbance time specification - p_{dt}
- Non-controlled system variable time specification - $p_{y_{nc} t}$

and their values were based on a scale of the max distance that the system response went outside of respective specification.

Finding the minimum of the objective function (Eq. 3.25) is not a simple task for a solver. It is non-linear due to that a set of controller parameters are not mapping linearly to all the different evaluation metrics and partly non-smooth due to penalization of breaching specification. Thus, different solvers were tested. Initially, the non-linear solver `fmincon`[18] from the MATLAB Optimization toolbox was used, but it is designed for problems with both continuous objective functions and continuous first derivatives, thus only working in a subset of the optimization problem, often getting stuck in local minima.

To solve this, the PSO solution was tested, which improved the solution but raised the needed runtime much due to the introduced stochasticity. A third solver, included in the MATLAB Global Optimization toolbox, `surrogateopt`[19] was tested, and shown to perform better. This solver allow non-smooth objective functions and tries to find the global minimum with some exploration, yet yielding results with low stochasticity. It also proved to be quicker than the other two optimization methods since it uses a surrogate function which approximates the objective function while running. The actual objective function is also evaluated recursively in the algorithm to guide the optimization.

A difference could be seen when doing a quick optimization of an H_∞ controller towards good disturbance rejection. All solvers quickly approximated a good solution to follow the reference within the specification which can be seen in Figure 3.23a. However, regarding the disturbance rejection, a slight shift in certain parameters could produce very different results, as can be seen in Figure 3.23b.



(a) Reference following **(b)** Disturbance rejection

Figure 3.23: Solvers yielding differing results

Unlike the two other solvers, `fmincon` required an initial starting point. The starting point used was the initial guess calculated as described in Section 3.4.2. The resulting solution was very close to the initial guess, thus stuck in a local minimum, it was good for reference following but less good at disturbance rejection. The PSO solution needed a longer time to converge to a good result for reference following and showed very stochastic results for disturbance rejection. The `surrogateopt` solver yielded a good result here.

The following procedure was executed when optimizing a controller:

Approximate optimization

Initially, an uncertainty grid with a minimum of 2 levels was used, i.e., the minimal and maximal value of each uncertain parameter. Then, 20 iterations of the objective function were evaluated to find a system response that acts approximately like, or converging towards the wanted behavior. In each iteration of the objective function, uncertainties were being sampled. The controller parameters yielding the best closed-loop system were saved.

Approximate uncertainties

The best system from the last step was used as the nominal system when sampling approximate uncertainties. The grid level was decided by the maximum level that generates up to 3000 samples. The step-wise minimal and maximal response in both time and frequency domain were saved and also converted to percentual uncertainties relative to the nominal system.

Nominal optimization

Since uncertainty calculation was the most time demanding part of the method (>90% of runtime), the percentual uncertainties calculated in the previous step were used as a fixed margin of uncertainty when evaluating the objective function in this step, which sped up optimization much. Hence, only the nominal system response had to be calculated and the range of system responses was calculated using the fixed percentual margin step-wise multiplied with the nominal system response. The objective function was evaluated for 200 iterations.

A downside to calculating the percentual margin was when the system response crosses zero several times. This can happen when for instance the reference was sinusoidal. Without fixing the problem, the uncertainty will take on very high values due to the denominator (nominal system response) closing in on zero. The problem was not fixed, and thus one of the case studies of the results, Single Lane Change (SLC), had to be run for only the **Approximate optimization** step, but with 200 iterations instead of 20.

Uncertainty calculation

After nominal optimization was done, the best system from the last step was used as the nominal system, and uncertainties were sampled again, but now with at max 20,000 samples. This number was decided from the decision taken in Section 3.3.4 with an extra margin of factor two. Responses in time and frequency domain were saved again, and also converted to percentual uncertainties.

(Re-optimization)

The solution from **Nominal Optimization** with the more accurate uncertainties was now checked to see if it lies within specifications. If not, it was re-optimized once, for 100 iterations. The resulting system was mostly within specification, for the cases where it is not, a recursive solution might have to be investigated.

Worst-case gain verification

Lastly, the worst-case gain of the system was calculated as a way of verifying the correctness of the sampled uncertainty. The worst-case gain at a particular frequency was calculated with MATLAB Robust Toolbox function `wcgain`, which computes the structured singular value μ [20]. The worst-case gain is supposed to be the highest gain possible given an uncertain system, and thus a warning was stated if the sampled uncertainty has higher gain than the worst-case gain.

A downside to `wcgain` was that it only calculates the worst-case gains at often around 20 frequency points, so in some cases, the sampled uncertainty has gain peaks between two frequency points, and those peaks are thus not verified. An example of the verification can be seen in Figure 3.24.

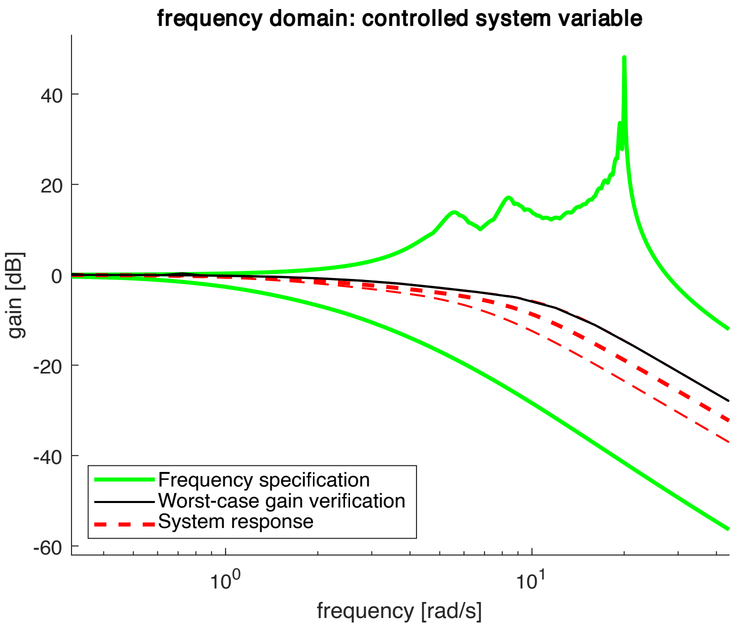


Figure 3.24: Worst-case gain verification

4

Results & Discussion

To evaluate the method of closed-loop system analysis, controller optimization as well as the similarities between the linear and non-linear models, four case studies were conducted and are presented in this chapter. In the first case, optimization for lowering the frequency range of the control signal is performed. In the second one, LQI control is compared to LQR control. In the third, optimization for disturbance rejection in both time and frequency domain is performed. Finally, in the fourth study a single lane change situation optimized for four different evaluation metrics. Further, the results are discussed with regards to methodology, modeling, and controller choice. Finally, the research questions are discussed.

4.1 Case study: Optimization for lower *bandwidth*

By lowering the bandwidth of the transfer functions from the external inputs disturbance, noise, and consequently reference, to the controlled state and control signal, the system should get less excited at higher frequencies. Testing that had been done on the real vehicle showed that inputs over a certain frequency tended to cause oscillatory behavior in the system. Thus, optimization for lowering the bandwidth was implemented. In Figure 4.1, results of this optimization is illustrated and the effect on the control signal especially investigated.

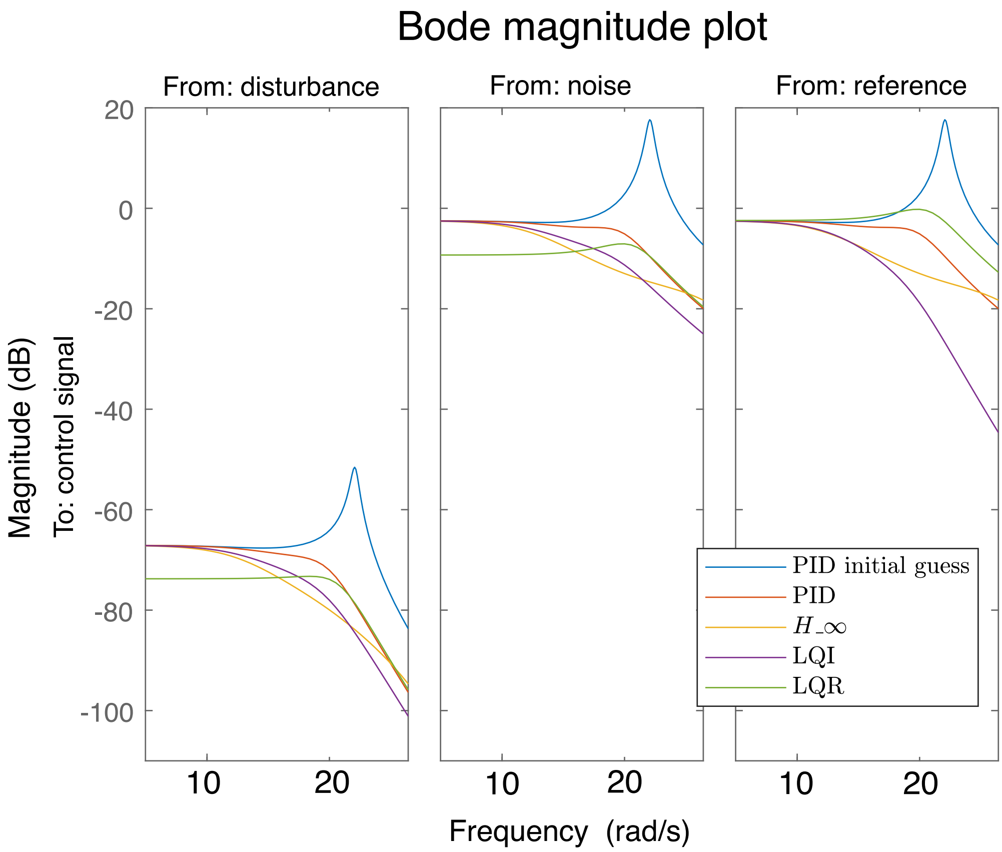
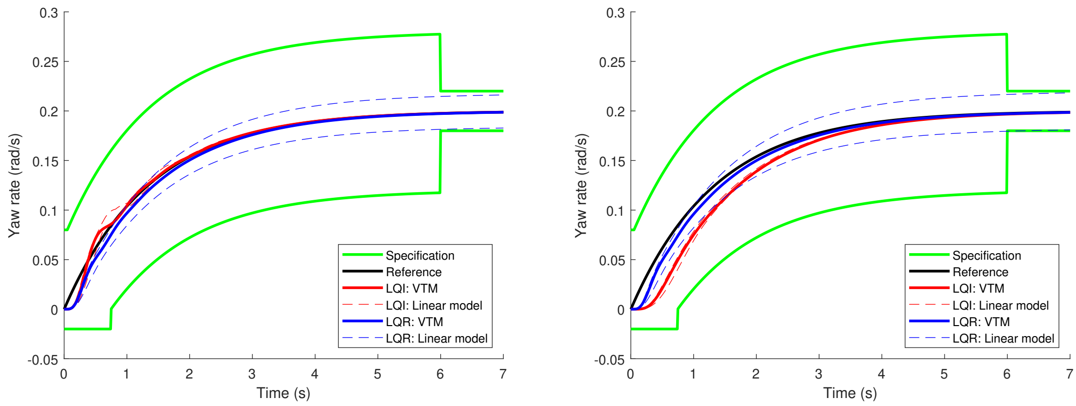


Figure 4.1: Bode magnitude plots of S_{ud}, S_{un}, S_{ur}

The PID solution from the initial parameter search was used as a reference solution as a non-optimized comparison. In comparison to the reference solution, all four controllers yielded a closed-loop system with lower bandwidth. The PID and LQR solution still had quite high bandwidth from reference inputs. The LQI and H_∞ solutions had similar bandwidth for such inputs but LQI dampened high-frequency inputs even more. Similarly, for disturbance and measurement noise inputs, the PID and LQR solution still had higher bandwidth than the other two. The H_∞ solution had the lowest bandwidth as well as a lower gain than the LQI in the mid-range of frequencies, but a higher gain at high frequencies.

4.2 Case study: LQI/LQR control

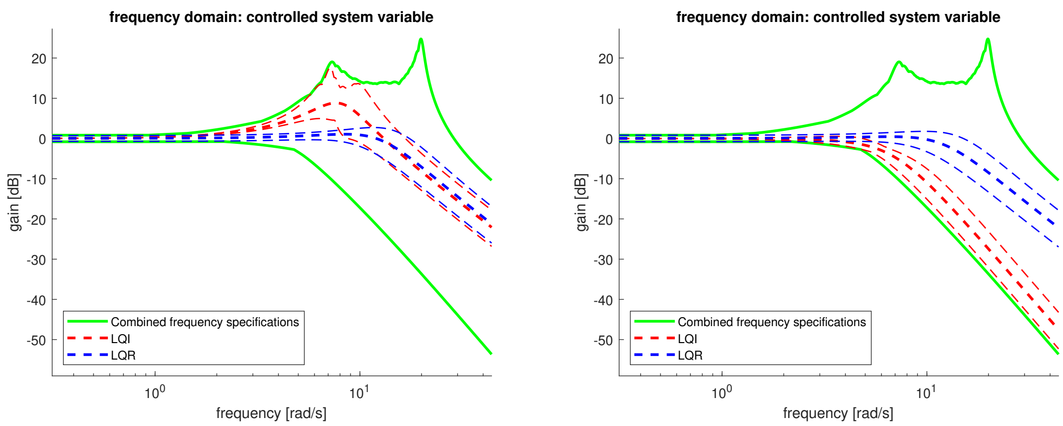
Due to the system architecture, which was explained in Chapters 1 & 2 and visualized in Figure 2.1, a vehicle controller with integral action was not necessarily needed. The reason for this is that the *Driver* acts as an outer loop integrating controller. Thus, the LQR was included for analysis in the thesis. Since the LQR and LQI controllers are based on the same theory, it could be interesting to see what difference there would be between these in optimization. Thus, they are compared in Figure 4.2, tuned for two different metrics, and evaluated both by the linear lateral tractor model and the VTM.



(a) LQI and LQR tuned for performance **(b)** LQI and LQR tuned for robustness

Figure 4.2: LQI and LQR optimization comparison, time domain

Since the LQR has no integral action, it has a quite uncertain reference following and poor steady-state performance. In this test, the uncertainties are actually almost as large as what fits inside the specification. Thus, it is better to use LQI if less uncertain reference following and steady-state performance are of the essence. In Figure 4.3 the frequency domain of S_{yr} for each solution is shown.



(a) LQI and LQR tuned for performance **(b)** LQI and LQR tuned for robustness

Figure 4.3: LQI and LQR optimization comparison, frequency domain

Since the frequency specifications are combined, it might look like the LQR can find other solutions but its frequency specification was much narrower, almost as narrow as its uncertainties (see Figure 3.22 for general reference). The peak in Figure 4.3a corresponds to the slightly quicker response in the time domain. Further, the more dampened frequency response in Figure 4.3b leads to the slower time response in the time domain.

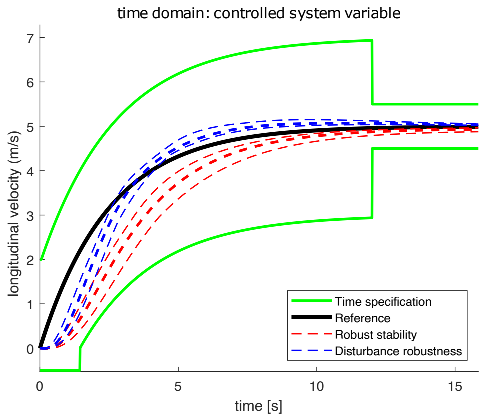
4.3 Case study: Disturbance rejection

For disturbance rejection of a general disturbance step, *robust stability* optimization was compared to *disturbance robustness* optimization. The metric *robust stability* measures peak gain of the GoF bode plots and *disturbance robustness* measures the mean distance between the system response and the wanted behavior (the line at zero) in the time domain. The results of optimizing for the two metrics and simulating the resulting controllers with the linear longitudinal model (Eq. 3.6) is shown in Figure 4.4. For this test, with the linear longitudinal model, the LQI controller was used. Note that disturbance rejection was performed in a separate simulation from that of reference following.

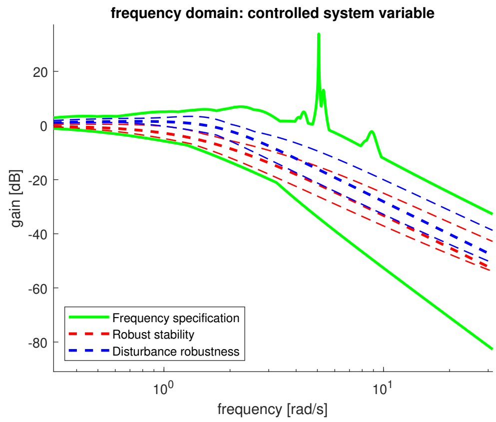
In Figure 4.4d it can be seen that optimizing for *robust stability* successfully lowered the peak gain of the most amplified transfer function S_{ud}. But to achieve good disturbance rejection, it was not sufficient to only optimize for this, since this solution results in a higher gain in the low-frequency region of S_{yd} than the *disturbance robustness* solution. Thus, the disturbance was not rejected, and instead, the longitudinal velocity becomes constantly affected by it. The crucial difference was most likely caused by that the cost for integral state offset is about 10^8 times higher in the *disturbance robustness* solution.

Likewise for PID, the integral gain was 0 for the *robust stability* solution, i.e. it had no integral action. For the H_∞ controller, a higher cutoff frequency for the disturbance filter was noticed, and thus, the disturbance might get treated as a reference signal. What was interesting was that both solutions have acceptable reference following, although differing so much in disturbance rejection.

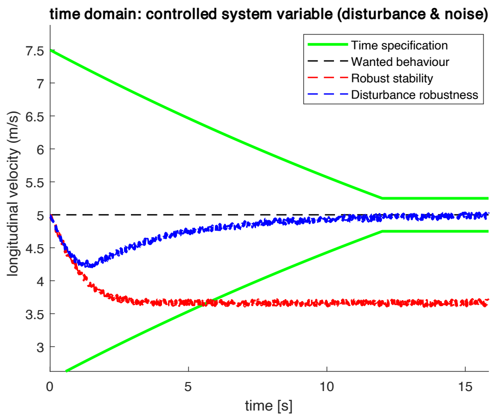
The system responses for all three controllers were simulated in VTM, with the results shown in Figure 4.5. At 5 seconds, a disturbance step of the same amplitude for all three controllers was applied. For the VTM, it can be seen that the *robust stability* solutions never reject the disturbance just as for the linear model. The H_∞ controller seems to be least affected by the disturbance and was also the quickest to compensate it. The reference seems to be followed better than the test with the linear model.



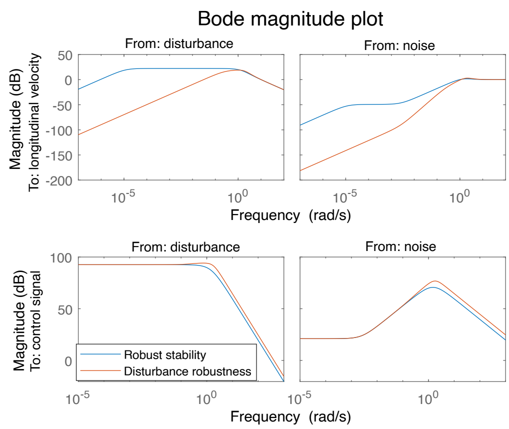
(a) Reference following, time domain



(b) Reference following, frequency domain



(c) Disturbance rejection, time domain



(d) Nominal system GoF bode plots

Figure 4.4: Simulation of linear longitudinal model: disturbance rejection

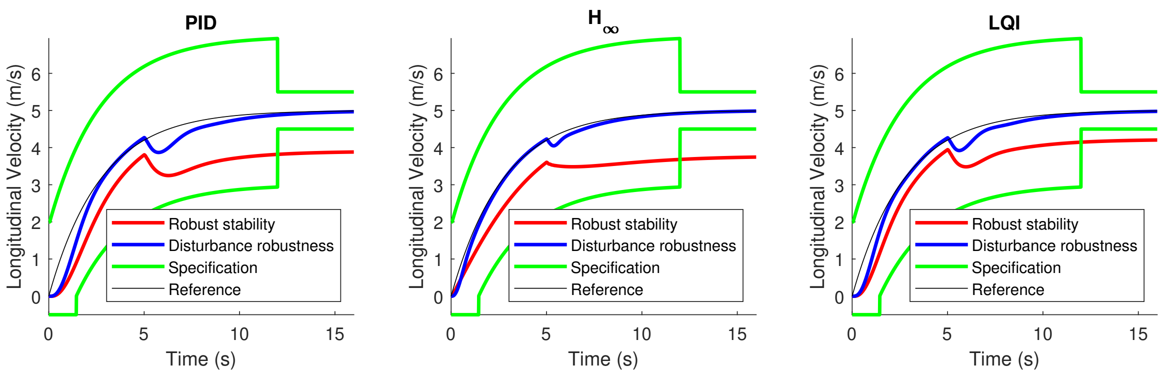
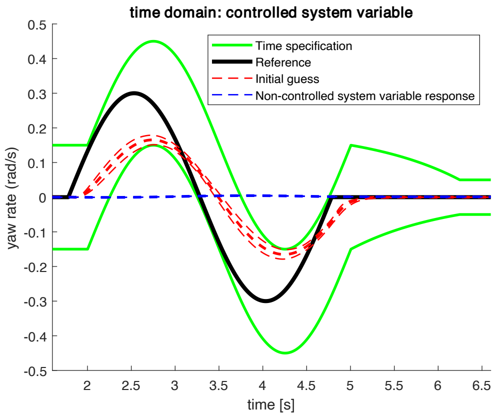


Figure 4.5: Simulation of VTM: disturbance rejection

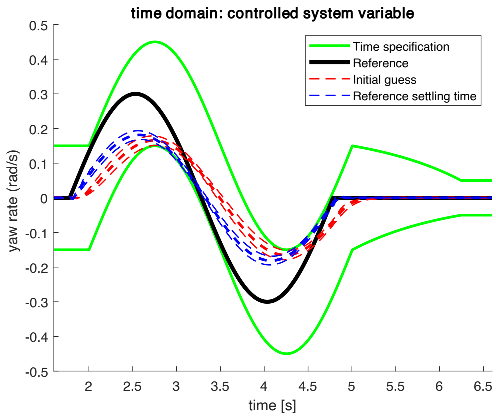
4.4 Case study: Single lane change

A study of a single lane change maneuver was performed, which is considered a standard test in vehicle dynamics testing. In this test, the reference sinusoidal period was set to three seconds and with 0.3 rad/s amplitude. The specification was generated as the reference ±0.15 rad/s, shifted 0.2 seconds later. The added metric settling time was defined as the time between the end of the sinusoidal part of the reference signal until the time where the controlled system variable had stabilized within $(0 \pm 0.005 A_{\text{max}})$ where $A_{\text{max}} = 0.3$ rad/s. In Figure 4.6, the result of simulating the linear lateral tractor model with four different H_∞ controllers is shown.

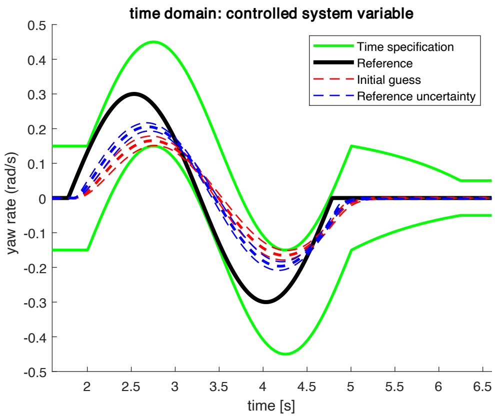
The solution from the initial parameter search was used as a reference solution.



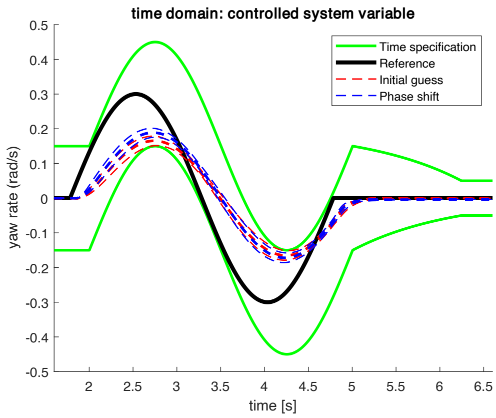
(a) Optimized for less lateral velocity



(b) Optimized for less settling time



(c) Optimized for less uncertainty



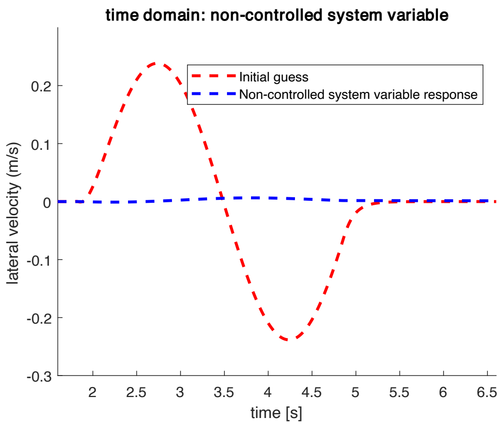
(d) Optimized for less phase shift

Figure 4.6: Simulation of truck performing a single lane change I

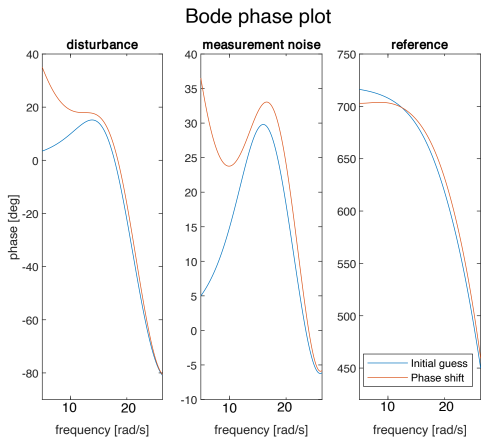
In case **(a)**, by just aiming to minimize the lateral velocity (non-controlled system variable), the resulting controller will not follow the reference at all and will also not fulfill the specification. That behavior is heavily penalized in the optimization algorithm, but it was probably stuck in a good enough local minimum. In case **(b)**, the settling time was lower than the reference solution. In case **(c)**, no difference in the amount of uncertainty can be seen. In case **(d)**, the difference was not big from the initial guess. The reference amplitude was not reached in any of the four cases. For better reference following, the optimization weightings would have to be changed.

In Figure 4.7a, the lateral velocity from the simulation shown in Figure 4.6a is plotted and in Figure 4.7b, the phase plot from disturbance, measurement noise, and reference to yaw rate from the simulation shown in Figure 4.6d is plotted.

The lateral velocity was lower, but at the cost of not following the reference



(a) Lateral velocity



(b) Phase plot from inputs to yaw rate

Figure 4.7: Simulation of truck performing a single lane change II

properly. Further, the phase shift was lower than the phase shift of the reference solution. The evaluation metrics are presented in Table 4.1.

Table 4.1: SLC evaluation metrics

	I	II	III	IV
ref	46.22	0.65	1263.68	17 719.14
(a)	1.58	0.08	227.58	13 082.34
(b)	50.01	0.21	1321.73	22 003.80
(c)	56.67	1.02	188.66	15 683.62
(d)	52.67	5.22	172.69	14 542.23

Metrics are, I - *Non-controlled system variable response*, II - Settling time, III - *Reference uncertainty*, IV - *Phase shift*, the solutions are marked as in Figure 4.6.

All metrics are defined in Appendix A. As for metric I, case **(a)** had the lowest value, and it was thus successful. It also had the lowest value for two other metrics, but that can be ruled out due to its unacceptable system response. As for metric II, case **(b)** had the lowest value and was thus successful. Comparing the results of metric III with the visual results, it is clear that this metric was not properly designed, since the 6 times higher values is not apparent from visual comparison. Further, case **(c)** fails to minimize it as well. As for metric IV, case **(d)** had the lowest value and was thus successful.

4.5 Methodology

When analyzing parametric uncertainty, the chosen method of using grid sampling proved good for models with not too many uncertain parameters. The trade-off lies between having a grid with high resolution and runtime, as the runtime will grow approximately at the same rate as the number of samples does, which is

$$n_s = g_r^{n_p} \tag{4.1}$$

where n_s is the number of samples, g_r is the number of grid points in each parameter and n_p is the number of uncertain parameters. Ideally, a grid with very high resolution would yield the best results (i.e. converging to continuous analysis). The problem is that with each extra parameter added, the runtime grows exponentially.

So for instance, when simulating the lateral tractor model (Eq. 3.1) with 5 uncertain parameters, a grid resolution of 2 was used for quick approximation and 7 at max for a more accurate analysis. There are 4 resolutions in between allowing for flexibility, and the corresponding number of samples is 32 and 16807.

Compare that to simulating the lateral tractor with trailer model (Eq. 3.3) with 10 uncertain parameters, resulting in that a resolution of 2 results in 1024 samples and for a resolution of 3, 59049 samples. Thus, using even the quickest approximation besides nominal response resulted in the total runtime of the method increasing by a factor 12. And there was no flexibility to test another resolution since 59049 samples result in unfeasible runtime.

For reference signals, mostly first-order system responses were used instead of step signals. This was because they were judged more realistic and used in controller optimization, generated less aggressive controllers which was positive for simulation of the VTM.

4.5.1 Model improvements and closed-loop validation

The idea was to use linear models with behavior similar to that of the VTM such that results and conclusions from analyzing them could be translated to the VTM, which was assumed to model behavior as close to reality as possible without obtaining real data in any form. The linear models used for this purpose were pre-existing models not derived specifically for this thesis. Making an initial comparison to the VTM by running them in open-loop without any uncertainties given the same parameters and input showed in some cases clear differences. This was expected as some dynamics are excluded from the linear models. The lateral models, as mentioned in [12] and [21], did not model, for example,

- Large articulation and tyre slip angles
- Deviations from Ackermann steering geometry
- Varying axle cornering stiffness
- Vehicle body roll motion
- Aerodynamic forces
- Suspension, chassis and cabin dynamics

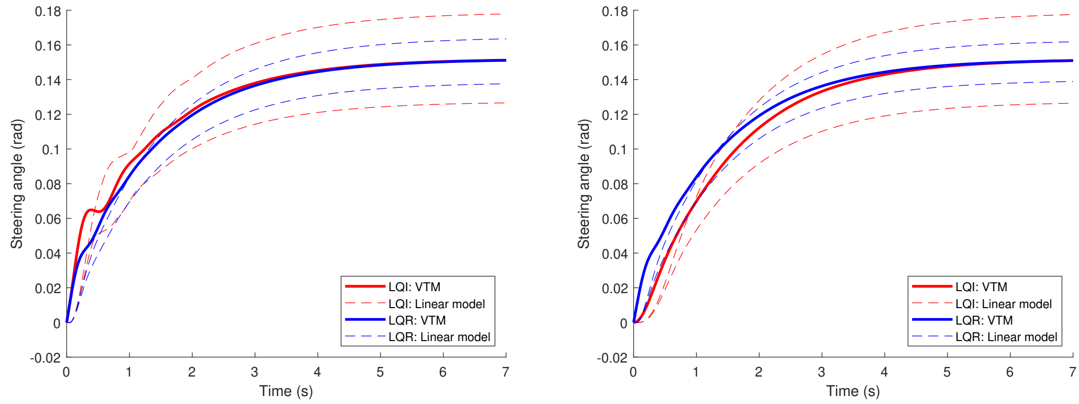
Also, the longitudinal model did not directly include e.g. any weight distribution, suspension, or wheel slip dynamics. The wheel slip dynamics were however partly included as the wheel torques were limited by approximate maximum acceleration and deceleration.

To better match the VTM response, realistic uncertainties were first added to the linear models, e.g., how much the cornering stiffness could vary depending on environmental reasons. While these uncertainties did add a range of input response behavior to the linear models, they still did not alone encapsulate the VTM behavior completely. To be able to reach that desired result, synthetic uncertainties were added to the linear models, which also had their outputs scaled. This could result in something like what was shown in Figure 3.4a, where it seemed like some combination of the included uncertainties would give a very similar response to the VTM. However, it could also give a result such as in Figure 3.4b, where all the possible linear model responses together encapsulated the VTM behavior to a large degree but where it would seem maybe a bit more unlikely that a single unique response would be close to the VTM response.

In closed-loop, the nominal linear model and VTM responses generally looked more similar. This was probably because both models strive to follow the same reference. For the same reason, the linear model uncertainty range gets thinner than the open-loop simulations done in Section 3.1.5. This can for instance be seen in Figure 4.2 where the LQI was compared to the LQR. As the LQR does not have any integral action, it has some steady-state error whenever the system was not nominal. Yet, the VTM in closed-loop with an LQR designed with feedforward based on the linear model resulted in close to no steady-state error, which was perhaps not anticipated.

Another result of the model behavior similarity in closed-loop was their respective control signals. For the aforementioned simulation visualized in Figure 4.2, the corresponding control signals are presented in Figure 4.8. While the steady-state control signals are equal, the transient behavior differ a bit. Most noticeably, the VTM control signals initial response was more aggressive and exhibits a slightly more oscillatory behavior, resulting in that the VTM signal was not being encapsulated by the linear model bounds in that region.

To further improve the similarity in behavior between the linear models and the



(a) LQI and LQR tuned for performance **(b)** LQI and LQR tuned for robustness

Figure 4.8: LQI and LQR optimization control signal comparison

VTM, there were several things that could have been done. Firstly, the models could have been augmented with additional forces and dynamics. This could make the models behave more like the VTM but with a risk of becoming over-parameterized and for the case of uncertainty analysis computationally heavy. Another way of increasing the similarity could be to tweak the parameters used for the linear models to match the VTM behavior rather than using just using VTM parameters. Taking this idea further, the models could have been derived using system identification methods with input-output data from the VTM or perhaps even from the system in reality.

4.5.2 Full-state feedback assumption

The LQR and LQI controller types presented earlier was subsequently used to generate results. These control theories bring some inherent robustness properties, such as guaranteeing an infinite gain margin and a phase margin $\geq 60°$ [22]. However, these two properties are only valid granted full-state feedback, i.e. that all system states can be measured. If for some reason this is not the case and some states instead have to be estimated using an observer, these robustness properties are lost [1]. The lateral vehicle models used in this thesis, i.e., Eq. 3.1 and 3.3, involves the tractor lateral velocity v_y as a state.

Whether this is measured or not relies on how accurate of a measurement that is required. In the real system, the lateral velocity is not directly measured, but rather the integration of a measured lateral acceleration. If this integration would introduce e.g. drift, would it classify as state feedback? Similarly, is a derivative of a measurement a good enough measurement? While it is important to consider these questions, defining the quality of measurement needed to allow for state feedback was not considered a part of the scope of this thesis. Thus, it was assumed that full state feedback was available.

4.5.3 Validity of analyzing linear model guarantees

A linear model will never act exactly like its nonlinear counterpart unless the input-output scenario is extremely limited, in which case the analysis might not be very useful. Thus, the idea of guaranteeing a behavior from a nonlinear model (such as the VTM) given the analysis of a linear model can be hard to argue for, as the word guarantee is a very strong statement. What helps to argue for this is the way the linear model is derived, with similar dynamics and adapted with uncertainties as described in Section 3.1 while not arguing for an exact behavior guarantee but rather for a range in which the behavior is expected.

Of course, there will always be dynamics present in a nonlinear model that can not be modeled exactly in its linear counterpart. So, to what extent does a linear model need to exhibit the same response as its nonlinear counterpart to be able to make any kind of a guarantee? An idea on how to improve the argument for a guarantee could be by e.g finding the unique set of uncertainties for the linear model that best fits the VTM response and then determine the rigidity of the similarity between these.

4.6 Guarantees for system in reality

As discussed in Section 4.5.3, guarantees for a nonlinear model given linear model analysis can in itself be tough. Furthering that guarantee to a real system is a completely different challenge and was not in the scope of this thesis. However, what could be done is to gather real data and compare it in simulation, which is something that is commonly done. In addition, the VTM could be augmented with dynamic uncertainty to make a similar guarantee as was done between the linear model and the VTM in this thesis.

A risk of layering uncertainties like this is that at the bottom layer (the linear model) the propagated uncertainties could grow too large to make any sense for analysis and controller synthesis. Of course, one could skip the middle layer (VTM) and instead, as mentioned in Section 4.5.1, adapt the linear model directly to real-world input-output data through system identification methods.

4.7 Controller choice

In general, both H_∞ control and LQI control showed great flexibility when optimized for different metrics. LQR usually resulted in similar solutions, slightly indifferent to optimization weights, maybe because of only having two parameters and feed-forward always enabled. Since the H_∞ controller is of a higher order than LQI, and quite high in general, it can follow very quick reference changes better but it has the downside that it needs more resources for its implementation and is more susceptible to numerical errors (e.g. too high gain at certain frequencies). Some of the H_∞ controllers created by the method worked well for the linear models but there was trouble when simulating with the VTM.

The large derivatives produced lead to that a fine time resolution and a high order solver was needed for simulation of the VTM, and thus time consumption rose. In some of these cases, the H_∞ controllers introduced a very large oscillatory control signal, though still a very dampened response as the VTM includes realistic, physically limited, actuator modeling. Thus, for the actual use of the H_∞ controllers produced by the method, these limitations need to be included in the optimization.

Tuning the H_∞ controller was not very intuitive, and thus generating controller parameters (as done in the method described in Section 3.4) proved to be a good way of doing it.

4.8 Evaluation metrics

The first research question of the thesis was *Which analytical methods can be used to quantify robustness, performance, and stability regarding lateral and longitudinal controllers in the context of autonomous truck vehicles? How do these compare?* The evaluation metrics can be seen as a means of quantifying the terms mentioned above. These metrics are discussed in this section.

By optimizing for certain metrics the resulting closed-loop behavior could vary a lot. For example, when only optimizing for *reference uncertainty*, the system response can be both fast and slow, oscillatory and non-oscillatory, overshoot or not, and e.g. phase shift can vary much. This might be due to that there are many minima of this metric in the objective function.

On the other hand, a metric like *reference performance* probably have a lot less unique minima or that most of its minima belong to a clear region of the function where minima of all metrics have resulting system responses that have good *reference performance*. Thus, *reference uncertainty* can be called a minor metric, and *reference performance* a major metric.

In Figure 4.9 this theory is visualized, as the more complex objective function is simplified to a two-dimensional function where the four different regions are where most of the e.g. 4 different major metrics minima occur, and the markers show where the minor metrics minima occur.

Optimizing only for major metrics yields more reproducible results and to optimize for a minor metric should be done in combination with a major metric to in a sense guide the optimization to a specific region of interest. The division of metrics between major and minor seems to differ a bit depending on controller choice and experiment setup, but in general, the following division was seen.
All metrics are defined in Appendix A. The major metrics are deemed to be

- Reference performance

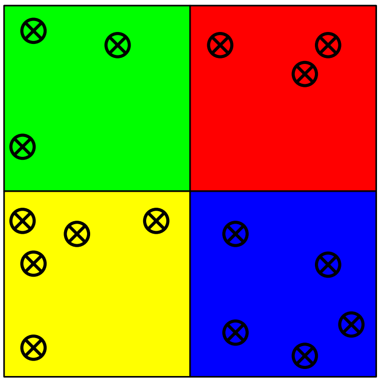


Figure 4.9: Simplified objective function supposed minima locations

- Reference robustness
- Reference reaction time
- Disturbance robustness
- Phase shift
- Bandwidth
- Non-controlled system variable response

and the minor ones are deemed to be

- Reference uncertainty
- Disturbance uncertainty
- Disturbance reaction time
- Reference oscillation
- Reference overshoot
- Minimal Nyquist distance
- Robust stability
- Critical frequency
- Critical gain

Another point of interest was how different evaluation metrics compare against each other, for instance, are they equally good at finding certain solutions?

When minimizing *non-controlled system variable response*, the results would often be close to those of optimizing for *reference robustness*, most likely due to that the system variables are linearly dependent of each other as a linear model was used and hence, when gains are lowered for reference following, the other system variable is also less excited. Another example was that when performing the single lane change case study, minimizing settling time usually gave results close to those from optimizing for *reference performance*. This might be due to that a high gain controller often reacts quickly to any existing control error.

Another thing that was observed was that optimizing for *phase shift* showed similar system responses as optimizing for *reference uncertainty*. They both resulted in controllers with high gains, and in terms of PID, the proportional gain was higher than similar controller parameters found in solutions optimized for e.g. *reference performance* which yielded system responses with much more uncertainty. These are just some examples of this, there might be more connections to be found.

4.9 Guaranteed bounds

The second research question of the thesis was *how can bounds for parametric uncertainties and external disturbances be derived, to ensure safety and stability in a closed-loop consisting of a plant model and controller?*.

As for *parametric uncertainties*, by sampling the uncertain plant and simulating it in combination with a controller, the range of system responses can be analyzed. Given a time domain specification that is deemed safe, it is then possible to calculate what margin a certain parameter has for staying inside the specification. A slightly conservative margin would be the minimal distance between the system response and the specification.

Vice versa, it might be possible to test for the maximum level of uncertainty that would make the system responses start ending up outside of the specification. Since different parameters affect the system in different ways, certain combinations of uncertainty in each of them could end up yielding the same system response, so it is a multifaceted problem.

An idea to solve it is to simulate the system with varying amounts of uncertainty in each variable, look for the combinations that make the total system uncertainty (i.e. percentual uncertainty w.r.t. nominal response) hit different levels (numerical but denoted as e.g. slightly uncertain, moderately uncertain, very uncertain and outside of specification), in a sense creating a level plot in the same dimension as the number of uncertain parameters. The question remains on how to perform this with a limited amount of simulation and still be able to trust the results.

As for *external disturbances*, in the time domain one way to discern e.g. max slope angle in the longitudinal case or max lateral wind in the lateral case was to simulate the system following a reference and suddenly receiving a disturbance step. Then the system response can be checked to see if it stays within specification or not.

This can be done in combination with the already added uncertainty. By then trying different levels of disturbance, a max level can be found. The downsides of this method are that the disturbance has to be modeled accurately, which can be tricky and time-consuming. Also, many different situations have to be simulated, which might be impossible to create specifications for or take too much time.

Another perhaps more general idea (in the frequency domain) is to look at the gain of S_{yd}. If the maximum allowed state shift Δy at each frequency is known in some way, then by dividing with the known gain, the maximum disturbance level at each frequency is also known. This has the downside that it requires accurate modeling of the disturbance, but with the upside of being able to create a profile of max disturbance w.r.t. frequency. So the question is how to determine $\Delta y(f)$, which remains to investigate.

5
Conclusion & Outlook

The problem that sparked this thesis originates from the desire to be able to analytically prove that a given closed-loop system, given certain inputs, would behave in a certain way. In this case, the closed-loop system is an autonomous truck. By being able to prove and predict the behavior of the truck, it can operate safely and allow other systems around it to operate safely and efficiently as well.

The aforementioned problem is certainly considerable and is thus only partly taken on in this thesis. More specifically, the thesis tries to cover how to systematically quantify the robustness, performance, and stability of a closed-loop system w.r.t lateral and longitudinal control of an autonomous truck. Further, the scope involves looking at how behavioral bounds of the truck can be determined when exposed to parametric uncertainties or external disturbances.

To enable the use of frequency analysis and controller synthesis methods, only linear models were used. They were chosen with simplicity in mind, i.e. they do not take into account the full dynamics of all subsystems (e.g. extensive tire dynamics). This is because the analysis was chosen to be on a high level, to get a rough overview of major dynamics instead of getting stuck on details that maybe would require a separate study.

As a benchmark for model accuracy as well as results analysis, the more detailed and nonlinear VTM was used as a model that is regarded close to reality. The idea was to make the linear model behave like the VTM so that analysis of the linear model gives results that can be argued to be valid for the VTM as well. Hence, the linear model was adapted to the VTM behavior response by introducing parametric uncertainty and scaling its output, creating a range of behavioral responses, given an input.

A general system architecture was proposed to standardize the control problem and sort out the relevant transfer functions. A controller design procedure was decided and early testing was performed to verify the expected model and controller behavior. Further, analytical methods to quantify system behavior by evaluation metrics both in time and frequency domain have been proposed. They relate to factors such as stability, performance, robustness, and disturbance rejection. Most are classical methods with well-known resulting metrics and some are newly introduced, quite simple constructions. Methods for analyzing uncertain systems were also briefly investigated.

A method was created to both evaluate a closed-loop combination of plant and controller, as well as generating a controller for a certain plant. The controller is optimized for the goal of reaching a certain specified system behavior which can be specified by the earlier defined evaluation metrics. It can also be specified by either time domain specifications or frequency specifications created from time domain specifications.

Whether the closed-loop passes the time specification was verified by simulation runs and the frequency specification by evaluating transfer function frequency response. Since the objective function is complicated, different optimizing methods have been investigated, and the one handling global optimization most efficiently was chosen. The optimization was done through several steps, designed to increase solution accuracy and minimize the computational load.

The linear models used for analysis were adapted to match the VTM behavior in such a way that comparing them in both open-loop and closed-loop simulation show many similarities, where the VTM lies well inside the uncertain linear model response bounds and the dynamic responses were similar. For some of the model states, in some scenarios, the similarities were however weaker. As for the controllers optimized towards some evaluation metric, some of them show good results in that the wanted system behavior was achieved, and in others, some investigation into improving optimization needs to be done.

Drawing conclusions regarding the VTM, given analysis and simulation results from the linear model, is not hard to argue for in some cases, where the simulations of the two models match well. However in some scenarios, at higher velocities or with aggressive inputs, where the similarities were weaker, it is harder to argue that the analytical results derived from the linear model would cohere with the VTM.

The evaluation metrics successfully quantify system behavior in terms of stability, robustness, performance, uncertainty, and disturbance rejection. The method of generating controllers to achieve a certain system behavior proved helpful in achieving certain system behavior. As for the behavioral bounds of the truck, those related to parametric uncertainty can be calculated in simulation and verified by worst-case gain analysis from the frequency domain. Those related to external disturbances can be calculated through simulation.

Future work

The method of generating controllers can be improved in the sense of making optimization faster and finding better optima of the objective function (closer to possible global optimum). The methods that require simulation is quite time-demanding and specific for the scenario being tested. Thus a more general solution in a sense of minimizing simulation use is yet to be found.

One of the studied test cases in Chapter 4 involved investigating a single lane change maneuver. This concept could be extended to other types of common tests, such as a double lane change, a sine with dwell, a constant radius turn, or a fishhook maneuver. These could both present results in model similarity and provide optimization goals for a deeper understanding of how a unique controller would handle different situations. Optimizing a controller for multiple cases would require the optimizing algorithm to be extended with this functionality. Most crucial would be how to solve the zero-crossing problem mentioned under *Nominal optimization* in Section 3.4.3 and investigate whether the frequency specification solution can be extended to more complicated references than step-like ones. Hybrid optimization, i.e. performing as much analysis and optimization as possible of the non-linear model and the remaining analysis on the linear model would be interesting.

Further extensions of the algorithm could involve optimizing for the case of two controllers working together, as an acceleration controller and a braking controller would. Additionally, since the lateral DSTM was linearized for constant longitudinal velocity, it would be of interest to investigate how a possible gain scheduling of the controllers can be implemented, and its implications.

Moreover, an investigation into the H_∞/LQI robustness guarantees would be useful to see if they can be used in creating behavioral bounds of the system or if they can be translated to a more hands-on metric of robustness in e.g. the time domain. The minimal performance level γ of each controller mentioned in Section 3.2 was manually tuned in use within the method until the controllers became sufficiently non-aggressive. Since γ is being minimized, it would be interesting to investigate what the viable range is and whether the minimal level can be used as a tuning parameter.

A different robust controller design method that appeared in the literature is that of Quantitative Feedback Theory (QFT). It seemed interesting to investigate, but no free tools to use it with MATLAB were found and that made it hard to incorporate into this thesis. Another thing would be to examine the field of Uncertainty Quantification (UQ), e.g., to get a better sense of how to analyze and simulate the uncertainty of the system, and how to analyze the reliability of the results.

Bibliography

[1] K. Åström and R. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 2010.

[2] M. M. Islam, N. Fröjd, S. Kharrazi, and B. Jacobson, "How well a single-track linear model captures the lateral dynamics of long combination vehicles," *Vehicle System Dynamics*, vol. 57, no. 12, pp. 1874–1896, 2019.

[3] J. Hebib and S. Dam, "Vehicle Dynamic Models for Virtual Testing of Autonomous Trucks," Master's thesis, Linköpings universitet, 2019.

[4] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*. Wiley, 2005.

[5] M. S. Kati, H. Köroğlu, and J. Fredriksson, "Robust lateral control of long-combination vehicles under moments of inertia and tyre cornering stiffness uncertainties," *Vehicle System Dynamics*, vol. 57, no. 12, pp. 1847–1873, 2019. [Online]. Available: https://doi.org/10.1080/00423114.2018.1552363

[6] M. D. Patil and P. Nataraj, "Automated synthesis of multivariable qft controller using interval constraint satisfaction technique," *Journal of Process Control*, vol. 22, no. 4, pp. 751 – 765, 2012.

[7] B. Palancz and L. Kovács, "Application of computer algebra for glucose-insulin control in H_2/H_∞ space using mathematica," *Periodica Polytechnica*, vol. 50, pp. 33–45, 01 2006.

[8] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*, ser. Feher/Prentice Hall Digital and. Prentice Hall, 1996.

[9] R. M. Redheffer, "On a certain linear fractional transformation," *Journal of Mathematics and Physics*, vol. 39, no. 1-4, pp. 269–286, 1960.

[10] Huaizhong Li and Minyue Fu, "A Linear Matrix Inequality approach to robust H_∞ filtering," *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2338–2350, 1997.

[11] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[12] M. Luijten, "Lateral Dynamic Behaviour of Articulated Commercial Vehicles," Master's thesis, Eindhoven University of Technology, 2010.

[13] H. Stenvall, "Driving resistance analysis of long haulage trucks at Volvo," Master's thesis, Chalmers University of Technology, 2010.

[14] K. Zhou and J. C. Doyle, "Essentials of robust control," 1997.

[15] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[16] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948 vol.4.

[17] J. Joglar-Alcubilla, “Generic tracking specifications translation from time domain to frequency domain,” *CoRR*, vol. abs/1601.00346, 2016.

[18] “Find minimum of constrained nonlinear multivariable function,” https://mathworks.com/help/optim/ug/fmincon.html, accessed: 2020-06-09.

[19] “Surrogate optimization for global minimization of time-consuming objective functions,” https://mathworks.com/help/gads/surrogateopt.html, accessed: 2020-06-09.

[20] A. Packard and J. Doyle, “The complex structured singular value,” *Automatica*, vol. 29, no. 1, pp. 71 – 109, 1993.

[21] D. Trombitas, “Roll stability control of autonomous truck combinations,” Master’s thesis, Delft University of Technology, 2019.

[22] W. Levine, *The Control Handbook*, ser. Electrical Engineering Handbook. Taylor & Francis, 1996.

A

Evaluation metrics

Note that the metrics connected to the Nyquist and Bode plot (figures A.3,A.4) are actually calculated by for all sampled systems and the worst result chosen. In these figures, only the nominal system is shown for clarity.

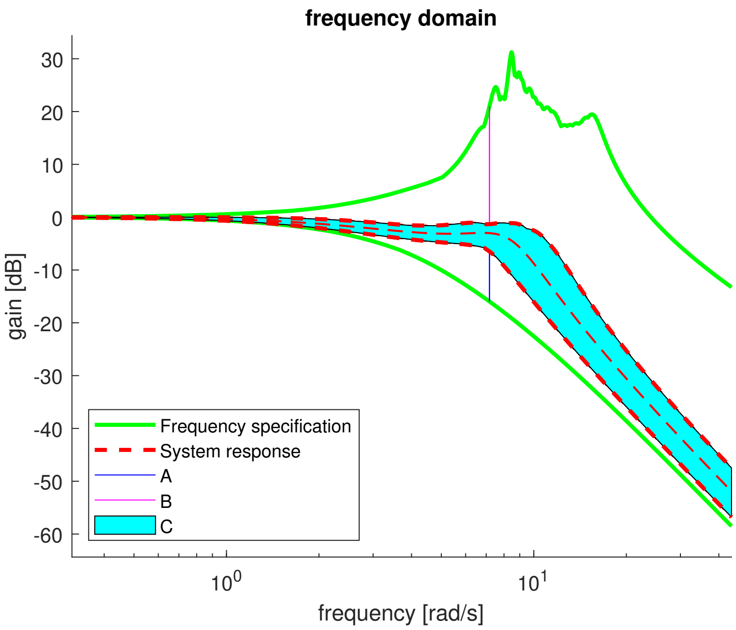


Figure A.1: Frequency domain metrics

$$\textit{Reference robustness:} \qquad (\sum_f |A|)/l_f \tag{A.1a}$$

$$\textit{Reference performance:} \qquad (\sum_f |B|)/l_f \tag{A.1b}$$

where A and B are logarithmically discretized and l_f is the number of discretization points.

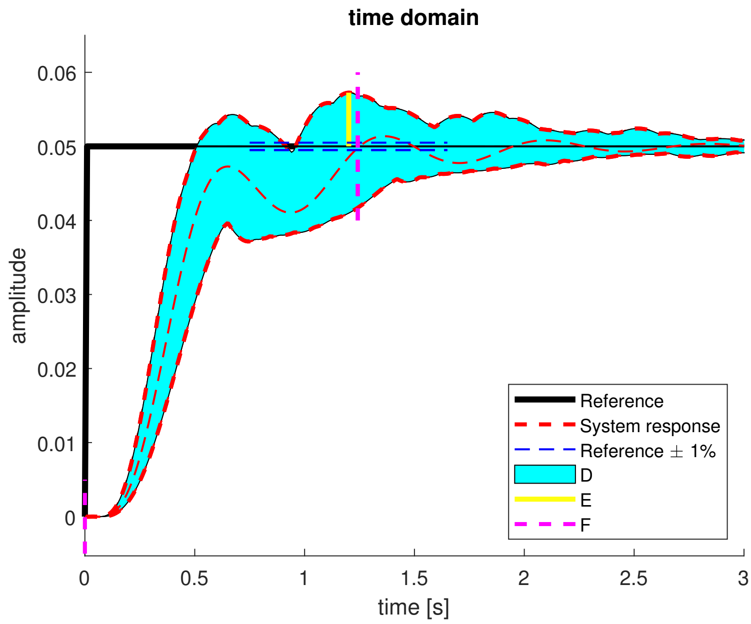


Figure A.2: Time domain metrics

$$\textit{Reference uncertainty:} \quad C + D \tag{A.2a}$$

$$\textit{Reference overshoot:} \quad \frac{A_r + E}{A_r} \tag{A.2b}$$

$$\textit{Reference reaction time:} \quad F \tag{A.2c}$$

$$\textit{Reference oscillation:} \quad \sum_t \dot{Y}_n \tag{A.2d}$$

where C is from Figure A.1, A_r is the amplitude of the reference, and Y_n is the nominal system response. Note that F is defined as the time between the start of the reference until the time where the nominal system response first is within $A_r \pm 1\%$ but not necessarily within afterwards.

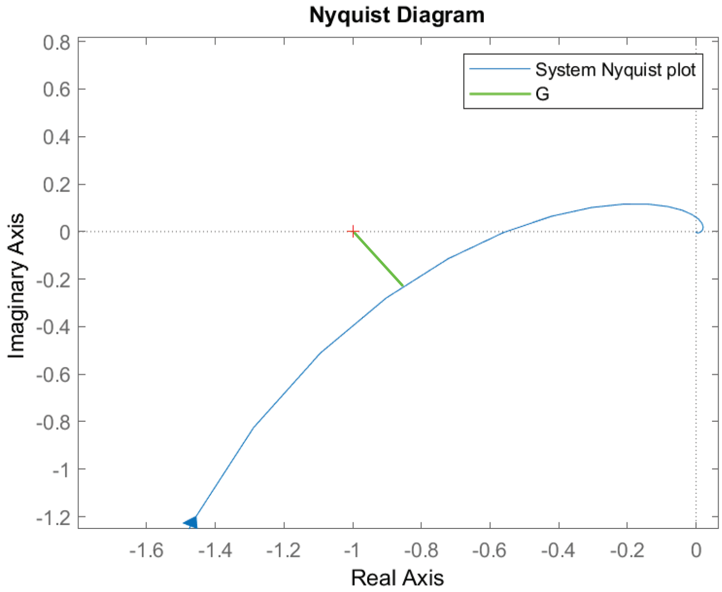


Figure A.3: Nyquist plot metric

$$\textit{Minimal Nyquist distance:} \qquad G \tag{A.3}$$

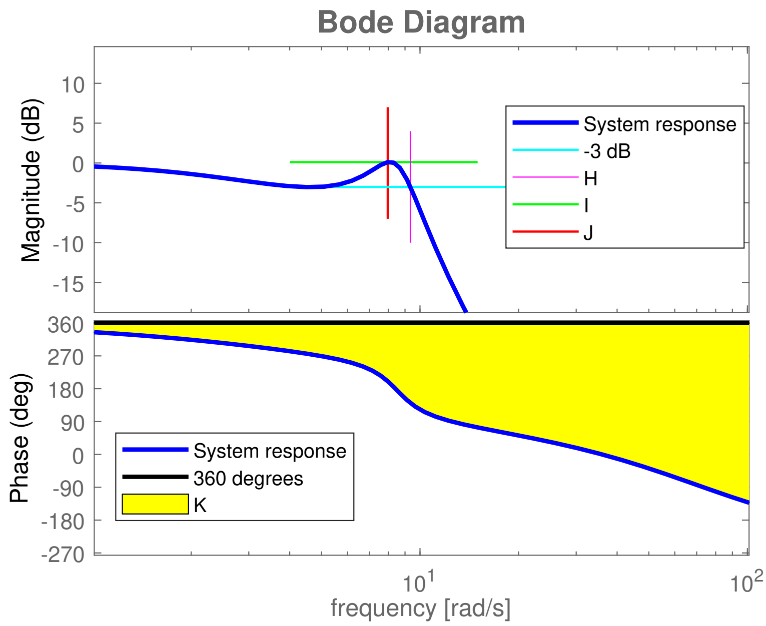


Figure A.4: Bode diagram metrics

$$\textit{Bandwidth:} \qquad H \tag{A.4a}$$

$$\textit{Critical gain:} \qquad I \tag{A.4b}$$

$$\textit{Critical frequency:} \qquad J \tag{A.4c}$$

$$\textit{Phase shift:} \qquad K \tag{A.4d}$$

Note that the bandwidth is defined at the lowest frequency where the system response has a gain of -3dB. Further, the line enclosing the area of phase shift lies at the first value higher than the phase shift of the system at low frequencies and the value should be divisible by 90. The metric *Robust stability* is calculated as the inverse of the infinity-norm of the GoF transfer functions (S_{yd}, S_{yn}, S_{ud}, and S_{un}) i.e. the inverse of their highest *Critical gain*.

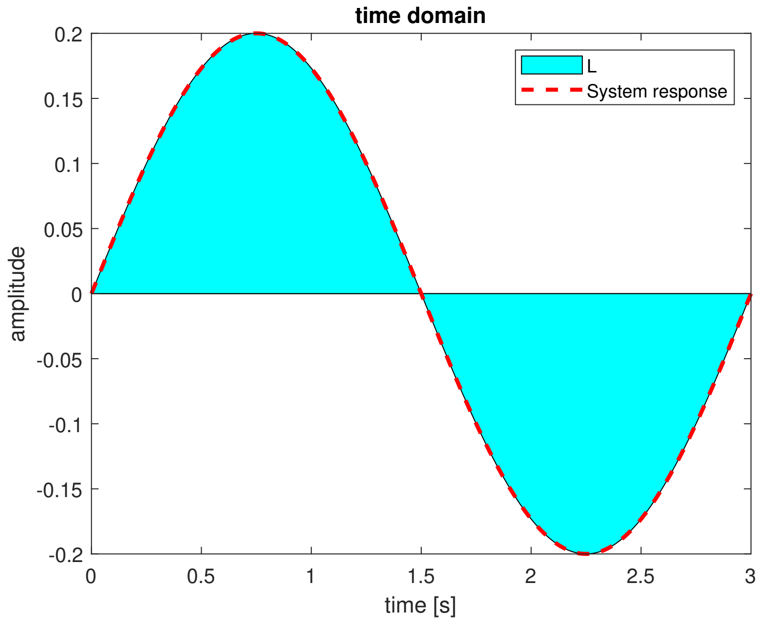


Figure A.5: Non-controlled system variable response

$$\textit{Non-controlled system variable response:} \qquad L \tag{A.5}$$

Note that the only the nominal system response was used to calculate this metric.

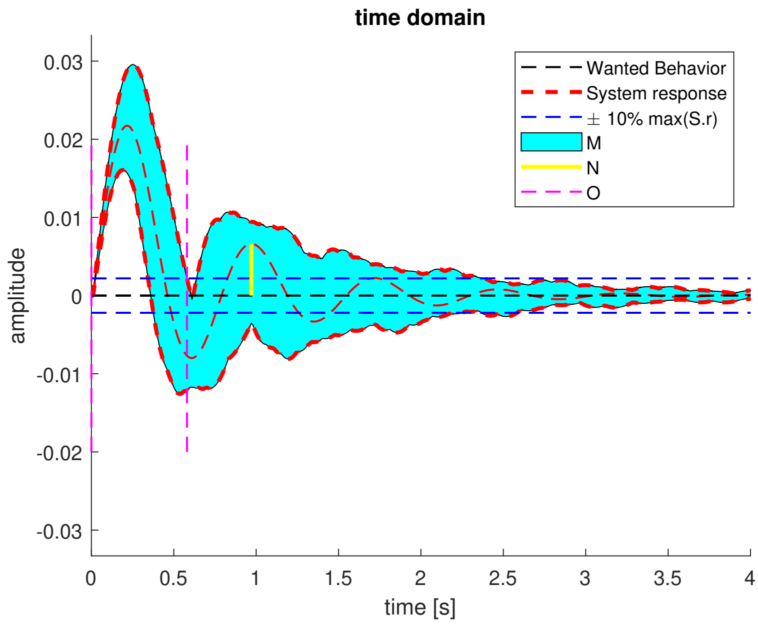


Figure A.6: Time domain: disturbance metrics

$$\textit{Disturbance uncertainty:} \qquad M \tag{A.6a}$$

$$\textit{Disturbance robustness:} \qquad (\sum_t |N|)/l_t \tag{A.6b}$$

$$\textit{Disturbance reaction time:} \qquad O \tag{A.6c}$$

where l_t is the length of the time vector. Note that O is defined as the time between that the disturbance enter the system until the time where the slowest system response first is within $(0 \pm 0.1 \cdot \max(Y_n))$ where Y_n is the nominal system response.