DEGREE PROJECT IN ELECTRICAL ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2020

Stabilization of Constant Power Loads Using Model Predictive Control

Applied to Train Propulsion Systems

MIKAEL GLAMHEDEN

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Stabilization of Constant Power Loads Using Model Predictive Control

Applied to Train Propulsion Systems

MIKAEL GLAMHEDEN

Master's Program in Systems, Control and Robotics
Date: September 3, 2020
Supervisor: Olle Trollberg & Henrik Mosskull
Examiner: Mikael Johansson
School of Electrical Engineering and Computer Science
Host company: Bombardier Transportation
Swedish title: Stabilisering av konstanta effektlaster med modellprediktiv reglering

Abstract

This thesis considers stabilization of constant power loads (CPLs) fed by a dc power source through an input filter, using model predictive control (MPC). Train propulsion systems generally utilize electrical motors whose output torque is tightly regulated by power converters. Often, these systems behave as CPLs. When a CPL is coupled with an input filter it can lead to a stability problem known as the negative impedance instability problem. Current state of the art regulators deal with this problem using classical frequency domain optimization-based controllers, such as H_∞. This thesis instead proposes a linear parameter-varying model predictive controller (LPV-MPC). This advanced control method solves the negative impedance instability problem while also being capable of explicitly addressing signal constraints, which often exist in power converter applications. The regulator is evaluated in MATLAB/Simulink as well as in a software-in-the-loop (SIL) simulator. It has furthermore been realized in a real-time hardware-in-the-loop (HIL) simulator and tested in a power laboratory. Theoretical results show improved performance over conventional H_∞ controllers, in terms of damping and control input use, under certain operating conditions where the control input is limited. The results can be used as a benchmark of theoretical performance limits for design of other regulators.

Sammanfattning

Detta examensarbete avhandlar stabilisering av konstanta effektlaster (CPL) matade med dc-effekt via ett ingångfilter, med hjälp av modellprediktiv reglering (MPC). Drivsystem i tåg använder vanligen elektriska motorer vars moment regleras hårt utav effektomriktare. Dessa system beter sig ofta som en CPL. När en CPL sammankopplas med ett ingångfilter kan det leda till ett stabilitetsproblem känt som *the negative impedance instability problem* (ung. negativ-impedans-instabilitetsproblemet). Dagens främsta regulatorer angriper detta problem genom att använda klassiska regulatorer baserade på optimering i frekvensdomän, till exempel H_∞. I detta examensarbete föreslås istället en linjär parametervarierande modellprediktiv regulator (LPV-MPC). Denna avancerade reglermetod löser stabilitetsproblemet och kan samtidigt hantera signalbegränsningar explicit. Signalbegränsningar är något som ofta finns i tillämpningar som involverar kraftomriktare. Regulatorn utvärderas i MATLAB/Simulink samt i en mjukvarusimuleringsmiljö. Regulatorn har dessutom förverkligats i en hårvarusimuleringsmiljö och testats i ett labb för kraftelektronik. Teoretiska resultat visar på förbättrad prestanda i jämförelse med konventionella H_∞-regulatorer, vad gäller dämpning och användning av styrsignal, i vissa arbetsfall när styrsignalen är begränsad. Resultaten kan användas som ett riktmärke som visar på gränser för teoretisk prestanda vid design av andra regulatorer.

Acknowledgements

I would like to express my gratitude towards the people involved in making this master thesis possible. Firstly, I would like to show my deepest gratitude towards my supervisors at Bombardier Transportation, Olle Trollberg and Henrik Mosskull, for their assistance throughout the project. Their great theoretical expertise, pedagogical aptitude, and frequent feedback has been much appreciated. I would also like to direct my gratitude to my examiner and academic supervisor Mikael Johansson without whom this thesis would not have been possible. Thank you for introducing me to the topic and recommending me to Bombardier Transportation. I also wish to thank Kent Öhrn and Bombardier Transportation for providing the thesis opportunity. Finally, I would like to thank the entire propulsion control team at Bombardier Transportation in Västerås for their interest in the project, and their assistance with the experiments.

Contents

Chapter 1

Introduction

This master thesis concerns stabilization and damping of a train propulsion system, using model predictive control (MPC). The MPC will be evaluated on systems which correspond to real train applications. In Figure 1.1 pictures of the considered train applications can be seen. Figure 1.1 (a) shows a picture of a London Central Line train whose propulsion system will mainly be considered in this thesis. Figure 1.1 (b) is a photo of a Changzhou metro train. The propulsion system of this train will be considered later in the thesis, when real-time implementation is investigated.

This chapter starts off by giving an overview of the propulsion system in electric trains. It then explains why stabilization is needed. Next, we broaden the perspective to other systems which exhibit similar behavior to what we see in train propulsion systems. From there follows an overview of different ways in which the stabilization problem has been addressed previously. We will finally explain why we think it is relevant to investigate the use of MPC; We list the objectives which were put up for this thesis, as well as the limit of scope.

1.1 Topology of Propulsion System in Electric Trains

The power to many electric trains is fed from a dc line, while the systems in the train, such as the motors, are ac power systems. Therefore, the power must be converted through a dc/ac converter, before it can be supplied to the motors and provide traction for the train. Figure 1.2 shows a computer render of a Bombardier converter module which converts dc to ac. The transistors, which

(a)

(b)

Figure 1.1: (a) Photo of London Central Line train in operation, from Bombardier Transportation [1]. (b) Photo of Changzhou metro train in operation, by SCJiang [2].

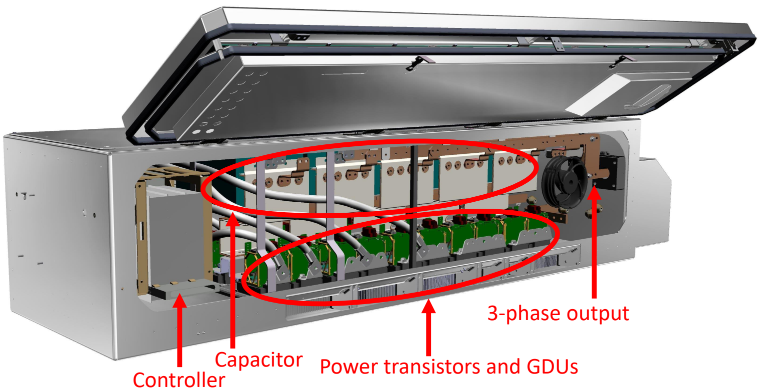


Figure 1.2: Computer render of a TC 1500 Bombardier converter module with open side cover, from Bombardier Transportation [3]. The large blue box, behind the white plates, in the top of the module, is the input capacitor. The green circuit boards are the GDUs for the transistor switches, which are the dark grey plates underneath the GDUs. The brains of the converter, the controller board, is located inside the large grey box to the left of the module. The copper bars which lead to the right are the connections for the three-phase motor load.

Figure 1.3: Photo of Bombardier permanent magnet motor, from Bombardier Transportation [4]. On the right side of the motor the connections for the three phases are visible. In the center, facing south west, is the connection to the rotor.

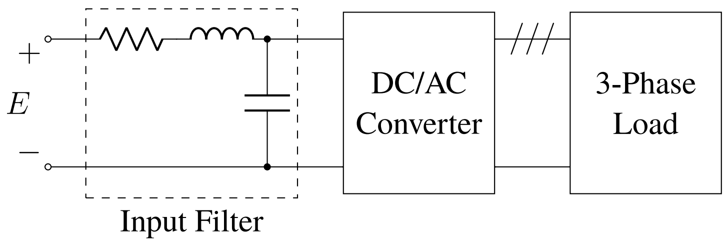


Figure 1.4: A three phase load is supplied by a dc/ac power converter. The power converter is connected to a dc supply voltage, E, via an input filter. This type of system often experiences stability issues related to the negative impedance problem at higher power loads.

do the switching in the converter, are the dark grey plates along the bottom part of the module. On top of them, green circuit boards can be seen. These are gate drive units (GDUs), which give the signals to turn off or turn on the transistors. The GDUs are controlled by a controller board which is located inside the large grey box in the leftmost part of the module. The copper bars leading to the right of the module is the three-phase output which goes to the motors. A photo of such a motor can be seen in Figure 1.3.

A simplified model of a dc fed train propulsion system can be seen in Figure 1.4, where the block labeled *3-Phase Load*, in the context of train propulsion, could be one or several ac motors. The input filter forms the connection between the dc voltage source and the converter module. It acts as a filter between the environment outside the train (the power lines) and the electronics inside. The capacitor is placed on the input of the converter in order to keep the voltage to the converter steady. The inductor is placed on the input to attenuate disturbances. The input capacitor can be seen in the picture of the converter module (Figure 1.2); it is the large blue box in the top part of the module. The line inductor of the input filter is located outside the converter module. The resistor in Figure 1.4 models the resistance which exists in the inductor winding and in the wires connecting the voltage source to the train. It is thus not an actual component in the circuit of the real propulsion system.

1.2 Problem Description

As mentioned above, the power supplied to the train motor is fed via a dc/ac power converter. The motor power is tightly regulated and is ultimately determined by the torque demanded by the train operator. The control objective of the converter is hence to provide a given motor torque, or more generally, a fixed voltage waveform on the load side, regardless of disturbances such as voltage fluctuations in the input filter.

When the control system in the converter is effective at following a given torque reference, this translates into it being effective at keeping the power throughout the converter close to the reference, and hence independent of the input voltage to the converter. This independence between the input voltage and the output power allows us to model the converter-load subsystem as a constant power load (CPL) [5] (see Figure 2.1). The term ”constant” is an unfortunate wording as the power reference certainly may vary over time, but independently from the voltage from the input filter. This is an approximation of the true system which assumes perfect torque reference following of the motor. However, we do not have perfect reference following. For example, it takes some time for the motor to respond to a change in reference. Furthermore, in reality, the output power of the converter is not completely independent of the input voltage. In many cases, however, the CPL approximation holds well, at least in a certain frequency interval.

Approximating the system as a CPL means that we model the converter-load subsystem as a non-linear component which draws current, i_d (see Figure 2.1) which is inversely proportional to the voltage across the component. For any electrical component it has to be true that the total power consumption of the component is equal to the voltage over it times the current through it. In a CPL the power appears constant, or rather independent of voltage, from the perspective of the input filter. This means that in a small signal perspective the current into the CPL is inversely proportional to the voltage over it. We know that for a resistor the relationship is the opposite; the current through the resistor is proportional to the voltage over it. The inverse relationship between the current and the voltage thus has the effect that the CPL will exhibit negative incremental resistance in the small signal perspective. This behavior, known as the negative impedance instability problem, degrades the stability margin of the feeder system [6]. The input filter is typically already poorly damped, in order to minimize power losses and to save space and cost [6, 7]. The CPL therefore tends to destabilize the input filter, especially for power draws above a certain threshold, and stabilization is required to prevent this from happening. The negative impedance problem is well known in the literature and many possible solutions for stabilizing a CPL with a poorly damped input filter exist [7, 8, 9, 10]. Common for most of them is that they modify the input impedance of the CPL actively in order to keep the closed loop system stable and well damped. Active stabilization is often achieved by introducing a control law which modifies the power draw of the CPL. By making changes to the power draw, around the reference point, the relationship between the voltage over the CPL and the current into the CPL is changed, which modifies the input impedance. With a correctly designed control law the input impedance is modified in such a way that stability is maintained even for operating conditions where the system normally would be unstable. However, this stabilization method introduces a trade-off between tracking the reference power and stabilizing the system, since stabilization introduces power disturbances which affect the reference tracking.

The method which is currently in use at Bombardier for stabilization of the input filter and converter system inside their dc-fed trains is based on H_∞ optimal control theory and uses the idea of a modification of the power draw of the CPL, as described above. As an alternative, in this thesis, we will consider how the negative impedance problem can be treated using an advanced control method known as model predictive control.

1.3 Prevalence of Negative Instability Problem

The negative impedance problem is not unique to the propulsion systems of electric trains. The same stability problem often appears in electrical systems where there are converters which one wants to regulate tightly, and hence exhibit CPL behavior. Singh et. al. [6] mentions that similar systems can be found in spacecraft, aircraft and electric vehicles, among many others. Tightly regulated converters are also prevalent in electric grids which use renewable energy sources, such as wind power and solar power plants. Typical for these systems is that the grid consists of many energy-generating components which often generate dc power. The power thus must be converted to ac before it can be connected to the larger transmission network. With the desire to decrease the size of components and to increase the efficiency of these systems comes the problem that the systems become less damped. Active stabilization of these systems therefore becomes an important task, and new insight into how that can be done efficiently has the potential to be beneficial to a variety of power converter applications.

1.4 Reason of Investigating Model Predictive Control

In most traditional control methods, as well as in H_∞ optimal control theory, there is no explicit way of defining signal limitations. This leads to unwanted behavior when limits do exist, i.e., excessive input and/or output values in the system which one want to control. Excessive inputs are often handled with a simple saturation of the signal, but excessive outputs are difficult to handle if the constraints are not explicitly accounted for in the control problem. Therefore it is common that systems are detuned and operated with a significant margin to the constraints, usually with a cost in terms of lost performance or efficiency. Using MPC, which can take both system input and system output constraints into consideration, such margins can be made smaller and hence one gets performance improvements. In the context of train propulsion systems there are output constraints in terms of current ratings on the motors and the electronics. There are also input power constraints for the same reasons. Another reason for limiting power modification is that torque jerks stresses the mechanical systems in the train and can also affect passenger comfort experience. Being able to handle these constraints properly could mean that the propulsion system would be more energy efficient, while also being a more comfortable experience for the passenger.

For this reason, this master thesis has investigated the use of MPC for stabilization and damping of CPL with input filter. MPC is a control method which solves an optimization problem online. In MPC it is possible to directly include signal limitations into the optimization problem. The method has been used for a long time in industries which deal with systems with slow dynamics, such as chemical plants. In more recent years however, with advancements in computational power and with the maturity of the theory of MPC itself, it has proven useful in other applications as well, such as the area of power electronics. In this master thesis an MPC regulator has been designed for stabilization and damping of the input filter and CPL system. It has been implemented and tested on the motor control systems at Bombardier Transportation in Västerås, Sweden. The performance of the controller has been evaluated against performance measures which are presented in Chapter 2.5. These are the same performance measures which were used in [7] to evaluate the controller which Bombardier already uses. In the next chapter we give a more detailed background to the stabilization problem, as well as the theory behind MPC and the optimal control theory framework.

1.5 Objectives

We will here list the main objectives of the thesis, which are to

- investigate the feasibility of MPC for stabilization of a CPL connected to a dc power source via an input filter.
- investigate the impact of explicit knowledge of control constraints on performance when such control constraints are present
- implement the MPC regulator in a real-time environment to show its feasibility and study its usefulness in control of power converters.

1.6 Limit of Scope

The scope of the thesis has been limited to

- studying solutions where the MPC optimization problem is solved online. See Chapter 2.9.3 for an alternative.
- studying the CPL model for stabilization of converter systems. Modelling of extra system dynamics have not been considered.
- studying MPC regulators with linear system models and quadratic cost functions.

Chapter 2

Background

This chapter begins by mathematically modeling the system which was introduced in Chapter 1. The stability problems which exist with the system are then explained in detail. This is done in Section 2.2 and 2.3. After that, in Section 2.4, a number of ways to address the instability problem are described. In Section 2.5 we introduce performance specifications which are relevant when evaluating how good a solution to the problem is. In Section 2.6 an introduction to frequency domain optimal control is given in order to explain the state of the art regulator from Section 2.7. Then in Section 2.8 follows a section describing how stabilization can be done using a Linear Quadratic Regulator (LQR). This will be used to connect the frequency domain methods to time domain stabilization methods. After that, in Section 2.9, comes an introduction to MPC. We will see how it relates to LQR and there will also be an overview of a few variants of MPC which are relevant to power systems applications.

2.1 Notation

We use A^H to denote the Hermitian transpose of the matrix A where Hermitian transpose is defined by

$$A^H = (\bar{A})^\top = (\bar{A^\top}) \tag{2.1}$$

$\bar{A}$ is the matrix where each entry of A is replaced by its complex conjugate. We use $tr(M)$ to denote the trace of the matrix M, which is the sum of the diagonal elements of M. We use $A \succ 0$ to denote that A is positive definite. Similarly, $A \succeq$ means that A is positive semi-definite.

$E[\cdot]$ is used as the expectation operator.

In discrete-time systems we use $x(k)$ as a shorthand for $x(kT_s)$, that is, the value of the continuous-time signal $x(t)$ at sample k, sampled with sample period T_s.

2.2 Modeling of Converter and Load

As introduced in Chapter 1, the propulsion system of an electric train can be modeled as CPL connected via an input filter to a dc power source. Assuming an RLC filter as input filter the system can be modeled according to Figure 2.1 where the converter and the load from Figure 1.4 has been combined into a CPL. The dynamics of the RLC filter are described by the following differential equations

$$\begin{aligned} \frac{di}{dt} &= \frac{1}{L}\left(-Ri - U_d + E\right) \\ \frac{dU_d}{dt} &= \frac{1}{C}(i - i_d) \end{aligned} \tag{2.2}$$

where L and R model the resistance and the inductance of the physical RLC filter (a known impedance) as well as the impedance of the dc line. Electric trains are powered through overhead lines, or from a third rail. This means that as the train moves, the distance to the voltage source changes, and so also the impedance described by R and L. The impedance may therefore vary in quite a large range and any stabilization method must be robust towards changes in these parameters. The capacitance C is the capacitance on the input of the converter. The states are the current through the inductor i, and the voltage over the capacitance, U_d. The voltage E is the line voltage from the dc feeder line, and i_d is the current into the CPL.

The CPL is described by the following relation

$$i_d(t) = \frac{P(t)}{U_d(t)} \tag{2.3}$$

where $P(t)$ is the total power draw from the load. This means, given a constant power P, the current i_d, drawn by the load, will have to increase/decrease to accommodate for a decrease/increase in U_d. Thus, the load acts like a negative impedance, in a small signal perspective. This is further seen if (2.3) is linearized in terms of P and U_d

$$\Delta i_d(t) = \underbrace{(-\frac{P_0}{U_{d0}^2})}_{Y_{CPL}} \Delta U_d(t) + \frac{1}{U_{d0}} \Delta P(t) \tag{2.4}$$

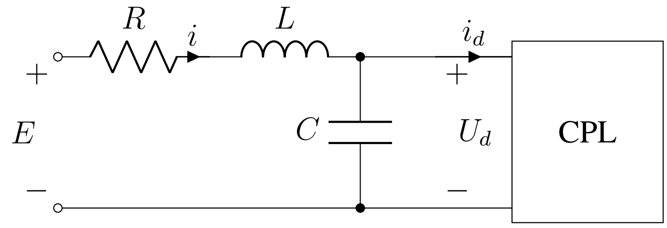


Figure 2.1: RLC input filter connected to a CPL.

where U_{d0} and P_0 is the input filter voltage and the total power draw of the CPL at the equilibrium, and $\Delta U_d(t) = U_d(t) - U_{d0}$, $\Delta P(t) = P(t) - P_0$ denote deviations from the equilibrium. We can now clearly see the small signal negative impedance relation between U_d and i_d from the fact that the admittance Y_{CPL} is negative for operating points where $P_0 > 0$.

2.3 The Negative Impedance Stability Problem

Because of the negative impedance effect, there exists a power load when the CPL destabilizes the closed loop system formed by the CPL and the RLC filter. This power limit can be determined analytically. From basic control theory it is known that the stability of a system can be studied by means of the poles of the system's transfer functions. If any pole is located in the the right half of the complex plane the system will be unstable.

Let us begin by writing the transfer functions from the excitation signals E and P to the states U_d and i. Let us begin with U_d. Laplace transformation of (2.2) and elimination of i yields

$$U_d(s) = -\underbrace{\frac{(Ls+R)\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}}_{Z_{DC}} i_d(s) + \underbrace{\frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}}_{G_E} E(s) \tag{2.5}$$

where the natural frequency ω_0 and the damping factor ζ are defined as

$$\omega_0 := \frac{1}{\sqrt{LC}}, \qquad \zeta := \frac{R}{2}\sqrt{\frac{C}{L}} \tag{2.6}$$

The transfer function Z_{DC} is the output impedance of the input filter and G_E is the filter transfer function from the input side. From now on let U_d, i, E and

P denote deviations from an equilibrium which the equation (2.3) has been linearized around. Using the linearized relationship (2.4) in (2.5) we can then write U_d in terms of E and P

$$\begin{aligned} U_d =& Z_{DC}(Y_{CPL}U_d + \frac{1}{U_0}P) + G_E E(s) \\ \Rightarrow U_d =& \frac{G_E}{1+Z_{DC}Y_{CPL}}E - \frac{Z_{DC}}{1+Z_{DC}Y_{CPL}}\frac{P}{U_{d0}} \end{aligned} \tag{2.7}$$

Similarly, we can write the current i, in Laplace domain, as a function of E and P

$$\begin{aligned} i &= Y_{in}E + G_E i_d = Y_{in}E + G_E(G_E E - U_d) = \\ &= \left(Y_{in} + \frac{G_e^2}{Z_{DC}}\right)E - \frac{G_E^2}{Z_{DC}}\left(\frac{G_E}{1+Z_{DC}Y_{CPL}}E - \frac{Z_{DC}}{1+Z_{DC}Y_{CPL}}\frac{P}{U_{d0}}\right) = \\ &= \left(Y_{in} + \frac{G_E^2}{Z_{DC}}\left(1 - \frac{1}{1+Z_{DC}Y_{CPL}}\right)\right)E + \frac{G_E}{1+Z_{DC}Y_{CPL}}\frac{P}{U_{d0}} \end{aligned} \tag{2.8}$$

where the transfer function, Y_{in}, from line voltage E to line current i is given by

$$Y_{in} = \frac{s/L}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{2.9}$$

If we study the transfer functions G_E, Z_{DC} in (2.5) and Y_{in} from (2.9) we see that they all have the same denominator and thus the same poles. The poles are given by

$$s = -\zeta\omega_0 \pm \sqrt{(\zeta\omega_0)^2 - \omega_0^2} = -\zeta\omega_0 \pm \omega_0\sqrt{\zeta^2 - 1} \tag{2.10}$$

For all physical RLC filters R, L and C are positive, and hence also ω_0 and ζ. Therefore, the roots of 2.10 must have negative real part and be located in the left half of the complex plane. Thus, all of these transfer functions are stable. Z_{DC} appears in the denominator of (2.8), but as can be seen in (2.5) its zero is in the left half plane and cannot lead to instability. Instability of (2.7) and (2.8) can hence only originate from the factor $1/(1+Z_{DC}Y_{CPL})$ which appears when we connect the CPL to input filter. The stability limit is determined by the poles of

$$\frac{1}{1+Z_{DC}Y_{CPL}} = \frac{s^2 + 2\zeta\omega_0 s + \omega_0^2}{s^2 + \left(2\zeta\omega_0 - \omega_0^2 L\frac{P_0}{U_{d0}^2}\right)s + \omega_0^2\left(1 - R\frac{P_0}{U_{d0}^2}\right)} \tag{2.11}$$

where instability occurs if at least one of the poles of (2.11) is in the right half of the complex plane. We will now study (2.11) to see when instability occurs. Let us define the following two expressions

$$\omega_s := \omega_0 \sqrt{1 - R\frac{P_0}{U_{d0}^2}} \tag{2.12}$$

$$\zeta_s := \frac{\zeta}{\sqrt{1 - R\frac{P_0}{U_{d0}^2}}} \left(1 - \frac{\omega_0 L}{2\zeta}\frac{P_0}{U_{d0}^2}\right) \tag{2.13}$$

We can then write the denominator of (2.11) as

$$s^2 + 2\zeta_s\omega_s s + \omega_s^2 \tag{2.14}$$

The polynomial (2.14) will have roots in the right half plane if the damping factor ζ_s is negative, which happens when

$$\frac{\omega_0 L}{2\zeta}\frac{P_0}{U_{d0}^2} \geq 1 \tag{2.15}$$

which can also be written as

$$P_0 \geq \frac{2\zeta}{\omega_0 L} U_{d0}^2 = \frac{RC}{L} U_{d0}^2 \tag{2.16}$$

This relation sets a limit on the possible power draw from the system. It is fully determined by the filter parameters R, L and C, as well as the input filter voltage U_d. We can write the equality relation for the limit as

$$P_{lim} = \frac{RC}{L} U_{d0}^2 \tag{2.17}$$

where P_{lim} denotes the power limit for the given input filter, at the operating point U_{d0}. The system is stable for all power draws smaller than P_{lim} and unstable for all power draws larger than P_{lim}.

2.3.1 Stability in terms of Nyquist Stability Criterion

As we have already seen, stability is determined by the poles of (2.11). According to the Nyquist stability criterion we can equivalently study the zeros the loop gain, namely

$$L_{DC} = Y_{CPL} Z_{DC} \tag{2.18}$$

The simplified Nyquist stability criterion states that the Nyquist diagram of the loop gain L_{DC} cannot encircle the point $(-1, 0)$ in the complex plane. If it

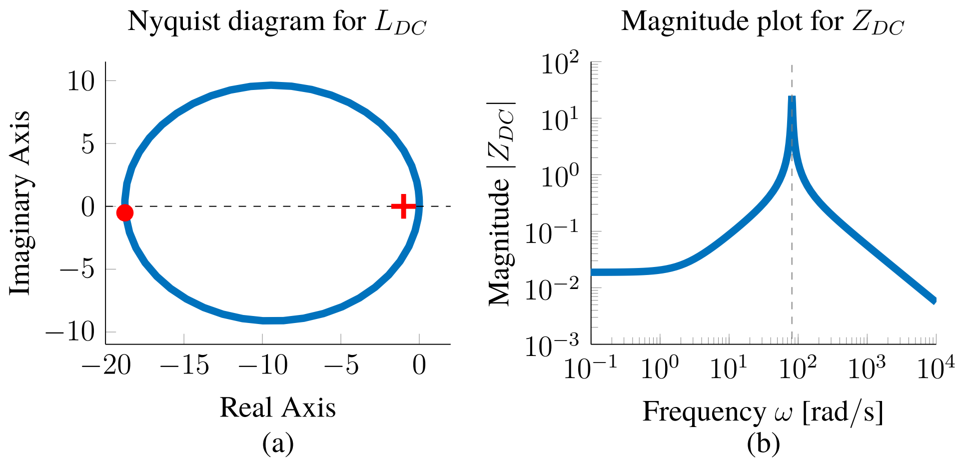


Figure 2.2: (a) Nyquist diagram for the loop gain L_{DC} for the CLT application, without any stabilization. The operating voltage , U_{d0}, is 630 V and the operating power, P_0, is 300 kW. The value of the natural frequency in the diagram is marked in red. The point $(-1, 0)$ is also marked with a red cross. (b) Magnitude plot of the output impedance, Z_{DC}, of the input filter. The natural frequency of the filter is marked with a vertical gray line.

does encircle the point $(-1, 0)$ the system is unstable. The output impedance of the input filter, described by the transfer function Z_{DC} is a stable transfer function, as has already been noted. Since the input filter is poorly damped it does however have a large resonance peak. This can be seen in Figure 2.2 (b). When multiplied with the admittance Y_{CPL} it is phase shifted 180°, if $P_0 > 0$. This can cause instability, if the magnitude of Y_{CPL} is too large. In Figure 2.2 (a) the Nyquist diagram of (2.18) has been plotted. We see that the point $(-1, 0)$ is encircled, and the system is unstable. To stabilize the system can be interpreted as rotating the Nyquist curve of the loop gain so that it no longer encircles the point $(-1, 0)$. This is done by modifying the admittance Y_{CPL}.

2.3.2 Verifying the Theoretical Stability Limit

To verify how equality (2.17) holds we can simulate the response of the system to a small step (here 1 V) in line voltage E while operating at different power draws. The parameter values in the simulations are according to Table 5.2. With these parameters (2.17) says that the stability limit is 44.5 kW. In Figure 2.3 we can see the step response of the system for three different power draws. First 49 kW which is 10 % above the limit, then 44.5 kW which is equal to

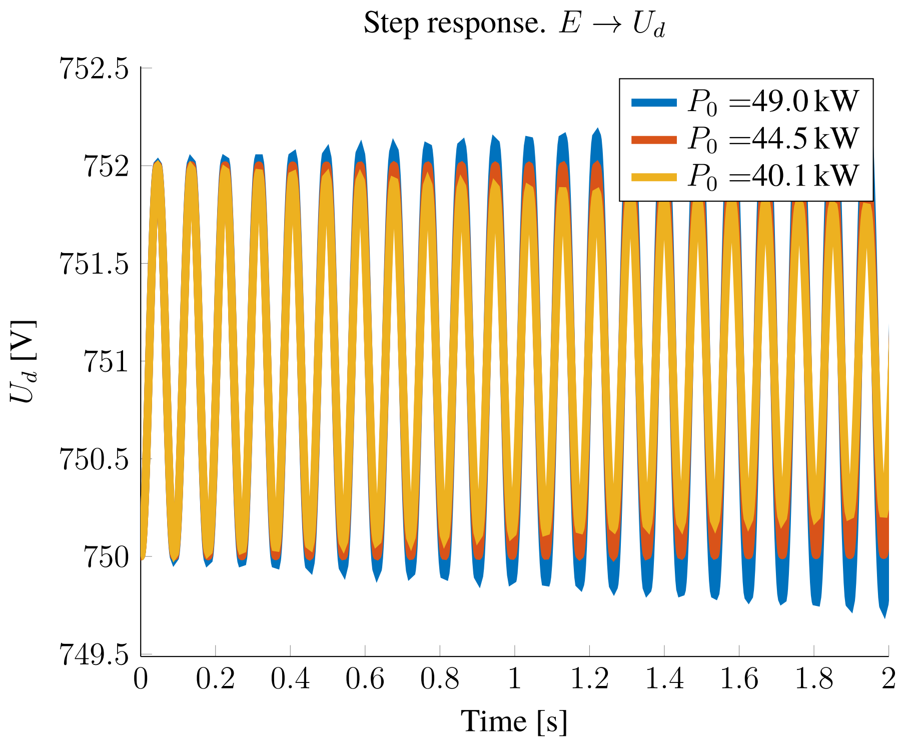


Figure 2.3: Step response of a CPL connected to an input filter without stabilization for three different operating points.

the limit, and finally 40.1 kW which is 10 % below the limit. The first step response shows an increasing amplitude, the second a constant amplitude and the third a decreasing amplitude, indicating that the system is unstable in the first case, conditionally unstable in the second case and stable in the third case. The result shows that the inequality (2.17) matches what we see in simulation.

Since the converter is meant to be operated at powers which might be 10 times higher than the stability limit, we realize that stabilization is necessary.

In the next section we will present some ways in which stability can be achieved.

2.4 Stabilization Strategies

There are multiple ways in which to compensate for the negative impedance effects of the CPL in order to stabilize the feeder system so that it can be used for power loads above its natural power limit. When the feeder side of the system consists of passive components, as is the case with the system under consideration, stabilization can generally be achieved in three different ways: (1) Using passive damping, where passive components are added to the circuit; or through active damping, where one can either achieve the damping by (2) adding an active component in parallel with the CPL - an auxiliary circuit - or (3) modify the CPL input impedance directly. These methods will be reviewed in the following subsections.

2.4.1 Passive Damping

One way to compensate for negative impedance effects caused by the CPL is to increase the damping of the system by modifying the input filter. This can be done by modifying the existing filter components or by adding new ones. Resistances, capacitors and/or inductors are added in parallel to either the inductor or the capacitor in the input filter. Figure 2.4 shows examples of how these components can be placed. Adding passive components this way leads to increased cost, weight and size of the system [6]. It also increases power losses, which is often undesirable. The input filter is usually made poorly damped in the first place to minimize such power losses. Such is the case in the train application considered in this thesis. There are implementations of loss free resistance, as mentioned in [6], which decreases the power losses. However, these components still increase the cost and complexity of the system.

2.4.2 Active Damping

Active damping creates the effect of parallel passive components by modifying the control structure in the active components of the system. However, since the control laws are implemented in software, the structure is not limited to that which can be realized with passive components. In the case where the feeder consists of passive components, this means modifications of the control structure of the load converter. It is also possible to add an auxiliary circuit in parallel with the load subsystem to dampen the system.

Active Damping Using Auxiliary Circuit

When stabilizing with an auxiliary, a new circuit, usually a dc/dc converter, is added in parallel with the load subsystem [5]. See Figure 2.5. This auxiliary circuit will actively modify its input impedance by changing how much current, i_a, it draws in order to stabilize the system. Meanwhile the load converter may work independently. The advantage of this method is that it leaves

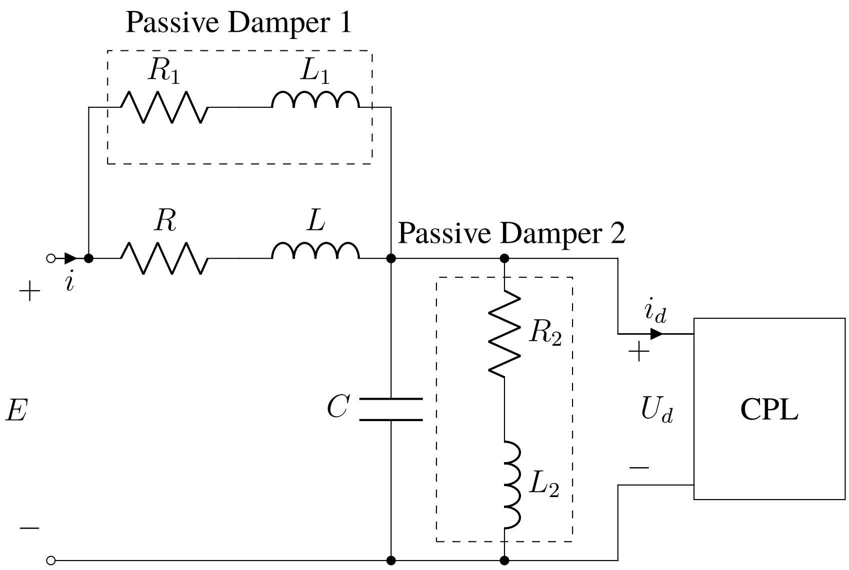


Figure 2.4: Circuit diagram of the input filter with added passive dampers.

the feeder and the load systems untouched. It also decouples stabilization from reference following, which is the overall objective of the load converter. However, like the passive damping approach it leads to increased cost and hardware complexity.

Active Damping Using Load Converter Circuit

This method achieves damping by directly modifying the power draw of the load converter in order to modify the input impedance of the CPL. This has an advantage in that it doesn't add any new components to the system. It therefore doesn't lead to increased cost and complexity in terms of components, in contrast to the previously presented methods. It does however lead to increased complexity of the software which handles the switching of the load converter. Another drawback is that the stability control loop can interfere with the overall control objective of making the power to the load follow the reference. By implementing active damping with the load converter, we get conflicting control objectives. On the one hand we want CPL behavior, which comes with perfect reference following. On the other hand, we need stability. Since the active damping modifies the power draw, we can no longer deliver perfect CPL behavior. We hence introduce a trade-off which does not exist with the other solutions [5, 6]. In the next section we will elaborate more on these trade-offs.

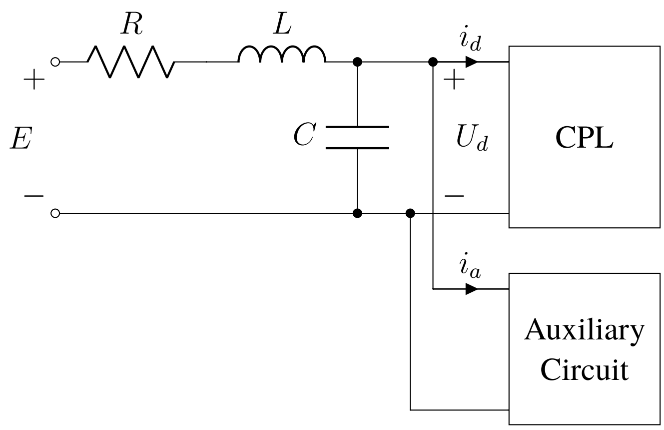


Figure 2.5: CPL with input filter, damped using auxiliary circuit.

2.5 Performance Specifications

We have already established that the main objective of any damping method is to stabilize the input filter. Beyond this main objective there are other performance specifications which we want to fulfill. In Mosskull's article on CPL stabilization [7] three performance specifications are introduced, which will be used in this thesis as well. In particular, we would like any good damping method to

1. give well damped system dynamics
2. limit interference with the overall control objective of following the power reference
3. be robust towards modelling errors

The first specification represents the idea that we want not only a stable system, but a system which is well damped. A well damped system is desirable in a train application since the system will shut down in case of overvoltage or undervoltage. Too high voltages can damage components and low voltages results in high currents which also may damage components. It is furthermore undesirable to have voltage ripple in the input filter since it can leak back onto the dc feeder line where it can interfere with different signaling systems.

Secondly, we do not want the stabilizing power modification to interfere too much with the objective of following the power reference set by the train operator. For this reason, we want the stabilizing power to be zero in steady state. Let $P_{stab}(t)$ be the stabilizing power modification. We want

$$\lim_{t\to\infty} P_{stab}(t) \to 0 \tag{2.19}$$

Furthermore, we want to minimize the magnitude of the power modification, $|P_{stab}|$, as long as the other specifications are fulfilled.

Finally, we want the damping to be robust towards modelling errors, meaning that stability should be maintained for variations of the passive components, R, L and C, within a certain tolerance level. Furthermore, as mentioned in Chapter 1 the CPL is an approximation of the true converter-load system dynamics. Even though we design the controller for a CPL, the real system is not the same. The controller should therefore also be robust towards non-ideal CPL dynamics, so that stability is maintained if the true converter-load system doesn't behave exactly like a CPL, i.e we do not exactly have

$$i_d(t) = \frac{P(t)}{U_d(t)}$$

2.6 Introduction to Frequency Domain Optimal Control

The controllers which are currently in use for stabilization of the input voltage to the converters in the propulsion systems of the trains at Bombardier Transportation are based on optimization in frequency domain. It is an H_∞ type of controller, which is derived by minimizing the ∞-norm of a certain function. This section introduces the theory behind H_∞, as well as H_2, optimal control.

Consider the problem of designing a output feedback controller $K(s)$ for the system $G(s)$, according to Figure 2.6. The system $G(s)$ has input u and output y which may be vectors, meaning $G(s)$ may be a multiple input-multiple output (MIMO) system. We then want to design the controller $K(s)$ such that certain criteria are met. These design criteria could be disturbance attenuation, noise attenuation, faster system dynamics, reference following, etc. H_∞ and H_2 optimal control provides a framework for finding $K(s)$ which satisfies the given conditions in an optimal sense (if such a $K(s)$ exists). In the next two subsections we will explain what the H_2 norm and the H_∞ are, before returning to the question of finding the optimal controller.

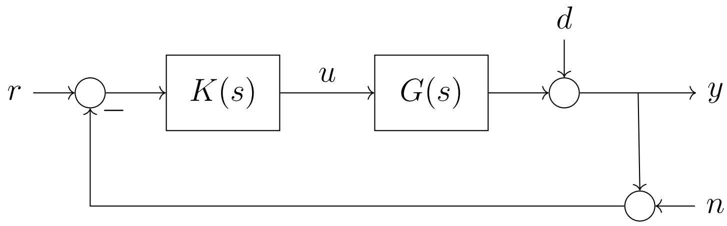


Figure 2.6: System $G(s)$ with negative feedback controller $K(s)$.

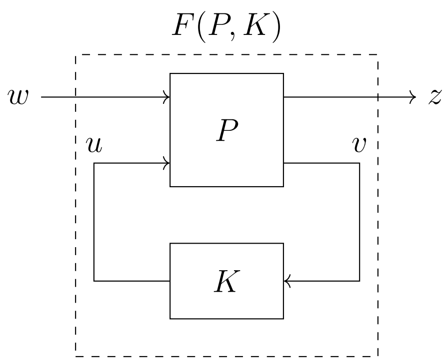


Figure 2.7: General system model formulation for controller optimization.

2.6.1 H_2 Norm Minimization

Assume that we have a transfer function $F(s)$, which for example could describe the transfer function from a disturbance signal to the output, or some other relation which one would like to minimize. In H_2 norm minimization, the following system norm is used to define the norm of the dynamical system $F(s)$

$$||F(s)||_2 = \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty}\sum_i \sigma_i^2(F(j\omega))d\omega} \tag{2.20}$$

where $\sigma_i(F(j\omega))$ is the ith singular value of $F(j\omega)$ (see Appendix A.1 for information on singular values). The H_2 norm of $F(s)$ is thus the sum of the singular values of $F(s)$ evaluated at $s = j\omega$, integrated over all frequencies. As noted by Skogestad and Postlethwaite in [11, Chapter 4.10], the H_2 norm is only defined for strictly proper dynamical systems, meaning that $F(j\omega) \to 0$ as $\omega \to \infty$. This is true for all physical systems.

The singular values of $F(j\omega)$ gives us information of the gain of $F(j\omega)$ at the frequency ω. In particular, for a matrix M, each singular value, $\sigma_i(M)$, corresponds to the gain of M in the input direction of v_i, where v_i are the orthonormal input vectors of the singular value decomposition (SVD) of M. That is

$$\sigma_i(M) = ||Mv_i||_2 = \frac{||Mv_i||_2}{||v_i||_2} \tag{2.21}$$

When we minimize the H_2 norm of a transfer function $F(s)$ we minimize the sum of squares of the singular values, across all frequencies. We are in a sense minimizing the gain of $F(s)$ for all possible input directions across all frequencies. The average behavior means that this type of controller sometimes generate high peaks at certain narrow frequency ranges if it means it can lower the gain at other frequencies sufficiently.

2.6.2 H_∞ Norm Minimization

In H_∞ norm minimization, the following system norm is used

$$||F(s)||_\infty := \max_\omega \bar{\sigma}(G(j\omega)) \tag{2.22}$$

where $\bar{\sigma}(F(j\omega))$ is the largest singular value of $F(j\omega)$. The H_∞ norm is the largest gain of the system $F(s)$, considered across all frequencies ω. It is in some sense the peak of the magnitude of the transfer function $F(s)$.

To minimize the H_∞ norm of the transfer function $F(s)$ therefore corresponds to minimizing the peak of the largest singular value. Hence, we see the difference between H_2 and H_∞. In H_∞ we only consider the worst case and try to improve that. In H_2 we consider the gain in all input directions and for all frequencies and try to minimize some sort of average. If $F(s)$ is a sensitivity function, describing the relationship between disturbance and output, H_∞ optimization will make sure disturbances are attenuated in the direction in which they have the worst impact. H_2 on the other hand, will consider the gain in all possible directions and attenuate disturbances in the directions which have the best average improvement.

2.6.3 Defining the Optimization Problem

With an understanding of what H_2- and H_∞-norm minimization are we will now return to the task of finding the optimal controller $K(s)$ for the system $G(s)$. In [11, Chapter 9], Skogestad and Postlethwaite present a framework for reformulating a feedback system like the one in Figure 2.6 onto the form shown in Figure 2.7, which we can apply well known optimization algorithms on, available for example in MATLAB's Robust Control Toolbox. In [11], more advanced control structures, which for example has higher degrees of freedom or take model uncertainty into account are considered as well. For the regulators which will be considered in this thesis, however, this simplified structure suffices.

In Figure 2.7 the signal w represents what [11] calls exogenous inputs, which could be reference signals, disturbances or noise. The signal z are the exogenous outputs; signals which we want to minimize such as the control error or the control inputs. The signal u are the control inputs, and v are the sensed outputs - the inputs to the controller. The plant P is then the MIMO transfer function from $\begin{bmatrix} w & u \end{bmatrix}^\top$ to $\begin{bmatrix} z & v \end{bmatrix}^\top$. That is

$$\begin{bmatrix} z \\ v \end{bmatrix} = P \begin{bmatrix} w \\ u \end{bmatrix} \tag{2.23}$$

Often P is partition as

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \tag{2.24}$$

so that we can write the relation between the inputs and the outputs for each output signal separately, in the following manner

$$\begin{aligned} z &= P_{11}w + P_{12}u \\ v &= P_{21}w + P_{22}u \end{aligned} \tag{2.25}$$

In P we may include weight functions on the exogenous inputs or the exogenous outputs, which allow us to penalize a signal differently in different frequency intervals. This allows is to shape the resulting controller $K(s)$, and in turn the qualities of the closed loop system.

The closed loop system $F(P,K)$ (see Figure 2.7) is a function of the plant P - which in turn is a function of the weights - and the controller K. The closed loop system can be written in terms of the elements of the partitioned P -system in the following way

$$F(P,K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \tag{2.26}$$

The optimization problem then consists of finding the controller K such that $F(P,K)$ is minimized with respect to some system norm; either H_2 or H_∞. If we optimize with respect to H_2 we want to minimize (2.20) with $F(P,K)$ as the argument, and similarly if we want to optimize with respect to H_∞ we minimize (2.22) with $F(P,K)$ as the argument.

Algorithms for minimizing $||F(P,K)||_2$ and $||F(P,K)||_\infty$ have been given by [12]. For the H_2 case there exists a unique solution and the optimal controller $K(s)$ is given by the solution to two algebraic Riccati equations. For the case of H_∞ it is possible to find all stabilizing controllers $K(s)$ which satisfy

$$||F(P,K)||_\infty < \gamma \tag{2.27}$$

for some $\gamma > \gamma_{min}$, where γ_{min} is the minimum value of $||F(P,K)||_\infty$. One can then reduce γ iteratively, using for example the bisection method, until a solution which is sufficiently close to the minimum has been found. The method for finding a controller $K(s)$ such that (2.27) is satisfied involves finding the solution to two algebraic Riccati equations, similar to what is done in the H_2 norm case.

2.6.4 Relating H_2 to Time Domain

The problem of minimizing the H_2 norm of the system $F(P,K)$ can be related to minimization of a function in time domain, through Parseval's theorem. If we assume that the exogenous input $w(t)$ is white noise with unit intensity, the expected power in the output signal $z(t)$ is [11]

$$E\left[\lim_{T\to\infty}\frac{1}{T}\int_0^T z(t)^\top z(t)dt\right] \tag{2.28}$$
$$= \text{tr } E\left[z(t)z(t)^\top\right] \tag{2.29}$$
$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} \text{tr}[F(j\omega)F(j\omega)^H]d\omega \tag{2.30}$$
$$= ||F(P,K)||_2^2 \tag{2.31}$$

Here we have used a different definition of H_2 norm, based on the trace of a matrix, which is equivalent to (2.20). Between the second and the third equality we have used Parseval's theorem which relates the power of a signal in frequency domain to the power in time domain. Thus we see that, minimizing (2.28) is equivalent to minimizing $||F(P,K)||_2$.

2.7 State of the Art: An H_∞ Controller

As mentioned in Section 2.6, the controllers which are used at Bombardier for stabilization of the input filter in the train propulsion system have been obtained using H_∞ optimization. In this section we present what those controllers are, and how they are derived. The derivations come from Mosskull in [7], where explicit solutions to the controller which stabilizes the input filter are found, both for H_2- and H_∞-norm optimization. This is done by letting the control signal be a deviation from the desired power reference. The closed loop transfer function from the reference power and the line voltage (exogenous inputs) to the control signal (exogenous output) are then found. The optimal controller is the controller which minimizes this transfer function.

2.7.1 The Optimization Problem for Negative Impedance Problem

Consider again the model derived in Section 2.2, i.e., the model where we considered a CPL connected to a dc-source via an input filter consisting of an RLC filter. In Section 2.3 we showed that it became unstable for power draws above a certain level, depending on the parameters of the input filter. Here we will derive a stabilizing controller using the methods of optimal control which were introduced in Section 2.6.

Consider now a small power deviation P_{stab} from the desired power draw, P_{CPL}, such that the total power-draw from the CPL is

$$P(t) = P_{CPL}(t) + P_{stab}(t) \tag{2.32}$$

The power deviation P_{stab} will be the control signal which is modified to keep stability. To express how the input filter and the CPL interacts with each other we will now express the linearized current (2.4) with the addition of the power deviation P_{stab}. It becomes

$$\Delta i_d(t) = \underbrace{(-\frac{P_0}{U_{d0}^2})}_{Y_{CPL}} \Delta U_d(t) + \frac{1}{U_{d0}} \Delta P_{CPL}(t) + \underbrace{(-\frac{P_{stab0}}{U_{d0}^2})}_{Y_{stab0}} \Delta U_d(t) + \frac{1}{U_{d0}} P_{stab}(t) \tag{2.33}$$

where $Y_{stab0} = 0$ typically, since a steady state power modification is undesirable. Just as in Section 2.3, let E, P_{CPL}, i and U_d denote deviations from an equilibrium which (2.33) has been linearized around. We can write the transfer functions from E, P_{CPL} and P_{stab} to the states i and U_d and get

$$\begin{bmatrix} i \\ U_d \end{bmatrix} = \begin{bmatrix} \left(Y_{in} + \frac{G_E^2}{Z_{DC}}\left(1 - \frac{1}{1+Z_{DC}Y_{CPL}}\right)\right) & \frac{G_E}{1+Z_{DC}Y_{CPL}} \\ \frac{G_E}{1+Z_{DC}Y_{CPL}} & -\frac{Z_{DC}}{1+Z_{DC}Y_{CPL}} \end{bmatrix} \begin{bmatrix} E \\ \frac{(P_{CPL}+P_{stab})}{U_{d0}} \end{bmatrix} \tag{2.34}$$

where we assumed $Y_{stab0} = 0$ according to the rationale above. G_E and Z_{DC} are the same as in (2.5), and Y_{in} is the same as (2.9).

We now want to design the control law. The most general control law possible is the following

$$P_{stab} = K_p(s)U_d + K_E(s)E + K_T(s)P_{CPL} \tag{2.35}$$

where we have feedback control from the input filter voltage U_d and feed-forward control from the reference power P_{CPL} and the line voltage E. This assumes that all of these three signals can be measured, which might not be the case, but it is a common starting point used to investigate performance limits. See Figure 2.8 for a block diagram of this general controller. In the figure the block labeled *Plant* represents the input filter and CPL system. This is a MIMO control system with inputs P_{stab}, P_{CPL} and E, and output U_d. The transfer functions $K_p(s)$, $K_E(s)$ and $K_T(s)$ are controllers to be designed. Assuming this general controller the closed loop system becomes

$$\begin{bmatrix} i \\ U_d \end{bmatrix} = \begin{bmatrix} \left(Y_{in} + \frac{G_E^2}{Z_{DC}}\left(1 - \frac{K_{pre}}{1+Z_{DC}Y_{DC}}\right)\right) & \frac{G_E F_{pre}}{1+Z_{DC}Y_{DC}} \\ \frac{G_E K_{pre}}{1+Z_{DC}Y_{DC}} & -\frac{Z_{DC}F_{pre}}{1+Z_{DC}Y_{DC}} \end{bmatrix} \begin{bmatrix} E \\ \frac{P_{CPL}}{U_{d0}} \end{bmatrix} \tag{2.36}$$

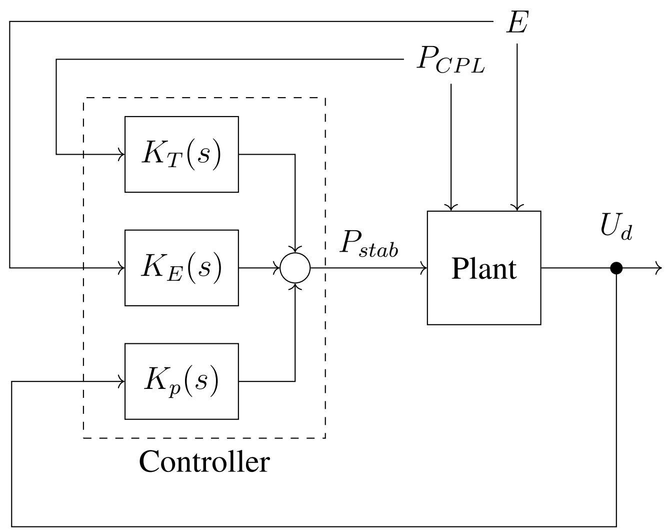


Figure 2.8: General controller for feedback and feed-forward control of input filter and CPL system. The block labeled *Plant* represents the input filter and the CPL.

where

$$K_{pre} = 1 - \frac{Z_{DC}K_E}{G_E U_{d0}} \tag{2.37}$$

$$F_{pre} = 1 + K_T \tag{2.38}$$

$$Y_{DC} = Y_{CPL} + \frac{K_p}{U_{d0}} \tag{2.39}$$

Notice how the input admittance Y_{DC}, which was discussed in Section 2.3.1, is modified in (2.39) through the introduction of the stabilizing control law.

Furthermore, we can write the transfer functions from line voltage deviations E and power reference deviations P_{CPL} to the control signal P_{stab}. It is

$$-\frac{P_{stab}}{U_{d0}} = \begin{bmatrix} \underbrace{\frac{G_E}{Z_{DC}}(S_{DC}K_{pre}(1+L_{DC0})-1)}_{F_E} & \underbrace{F_{pre}S_{DC}(1+L_{DC0})-1}_{F_p} \end{bmatrix} \begin{bmatrix} E \\ -\frac{P_{CPL}}{U_{d0}} \end{bmatrix} \tag{2.40}$$

where $L_{DC0} = Z_{DC}Y_{CPL}$ is the loop gain without control, and $S_{DC} = 1/(1 + Z_{DC}Y_{DC})$ is the sensitivity function. If we refer back to Figure 2.8 we see that the transfer function from P_{CPL} to P_{stab}, namely F_p, describes how variations in the power reference P_{CPL} affects the power modification P_{stab} both through the feedback $K_p(s)$ and through the feed forward $K_T(s)$. Similarly, the transfer function from E to P_{stab}, F_E, describes how line voltage variations E affects the power modification P_{stab} through the feedback $K_p(s)$ and through the feed-forward $K_E(s)$. Because of (2.32) the transfer function F_p can be seen as the reference tracking error. This is a transfer function we would like to minimize. The impact of variations in line voltage on the power modification is also something which we would like to minimize, since one of the performance specifications in Section 2.5 was to minimize use of power modification, and we want a well damped system.

Instead of minimizing K_p, K_E and K_T directly, the control problem can be seen as the optimization problem of finding regulators S_{DC}, K_{pre} and F_{pre} such that the transfer functions F_E and F_p are minimized in a system norm sense. In [7] Mosskull shows that this optimization problem can be solved analytically both for H_2 and for H_∞ optimization. In the following subsections the explicit solutions to the optimization problem of minimizing $\begin{bmatrix} F_E & F_p \end{bmatrix}$ are given. In the solutions below feed-forward from P_{CPL} is not considered. This means that $K_T(s) = 0$ and thus $F_{pre} = 1$. Optimization is done assuming feedback from the input filter voltage U_d and feed-forward from line voltage E. The controller in Section 2.7.4 is what Bombardier's conventional stabilization is based upon and this is what we will use as a benchmark for our own regulator.

2.7.2 H_2 Optimal Controller

For H_2 optimization [7] shows that the explicit solution to the optimization problem is the following controller

$$\frac{K_p(s)}{U_{d0}} = \underbrace{4\zeta\omega_0 C\left(\frac{P_0}{P_{lim}} - 1\right)}_{K_{\text{stab},H_2}} \frac{s}{s+\omega_c}, \quad K_E(s) = -K_p(s) \tag{2.41}$$

where $\omega_c = R/L$. This is a first order high-pass filter with a cut-off frequency which only depends on the input filter parameters, scaled by a constant gain dependent on the operating point. This controller does not use feed-forward from P_{CPL}. It does however require that both U_d and E are measured.

2.7.3 H_∞ Optimal Controller

For H_∞ optimization [7] shows that the optimal control law consists of the following two controllers

$$\frac{K_p(s)}{U_{d0}} = \underbrace{2\omega_0 C\left(\zeta\left(\frac{P_0}{P_{lim}} - 1\right) + \frac{9}{8}\right)}_{k_{\text{stab},H_\infty}} \underbrace{\frac{s}{s+\omega_c}}_{B(s)} \tag{2.42}$$

and

$$\frac{K_E(s)}{U_{d0}} = -k'_{\text{stab},H_\infty} \underbrace{\frac{\omega'_0\zeta'_B s}{s^2 + \omega'_0\zeta'_B s + \omega'^2_0}}_{B'(s)} \tag{2.43}$$

The constant gain $k'_{\text{stab},H_\infty}$ is defined as

$$k'_{\text{stab},H_\infty} = \frac{p_1 - \omega_c}{p_1 + \omega_c} k_{\text{stab},H_\infty} \tag{2.44}$$

and ω'_0 and ζ'_0 are defined as

$$\omega'_0 = \sqrt{\omega_c p_1}, \quad \zeta'_B = \frac{p_1 + \omega_c}{\sqrt{\omega_c p_1}} \tag{2.45}$$

The design parameter p_1 minimizes the peak value of $K_E(s)$ and is given by

$$\frac{p_1}{\omega_0} = \sqrt{1 + \zeta^2\left(\frac{P_0}{P_{lim}} - 1\right)^2} + \zeta\left(1 + \frac{P_0}{P_{lim}}\right) \tag{2.46}$$

The resulting controller $K_p(s)$ consists of the same first order high-pass filter found in (2.41) but with a different gain. The controller $K_E(s)$ is a second order band-pass filter where the cut-off frequencies depend both on the operating point and the input filter parameters.

2.7.4 Modified H_∞ Optimal Controller

In practice, often only the input filter voltage U_d is measured and available for feedback [7]. This means that $K_E(s)$ is set to zero. Without $K_E(s)$ the H_∞ controller in 2.7.3 performs poorly when it comes to attenuating disturbances in line voltage E. In [7] this issue is overcome by designing what they call a suboptimal H_∞ controller, from now on referred to as H_∞^{sub}, which minimizes the cost function F_E when only K_p is available. The resulting controller is

$$\frac{K_p(s)}{U_{d0}} = \underbrace{\left(2\left(1 - \frac{3}{(3.7)^2}\right)\zeta\frac{P_0}{P_{lim}} + \frac{3}{3.7}\right)\sqrt{\frac{C}{L}}}_{K_{\text{stab},H_\infty^{\text{sub}}}} B(s) \tag{2.47}$$

where

$$B(s) = \frac{\omega_0 \zeta_B s}{s^2 + \omega_0 \zeta_B s + \omega_0^2} \tag{2.48}$$

and

$$\zeta_B = 3.7 + 2\zeta \frac{P_0}{P_{lim}} \tag{2.49}$$

The resulting controller is a second order band-pass filter where both the filter coefficients and the gain $K_{\text{stab},H_\infty^{\text{sub}}}$ depend on the operating point and the input filter parameters. It is this controller which is implemented in the control structure of Bombardier's motor converter modules.

2.8 Linear Quadratic Gaussian Control

The controllers in Section 2.6 are designed from a frequency domain perspective. MPC, which will be the type of controller under investigation during most of this thesis, is based on time domain optimization. Before approaching MPC we will first, in this section, introduce a related control design method, namely linear quadratic Gaussian (LQG) control. We will show what it is and how it relates to H_2 optimal control.

In LQG control linear system dynamics are assumed. If they are not linear the system model can be linearized. It is also assumed that the system dynamics are known. This means that in standard LQG, as described below, there is no way to explicitly take model uncertainty into account. Finally, it is assumed that the measurement noise and the disturbance noise are uncorrelated zero-mean Gaussian stochastic processes with constant power spectral density. This is almost never true, but it is a useful approximation which has turned out to work pretty well in practice. Mathematically we have the following model

$$\dot{x}(t) = Ax(t) + Bu(t) + w_d(t) \tag{2.50}$$

$$y(t) = Cx(t) + w_n(t) \tag{2.51}$$

where $w_d(t)$ and $w_n(t)$ is disturbance noise and measurement noise with the following properties

$$E\left[w_d(t) w_d(\tau)^\top\right] = W\delta(t-\tau) \tag{2.52}$$

$$E\left[w_n(t) w_n(\tau)^\top\right] = V\delta(t-\tau) \tag{2.53}$$

$$E\left[w_d(t) w_n(\tau)^\top\right] = E\left[w_n(t) w_d(\tau)^\top\right] = 0 \tag{2.54}$$

where $\delta(t-\tau)$ is the Dirac delta function, and W and V are constant real-valued matrices describing the spectral density of $w_d(t)$ and $w_n(t)$ respectively.

The LQG control problem is to find the optimal control $u(t)$ which minimizes the following quadratic cost function

$$J = E\left[\lim_{T\to\infty}\frac{1}{T}\int_0^T [x(t)^\top Qx(t) + u(t)^\top Ru(t)]dt\right] \tag{2.55}$$

where $Q = Q^\top \succeq 0$ and $R = R^\top \succ 0$ are constant weighting matrices. These are design parameters which can be chosen to alter the importance of minimizing the states $x(t)$ versus the control input $u(t)$. Note that if one doesn't want to apply cost to the state $x(t)$ per se, but on the output $y(t)$ one can simply let $Q = C^\top Q'C$ for some weight matrix Q' since

$$y(t)^\top Q'y(t) = (Cx(t))^\top Q'(Cx(t)) = x(t)^\top C^\top Q'Cx(t) \tag{2.56}$$

If we consider the input filter and CPL system, which we will apply LQG to later in Chapter 3, the state is naturally chosen as the line current i and the input filter voltage U_d. The control input would be the power modification P_{stab}. So if we want to have a well damped input filter voltage we would want to make the elements of Q which are related to U_d large. In a similar way, we want to make R large if we want to restrict use of control input P_{stab}. In the end, control input usage and damping of states are conflicting objectives and making both Q and R large doesn't mean we achieve both control objectives perfectly. instead it is the relative difference between Q and R which determines what we deem most important: a well damped system or limited control usage. For this reason one often lets one of these weight matrices, either R or Q be the identity matrix, and only alters the other one. Since it is the relative weight difference between R and Q which matters, this simplifies design, as it removes one degree of freedom.

It is worth pointing out one outcome which comes from the fact that the cost function 2.55 has quadratic cost terms. The cost function will make the controller punish large deviations disproportionately compared to low deviations. Small deviations however carry almost no cost. This can sometimes lead to system dynamics which have small oscillations around the equilibrium, since the controller doesn't care much about damping those oscillations.

The solution to the LQG problem is given by the separation principle (see Appendix A.2). It states that we can find the optimal controller by first assuming a deterministic system (no noise) and find the optimal state feedback for that system. This is what is called the optimal linear quadratic regulator (LQR). After that we can design a Kalman filter for optimal state estimation.

The optimal state feedback

$$u(t) = -Lx(t) \tag{2.57}$$

is given by the L which minimizes the deterministic cost function

$$J_{det} = \int_0^\infty [x(t)^\top Qx(t) + u(t)^\top Ru(t)]dt \tag{2.58}$$

and the solution is

$$L = R^{-1}B^\top P \tag{2.59}$$

where $P = P^\top \succeq 0$ is the positive solution to the algebraic Riccati equation

$$A^\top P + PA - PBR^{-1}B^\top P + Q = 0 \tag{2.60}$$

The optimal state estimator is

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K(y(t) - C\hat{x}(t)) \tag{2.61}$$

where K is chosen such that $E\left[(x - \hat{x})^\top(x - \hat{x})\right]$ is minimized. The expression is minimized when

$$K = SC^\top V^{-1} \tag{2.62}$$

where $S = S^\top \succeq$ is the solution to the algebraic Riccati equation

$$SA^\top + AS - SC^\top V^{-1}CS + W = 0 \tag{2.63}$$

and V and W are the spectral density matrices from (2.52) and (2.53).

2.8.1 Relating LQG and H_2

Let us look back on (2.28). We restate it here again for convenience

$$E\left[\lim_{T\to\infty} \frac{1}{T}\int_0^T z(t)^\top z(t)dt\right] \tag{2.64}$$

We see that the time domain expression for the H_2 optimization problem looks a lot like the LQG cost function 2.55. In fact, LQG is a special case of H_2. If we let the exogenous output z in the H_2 problem be

$$z = \begin{bmatrix} Q^{\frac{1}{2}} & 0 \\ 0 & R^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \tag{2.65}$$

and if we relate the stochastic inputs w_d and w_n to w in the following way

$$\begin{bmatrix} w_d \\ w_n \end{bmatrix} = \begin{bmatrix} W^{\frac{1}{2}} & 0 \\ 0 & V^{\frac{1}{2}} \end{bmatrix} w \tag{2.66}$$

we see that we do in fact have the same problem formulation.

2.9 Introduction to Model Predictive Control

Model predictive control (MPC) is a control method with its origins in the process industry of the 1980s. Its capability of dealing with constrained MIMO control systems made it attractive to the oil and chemical industries [13]. The computational burden of MPC is relatively high, but since the processes in these industries have slow dynamics, they were able to benefit from this advanced control method despite the limited computational power of computers of the time. With improvements in computer technology, combined with the development of custom algorithms, the computational speed of the MPC optimization algorithms have greatly improved. It can therefore be used at sample rates which are orders of magnitude faster, than when first introduced. For this reason MPC has found its way into a much wider range of applications, among which are control of power converters and drives [13], [14], which will be explored in this thesis.

Common for all MPC approaches is that they use a model of the process one wishes to control to predict the future trajectory of the process. By optimizing over the input control sequence, it is possible to find the control input which results in the best trajectory, for a finite number of samples into the future (see Figure 2.10). Feedback is achieved by only applying the first input in the computed control sequence and then repeating the optimization each sample period, while also shifting the prediction horizon one sample forward. This way the control input sequence is updated at each sample to account for modelling errors or disturbances. Because of the way the horizon is shifted, MPC is sometimes referred to as *receding horizon control*. In the optimization problem it is also possible to include constraints on the input or on the internal states of the model. This is quite useful if there for example exist known hard limits on some of the states or if we want to keep the control inputs within some bound. Being able to incorporate constraints directly into the problem formulation is something which is not possible in control methods such as H_∞ or LQG and is one of the biggest appeals of MPC. In the next subsection we will describe the classical MPC problem formulation.

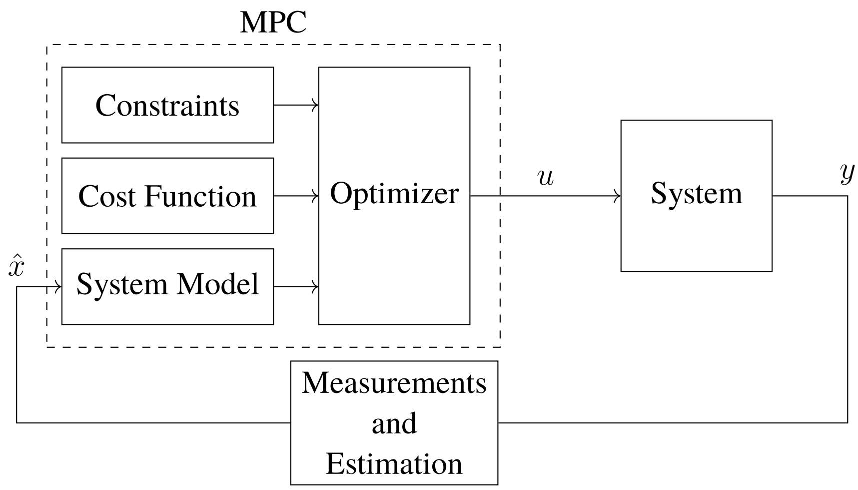


Figure 2.9: Block diagram of a general MPC regulator structure.

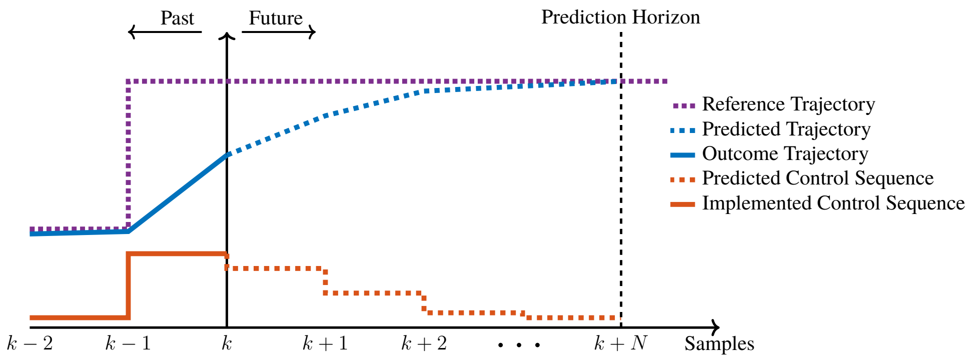


Figure 2.10: Illustration of the predicted trajectory and the outcome trajectory under MPC.

2.9.1 Classical Problem Formulation of Model Predictive Control

In classical MPC a deterministic linear time invariant (LTI) discrete-time model is used to model the process. It can be written as

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{2.67}$$

where $x(k)$ is the internal state, $u(k)$ is the control input, $y(k)$ is the system output i.e. the states which we can measure, and k is the discrete time index.

One nice property of the discrete-time LTI state space model is that the state at time index k can be written as a linear combination of the initial state $x(0)$ and the control inputs by iterating the system equations (2.67) forward. We have

$$\begin{aligned} x(1) =& Ax(0) + Bu(0) \\ x(2) =& Ax(1) + Bu(1) = A(Ax(0) + Bu(0)) + Bu(1) = A^2x(0) + ABu(0) + Bu(1) \\ &\vdots \\ x(k) =& A^k x(0) + \sum_{i=0}^{k-1} A^i Bu(k-1-i) \end{aligned} \tag{2.68}$$

We can write (2.68) on vector form as

$$x(k) = A^k x(0) + \begin{bmatrix} A^{k-1}B & A^{k-2}B & \cdots & AB & B \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(k-1) \end{bmatrix} \tag{2.69}$$

Using (2.69) we can predict what the state $x(k)$ will be any finite number of time steps into the future, given the initial state and the control input. With (2.68) we can not only predict where the state will end up, but since we can stop the summation at any arbitrary time step before k we can predict its entire trajectory. This principle is what is used in MPC. Given that we know, or can estimate, what the state is at time index k we try to find a sequence of control inputs $[u(0), u(1), \cdots, u(k-1)]^\top$ such that the trajectory of the state x, N time steps into the future, is "good" according to some measure of our choice. In classical MPC we use a quadratic cost function to define what we consider "good". The cost function which often is used looks like

$$J = \sum_{k=0}^{N-1} (x(k)^\top Qx(k) + u(k)^\top Ru(k)) + x(N)^\top Q_f x(N) \tag{2.70}$$

where $Q \succeq 0$, $R \succ 0$ and $Q_f = Q_f^\top \succeq 0$ are weights on the state, control input and the terminal state respectively, and N is the prediction horizon. The first part of (2.70) is the cost of the first N time steps. It is essentially a finite version of the infinite horizon LQR cost function (2.58). The second part is a separate cost placed on the final state in the horizon added to represent the cost from time N to infinity. This allows the MPC problem to be solved for a limited horizon while still resembling the optimal solution to the infinite horizon problem in the LQR problem (see (2.58)). In fact, without constraints the classical MPC problem, using quadratic cost, is equivalent to the discrete-time version of the LQR problem, if Q_f is set to be the solution to the discrete-time algebraic Riccati equation (see Appendix A.3). This is useful, because, depending on how the weights Q and R are chosen, the best control input in the short term, as determined by the summation part of cost function, could be something which will drive the system further away from the equilibrium. By adding a cost to the final state representing the cost for all future time, it becomes costly to move away from the equilibrium and we make sure the "good" solution moves towards the equilibrium instead. This principle of first computing the cost for the first N terms and then adding a cost for all future time is called the dual mode principle and is for example described by Kouvaritakis and Cannon in [15, Chapter 2.3].

The last piece of the MPC problem formulation is to add the constraints. The constraints are usually restricted to polyhedral sets (sets of linear inequalities) in terms of the state and the control policy. This is done to make sure the problem is convex, which makes it easier to find the solution. If we bring everything which has been introduced in this section together, the discrete-time model, the cost function and the constraints, we get an optimization problem on the form

$$\begin{array}{lll} \text{minimize} & \displaystyle\sum_{i=0}^{N-1}(x_i^\top Q x_i + u_i^\top R u_i) + x_N^\top Q_f x_N & \\ \text{subject to} & x_{i+1} = Ax_i + Bu_i & i = 0, 1, \cdots, N-1 \\ & x_i \in \mathcal{X}, \quad u_i \in \mathcal{U} & i = 0, 1, \cdots, N-1 \\ & x_N \in \mathcal{X}_f, \quad x_0 = x(k) & \end{array} \tag{2.71}$$

where we have used the notation x_i to denote the ith sample into the future starting from sample $x(k)$. This is clarified by the constraint $x_0 = x(k)$. The polyhedral sets $\mathcal{X}$ and $\mathcal{U}$ constrain the state and the control policy respectively. One sometimes adds a terminal set $\mathcal{X}_f$ which constrains the position of state x at the end of the horizon. It is added to force the state at time step N into a set in which none of the constraints $\mathcal{U}$ and $\mathcal{X}$ are violated for all future time. This means that linear state feedback will be the optimal control solution from time N and forward, and that the choice of the solution to the LQR problem is a valid choice for the final cost Q_f.

Typical choices for the constraints $\mathcal{X}$ and $\mathcal{U}$ are

$$\begin{aligned} x_{\text{min}} \leq x_i \leq x_{\text{max}}, &\quad i = 0, 1, \cdots, N \\ u_{\text{min}} \leq u_i \leq u_{\text{max}}, &\quad i = 0, 1, \cdots, N-1 \\ |u_{i+1} - u_i| \leq \Delta u_{\text{max}}, &\quad i = 0, 1, \cdots, N-1 \end{aligned} \tag{2.72}$$

The optimization problem (2.71) is solved with the state at the current sample, $x(k)$, as initial state. The problem is a convex quadratic program (QP), for which there are many efficient solvers [15].

Because of (2.69) we can substitute x_i in (2.71) and write the state as a linear combination of the initial state, the control input, and the system matrices A and B. The solution to (2.71) is the optimal control sequence $U^* = [u_0, \cdots, u_{N-1}]^\top$, i.e the optimal control for N steps forward in time. In MPC, only the first control input, u_0, is applied. Then in the next sample period the optimization problem is solved again, giving the optimal control sequence N samples forward in time. As before only the first input in the sequence is realized. Hence, at each sample instance the optimization problem is solved for the state $x(k)$ which the system is currently in, the optimal control sequence U^* given the cost function and the constraints is found, and the first control input u_0 is applied. This is the receding horizon principle, which was explained before, and the way in which feedback is introduced in MPC. Since the optimization problem is parameterized in the current state $x(k)$, according to (2.69) we can write the optimization problem (2.71) as a function $J_{MPC}(x(k))$. The control input at time index k is then given by

$$u(k) = \left(\arg\min_U J_{MPC}(x(k)) \right)_0 \tag{2.73}$$

where $(\cdot)_0$ represents the first element in an array. The entire solution may then be seen as a state feedback law, which however is difficult to express in closed form.

Oftentimes the state $x(k)$ is not directly available, but we have a situation like in (2.67) where we measure $y(k)$ - a linear combination of the states. Therefore, the MPC has to be combined with some sort of state estimation. The state estimation could for example be done using the Kalman filter, which was explained in Section 2.8. We then get a closed loop system of the form which is shown in Figure 2.9.

Furthermore, note that the optimization problem (2.71) will find the control sequence which drives the state $x(k)$ to zero. If one for example, as is common, want the output $y(k)$ to follow a reference $r(k)$ instead one can simply define

$$e(k) = y(k) - r(k) = Cx(k) - r(k)$$

Then one can replace the cost on the state in the cost function with

$$e(k)^\top Q' e(k) = (Cx(k) - r(k))^\top Q'(Cx(k) - r(k))$$

similar to what was done in (2.56) to make the LQR cost be on the output instead of the state.

In the next few subsections we present an overview of different ways in which MPC has been implemented. The focus will primarily be on applications in power converters and drives.

2.9.2 Implementation Considerations for Online Optimization of MPC

MPC methods which compute the control input online in each sample period, are considered traditional MPC methods. The dynamics in electrical systems are typically quite fast; at least compared to the slow chemical processes where MPC originated. Making sure the MPC optimizer is fast enough can therefore be a challenge. Special methods need to be employed to minimize the online computation time. In this subsection we will therefore consider some ways in which the online computation time can be decreased. The methods presented here come from Wang and Boyd in [16]. We refer to that paper for a more in-depth explanation of the methods.

Primal Barrier Interior-point Method

In the primal barrier interior-point method the inequality constraints of the MPC problem are replaced by a so-called barrier term, which is added to the cost function. This makes the problem solvable using Newton's method. The method is described in [16]. First the optimization problem (2.71) is rewritten into a more compact form. This is done by defining an optimization variable which is the combination of x and u

$$z = \begin{bmatrix} u_k & x_{k+1} & \cdots & u_{k+N-1} & x_{i+N} \end{bmatrix}^\top \in \mathbb{R}^{N\times(m+n)} \tag{2.74}$$

where n is the size of the state x and m is the size of the control input u. The QP (2.71) is then

$$\begin{aligned} &\text{minimize} \quad z^\top Hz \\ &\text{subject to} \quad Pz \leq h, \quad Cz = b \end{aligned} \tag{2.75}$$

where the definition of H, P, h, C and b may be found in Appendix A.4.

Next, the inequality constraints are replaced by a so called barrier term, which is added to the cost function. We then get an approximation of the original problem

$$\begin{aligned} &\text{minimize} \quad z^\top Hz + \kappa\phi(z) \\ &\text{subject to} \quad\quad Cz = b \end{aligned} \tag{2.76}$$

where $\kappa > 0$ is a tunable barrier parameter. The smaller the value of κ the better the approximation will be. A small κ will however also make the problem take more iterations to converge. The choice of κ in an MPC application is thus a trade-off between solution accuracy and computational speed and the choice will depend on the needs of the specific application. The function $\phi(z)$ is the log barrier - a function which is associated with the inequality constraints. It is defined as

$$\phi(z) = \sum_j -\log(h_j - p_j^\top z) \tag{2.77}$$

where p_j is the jth row of P. When $p_j^\top z$ approaches h_j, i.e., when we get close to the constraint, the barrier function increases towards infinity. By adding this to the cost and removing the constrains, any solution too close to the constraint will have a really high cost and hence not be optimal in the approximate problem. Once the problem is on the form (2.76) it can, for example, be solved with Newton's method, which has quadratic convergence rate.

Infeasible Start Newton Method

Once a QP is on the form of (2.76) it can be solved with a so called infeasible start Newton method, which is suggested in [16]. It consists of forming the optimality conditions for (2.76) which are

$$\begin{aligned} r_d &= 2Hz\kappa P^\top d + C^\top v = 0 \\ r_p &= Cz - b = 0 \end{aligned} \tag{2.78}$$

where $d_j = 1/(h_j - p_j^\top z)$ and v is a variable associated with the equality constraint. The Newton step then consists of solving the linear equations

$$\begin{bmatrix} 2H\kappa P^\top \text{diag}(d)^2 P & C^\top \\ C & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta v \end{bmatrix} = - \begin{bmatrix} r_d \\ r_p \end{bmatrix} \tag{2.79}$$

using an initial guess z_0, v_0. The approximate solution $\bar{z}$, $\bar{v}$ is updated in each step according to $\bar{z} := \bar{z} + s\Delta z$ and $\bar{v} := \bar{v} + s\Delta v$, where $s \in (0, 1]$. Before the next Newton step the residual vector $(r_d, r_p)^\top$ is updated by computing the optimality conditions (2.78) with the current approximate solution. Once the residuals are below a certain threshold which we define, we stop. The initial guess, (z_0, v_0), may be infeasible, by violating the equality constraints $Cz = b$, which is where the method gets its name from. The initial guess does however need to satisfy the inequality constraints $Pz < h$ or else the barrier function (2.77) is undefined as it would involve taking the log of a negative number.

Improving Computation Speed of the Newton Step

The speed at which the system of equations (2.79) are solved can be greatly improved by exploiting the structure of the problem. Namely that

$$2H + \kappa P^\top \text{diag}(d)^2 P$$

is a block diagonal matrix. The cost of solving the system of equations using LU decomposition (Gaussian elimination) is $\mathcal{O}(N^3(n+m)^3)$ flops. When the system is known to have the aforementioned structure, a method called block elimination can be used instead. Block elimination has at most computational cost of $\mathcal{O}(N(n+m)^3)$ flops for a problem with this structure [16]. The cost thus goes from being cubic to linear in terms of the horizon. If the cost function in the MPC optimization problem doesn't include cross-terms, i.e terms which depend both on the state x and the control input u, the complexity decreases even further to be $\mathcal{O}(N(n^3 + n^2 m))$. This cost is linear in terms of both the horizon N and the number of control inputs m. Thus, in time critical applications we are more limited in terms of which cost functions we can use. This poses limitations on the choices we have when we want to describe the notion of desired performance in the MPC framework. If computational speed is less of a problem we could consider more complex cost functions which might better reflect the desired performance. The way the MPC problem is formulated in (2.7) there are no cross-terms in the cost function.

Warm start

Another way in which the online computational speed of the MPC problem can be improved is by warm start. The number of iterations needed to find the solution using Newton's method decreases the closer the initial guess is to the solution. In MPC we solve a QP at each sample and obtain a solution which is the optimal trajectory for that time. Let the computed trajectory at time step $k-1$ be

$$z^* = \begin{bmatrix} u^*_{k-1} & x^*_k & \cdots & u^*_{k+N-2} & x^*_{k+N-1} \end{bmatrix}^\top \tag{2.80}$$

In the next time step, at k, it is reasonable to assume that the previous solution is close to the new solution. Therefore, we can initialize the primal barrier method at time step k with

$$z_0 = \begin{bmatrix} u^*_k & x^*_{k+1} & \cdots & x^*_{k+N-1} & \hat{u} & \hat{x} \end{bmatrix}^\top \tag{2.81}$$

namely the same trajectory as z^* but shifted one timestep. The last two elements of the initial guess, $\hat{u}$ and $\hat{x}$, can for example be chosen as the average of all the control inputs and the average of all the state values in the rest of the trajectory. This initial guess z_0 is likely to be close to the optimal solution to the QP, meaning the solver will converge quickly, after only a few iterations. Furthermore, if z^* satisfies the inequality constraints $Pz < h$ so will z_0, except possibly at the first and last timesteps. For those two timesteps it is not guaranteed that the inequality constraints will hold. The initial guess may then be modified at these two timesteps in order to satisfy the constraints.

2.9.3 Explicit Model Predictive Control

An entirely different way of reducing the online computation time of MPC is to move most of the computation offline. Explicit MPC is one way to do that. In explicit MPC the optimization problem is parameterized in terms of the state and the problem is then solved offline. If the cost function is quadratic, the constraints linear inequalities, and the model is LTI, the resulting solution will be a piece-wise linear function in terms of the state [17]. The state-space will be partitioned into polyhedral regions such that the optimal control in each region is a linear function in terms of the state. This means that if we want the optimal control u^* for some state x_0, it is given by

$$u^* = K_i x_0 \tag{2.82}$$

where the feedback K_i will differ depending on where in the state-space x_0 is located. The polyhedral regions, where each feedback K_i is valid, are separated by hyperplanes. The regions can be sorted into a binary tree, where, in each node of the tree one evaluates which side of a hyper-plane the state lies. This means that the problem of finding the optimal control is reduced to a binary search in each time step. Since binary search has complexity $\mathcal{O}(\log_2 H)$, where H is the number of hyper-planes dividing the state space, this method has the potential to be very fast, as long as the number of regions is manageable. One disadvantage with explicit MPC is that it requires more memory than implicit solvers. If memory is limited, explicit MPC might not be an option if the number of regions is too large. According to [16] explicit MPC is manageable when $n \leq 5, m \leq 3$, and $N \leq 12$, where n is the number of states, m number of inputs, and N the prediction horizon. This result is however from 2010 and could be considered somewhat dated. More recently, in 2017, [18] showed that explicit MPC is manageable on modern hardware with a problem space of $n = 10$, $u = 5$, and $N = 40$. With a problem space of that size the binary tree shouldn't grow too large.

2.9.4 Finite Control Set Model Predictive Control

Finite control set MPC (FCS-MPC) is a variant of MPC which has been widely investigated for use in power converter applications. See, for example [19, 20, 21, 22, 23]. FCS-MPC is a low level type of control where the discrete nature of the power converter is taken into account and the MPC problem is formulated in terms of the switches inside the converter. The most popular method is called optimal switching vector MPC (OSV-MPC) [14]. This is something which we looked at in this thesis. We decided not to implement it since it didn't fit into the existing control structure at Bombardier.

In this method we define a set S which is the set of all valid combinations the transistors in the converter can be in (see Figure 2.11). Each element, S_i, of S is a vector of length three which represents which switch is open in each of the legs in the converter circuit. For example $S_1 = (1, 0, 0)$. This means that S_a, $\bar{S}_b$ and $\bar{S}_c$ are open and $\bar{S}_a$, S_b and S_c are closed. For a typical three-legged dc/ac converter, as the one in Figure 2.11, there are 8 valid configurations for the switches, since it must be true that one and only one switch per leg is open at any given time. This means that $|S| = 2^3 = 8$. Each switch configuration S_i results in different voltages across the phases in the output filter. These discrete voltage vectors can be seen as the control inputs to the system.

Thus, the control input isn't part of a continuum as is normally the case in MPC, but instead part of some discrete set S with a finite number of possible states. The optimization problem then reduces into finding the best input voltage vector in the set, at each sample instance. If the set is small enough it can be done by simply computing the cost of each possible option and picking the best one. The computational complexity of FCS-MPC is $\mathcal{O}(|S|^N)$ where $|S|$ is the size of the control set S, and N is the prediction horizon. We see that the complexity grows exponentially with the prediction horizon. Meanwhile

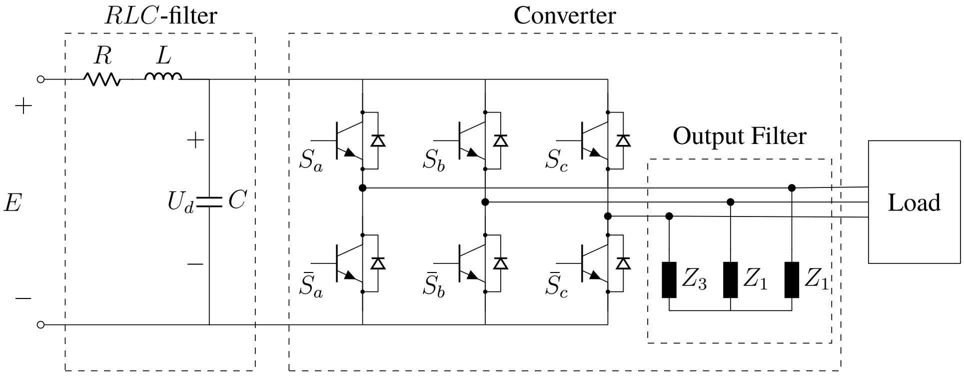


Figure 2.11: Converter from dc to three-phase ac with load and input RLC filter.

the switching frequency of the converter transistors are typically in the order of tens of kHz. For this reason, FCS-MPC generally can't address long prediction horizons and in most power electronics applications, prediction horizons of length one is used. Furthermore, OSV-MPC will result in a variable switching frequency, unless constraints addressing this are added into the problem formulation. There are alternative FCS-MPC methods such as optimal switching sequence MPC (OSS-MPC) that directly takes this into account. However, OSS-MPC has greater computational cost than OSV-MPC [14].

Chapter 3

Linear Quadratic Regulator for the Negative Impedance Problem

In this chapter we will show how the negative impedance problem can be approached using the LQG framework presented in Chapter 2.8. Assuming a deterministic model, we will design the optimal full state feedback law.

3.1 LQR Controller Design

Consider again a CPL connected to a dc voltage source via a RLC filter which was modeled in Chapter 2.2 and described by equations (2.2). They are repeated here

$$\frac{di}{dt} = \frac{1}{L}\left(-Ri - U_d + E\right)$$
$$\frac{dU_d}{dt} = \frac{1}{C}(i - i_d)$$

Consider also the same linearization of the current i_d as was done in Chapter 2.7.1, namely (2.33), which is also also repeated here

$$\Delta i_d(t) = -\frac{P_0}{U_{d0}^2}\Delta U_d(t) + \frac{1}{U_{d0}}\Delta P_{CPL}(t) + \frac{1}{U_{d0}}P_{stab}(t)$$

where Y_{stab0} is set to zero. As before $P_{stab}(t)$ is a power modification which is used for stabilization. Using the equation for the linearized current the differential equations (2.2) can be written on state space form

$$\dot{x} = \underbrace{\begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & \frac{P_0}{CU_{d0}^2} \end{bmatrix}}_{A} x + \underbrace{\begin{bmatrix} 0 \\ -\frac{1}{C} \end{bmatrix}}_{B} u + \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & -\frac{1}{C} \end{bmatrix} \begin{bmatrix} E \\ \frac{P_{CPL}}{U_{d0}} \end{bmatrix} \tag{3.1}$$

In (3.1) the state and control input are defined as

$$x = \begin{bmatrix} \Delta i \\ \Delta U_d \end{bmatrix}, \quad u = P_{stab}/U_{d0} \tag{3.2}$$

The state x is hence defined in terms of deviations from an equilibrium, around which the equations have been linearized. More precisely

$$\Delta i = i - i_0, \quad \Delta U_d = U_d - U_{d0} \tag{3.3}$$

where i_0 and U_{d0} is the line current and input filter voltage at the equilibrium point. The two excitation signals, E and P_{CPL} are not considered in the LQR controller design, since it is designed in terms of the control input u and the state x. We therefore disregard them for now so that we get a system on the form

$$\dot{x} = Ax + Bu \tag{3.4}$$

which we assume to be fully deterministic, meaning that we have full knowledge of the state x.

The weights in the cost function (2.55), which is repeated here

$$J_{det} = \int_0^\infty [x(t)^\top Qx(t) + u(t)^\top Ru(t)]dt$$

, are designed in order to meet the performance specifications listed in Chapter 2.5. First we add a weight q_{U_d} reflecting that we would like to minimize ripple in the voltage U_d. Secondly, we add a weight r_u to the control input u to reflect that we want to limit the use of control input. The weights for the cost function (2.55) are thus

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & q_{U_d} \end{bmatrix}, \quad R = r_u \tag{3.5}$$

where the values q_{U_d} and r_u can be adjusted in order to obtain desired closed loop dynamics.

The feedback gain L is then given by solving the Riccati equation (2.60) for the system matrices A and B as defined by (3.1) and the weights (3.5).

3.2 Varying Operating Point

As we said before, the controller assumes that the input to it is given as deviations from an equilibrium, or an operating point. To remove steady state bias, the operating point has to move when reference changes are made in the line voltage E and the power reference P_{CPL}.

Furthermore, to have a model which is valid in each operating point, the state space model is parameterized in terms of the operating point. See Chapter 4.3 for more details on how this is done.

The operating point can't be changed too fast or else the system can become unstable. One also cannot change the operating point too slowly, or the model might not reflect the system well. To vary the operating point, we design a first order low pass filter

$$F(s) = \frac{1}{\tau s + 1} \tag{3.6}$$

where τ is the time constant of the filter. We will filter the state vector $(i(t), U_d(t))^\top$ through the filter $F(s)$ and define the output of the filter to be $(i_0(t), U_{d0}(t))^\top$. The input to the LQ regulator is then $x(t) = (i(t), U_d(t))^\top - (i_0(t), U_{d0}(t))^\top$.

Chapter 4

Model Predictive Control for the Negative Impedance Problem

The goal of this thesis was to investigate the use of MPC to stabilize the system which consists of a CPL fed by a dc power source via an RLC filter. As we have seen already, this system suffers from the negative impedance instability problem. In this chapter we will show how the system can be stabilized using an MPC regulator. We start off by designing a classic MPC regulator according to the theory in Chapter 2.9. The design will build on the LQ regulator from Chapter 3. Then, in Section 4.3 we extend the MPC model into a parameter-varying design.

4.1 Problem Formulation in Model Predictive Control Framework

As explained in Chapter 2.9, MPC relies on an discrete-time system model. In order to design a MPC controller we will therefore have to come up with a discrete-time model for the input filter and CPL system described by the differential equations (2.2)

$$\frac{di}{dt} = \frac{1}{L}\left(-Ri - U_d + E\right)$$
$$\frac{dU_d}{dt} = \frac{1}{C}(i - i_d)$$

and equation (2.3)

$$i_d(t) = \frac{P(t)}{U_d(t)}$$

describing the relationship between the input voltage and the input current to the CPL. The equations were repeated here to aid the reader. To get a discrete-time LTI model we will start with the continuous-time LTI model used in Chapter 3. It is repeated here for simplicity. The model is

$$\dot{x}(t) = A_c x(t) + B_c u(t) \tag{4.1}$$

where

$$A_c = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & \frac{P_0}{CU_{d0}^2} \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ -\frac{1}{C} \end{bmatrix} \tag{4.2}$$

are the continuous-time state space matrices. The control input is $P_{stab}(t)/U_{d0}$ and the state is $x(t) = \begin{bmatrix} \Delta i(t) & \Delta U_d(t) \end{bmatrix}^\top$, where $\Delta i(t)$ and $\Delta U_d(t)$ are defined as deviation variables from an equilibrium according to (3.3).

Using zero order hold sampling the system can be transformed to a discrete-time LTI system on the form (2.67)

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned}$$

where the discrete-time system matrices are defined as

$$A = \exp(A_c T_s), \quad B = \int_0^{T_s} \exp(A_c s) B_c ds \tag{4.3}$$

where A_c and B_c are the continuous-time system matrices (4.2), T_s is the sampling period and $\exp(\cdot)$ is the exponential function.

The MPC cost function is defined according to (2.70)

$$J = \sum_{k=0}^{N-1} (x(k)^\top Q x(k) + u(k)^\top R u(k)) + x(N)^\top Q_f x(N)$$

where we set Q_f equal to the solution of the discrete-time algebraic Riccati equation (DARE)

$$Q_f = A^\top Q_f A - A^\top Q_f B(\bar{R} + B^\top Q_f B)^{-1} B^\top Q_f A + \bar{Q} \tag{4.4}$$

This choice of Q_f corresponds to the cost of using the LQR from the final state in the MPC time horizon, $x(N)$, to infinity, as was explained in Chapter 2.9.1. The weight matrices $\bar{Q}$ and $\bar{R}$ are the weights on the state and the control input respectively for the infinite horizon LQR problem. These matrices do not necessarily have to be the same as Q and R in (2.70). They thus become two new tuning parameters which affect the MPC.

For constraints we will assume control input constraints on the following form

$$\underline{u} \leq u_k \leq \overline{u} \tag{4.5}$$

where $\overline{u} \geq 0$ and $\underline{u} < 0$ are the upper and lower limits on u.

This concludes the nominal definition of the MPC controller. As is, the controller assumes that both the states can be measured, which isn't necessarily the case. The controller is also linearized around one particular operating point defined by $(P_0, i_0, U_{d0}) = (P_0, P_0/U_{d0}, U_{d0})$, while the system itself may operate at any operating point as the reference power or the line voltage changes. These are problems which will be addressed in the next two sections.

4.2 State Estimation

The MPC in Section 4.1 assumes that both the line current i and the input filter voltage U_d can be measured. However, we would like to have a controller which works when only U_d is available for measurement, similar to the H_∞^{sub} regulator in Chapter 2.7.4. For this reason, we will design an estimator for the line current i. Based on the separation principle explained in Chapter 2.8, the state estimator can be designed separately from the controller.

We will however not use a Kalman filter in this case for the estimation of i. The reason for this is that the Kalman filter uses the linear state estimator (2.61). Since we have a non-linear system we would prefer an estimator which is independent of the operating point (P_0, i_0, U_{d0}). To estimate the current independently of the operating point we can use the original, nonlinear system equations (2.2). We know that in continuous time

$$i(t) = C\frac{dU_d(t)}{dt} + \frac{1}{U_d(t)}(P_{CPL}(t) + P_{stab}(t)) \tag{4.6}$$

Since we can measure U_d and P_{CPL}, and realize P_{stab} in our regulator, we have access to sampled versions of these signals and from them we can form a discrete-time version of the current i. We get

$$i(k) = CD(z)U_d(k) + \frac{1}{U_d(k)}(P_{CPL}(k) + P_{stab}(k)) \tag{4.7}$$

where

$$D(z) = \left(\frac{\tau z^{-1}}{1-(1-\tau)z^{-1}}\right)\left(\frac{1-z^{-1}}{T_s}\right) \tag{4.8}$$

is a discrete-time filter which approximates the derivative of U_d. It is a backward difference approximation of the derivative combined with a first order low pass filter with cutoff frequency τ. The low pass filter is added to attenuate high frequency noise, which in general is amplified by a derivative filter. In (4.8) T_s is the sampling time and z denotes the variable of the $\mathcal{Z}$-transform. With the estimator (4.7) we can now get the state vector $x(k) = \begin{bmatrix}\Delta i(k) & \Delta U_d(k)\end{bmatrix}^\top$ by simply subtracting the operating point from $U_d(k)$ (which we measure) and $i(k)$ (which we estimate) in the following manner

$$x(k) = \begin{bmatrix}\Delta i(k) \\ \Delta U_d(k)\end{bmatrix} = \begin{bmatrix}i(k) - i_0 \\ U_d(k) - U_{d0}\end{bmatrix} \tag{4.9}$$

4.3 Linear Parameter-Varying Model

As mentioned in Section 4.1 we would like a controller which works for any operating point. However, since the model (4.1) has been linearized it is only valid around the point for which it has been linearized. To deal with this problem we will extend our controller to a linear parameter-varying MPC (LPV-MPC), where the system matrices A and B are parameter-dependent and vary over time according to the operating point.

4.3.1 Parameterized Model

The first step in switching to an LPV-MPC is to update the linear model based on the operating point. Let

$$\theta = \frac{P_0}{U_{d0}^2} \tag{4.10}$$

be a scalar parameter defined by the operating power P_0 and the operating voltage U_{d0}. Consider the matrices of the continuous-time state space model (4.2). The matrix A_c can be written as

$$A_c(\theta) = \begin{bmatrix}-\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & \frac{\theta}{C}\end{bmatrix} \tag{4.11}$$

where only one element in the matrix depends on θ. The matrix B_c is constant with respect to θ. In MPC, however, we use a discrete-time model. This small difference unfortunately has the consequence that the elements of A and B are affected more. Through the discretization (4.3) we see that a change in A_c affects elements in both A and B. The exponential function furthermore makes it difficult to find closed form expressions for these values.

Instead of finding closed form expressions for $A(\theta)$ and $B(\theta)$ we will find the best piece-wise linear functions which fit computed values for $A(\theta)$ and

$B(\theta)$. By computing (4.3) for a range of operating points θ_i, d units apart, the discrete-time state-space matrices can be plotted as a function of θ. By assuming that the functions $A(\theta), B(\theta)$ are linear in the interval $\left[\theta_i - d/2 \quad \theta_i + d/2\right]$, we can express the discrete-time system matrices A and B as piece-wise linear functions of θ. A piece-wise linear function was chosen since it is easy to implement, while it also avoid jumps in the function, as we would have with a piece-wise constant function for example. In Figure 4.1 and 4.2 the resulting piece-wise constant functions are plotted along with the discrete data points from which the function was created. We end up with one function for each element in the A-matrix and the B-matrix. Using these functions, it is possible to quickly compute the discrete-time linear system model for a given operating point.

Since the terminal cost Q_f depends on the discrete-time state-space matrices through (4.4), it too will vary with the operating point. Solving the DARE is a relatively costly and complicated operation. To speed up computation time, and to simplify implementation, this matrix is also parameterized in θ. This is done by fixing the weight matrices Q and R and then solving the DARE for different A- and B-matrices. The elements of Q_f can be plotted as functions of θ. This has been done in Figure 4.3. Using the same piece-wise linear function as before, we can fit a function to the data which allows us to compute an approximate terminal cost very fast.

The matrices $A(\theta)$, $B(\theta)$ and $Q_f(\theta)$ are updated each sample period but are assumed constant across the prediction horizon of the MPC optimizer.

4.3.2 Slowly varying the operating point

Just as with the LQ regulator in Chapter 3 we will need to vary the operating point over time to remove steady state error. The operating point should not be changed too fast or else the system might become unstable. We also do not want the operating point to be varied too slowly. We might then operate with a model which does not reflect the state of the true system. This can affect performance, and in worst case lead to instability.

Like what was done for LQR, we will design a low pass filter through which we will filter the signals $P_{CPL}(k)$, $i(k)$ and $U_d(k)$. The filter is designed as a first order discrete-time low pass filter

$$F(\nu) = \frac{\nu z^{-1}}{1 - (1-\nu)z^{-1}} \tag{4.12}$$

where ν is normalized frequency. We then define the operating point $P_0(k)$,

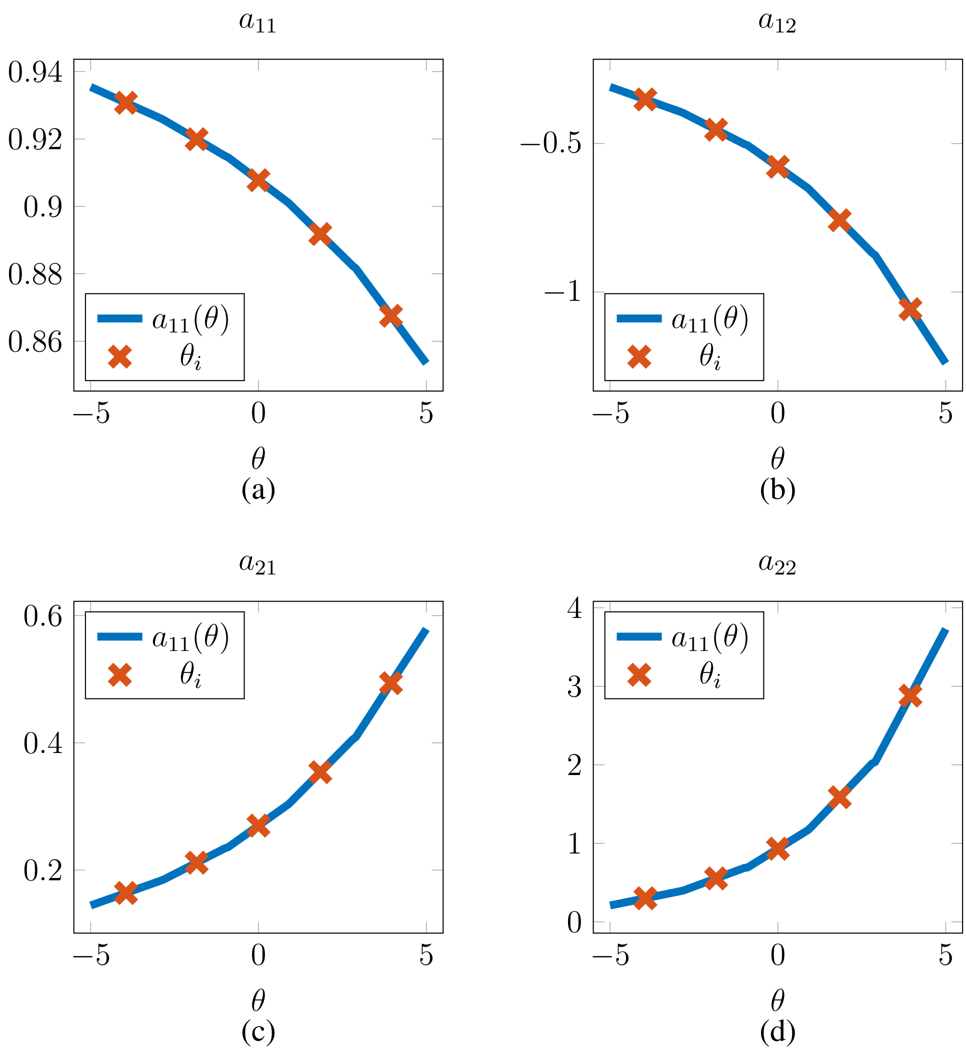


Figure 4.1: Piece-wise linear parametrization of the discrete-time state-space matrix $A(\theta)$, where each plot (a)-(d) represent one element of $A(\theta)$. The red cross marks are the computed middle points θ_i such that the function is constant in the interval $[\theta_i - d/2 \quad \theta_i + d/2]$, while the blue curves show the value of $A(\theta)$.

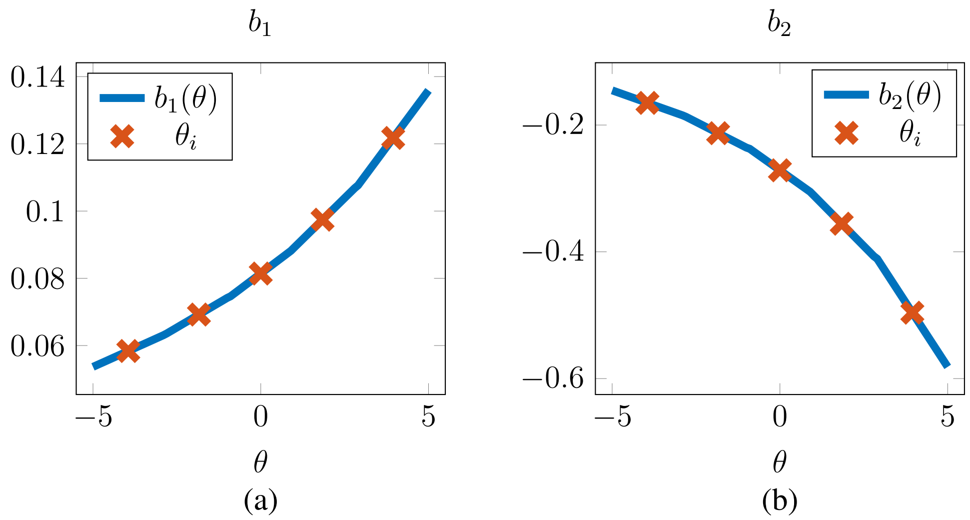


Figure 4.2: Piece-wise linear parametrization of the discrete-time state-space matrix $B(\theta)$, where each plot (a)-(b) represent one element of $B(\theta)$. The red cross marks the computed middle points θ_i such that the function is constant in the interval $\left[\theta_i - d/2 \quad \theta_i + d/2\right]$, while the blue curves show the value of $B(\theta)$.

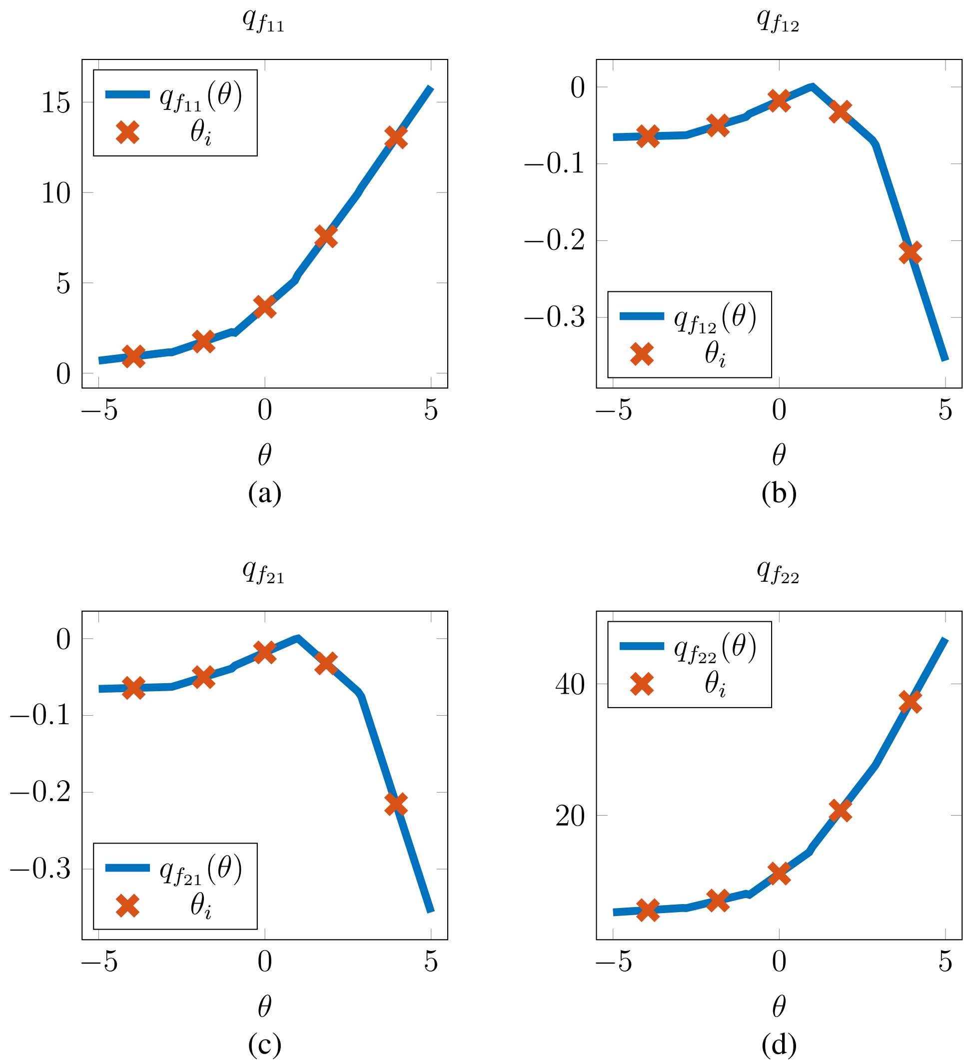


Figure 4.3: Piece-wise linear parametrization of the discrete-time state-space matrix $Q_f(\theta)$, where each plot (a)-(d) represent one element of Q_f. The red cross marks are the computed middle points θ_i such that the function is constant in the interval $\left[\theta_i - d/2 \quad \theta_i + d/2\right]$, while the blue curves show the value of $Q_f(\theta)$.

$i_0(k)$ and $U_{d0}(k)$ as the filtered versions of $P_{CPL}(k)$, $i(k)$ and $U_d(k)$ respectively. Furthermore, we let the parameter $\theta(k) = P_0(k)/U_{d0}(k)^2$.

4.3.3 Varying Input Limitations

Since our control input u in the MPC depends on the operating point - it is defined as $u = P_{stab}/U_{d0}$ - the limits given to the MPC controller have to be scaled with the operating voltage. This means that

$$\begin{aligned} \overline{u}(k) &= \frac{\overline{P}}{U_{d0}(k)} \\ \underline{u}(k) &= \frac{\underline{P}}{U_{d0}(k)} \end{aligned} \tag{4.13}$$

where $\overline{P}$ and $\underline{P}$ are the upper and lower bounds on the power modification. These limits could potentially vary with time as well. The upper and lower limits in the reference frame of the MPC, $\overline{u}(k)$ and $\underline{u}(k)$, will be updated each time step, but assumed constant across one prediction horizon in the optimization problem. This was done to limit the scope to consider typical constraints as listed in (2.72).

4.3.4 Optimization Problem Formulation for LPV-MPC

We will now present the updated QP for the LPV-MPC. It looks as follows

$$\begin{array}{lll} \text{minimize} & \sum_{i=0}^{N-1} \left(x_i^\top Q x_i + u_i^\top R u_i \right) + x_N^\top Q_f x_N & \\ \text{subject to} & x_{i+1} = A x_i + B u_i & i = 0, 1, \cdots, N-1 \\ & \underline{u} \leq u_i \leq \overline{u} & i = 0, 1, \cdots, N-1 \\ & A = A(\theta_k), \quad B = B(\theta_k), \quad Q_f = Q_f(\theta_k) & \\ & \underline{u} = \underline{u}(k), \quad \overline{u} = \overline{u}(k), \quad x_0 = x(k) & \end{array} \tag{4.14}$$

where $\theta_k = \theta(k)$. So to summarize, each time step k the QP (4.14) is solved with state-space matrices A and B and terminal cost Q_f fixed to the value of which they had at time step k. The limits on the control inputs are also fixed for the entire prediction horizon N. The solution to the QP is the optimal control input sequence N samples into the future. Only the first computed input u_0 is applied, at time step k. Then in the next time step the QP is solved once again.

Figure 4.4 shows a block diagram of the MPC and its prefilter. The prefilter takes $U_d(k)$ and $P_{CPL}(k)$, obtains the operating point using the filter (4.12),

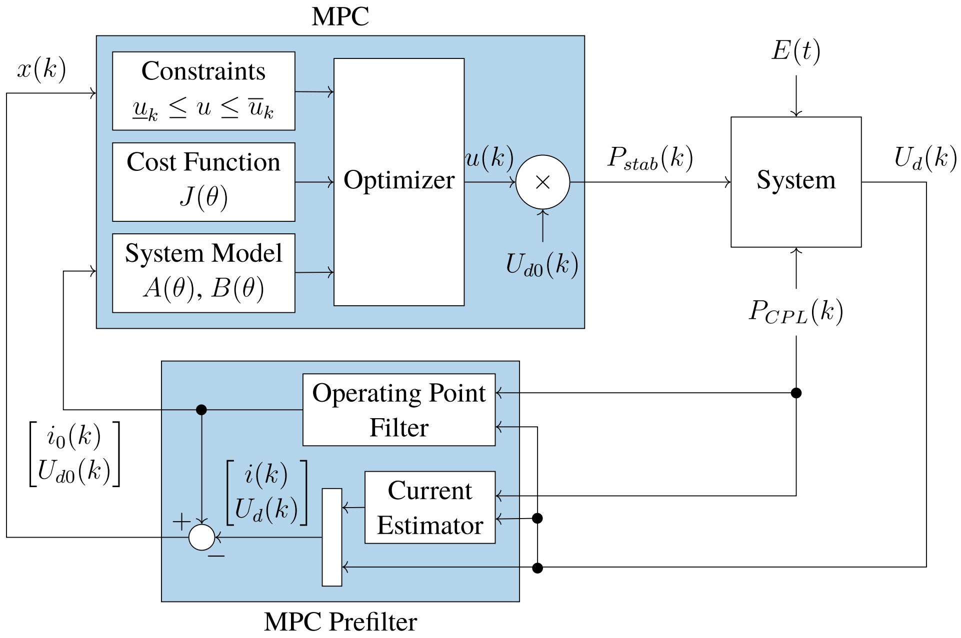


Figure 4.4: Block diagram of input filter and converter system controlled with LPV-MPC.

estimates the line current using the current estimator (4.7), and computes the state vector (4.9). The state $x(k)$, and the operating point are sent to the MPC, which computes the optimal control input $u(k)$, given the constraints, the cost function $J(\theta)$ and the system model. The control input is then multiplied with the operating voltage $U_{d0}(k)$ to obtain $P_{stab}(k)$, the power modification, since the control input $u(k)$ which the MPC computes is equal to the converter input current. The power modification $P_{stab}(k)$ is then sent to be realized in the converter.

4.3.5 Software Implementation

The controller, defined by the QP (4.14) was implemented using CVXGEN [24]. This is a web interface created by Jacob Mattinley which can create code to solve QPs such as (2.7). This web tool allows the user to define an optimization problem in simple and intuitive syntax, which is close to the mathematical formulation. It then generates a solver written in C-code for the problem which was specified. This script can be called in MATLAB or integrated into a larger C program. CVXGEN implements all the methods for speeding up the online optimization which were brought up in Chapter 2.9.2. This makes it possible to run the MPC in real time even with high sampling frequency.

Chapter 5

Experimental Setup

This chapter explains how the MPC regulator from Chapter 4 has been evaluated. In Section 5.1 we explain how the performance of the MPC regulator will be tested, to see how it compares to the specifications set up in Section 2.5. Then, in Section 5.2 we discuss practical implementation aspects such as computational complexity and hardware integration.

5.1 Performance Evaluation

The performance of the MPC regulator will be tested by analyzing the step response of the closed loop system in various operating points. We will do steps in both line voltage E and in reference power P_{CPL} and analyze how it affects the input filter voltage U_d and the stabilizing power P_{stab}.

We will also see how limiting the control input in different ways affects the step response. We will consider two types of control input limitations. First we will limit the control to a band $\pm P_{\max}$ kW which is a typical operating scenario. The control is limited to a range in order to limit the impact of the power modification on other control objectives, such as limiting the torque contribution which introduces mechanical stress. The second scenario considered here is when only negative power modifications are allowed. This a typical situation when the system is close to a current limitation for example.

The conventional way of dealing with these limits at Bombardier is to let the H_∞^{sub} regulator compute its control input. Then if it is not possible to realize the control input because of some limitation, the input is truncated. The contribution of the power modification which was cut off can then be realized using a shunt resistor which can be connected in parallel with the converter. This method leads to power losses, since power is wasted through the shunt resistor. It could therefore be advantageous to be able to stabilize the system without this shunt resistor.

The simulations evaluating the step response of the system have been done in MATLAB/Simulink and in Bombardiers own software-in-the-loop (SIL) simulator. Both environments are explained further below.

5.1.1 MATLAB/Simulink

MATLAB and Simulink have been used to test the MPC regulator on an ideal CPL system. The CPL and input filter from Figure 2.1 were modelled in Simulink using the electronics simulation platform PLECS. This environment has been used to evaluate the regulator in a scenario where the internal model of the MPC corresponds precisely with the target system, i.e. an ideal CPL. The performance will therefore mainly be affected by the controller structure and the tuning of the controller, without being degraded by modelling errors.

The Simulink model can also be used to precisely test the regulator's sensitivity to errors in parameter values. We can do that by giving the MPC regulator the wrong information about the system, such as the wrong value for the inductance of the input filter and see how that affects the performance.

5.1.2 The Software-in-the-Loop Simulator

For (SIL) simulations we use a proprietary software of Bombardier Transportation, capable of testing the actual source code which runs in Bombardiers converter modules, on a desktop computer. The SIL simulates the train environment (input filter, converter and motors) so that code can be tested in an environment which resembles a real train propulsion system quite well. The SIL has been used to evaluate the regulator on control software which is much closer to the real propulsion system in the train.

The train application which the MPC regulator was evaluated on is called London Central Line (CLT) - a train line in the London Underground. It was chosen because it has converter-load-dynamics which are quite well approximated by a CPL, around the natural frequency ω_0. Its dynamics are quite representative of the dc powered trains Bombardier works with. The input filter in CLT also has a low natural frequency, which means that the bandwidth of the controller doesn't need to be too high. This puts less restrictions on the sampling frequency of the MPC, and we can use a lower sampling frequency without impacting performance too much.

5.1.3 Evaluation Details

We will evaluate the step response of the system at three different operating points. These points correspond to driving the train at a fixed speed while varying the applied motor torque. For a CPL the specific value of the torque and the speed do not matter for performance, but only the total power. The total power is approximately given by

$$P = N_m T_m \omega_m \tag{5.1}$$

where T_m is the torque per motor, ω_m is the mechanical motor speed and N_m is the number of motors (which for CLT is 4). This approximation disregards potential power losses but modelling those are out of the scope of this thesis.

The operating points which are considered are found in Table 5.1. The first three points correspond to driving the train at full traction (maximum torque), coasting (zero torque), and full brake (maximum negative torque). The operating points come from the speed-torque curve in Figure 5.1, which indicates the maximum torque allowed for different speeds, in traction and in braking, for the CLT train application. The chosen operating points are indicated with blue diamonds in Figure 5.1. The simulations of the step response according to these operating points will be done in Simulink and in the SIL simulator. In Simulink, where we are simulating a true CPL, the torque and the speed do not matter. The operating point is fully determined by the nominal voltage and the nominal power. The nominal power for the various operating points are given in the right-most column in Table 5.1. The nominal voltage for the CLT application will be according to Table 5.2 for all simulations. We use the H_∞^{sub} regulator as a benchmark since it is the regulator which is conventionally used at Bombardier for stabilization, and is already integrated into the code for the SIL simulator. In Simulink we are also able to simulate the H_∞ regulator from Chapter 2.7.3. Tests will be done with and without control input limitations.

Even though the CLT system behaves a lot like a CPL, at low frequencies, it is not a true CPL. For example, the torque regulation dynamics of the converter are dependent on the speed of the motor. One consequence of this is that it can display different dynamics for the same operating power, depending on which torque and speed is applied. For higher motor speeds the torque dynamics are not as fast, and the CPL approximation does not work as well as for lower speeds. To verify this behavior and to test the MPC regulator's robustness to it we will therefore evaluate the step response of the system for one more operating point which results in the same total power as operating

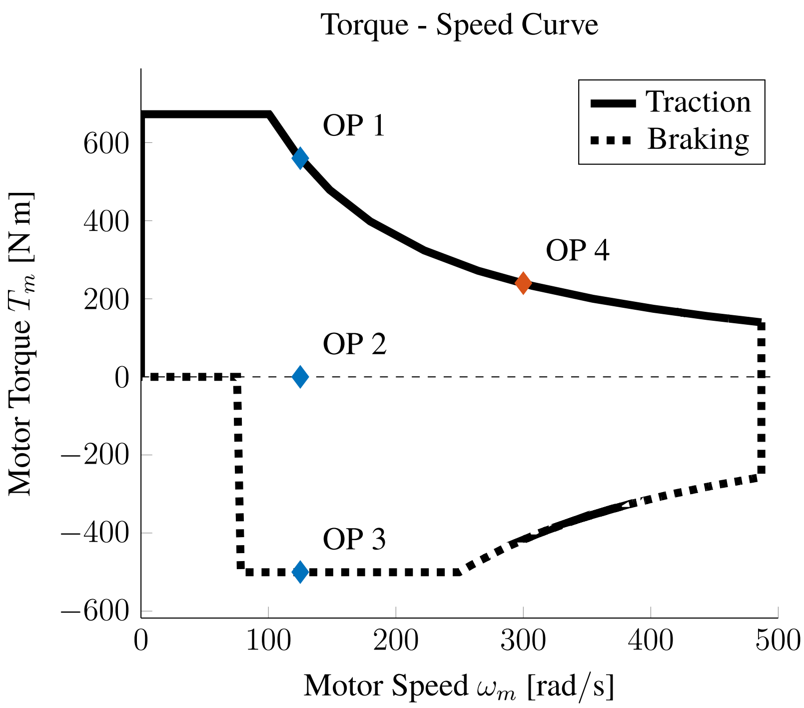


Figure 5.1: Speed-torque curve for CLT application. The operating points (OPs) from Table 5.1 are marked and labeled in the curve.

Table 5.1: Operating Points for CLT Simulations

		Operating Point Parameters		
№	Operating Point	Torque / Motor	Motor Speed	Power P_0
1.	Full Traction	560 N m	125 rad/s	300 kW
2.	Coasting	0 N m	125 rad/s	0 kW
3.	Full Brake	−500 N m	125 rad/s	−234 kW
4.	High Speed	240 N m	300 rad/s	300 kW

point 1 in Table 5.1, but has a different speed than what was used in the other three operating points. This fourth operating point which we will evaluate performance in is indicated with an orange diamond in Figure 5.1. This way we can compare the response in the first operating point to that of the fourth to see the impact of the varying degree of CPL behavior the CLT system displays. The response which we will see is related to the regulator's robustness towards modelling errors on the load side. As we know, the MPC regulator does not take torque dynamics into account. By comparing the response in operating points with different torque dynamics we can see the regulators robustness towards those. We refer to Table 5.1 again for the precise values for the operating point.

5.1.4 Frequency Domain Analysis

For the LQR, and the H_∞- and H_∞^{sub}-regulators in Chapter 2.7, we have explicit expressions for the regulators. We can therefore analyze the frequency response of the closed loop system while controlled with each one of these regulators. The transfer functions which we will consider are F_e and F_p from (2.40), which are the transfer functions from line voltage E and power reference P_{CPL} to the control input P_{stab}. Ideally these should be zero for all frequencies, but this is not possible. We can however compare how well the controllers suppress the peak of these transfer functions. We will also look at the transfer functions from E and P_{CPL} to the states i and U_d. For a well damped system these transfer functions should not have large resonance peaks.

Furthermore, we will look at the Nyquist curves of the admittance Y_{DC} and the loop gain L_{DC}. It is interesting to see how the regulators affect the admittance Y_{DC} since we know from (2.4) that, for a true CPL, it is equal to $-P_0/U_{d0}^2$ for all frequencies, if no stabilization is applied. It is hence fixed on the negative real axis for positive values of P_0. The active stabilization

Table 5.2: Filter Parameter Values for Simulation

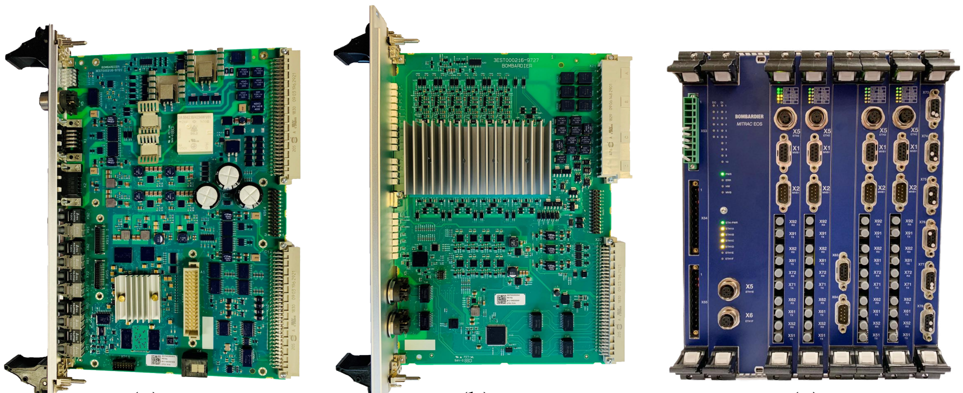
Quantity	Symbol	Value		Unit
		CLT	MCX	
Filter resistance	R	18.8	5	mΩ
Filter capacitance	C	18.0	8	mF
Filter inductance	L	8.4	1.6	mH
Filter resonance frequency	ω_0	81.3	279.5	rad/s
Filter damping coefficient	ζ	0.0138	0.0056	
Nominal input filter voltage	U_{d0}	630	1500	V

which we apply will modify the admittance. From Chapter 2.3.1 we know that the loop gain L_{DC} is given by $Y_{DC}Z_{DC}$, and that without stabilization the Nyquist curve of the loop gain encircles the point $(-1, 0)$ in the complex plane. According to the simplified Nyquist stability criterion, this makes the system unstable. By plotting the loop gain for the system with stabilization applied we can expect to see how the loop gain differs from the case where there is no stability.

In the SIL simulator it is possible to identify the frequency response from different inputs and outputs. This is done by exciting a desired system input with a sinusoidal disturbance signal of known amplitude and measuring how it affects different output signals. The frequency of the sinusoidal is varied across a spectrum to obtain an empirical frequency response. This way we can obtain the frequency response of the MPC regulator and compare it to the same transfer functions for the H_∞ regulator as well as the LQR regulator. In particular, we will identify the transfer functions of the admittance Y_{DC} and the loop gain L_{DC} for the MPC regulator. Since the MPC optimization problem is approximately the same as the infinite horizon LQR problem in the case of no control signal limitations we can also expect the results from the analytical solution of the LQR regulator to give an indication of how the frequency response of the MPC would be in the same linear scenario.

5.2 Real-Time Implementation Aspects

Since the MPC regulator solves an online optimization problem it is important to verify that the regulator can run in real time on the intended hardware. At Bombardier Transportation the processing is done by a processor located on



(a) (b) (c)

Figure 5.2: Photos of controller boards, from Bombardier Transportation [25]. Photo (a) shows the controller board, (b) shows the IO board, and (c) shows the two boards mounted vertically, next to each other, in a rack.

a controller board which is integrated into the converter unit. This controller board, which can be seen in Figure 5.2, is the lowest level control unit for the train propulsion system. Figure 5.3 shows a schematic of the controller board, integrated in a motor control application, where the part of the circuit which corresponds to the input filter has been labeled. The motor converter and the motors to the right in the drawing are what is simplified to the CPL in Figure 2.1. For the MPC regulator to be viable for use in real-time it has to be able to run on this controller board alongside all the other processes.

5.2.1 Evaluation in Real-Time Environments

To make sure the MPC regulator can run in real time it will be evaluated in Bombardier's real-time simulator. This simulator is a hardware-in-the-loop (HIL) simulator, meaning that the code runs on the hardware specific processing unit, namely the controller board, while connected to a machine capable of simulating the behavior of the train propulsion system, in real time. For HIL simulations we have used dSPACE's LabBox, of which a photo can be seen in Figure 5.4. The controller board (see in Figure 5.2) is hooked up to the LabBox and it is then able to simulate the propulsion system in real-time. For the application which we have used it uses a sample time of 12 µs.

The MPC controller will also be tested in Bombardier's power laboratory (PowerLab). This is a testing lab where the entire propulsion system of the train can be tested. In this testing environment the MPC regulator, which has been integrated into the rest of the propulsion system software and is running on the controller board, is tested together with the real converter and motor

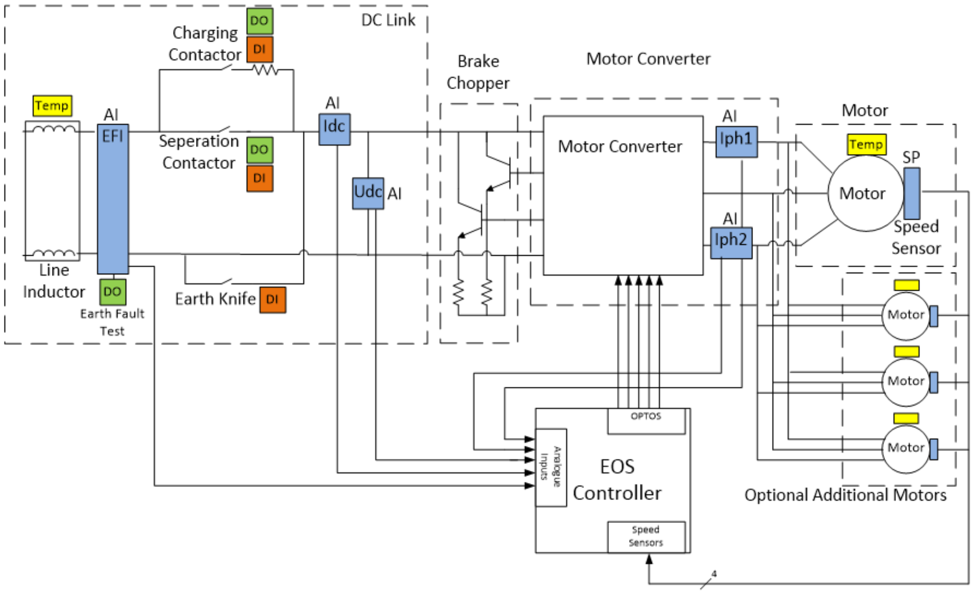


Figure 5.3: Diagram of the controller in an motor converter application, from Bombardier Transportation [26]. The controller itself can be seen in the bottom part of the drawing. It receives input signals from the input filter (labeled DC-link) in the left part of the drawing, and from the motors to the right. The computed control input is realized through the switching of transistors in the motor converter.

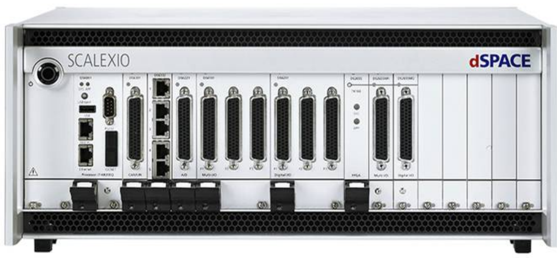


Figure 5.4: Photo of a dSPACE SCALEXIO LabBox, which has been used for real-time simulations. Photo from dSPACE [27].

hardware. In this test the regulator is controlling the real converter seen in Figure 1.2 and the load is a real motor, namely the one seen in Figure 1.3. The propulsion motor is connected back-to-back with another motor which can be used to introduce a counterforce to emulate a load. This testing environment is as close as it gets to the "real world" without having to test on a moving train.

These tests have primarily been used to evaluate practical aspects of running online MPC. For example, they have been used to verify that it was possible to implement the MPC regulator and fully integrate it as a functioning part together with all the other control systems which run on the controller board. This is an important aspect to evaluate since a regulator in an industrial application most likely has to share computational resources in a processor environment which is already well utilized. This is the case with the controller board in Bombardier's propulsion system. With the controller board it is also possible to measure the execution time of each processing loop. From this we have been able to evaluate the computational burden of running MPC. The CPU on the controller board 5.2 is an ARM Cotex A9 with a clock frequency of 667 MHz.

The application which was available for testing in the HIL simulator and in PowerLab, at the time this master thesis, is different from the CLT application. This application is called MCX. This system has an input filter with a much higher natural frequency ω_0, and it is not as well approximated by a CPL, in this frequency range. Unlike a CPL the system is even stable without active stabilization. Had a system with dynamics which are well approximated by a CPL in the region around the natural frequency been configured in the PowerLab, performance could have been evaluated there as well. Since the behavior of the MCX system is so different from an ideal CPL around the natural frequency ω_0, the evaluation in the HIL simulator and in PowerLab can also be seen as another test of the robustness of the MPC regulator. The MPC regulator is expecting a system which behaves like a CPL but is getting something completely different. The input filter parameters for the MCX application is given in its corresponding column in Table 5.2.

5.2.2 System Modifications for Real-Time Functionality

As mentioned in the previous subsection, the MCX application has system dynamics around its natural frequency which are quite different from a CPL. One way of dealing with this mismatch would be to model the non-CPL behavior of the system and incorporate it into the internal model of the MPC. This would however increase the order of the state space model. It is also outside the scope of this thesis. As mentioned in Chapter 2.9.2 the computational complexity of the MPC optimization problem is

$$\mathcal{O}(N(n^3 + n^2 m)) \tag{5.2}$$

at best. Extending the model, to reflect more complicated dynamics, would mean adding at least one more state, and likely two more states. According to (5.2) the complexity grows as n^3 where n is the number of states. So, to go from 2 to 3 states would increase the complexity by a factor $3^3/2^3 \approx 3.4$ and adding two more states would increase the complexity by a factor $4^3/2^3 = 8$. Extending the model is therefore something which one wants to avoid in MPC, if computational speed is important.

The H_∞^{sub} regulator in Chapter 2.7.4, which is the conventional input filter stabilization regulator at Bombardier, executes on the control board with an execution period of 50 µs. Since the MPC optimization problem is quite computationally burdensome, even for a system of lower degree, it was not possible to have it execute that fast. Furthermore, from (5.2) we know that the complexity of the MPC optimization is linear in prediction horizon N. The prediction horizon N is proportional to the natural frequency of the open loop system and the sampling frequency f_s according to

$$N \propto \frac{f_s}{\omega_0}$$

Hence, if the sampling frequency is increased, the number of samples has to increase as well in order for the MPC prediction horizon to include enough time in its prediction. This means that increasing the sampling frequency not only decreases the time available for the optimizer to find a solution, it also makes the optimization problem more complex.

For this thesis the MPC regulator was placed on a level on the controller board which executes at 5 ms. The reason for this choice was that it was an execution speed in the range of what was deemed achievable with MPC with the computational power available. For example [16] was able run MPC at the same frequency with a higher order system. One difference is however that in our case the MPC has to share computational resources with other tasks which are also running on the EOS board. Another reason for why 5 ms in particular was chosen was that there already existed a task level in the software with that execution period, simplifying implementation.

One practical implementation problem of placing the MPC on such a slow task compared to the 50 µs level which the other control loops operate on, is that we get aliasing, or frequency folding, if the MPC directly samples the signals from the faster tasks. To prevent aliasing a first order low pass filter was placed on the input of the MPC. If we refer back to Figure 4.4, the low pass filter was added to the state $x(k)$, between the output of the MPC prefilter and the input of the MPC regulator. Hence, the MPC prefilter and the low pass filter are executed on the fast 50 µs task while the MPC regulator itself is slower and operates on the 5 ms task. However, the low pass filter degrades the phase margin of the closed loop system. The phase margin is already not great because the real-time system introduces some delays which were not present in the Simulink model. We would therefore like to compensate for these and increase the phase margin.

In order to increase the phase margin a lead filter was added to the input of the MPC regulator, in series with the previously mentioned low pass filter. The lead filter was designed to add phase margin at the natural frequency, and invert the phase of the unmodeled dynamics, making the load more CPL-like, and thus increasing the stability margin of the MPC regulator. The lead filter is designed as a continuous time filter and is on the form

$$K\left(\frac{\alpha\tau s+1}{\tau s+1}\right) \tag{5.3}$$

where the upper and lower cutoff frequencies of the filter are given by $1/\tau$ and $1/(\alpha\tau)$. The parameter α is related to the total phase gain. The maximum phase gain is

$$\arcsin\left(\frac{\alpha-1}{\alpha+1}\right) \tag{5.4}$$

Finally, the gain K is chosen as $1/\sqrt{\alpha}$ which results in a gain of 1 at the point of maximum phase gain, which in this case is the resonance frequency. One drawback of the lead filter is that it amplifies high frequencies, which means that noise with a high frequency is amplified as well.

Another consequence of having the MPC regulator on a slower task of 5 ms (or equivalently 200 Hz) is that the output of the MPC regulator will be seen as a jagged stair signal with sharp edges, from the perspective of the faster 50 µs task. The stair-like signal generated by the MPC regulator will introduce a 200 Hz signal (and higher frequency modes of that signal) into the system. This could present itself to be an issue if the system dynamics aren't well damped enough. If that should be the case this 200 Hz mode will be induced into the system.

Chapter 6

Evaluation on Ideal CPL

In this chapter we present the results of tests which have been made on an ideal CPL, as described in Chapter 5. The MPC regulator results are compared with the performance of the H_∞ regulators from Chapter 2.7. As explained in Chapter 5, simulations have been done in MATLAB/Simulink.

For the performance test the MPC has been evaluated using model parameters corresponding to the CLT train application. The model parameters corresponding to that train application can be found in Table 5.2. Throughout this chapter the LQR uses parameters according to Table 6.1, while the MPC regulator uses parameters according to Table 6.2, unless otherwise stated. The length of the prediction horizon, N, which is 20, corresponds to approximately $1.3 \times \frac{2\pi}{\omega_0}$. This is 1.3 times the period of the natural oscillations of the unstabilized system. Simulations showed that the performance was degraded if the prediction horizon covered less than one period. Very long prediction horizons also didn't make the performance noticeably better. This particular value for N was chosen since it was a nice round number, longer than one period.

In Section 6.1 we present results of frequency plots, such as Nyquist diagrams and Bode plots. In Section 6.2 we present the results of MATLAB/Simulink simulations of step responses in various operating points. Finally in Section 6.3 we present results where we have tested the MPC regulator's robustness towards errors in the input filter model. A discussion of the results with interpretations are given later, in Chapter 8.

6.1 Frequency Domain Results

In this section we present results which are in form of frequency data. We present Nyquist diagrams of the admittance Y_{DC} and the loop gain L_{DC} for the

Table 6.1: LQR Parameters

Parameter Description	Symbol	Value
Input filter voltage weight	q_{Ud}	5
Control input weight	q_r	1
Filter time constant	τ	$(\frac{\omega_0}{4})^{-1}$

Table 6.2: MPC Parameters

Parameter Description	Symbol	Value
Input filter voltage weight	q_{Ud}	5
Control input weight	q_r	1
Final weight states	$\bar{Q}$	$\text{diag}([0, 5])$
Final weight control input	$\bar{R}$	1
Prediction horizon	N	20
Sampling frequency	f_s	200 Hz

different controllers. We present magnitude plots of the frequency response from the external excitation signals E and P_{CPL} to the control input P_{stab}, and to the states U_d and i. The frequency domain results are plotted for the LQ-, the H_∞-, and the H_∞^{sub}- regulators. For the LQR we use weights according to Table 6.1. The filter constant for the operating point filter (3.6) has the value of $\tau = \omega_0/4$, namely one fourth of the natural frequency. Throughout this section we will use solid lines to indicate that a result comes from an analytical expression (a transfer function, which has been drawn), and dashed lines to indicate that the result comes from identification in the SIL simulator.

6.1.1 Nyquist Diagrams

In this subsection we present Nyquist plots for the admittance Y_{DC} and the loop gain L_{DC} for the different regulators.

From Chapter 2.3.1 we know that the admittance is equal to $-P_0/U_{d0}^2$ and is negative real for positive powers, when no stabilization is applied. In Figure 6.1 we see how the different regulators (LQR, MPC, H_∞ and H_∞^{sub}) modify the admittance Y_{DC}. For the system to be stable the admittance should have positive real part at the natural frequency ω_0. This is achieved for all the regulators. The natural frequency is marked with a red dot in the Nyquist curve. For LQR and H_∞ we have the analytical result from the transfer functions.

For H_∞^{sub} we have plotted both the result from the transfer function, as well as the identified admittance from SIL simulations. For the MPC regulator we do not have an analytical expression for the regulator, so we only have the identified admittance from the SIL simulator. We know from the theory presented in Chapter 4 that the MPC regulator should achieve a result which resembles the LQR in linear operation, i.e. when there are no control input limitations. If we look at the Nyquist diagram of Y_{DC} we see, however, that the curve for LQR and MPC differ a bit. We do expect some difference in identified admittance though, since we haven't accounted for the power modification which is caused by power losses. If we compare empirical and theoretical admittance for the H_∞^{sub} we see that there is some difference there as well. The difference between the theoretical admittance for the LQR and the identified admittance for the MPC regulator appear to be of the same order as what we see for the H_∞^{sub} and hence we conclude that the LQR and the MPC regulator likely behave very similarly in linear operation. The jitter and the jumps which can be seen in the dashed curves is due to measurement noise.

Recall that the loop gain is given by $Z_{DC}Y_{DC}$, that is the output impedance of the input filter multiplied with the input admittance of the CPL. Without any stabilization the Nyquist curve for the loop gain looks like Figure 2.2 from Chapter 2.3.1. Figure 6.2 shows a similar Nyquist plot for the loop gain L_{DC}, when stabilization is applied. Figure (a) shows the entire Nyquist curve while (b) shows the curves zoomed in around the point $(-1, 0)$. For stability the Nyquist curve should not encircle this point. We see that the LQR and the MPC regulators achieve this goal, as do the H_∞ and H_∞^{sub} regulators. Again, there is a mismatch between the empirical results drawn with dashed lines and the theoretical results drawn in solid lines, which can be attributed to power modifications caused by power losses which we have neglected in our model.

6.1.2 Magnitude Plots From Excitation Signals to Control Input

Part of the performance specifications in Chapter 2.5 was to limit the use of power modification P_{stab}. In Figure 6.3 the magnitude of the frequency response of the transfer functions (a) F_E and (b) F_p, found in (2.40), are plotted for the CLT application. These are the transfer functions from the line voltage E and the reference power P_{CPL} to the control input P_{stab}. F_E is a measure of how much variations in line voltage E affect the power modification P_{stab}. For perfect disturbance attenuation this transfer function should be zero, but since the power modification P_{stab} is needed for stability this is not possible.

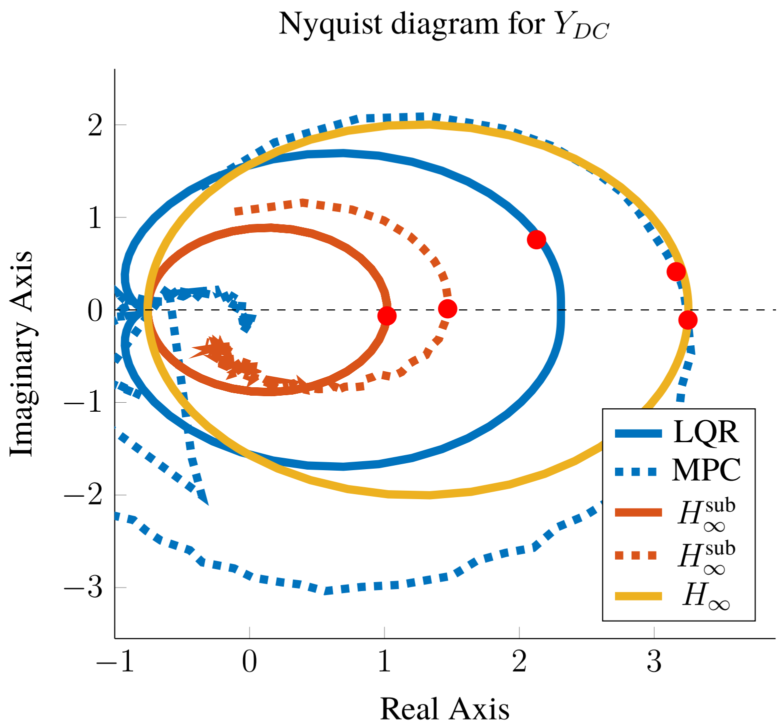


Figure 6.1: Nyquist diagram of the admittance Y_{DC} for the CLT application. The operating point is according to 1 in Table 5.1. Nyquist curves drawn in solid lines come from analytical results. Curves drawn with dashed lines come from identification in the SIL simulator. Red dots mark the location of the resonance frequency.

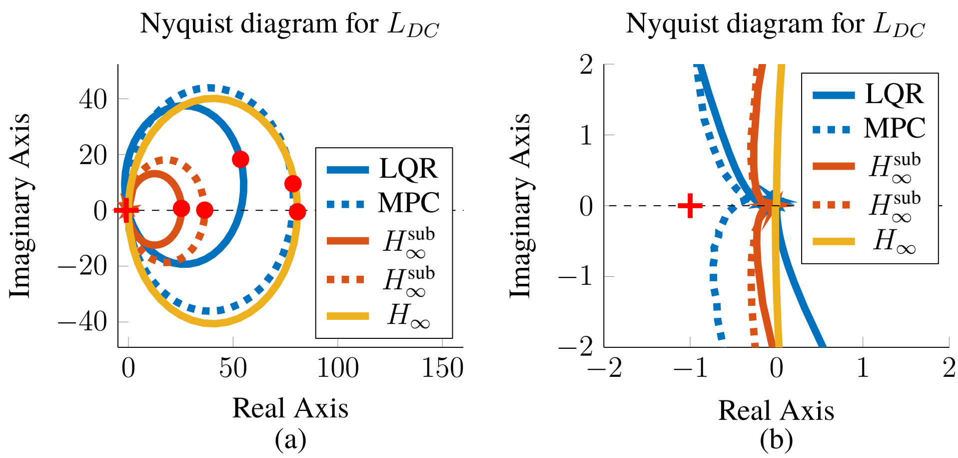


Figure 6.2: Nyquist diagram for the loop gain L_{DC} for the CLT application. The operating point is full traction according to 1 in Table 5.1. Plots in solid lines are theoretical results from the transfer functions. Plots drawn with dashed lines come from identification in the SIL simulator. Red dots mark the location of the resonance frequency. The point $(-1, 0)$ is marked with a red cross. (a) shows the entire Nyquist curve. (b) shows the same curves but zoomed in around the point $(-1, 0)$ in the complex plane.

In Figure 6.3 the horizontal gray lines mark where the gain is $1/\sqrt{2}$. Above this line we have amplification of signals. The LQR regulator, and in turn the MPC regulator was tuned to give a response which resembles the H_∞ and H_∞^{sub} regulators.

If we look at Figure 6.3 (a) we see that for the LQR regulator the peak is slightly higher and the bandwidth of the amplification wider than for the H_∞^{sub} regulator. The gain goes to zero in stationarity which means $P_{stab}(t) \to 0$ as time goes to infinity. This is what we want. The way the LQR is tuned we can expect it to use more control input than the H_∞^{sub} and the H_∞ regulators when trying to dampen variations in the line voltage E. The response can also be expected to be somewhat slower. Since the MPC regulator is tuned in the same way as the LQR, and since they seem to have loop gains which correspond with each other it is expected that the MPC regulator should behave in a similar manner.

If we instead look at Figure 6.3 (b) we see the magnitude of the frequency response of F_p. This is the transfer function from the reference power P_{CPL} to the power modification P_{stab}. Recall that the performance specifications in Chapter 2.5 states that we want to limit interference with the control objective of following the power reference. Perfect reference following would mean that this transfer function F_p is zero for all frequencies (recall that $P = P_{CPL} + P_{stab}$). Since F_p is non-minimum phase this is not possible; the gain has to be greater than 1 at the natural frequency ω_0. Again, the gray horizontal line marks where the gain is larger than $1/\sqrt{2}$, where we have amplification. This is the frequency range where the stabilization will disturb the reference following. We see that the LQR regulator reaches above this threshold for a wider frequency range than the H_∞^{sub} regulator. The peak however is slightly lower. The H_∞ regulator on the other hand has the lowest peak out of all the regulators but the peak is above $1/\sqrt{2}$ for a wider frequency range. Like before, we expect the MPC regulator to exhibit similar behavior as the LQR regulator in linear operation.

6.1.3 Magnitude Plots From Excitation Signals to States

In Figure 6.4 the magnitude of the frequency response of the transfer functions from the line voltage E and the power reference P_{CPL} to the states i and U_d have been plotted for the CLT application. The transfer functions have been plotted for LQR, H_∞ and H_∞^{sub}. An operating power of 300 kW, corresponding to full traction according to Table 5.1, is used and all parameters are according to the CLT column in Table 5.2.

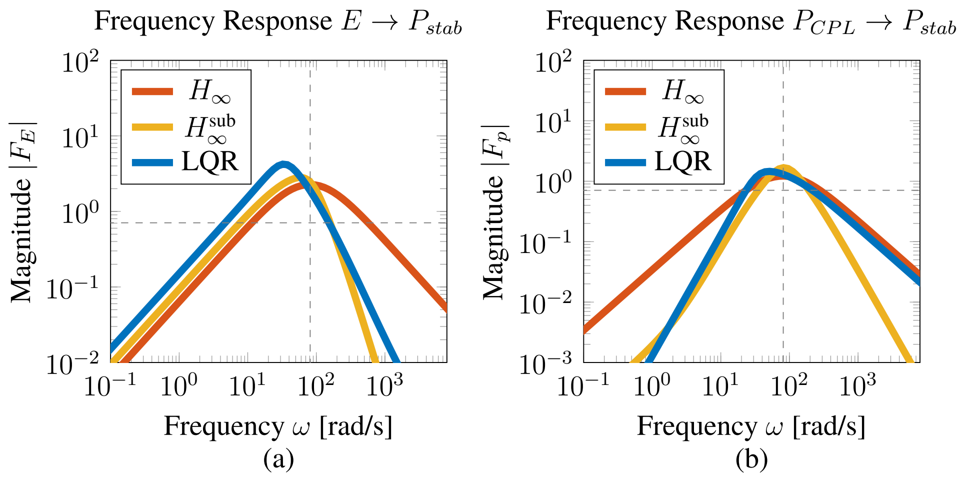


Figure 6.3: Frequency response of the transfer functions F_E and F_p from (2.40) for the CLT application, plotted in log-log scale. Plot (a) shows $|F_E|$ and (b) shows $|F_p|$. The operating power is 300 kW. The gray horizontal line marks the magnitude $1/\sqrt{2}$ and the vertical line marks the value of the natural frequency ω_0.

One of the performance specifications in Chapter 2.5 was to have well damped system dynamics. Well damped system dynamics are characterized by low resonance peaks in transfer functions from external signals. Figure 6.4 (a) shows the transfer function from line voltage E to the line current i. We see that for all the regulators there is a peak around the natural frequency of the input filter. The LQR, however, gives a higher peak than the other two regulators. This might be expected since we did not put a cost on the line current in the LQR cost function. If we instead look at the transfer functions from E and P_{CPL} to the input filter voltage U_d (see Figure 6.4 (b) and (d)) we see that the LQR does better when it comes to disturbance attenuation. The peak is lower for the LQR than for the H_∞^{sub} in both cases. This is reasonable since that is what the LQR explicitly has been designed for, through the choices of weights in its cost function (2.55). We do however see that the bandwidth of the transfer function from E to U_d is slightly lower for the LQR than for the other two regulators. This suggests that we can expect a slightly slower step response from E to U_d, for the LQR and also for the MPC regulator. It does however mean that the LQR and hence also the MPC regulator should attenuate disturbances in the input filter voltage U_d better than what the H_∞^{sub} does. This is an important characteristic needed to avoid over-voltages and under-voltages. Finally, the H_∞ regulator, achieves better damping than both the LQR and the H_∞^{sub} regulator. It is worth noting once again though that this regulator assumes that the line voltage E can be measured. This is a signal which neither of the other regulators have available.

6.2 MATLAB/Simulink Simulation Results

This section presents simulation results for the MPC regulator which were done in MATLAB/Simulink. We use parameters according to the CLT application (see Table 5.2). The MPC uses a sample frequency of 200 Hz, which is a sample frequency which we could also use in our real-time implementation (see Chapter 5.2 and Chapter 7). In the MATLAB/Simulink simulations a filter constant of

$$\nu = \frac{1}{4}\frac{\omega_0}{2\pi f_s}$$

was used for the operating point filter (4.12) which was described in Chapter 4.3.2. This corresponds to a filter with a cut-off frequency which is one fourth of the natural frequency, ω_0, normalized by the sampling frequency f_s.

The MPC regulator is evaluated by making steps in the excitation signals E and P_{CPL} and evaluating the response in terms of input filter voltage U_d and control signal P_{stab}. That way we can see how well damped the system response is. We will also see how much control input is used, in terms of the power modification P_{stab}. By introducing control input limitations, we can also see how the MPC regulator behaves under those circumstances. The performance is compared to that of the H_∞- and H_∞^{sub}- regulators from Chapter 2.7.

6.2.1 Response to Line Voltage Steps

In this subsection we present simulation results relating to the response of the system when it is subject to a step in line voltage E. We have simulated the step response for the operating points corresponding to full traction, coasting and full brake in Table 5.1. Figure 6.5 shows the response in full traction, Figure 6.6 the response in coasting and 6.7 shows the response during full brake. The step in line voltage is of magnitude 50 V. Since the Simulink model is a true CPL, the operating point is determined fully by the nominal power (listed in Table 5.1) and the nominal input filter voltage, which is found in Table 5.2. For each operating point the response has been simulated for three different

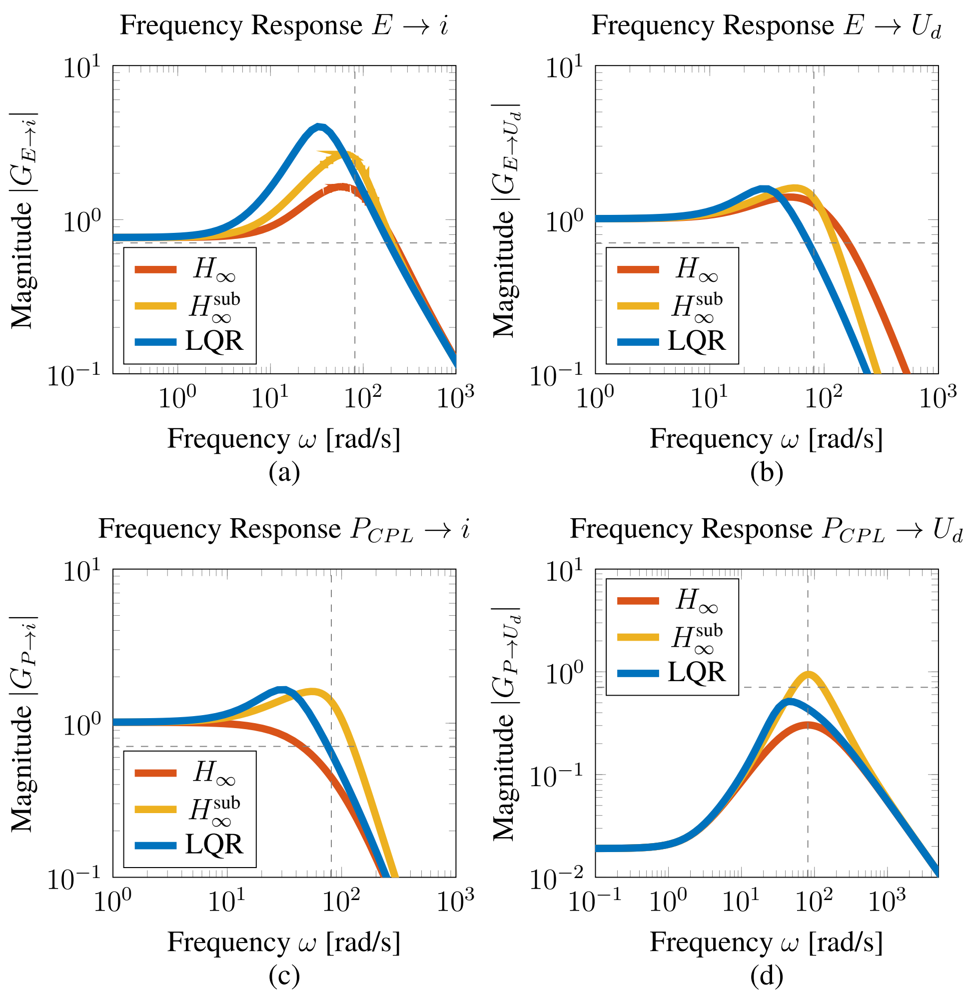


Figure 6.4: Magnitude of the frequency response from the input signals E and P_{CPL} to the states U_d and i, for the CLT application. Plotted in log-log scale. Plotted for three different regulators, H_∞, H_∞^{sub} and LQR, at operating power of 300 kW. The gray horizontal line marks the magnitude $1/\sqrt{2}$ and the vertical line marks the value of the natural frequency ω_0

types of control input limitations: no limits, band limited control input, and upper limited control input.

Looking at Figure 6.5 which shows the response when the load draws full power, we see that the MPC regulator gives a well damped response in input filter voltage U_d. This is most notable in Figure 6.5 (a) where there are no control input limitations. The peak of the response is lower than what the H_∞^{sub} regulator achieves. The response of the MPC regulator is however a little bit slower. It also uses more control input than the H_∞^{sub} regulator. This corresponds to what we saw in the frequency domain results in Section 6.1.2 and Section 6.1.3.

When the control input is band limited such as can be seen in Figure 6.5 (c) and (d) the MPC and the H_∞^{sub} achieves similar damping of the voltage U_d. We see however that the way the MPC regulator has been tuned it will use more control input than the H_∞^{sub} regulator, in order to minimize the ripple in the voltage U_d.

When the control input is limited to only negative values such as can be seen in Figure 6.5 (e) and (f), the MPC achieves a much more well damped response than what the H_∞^{sub} regulator does. Looking at the response in power modification we see that the MPC uses a large power modification right away to quickly damp the system. This reduces the ripple in the voltage U_d while also requiring less power modification in total, if compared with the H_∞^{sub} regulator.

The H_∞ regulator, however, which measures both the input filter voltage U_d and the line voltage E, outperforms both the MPC regulator and the H_∞^{sub} in all scenarios.

If we turn our focus to Figure 6.6 and Figure 6.7 we see that the MPC regulator behaves similar to how it behaved in the case of full traction. In general it demands more power modification than the H_∞^{sub} regulator but also achieves a smoother response in input filter voltage U_d. It can be argued that the amount of power modification that the MPC demands in these conditions is unnecessarily large given that the system is stable in these circumstances. The MPC, however, has been designed to maximize damping during traction, and the weights are thus chosen accordingly. If one wants less power modification in stable operating conditions the weights could for example be made adaptive. To simplify the design this was not done.

In Table 6.3 the summed reference error for the voltage step response is shown, for the operating points full traction, coasting and braking, and for the three different types of control input limitations which have been considered. The table includes the summed error for the regulators MPC, H_∞ and H_∞^{sub}.

Table 6.3: Summed Error During Voltage Step in Simulink

		E_Σ [V]		
Operating Point	Control Limits [kW]	MPC	H_∞	H_∞^{sub}
	$\pm\infty$	9.73	7.71	9.75
Full traction	± 40	15.72	8.39	13.87
	$[-\infty, 0]$	23.56	18.63	30.60
	$\pm\infty$	9.19	7.51	9.04
Coasting	± 20	11.58	8.08	10.66
	$[-\infty, 0]$	13.28	10.76	14.16
	$\pm\infty$	8.96	7.67	8.73
Full brake	± 20	9.37	7.67	8.73
	$[-\infty, 0]$	9.94	8.65	9.95

The summed error E_Σ is computed according to

$$E_\Sigma = \sqrt{\frac{1}{N}\sum_{k=0}^{N} e(k)^2} \tag{6.1}$$

where $e(k) = U_d(k) - U_d^{ref}$. It is a way of assigning a number to the quality of the step response. The more well damped the system dynamics are, the smaller E_Σ is.

Similarly the summed power modification, P_Σ, for each voltage step is shown in Table 6.4. It is computed according to

$$P_\Sigma = \sqrt{\frac{1}{N}\sum_{k=0}^{N} P_{stab}(k)^2} \tag{6.2}$$

The smaller the total power modification is during the time of the step change, the smaller P_Σ is.

From Table 6.3 and Table 6.4 we see that the operating condition where the MPC regulator results in both a smaller summed voltage error, and a smaller summed control input, compared to H_∞^{sub}, is in full traction with the control input limited to negative values. This suggests that the knowledge of control constraints gives the MPC an advantage in this scenario.

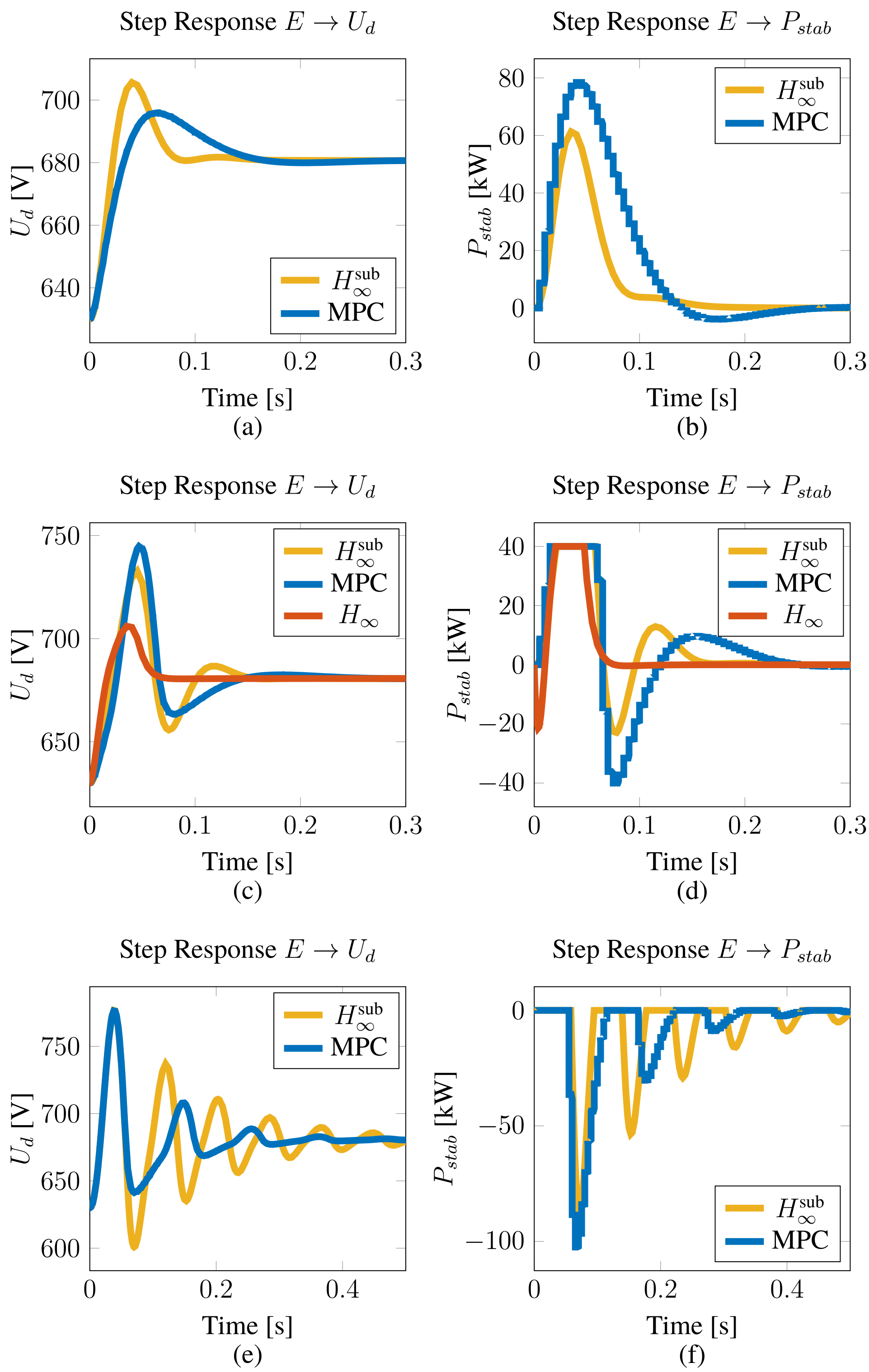


Figure 6.5: Response to a 50 V change in line voltage. Operating point is full traction according to 1 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 40 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

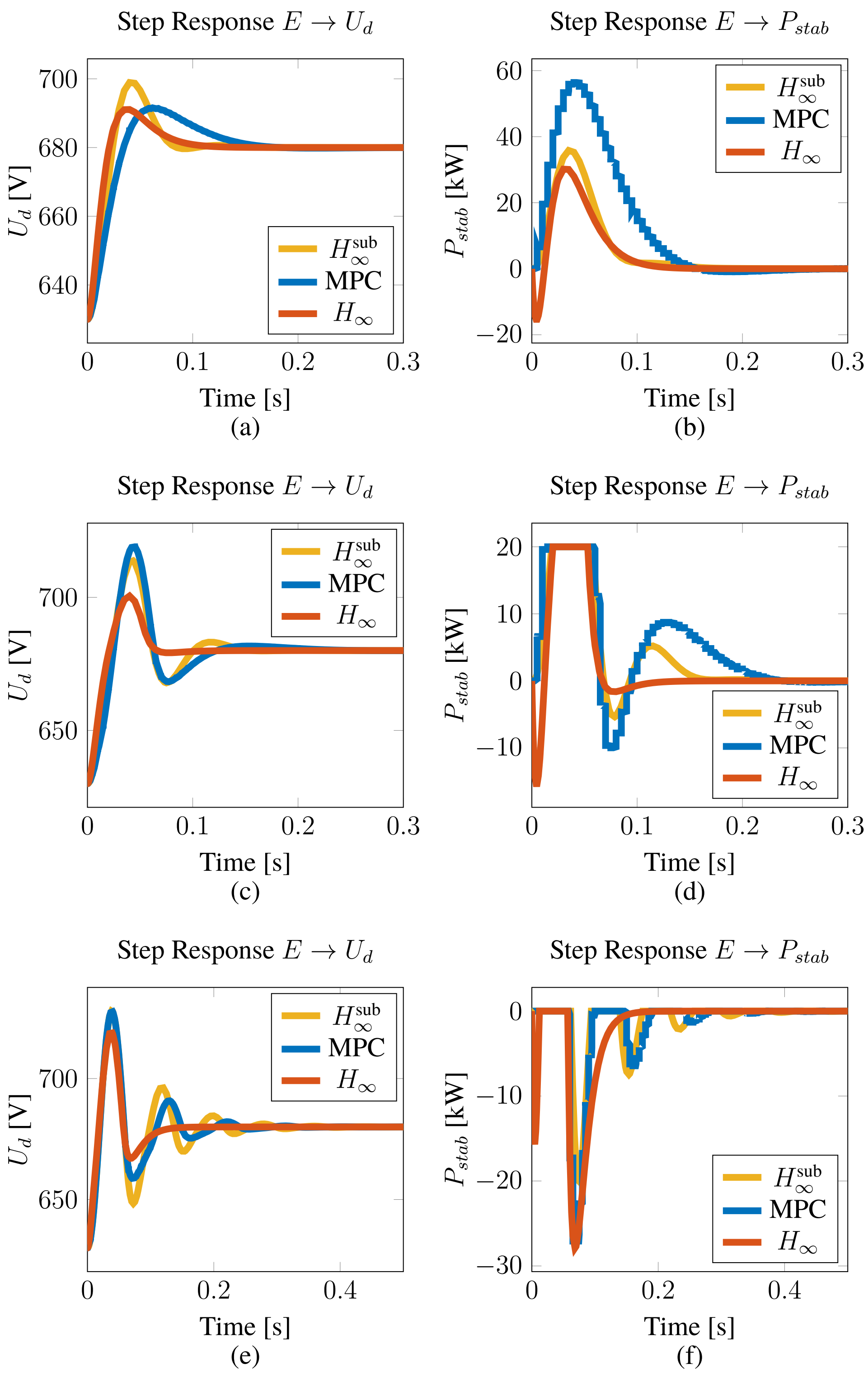


Figure 6.6: Response to a 50 V change in line voltage. Operating point is coasting according to 2 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 20 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

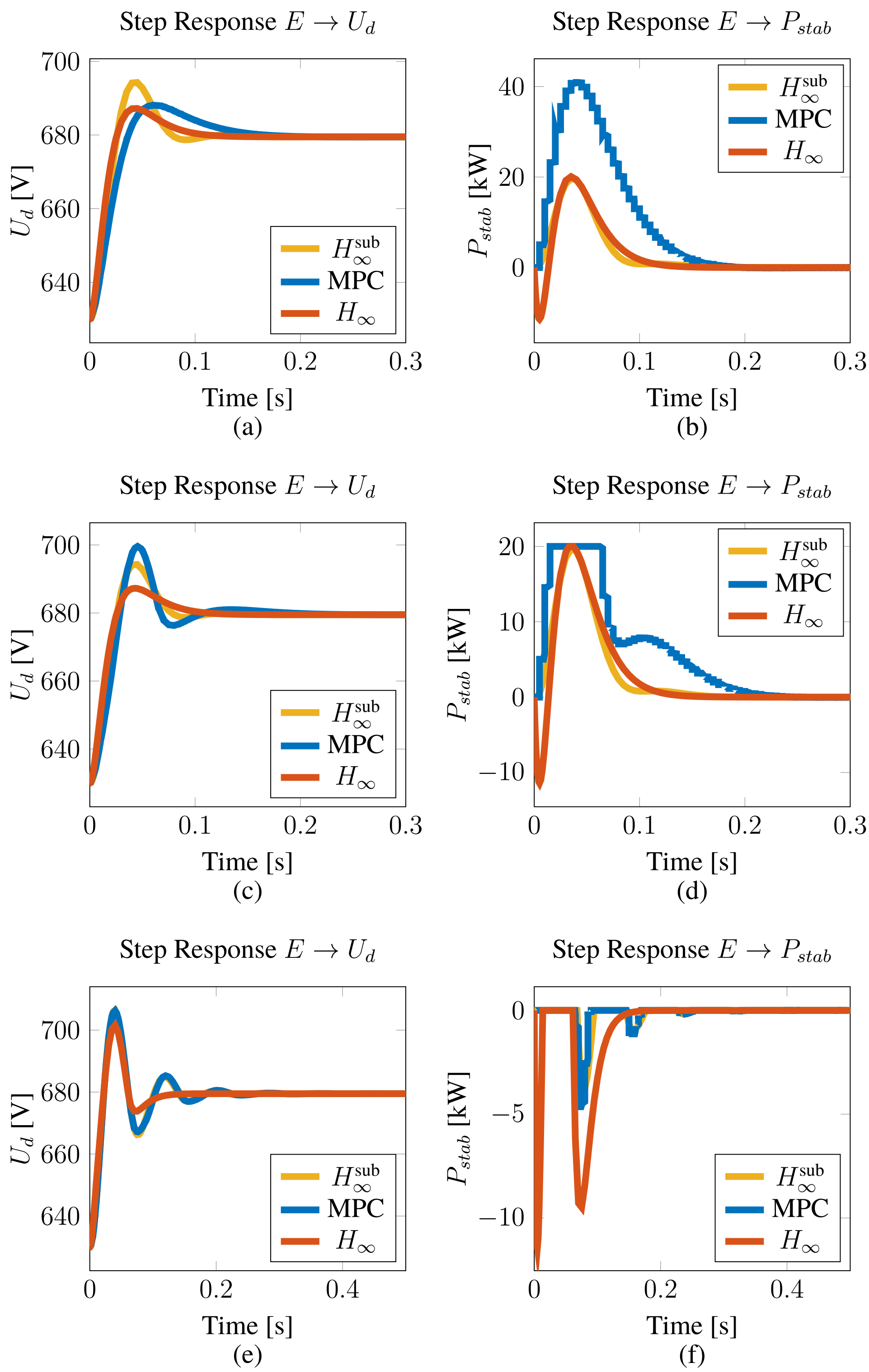


Figure 6.7: Response to a 50 V change in line voltage. Operating point is full brake according to 3 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 20 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

Table 6.4: Summed Power Modification During Voltage Step in Simulink

		P_Σ [kW]		
Operating Point	Control Limits [kW]	MPC	H_∞	H_∞^{sub}
Full traction	$\pm\infty$	25.10	12.01	15.54
	± 40	15.42	10.69	12.78
	$[-\infty, 0]$	22.78	19.49	20.09
Coasting	$\pm\infty$	17.25	7.70	9.10
	± 20	7.16	5.79	6.20
	$[-\infty, 0]$	5.26	6.18	3.82
Full brake	$\pm\infty$	12.85	5.28	4.98
	± 20	7.20	5.28	4.98
	$[-\infty, 0]$	0.71	2.39	0.65

6.2.2 Response to Power Reference Steps

In this subsection we present simulation results related to the response of the system when it is subject to a step in power reference P_{CPL}. We have simulated the step response for the operating points corresponding to full traction, coasting and full brake in Table 5.1, as was done in the previous subsection for the voltage steps. Figure 6.8, 6.9 and 6.10 show the response of the CPL and input filter system to a step in power reference P_{CPL}. The step is of magnitude 30 kW or 10 % of the operating power when in full traction. For each operating point the response has been simulated for three different types of control input limitations: no limits, band limited control input, and upper limited control input.

Looking at Figure 6.8 which shows the response when the load draws full power, we see that the MPC regulator gives a well damped response in input filter voltage U_d. This is most notable in Figure 6.8 (a) where there are no control input limitations. The peak of the response is lower than what the H_∞^{sub} regulator achieves. The response of the MPC regulator is however a little bit slower. It also uses more control input. This is similar to what we saw in Section 6.2.1 for the step in line voltage.

When the control input is band limited such as can be seen in Figure 6.8 (c) and (d) the MPC achieves less damping of the input filter voltage U_d than what the H_∞^{sub} does. The MPC regulator also demands more control input than what the H_∞^{sub} regulator does.

When the control input is limited to only negative values such as can be seen in Figure 6.8 (e) and (f), the MPC achieves a much more well damped response than what the H_∞^{sub} regulator does. The ripple dies out much sooner than what it does for the H_∞^{sub} regulator. Looking at the response in power modification we see that although the power modifications made by the MPC are large in magnitude, the total modification made is smaller than what the H_∞^{sub} does.

If we turn our focus to Figure 6.9 and Figure 6.10 we see that the MPC regulator behaves similar to how it behaved in the case of full traction. In general it demands more power modification than the H_∞^{sub} regulator but also achieves a response with lower peak in the input filter voltage U_d. Again, it can be argued that the amount of power modification that the MPC demands in these conditions is unnecessarily large given that the system is stable in these circumstances. As was argued in Section 6.2.1, the MPC, has been designed to maximize damping during traction, and the weights are thus chosen accordingly. If one would use less power modification in stable operating conditions the weights could be made adaptive. To simplify the design this was not done.

In Table 6.5 the summed reference error for the response in input filter voltage U_d is shown. As before the table includes the summed error, E_Σ, for the different regulators. The error value is computed according to (6.1), in the same was as was done in Section 6.2.1. Similarly, Table 6.6 shows the summed power modification P_Σ for each of the step responses. The value is computed according to (6.2), as was done in Section 6.2.1. Again, as in the case of the voltage step, the advantage of the MPC is seen in the case of full traction, with negative control constraints. In most other operating conditions the H_∞^{sub} performs better than the MPC since it results in a smaller summed error value while also having a smaller summed control input.

6.3 Robustness Towards Model Errors in the Input Filter

In this section we present results which are meant to evaluate how robust the MPC regulator is towards a certain type of modelling errors. In particular we are considering errors in the values of the RLC filter parameters. We will also consider errors in the value of the parameter θ, which also impacts the MPC state space model.

Figure 6.11 shows the response in input filter voltage U_d and control input P_{stab} to a step in line voltage E, for the ideal CPL model in Simulink, using the

Table 6.5: Summed Error During Power Reference Step in Simulink

		E_Σ [V]		
Operating Point	Control Limits [kW]	MPC	H_∞	H_∞^{sub}
Full traction	$\pm\infty$	5.15	2.71	5.11
	± 20	7.41	4.01	5.49
	$[-\infty, 0]$	6.36	2.71	10.95
Coasting	$\pm\infty$	4.42	2.69	4.55
	± 10	5.50	4.54	4.91
	$[-\infty, 0]$	4.28	2.70	5.29
Full brake	$\pm\infty$	3.97	2.67	4.18
	± 10	4.09	3.43	4.18
	$[-\infty, 0]$	3.82	2.67	4.35

Table 6.6: Summed Power Modification During Power Reference Step in Simulink

		P_Σ [kW]		
Operating Point	Control Limits [kW]	MPC	H_∞	H_∞^{sub}
Full traction	$\pm\infty$	7.92	6.11	5.14
	± 20	8.05	5.18	5.24
	$[-\infty, 0]$	7.63	6.11	8.60
Coasting	$\pm\infty$	4.92	4.92	2.97
	± 10	3.90	3.15	2.85
	$[-\infty, 0]$	3.57	4.92	2.72
Full brake	$\pm\infty$	3.33	4.01	1.63
	± 10	3.38	2.54	1.63
	$[-\infty, 0]$	2.20	4.01	1.38

MPC regulator. Errors have been introduced into the MPC system model. The errors are introduced through perturbing the value of the input filter parameters R and L, as well through varying the operating point θ. The error is indicated as a relative difference from the nominal value in Table 5.2. In Figure 6.11, R_0 and L_0 refer to the nominal values. θ_0 refers to the correct operating point parameter as given by the fraction P_0/U_{d0}^2. The voltage plot has been centered around the peak in order to magnify the differences in the step response.

From Figure 5.1 we see that the MPC is robust towards quite large variations in the input filter parameters. For example, overestimating the resistance R by 10 times barely has a visible impact on the step response. A tenfold underestimation of the inductance L results in a less well damped system response but does not affect stability. The model is more sensitive to errors in the operating point parameter θ but can still handle quite large deviations in this case as well. Relative errors of a factor 2 noticeably affects the step response but keeps it well damped.

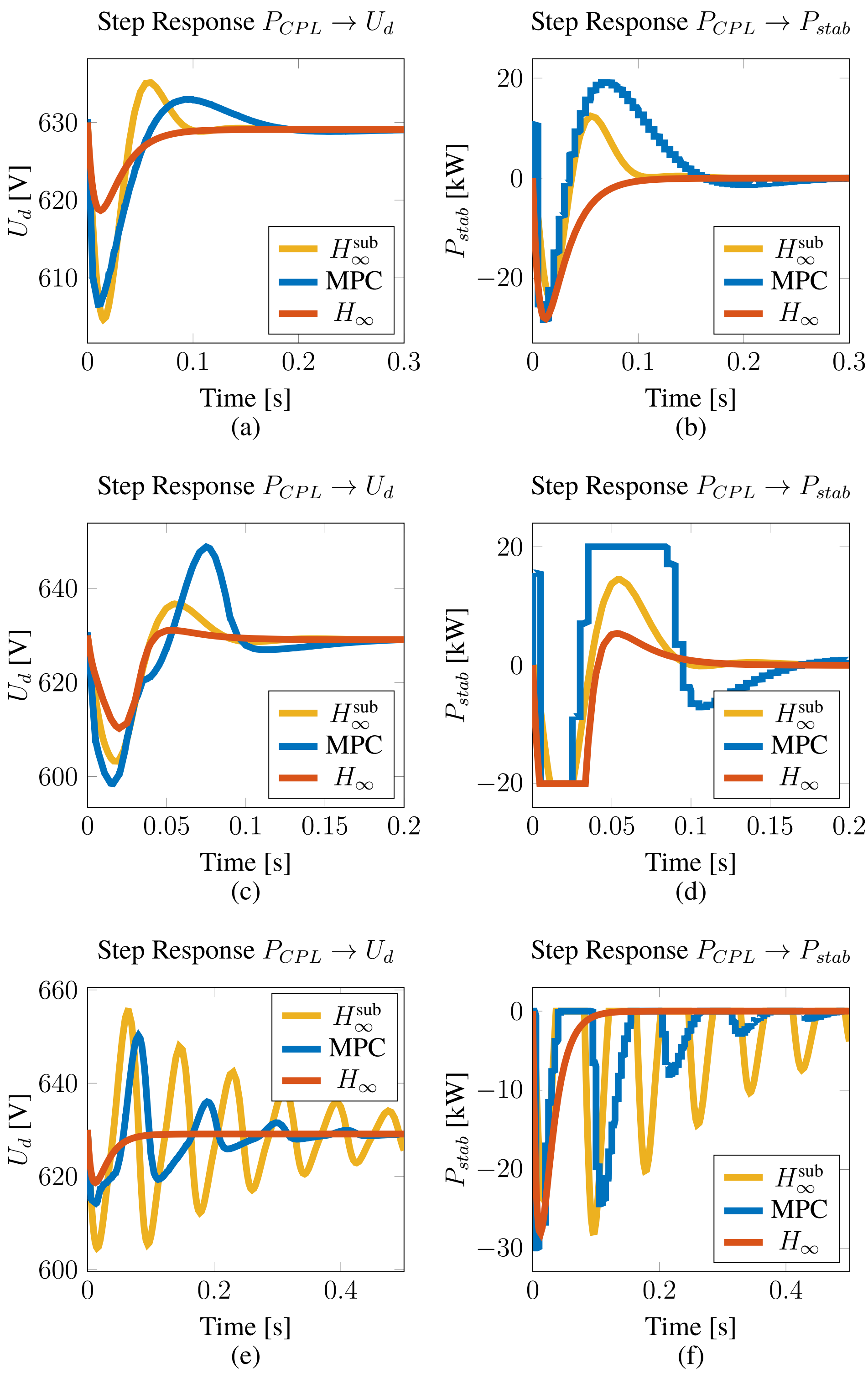


Figure 6.8: Response to a 30 kW power reference change. Operating point is full traction according to 1 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 20 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

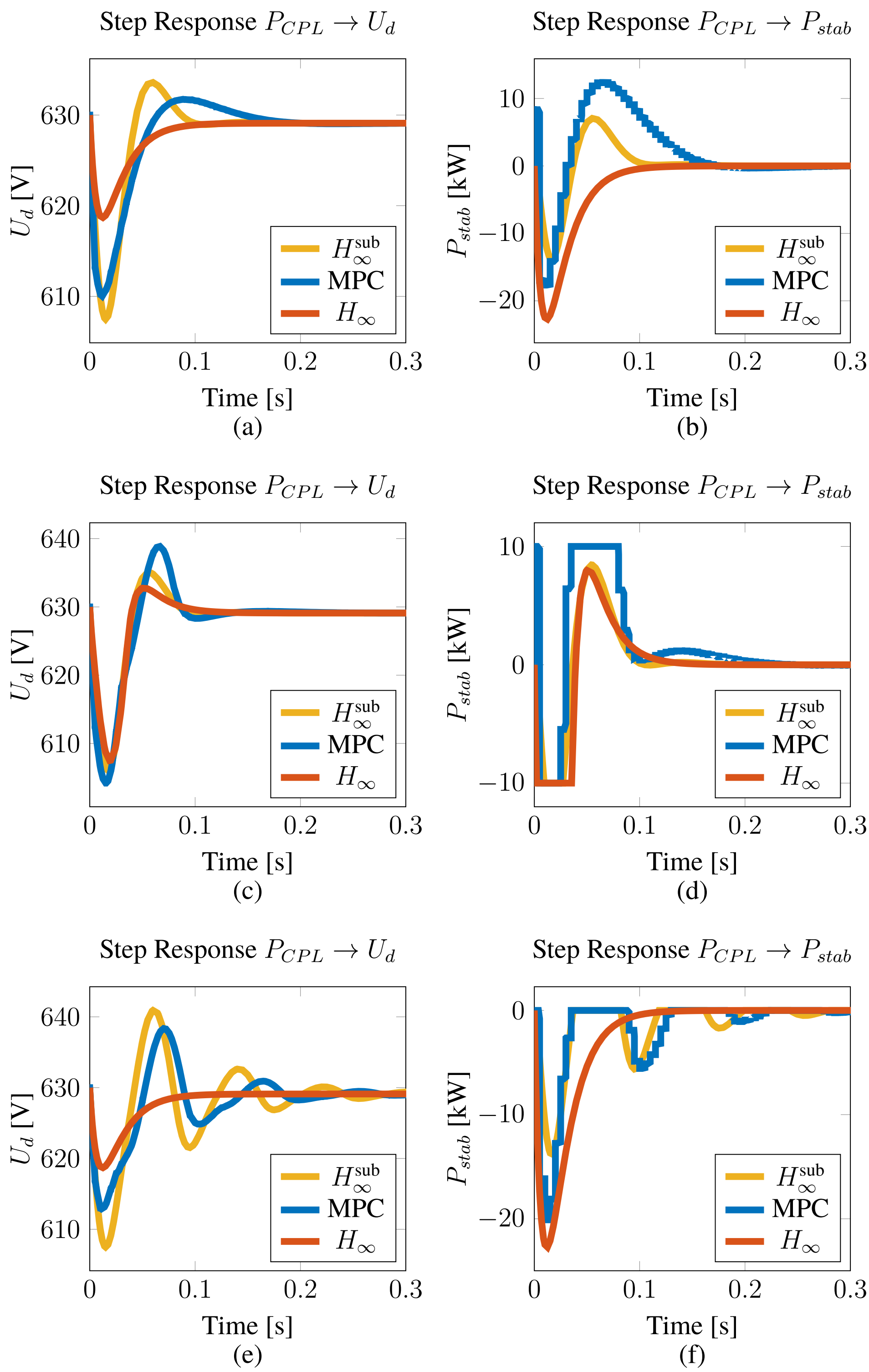


Figure 6.9: Response to a 30 kW power reference change. Operating point is coasting according to 2 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 10 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

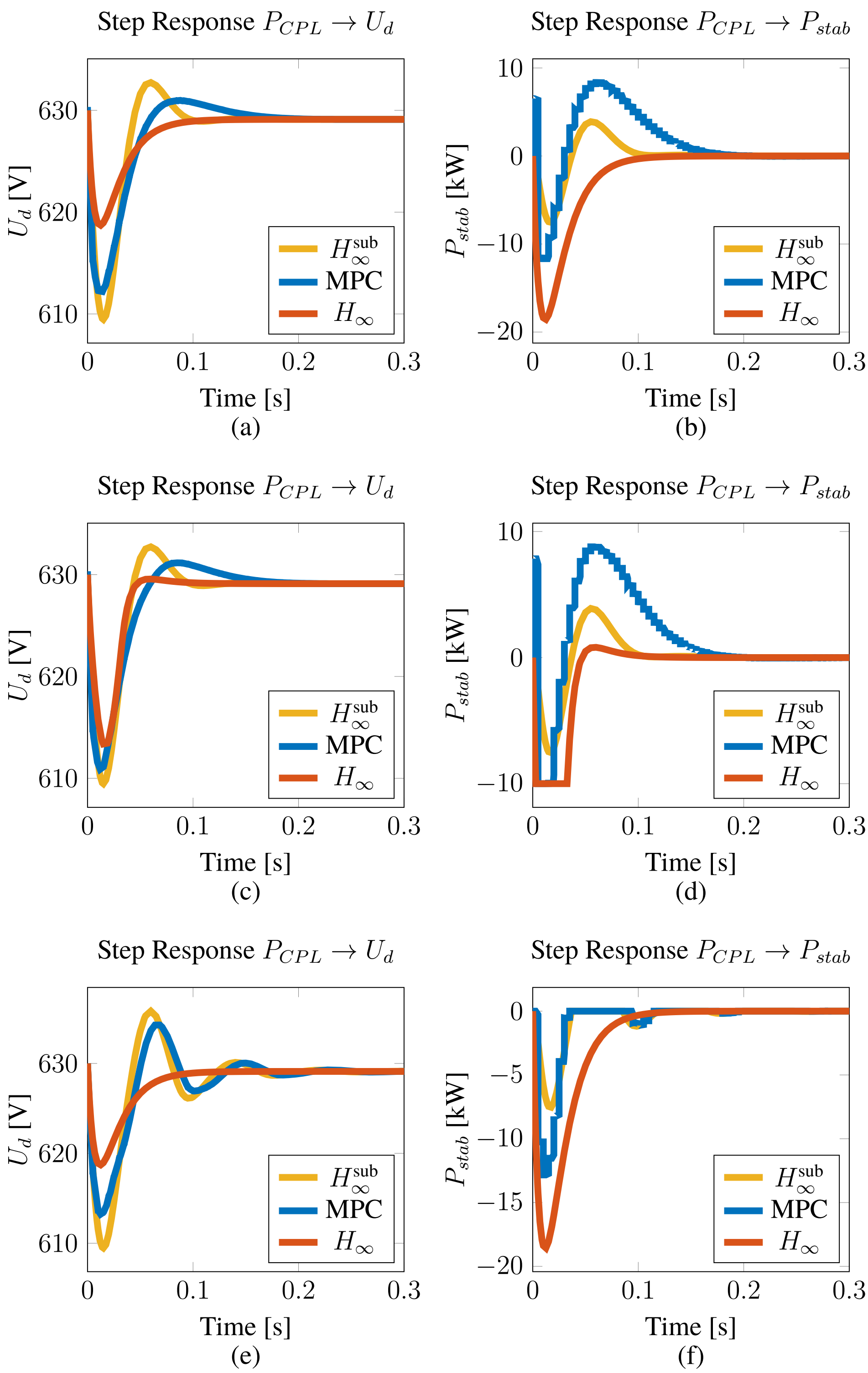


Figure 6.10: Response to a 30 kW power reference change. Operating point is full brake according to 3 in Table 5.1. Plots (a) and (b) show response with no control input limitations. Plots (c) and (d) show response when the control input is limited to ± 10 kW. Plots (e) and (f) show response when control input is limited to the interval $[-\infty, 0]$ W. The MPC regulator is compared to the H_∞^{sub} regulator and the H_∞ regulator.

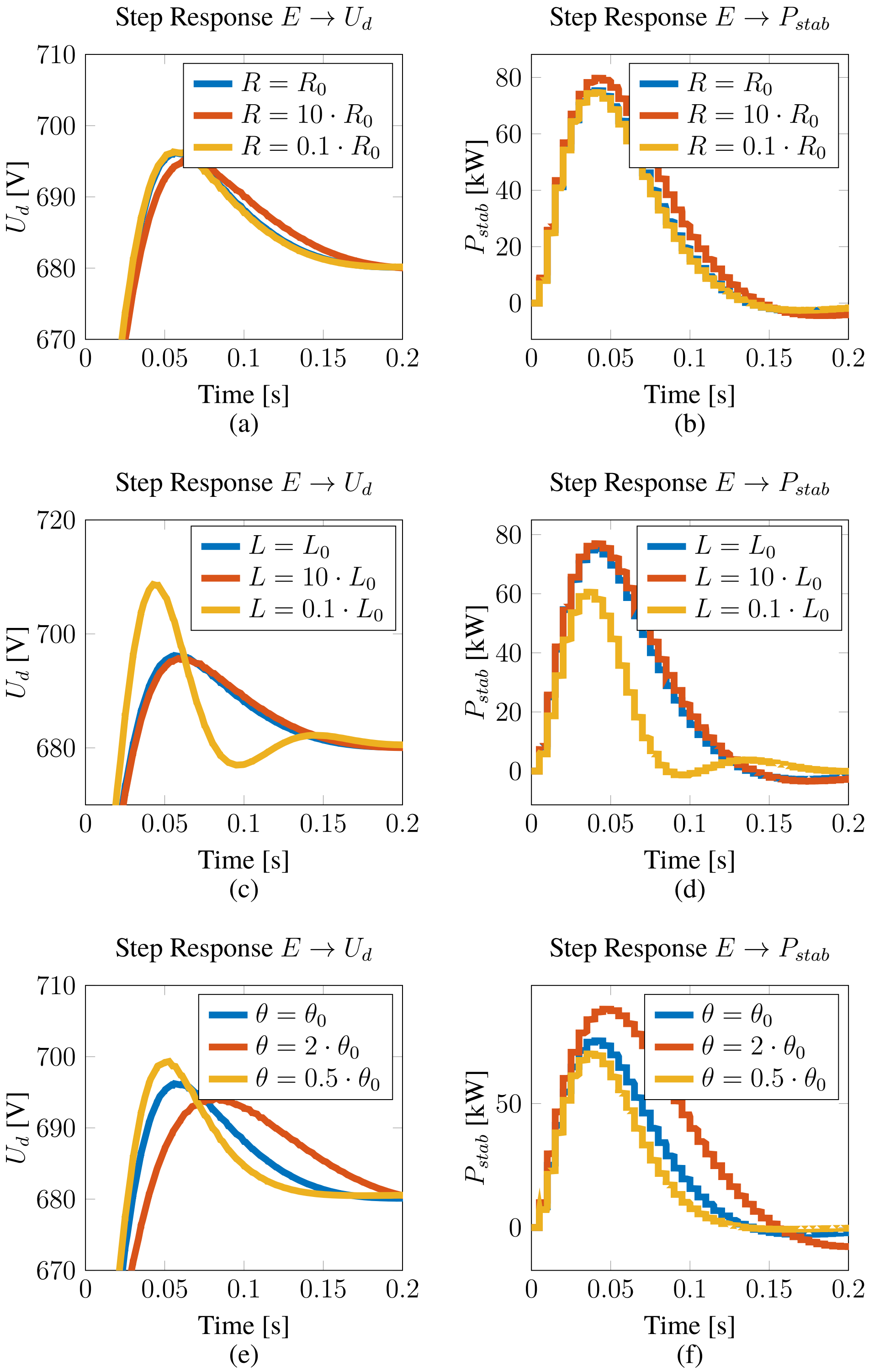


Figure 6.11: Response to a 50 V change in line voltage. Errors have been introduced into the MPC system model. Plots (a) and (b) show the response when error is introduced in the value of the resistance R. Plots (c) and (d) show the response when error is introduced in the value of the inductance L. Plots (e) and (f) show the response when error is introduced in the value of the operating point parameter θ. Operating points is full traction according to 1 in Table 5.1.

Chapter 7

Evaluation in Train Propulsion System

In this chapter we present results regarding implementation of an online LPV-MPC regulator in a real propulsion system.

In Section 7.1 we present results from implementation in the SIL simulator. As explained in Chapter 5.1.2, this environment simulates the propulsion system of the train on a desktop computer. We will use the CLT application for these simulations since its dynamics are fairly close to an ideal CPL around the natural frequency.

In Section 7.2 we will take it a step further and simulate in the HIL simulator. Here the code runs in real-time on application specific hardware. Then, in Section 7.3 we show that the regulator also works in the PowerLab. As explained in Chapter 5.2 the real-time evaluations were made on a train system which did not behave much like an ideal CPL, since that was the testing environment available at the time of the thesis. These tests are therefore used to evaluate different aspects regarding implementation in real-time environments. In Section 7.4 we give results regarding execution time with the MPC regulator on the controller board. Finally in Section 7.5 we give results regarding required memory requirements for the MPC program. The results are analyzed and discussed in Chapter 8.

7.1 Evaluation in the SIL Simulator

This section gives simulation results for step response in the SIL simulator, using the CLT system model. As mentioned in Chapter 5, the SIL simulator models the actual train environment and hence is not an ideal CPL model.

Section 7.1.1 shows a comparison of the simulated step response in the SIL simulator compared to that in Simulink. Then in Section 7.1.2 we evaluate how different values of the operating point filter parameter ν affects the step response. In Section 7.1.3 we evaluate the current estimator which was designed in 4.2. Later, in Section 7.1.4 we study the impact of varying the weights in the MPC cost function. We will also see how the MPC regulator handles operating conditions where the dynamics of the CLT application are less like a CPL. Lastly, in Section 7.1.5 we present power reference steps for the system, for various operating points, showing that the MPC works as intended the SIL simulator for all operating points.

7.1.1 Robustness Towards CPL Modeling Errors

Figure 7.1 shows a comparison between the system response in Simulink, using the ideal CPL model, and the response in the SIL simulator, where different non-ideal CPL dynamics are present. The figure shows the response in input filter voltage U_d and control input P_{stab} to a 50 V step in line voltage E. The response has been simulated in Simulink using the LQR and the MPC regulator, and in the SIL simulator using the MPC regulator. The MPC regulator uses a filter constant for the operating point filter (4.12) which is $\nu = \frac{1}{4}\frac{\omega_0}{2\pi f_s}$. The differences in the response between the LQR and the MPC as simulated in Simulink may be attributed to the fact that the LQR is designed as a continuous-time regulator, while the MPC regulator is designed as a discrete-time regulator with a sample rate of 200 Hz. The LQR also has direct access to the line current i while the MPC regulator estimates this signal. The step response of the system, controlled by the MPC, is slightly less damped in the SIL simulator than what it is in Simulink. By tuning the operating point filter parameter for the SIL model a more well damped response can be achieved (see Section 7.1.2). Overall the response is still quite similar. This shows that the CPL assumption holds quite well for a real train application, at least for this operating point.

7.1.2 Impact of the Operating Point Filter Parameter

Recall the MPC operating point filter (4.12) with parameter ν which is normalized frequency. Figure 7.2 shows the system response to a step in line voltage E, in the SIL simulator, for a few different values of ν. The MPC regulator is used for stabilization. Figure 7.2 (d) shows the response in the operating point parameter θ to the step. Looking at the yellow line in Figure 7.2 (a), we

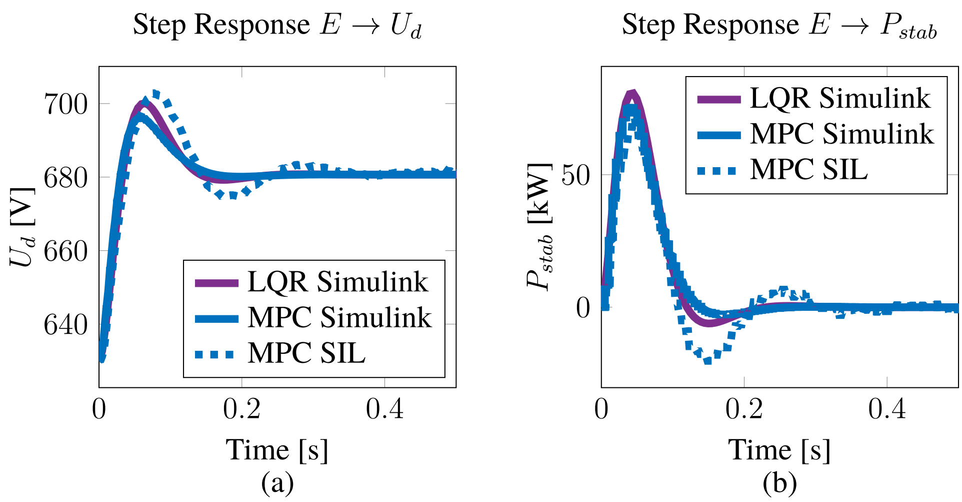


Figure 7.1: Comparing step response for LQR and MPC in Simulink to MPC in SIL simulator. Response to a 50 V change in line voltage. Operating point is full traction according to 1 in Table 5.1. There are no control input limitations.

see that a small value of ν, corresponding to a low cutoff frequency, leads to the smoothest response in input filter voltage U_d. However, this leads to a very slow settling time for the line current i (see Figure 7.2 (b)) and the power modification P_{stab} (see Figure 7.2 (c)). Since one of the performance specifications is to limit the use of power modification such a cutoff frequency is undesirable. If on the other hand the cutoff frequency ν is too close to the natural frequency ω_0 the system becomes less damped, although quite fast (see the blue curves in Figure 7.2). Through tuning we noticed that a value of $\nu = \frac{1}{8}\frac{\omega_0}{f_s}$ worked well in the SIL simulator. This gave a well damped input filter voltage U_d while keeping the line current i and the power modification well damped as well.

7.1.3 Measured Versus Estimated Current

In this subsection we evaluate the current estimator which was designed in Chapter 4.2. Figure 7.3 shows the measured and the estimated line current during a 50 V step in line voltage E. We see that the estimated current is noisy but follows the measured current well.

Figure 7.4 shows the response in input filter voltage U_d and control input P_{stab} for estimated and measured current respectively. The response is to a step in line voltage, E. From the figure we see almost no difference in the two responses. The reason why it doesn't matter that the current estimator is noisy is that the current is band-pass filtered before used in the MPC. This is done no matter if estimated or measured current is used. The band-pass filtering is a result of the fact that the MPC (1) subtracts steady-state signals since it works with deviations from an equilibrium, and (2) it operates on a frequency of 200 Hz and thus low-pass filters all input signals to avoid aliasing. The aliasing filter will at the same time remove the high frequency noise from the estimated current.

7.1.4 Impact of Weights and Varying CPL Dynamics

In this subsection we show how varying the weights in the MPC cost function affects the system dynamics in the SIL simulator. In Figure 7.5 we see the response of the system for two different weights on the input filter voltage U_d. The response is also shown for two different operating points. The first operating point (response plotted in Figure 7.5 (a) and (b)) is the first operating point in Table 5.1 and the second operating point (response plotted in Figure 7.5 (c) and (d)) is the fourth operating point listed in Table 5.1. Both of these operating points correspond to full traction, but at two different motor speeds. As explained in Chapter 5.1.3 the torque dynamics are different for higher motor speeds and the CPL approximation does not hold as well there. We want a regulator which is robust to these variations in system dynamics.

Looking at Figure 7.5 (a) and (b) we see that having a larger weight on U_d pushes down the peak of the voltage U_d more, but also uses more control input. In the second operating point the response becomes a lot less well damped in the case when $q_{U_d} = 1$ (see Figure 7.5 (c) and (d)). This is despite it using almost as much power modification P_{stab}. We hence see that designing the controller in order to make the system dynamics more damped, by increasing the weight on the input filter voltage U_d, also makes the controller more robust towards modelling errors in the CPL.

7.1.5 Power Reference Steps

In this subsection we present the response to a step in power reference P_{CPL} for three different operating points, full traction, coasting, and full brake, where the operating points are defined in Table 5.1. The response is plotted in Figure 7.6. The size of the step in power reference is 50 kW. Figure 7.6 (a) and (b) show the response while operating in full traction, Figure 7.6 (c) and (d) show the response while operating in coasting, and Figure 7.6 (e) (f) shows the response while operating in full brake. The vertical axis has been scaled in the same way for all the plots so that it is clear that power modifications of very different amplitudes are needed depending on what the operating condition is. Similarly, the amplitude of the spike in the voltage U_d differs a lot depending on how well damped the system is on its own. In coasting and in braking the system is as we know stable. We see that the MPC regulator behaves a lot like what we saw in the Simulink simulations. In these simulations we have set the operating point filter parameter to be $\nu = \frac{1}{8}\frac{\omega_0}{2\pi f_s}$.

7.2 Real-Time Simulations for MPC Regulator

This subsection gives real-time simulation results for the MCX project described in Chapter 5.2. As mentioned previously this system is already well damped, so this test is aimed at checking feasibility of the MPC regulator design. The simulations have been performed in the HIL simulator and the MPC regulator is thus running in real-time on application specific Bombardier hardware, together with all the other control systems which exist in the MCX propulsion system. The MPC regulator is running at a sampling frequency of 200 Hz and the prediction horizon is 5 time steps. The length of the prediction horizon, N, corresponds to approximately $1.1 \times \frac{2\pi}{\omega_0}$ for the MCX application. This is 1.1 times the period of the natural oscillations of the unstabilized system. Simulations showed that the performance was degraded if the prediction horizon covered less than one period. Very long prediction horizons also didn't make the performance noticeably better. This particular value for N was chosen since it was a nice round number, longer than one period which was possible to realize with the available hardware.

Figure 7.7 show HIL simulations of the response in input filter voltage and torque modification when a step is made in the torque reference and in the line voltage. The operating point is full traction which in this application corresponds to a reference torque of 1560 N m. The reference speed is 30 km/h which corresponds to a motor speed of about 140 rad/s. The plots show the response in input filter voltage as well as the applied stabilizing torque T_{stab}, which is what the stabilizing power modification P_{stab} is converted into before implementation in the converter. The torque modification was limited to $\pm$ 500 N m.

7.3 PowerLab Experiment

The MPC regulator was also tested on MCX in PowerLab. As mentioned before, this system is really well damped, even without any stabilization, so this test is aimed at checking feasibility of the MPC regulator design. Figure 7.8 shows the response of the MCX application during a 100 N m step in torque reference. The response in input filter voltage U_d and the applied control input is shown. The result of the MPC stabilization is plotted together with the H_∞^{sub} stabilization, and no stabilization at all.

7.4 Execution Time

As well as evaluating step responses for the MPC regulator through HIL simulations and in PowerLab, we also measured the execution time of running the MPC optimization code on the controller board. This was done in the HIL simulator. The test was made by measuring the execution time of the 5 ms task on the control board, while operating in full traction and varying the train speed from 0 km/h to 80 km/h. By execution time we refer to the time it takes for the processor to perform all calculations which are needed for the next sample period. Thus by measuring the execution time of the 5 ms task level of the controller board we measure the time it takes to complete all processes which are located on the 5 ms level and levels fast than that. For the program to be able to run all calculations have to be finished within the time span of the sample period, namely 5 ms.

During the test the maximum and the average execution time were noted. This was done for three different values for the tolerance in the infeasible start Newton method which has been used in the MPC optimizer (see Chapter 2.9.2). The execution time was also compared to a code which does not have the MPC optimizer in it. Table 7.1 lists the average and maximum execution time for these tests. We see that the MPC optimizer adds about an order of magnitude to the maximum execution time. We also see that for this particular problem, neither the average nor the maximum execution time for the MPC changes dramatically within the tolerance range which we have tested. The total time stays within the same order of magnitude for all tested tolerances. We can also see that even for the strict tolerance of 1×10^{-5} there is a decent margin between the maximum execution time and the task cycle time which is 5 ms. This is however a tolerance which is much stricter than what is needed for good control performance.

Table 7.1: Execution Time

Regulator	Tolerance	Max. Time [ms]	Average Time [ms]
MPC	1×10^{-5}	0.780	0.735
MPC	1×10^{-2}	0.664	0.619
MPC	1	0.522	0.440
H_{∞}^{sub}	-	0.051	0.005

7.5 Impact on Storage Utilization

Without the MPC code, the file which runs on the controller board is 130 kB in size. With the MPC code it is 153 kB. It is thus a total memory storage increase of 23 kB or approximately 18 %. The MPC regulator thus adds a substantial amount to the total code. We did not look at dynamic memory utilization in this thesis.

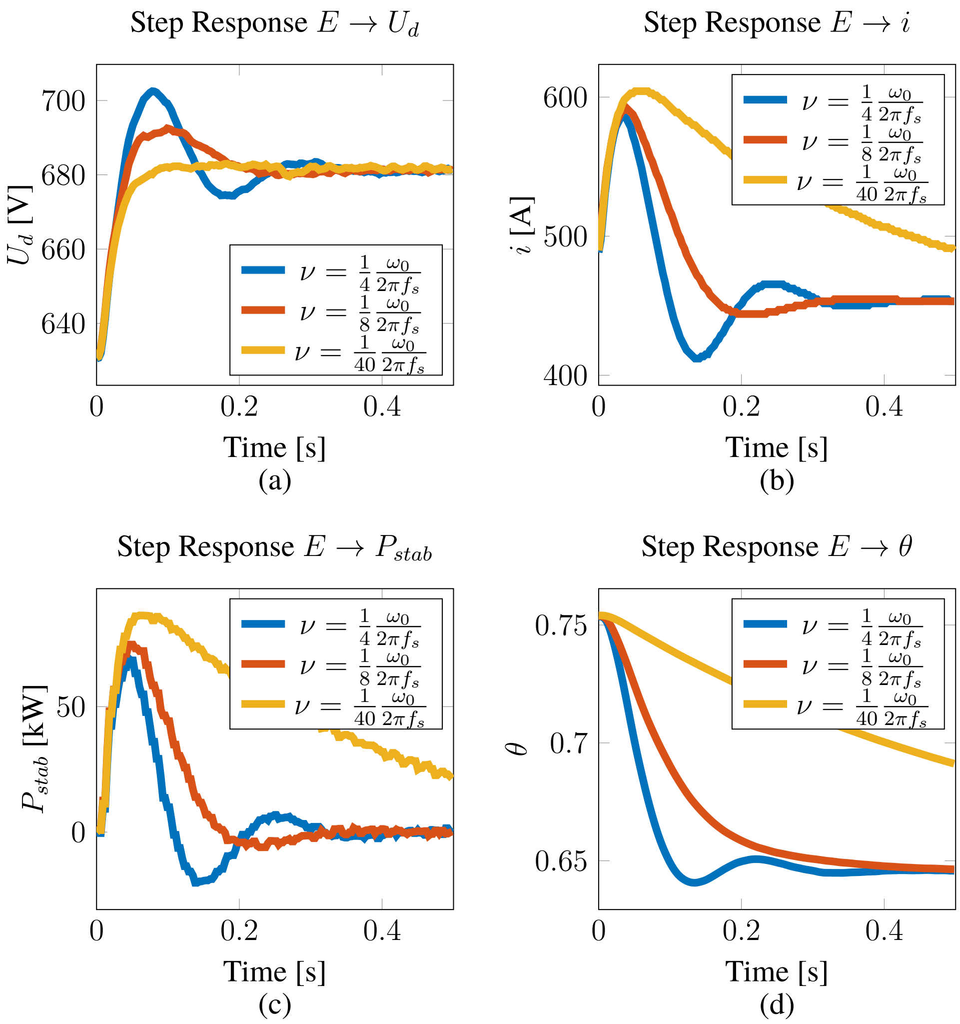


Figure 7.2: Comparison of the step response using MPC for different values of the operating point filter parameter. Response to a 50 V change in line voltage. Operating point is full traction according to 1 in Table 5.1. (a) shows the response in input filter voltage U_d, (b) the the response in line current i, (c) the response in power modification P_{stab}, and (d) shows the response in the operating point parameter θ.

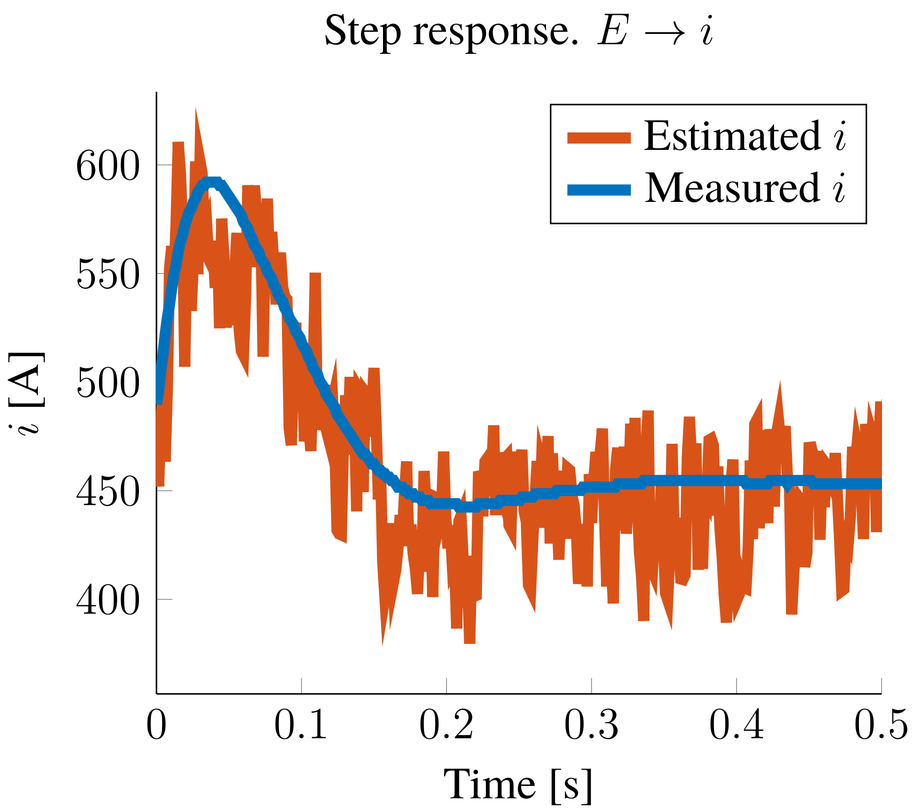


Figure 7.3: Estimated versus measured current in the SIL simulator. Response to a 50 V change in line voltage. Operating point is full traction according to 1 in Table 5.1. There are no control input limitations.

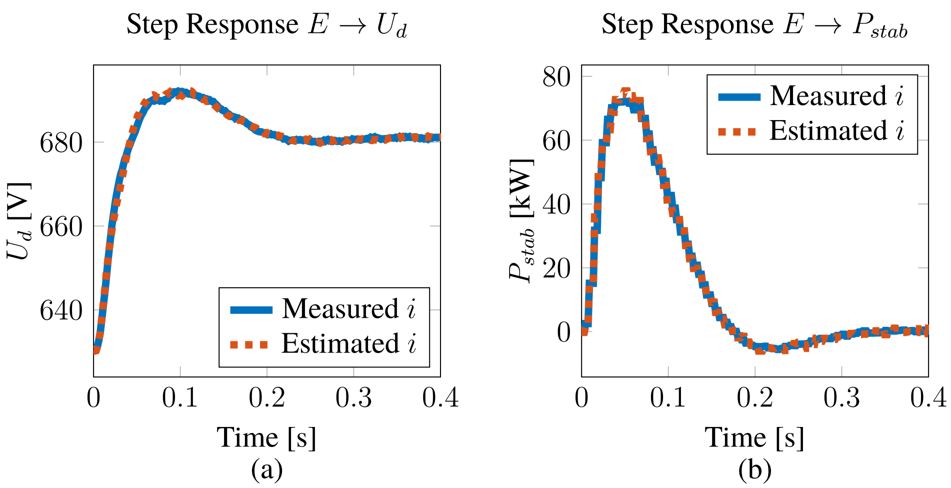


Figure 7.4: Comparing step response for MPC with measured and estimated current in the SIL simulator. Response to a 50 V change in line voltage. Operating point is full traction according to 1 in Table 5.1. There are no control input limitations.

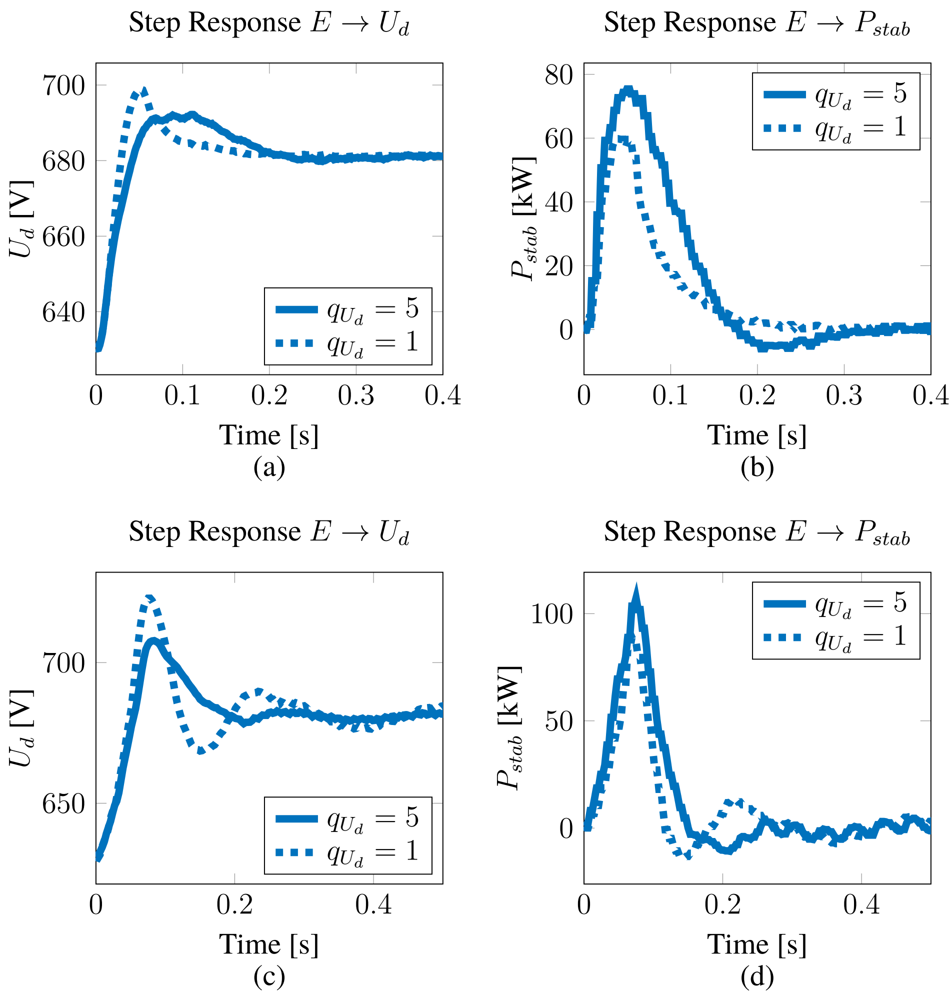


Figure 7.5: Comparing step response for MPC with two different sets of weights, in the SIL simulator. Response to a 50 V change in line voltage. In plots (a) and (b) the operating point is full traction according to 1 in Table 5.1. In plots (c) and (d) operating point is full traction according to 4 in Table 5.1. There are no control input limitations.

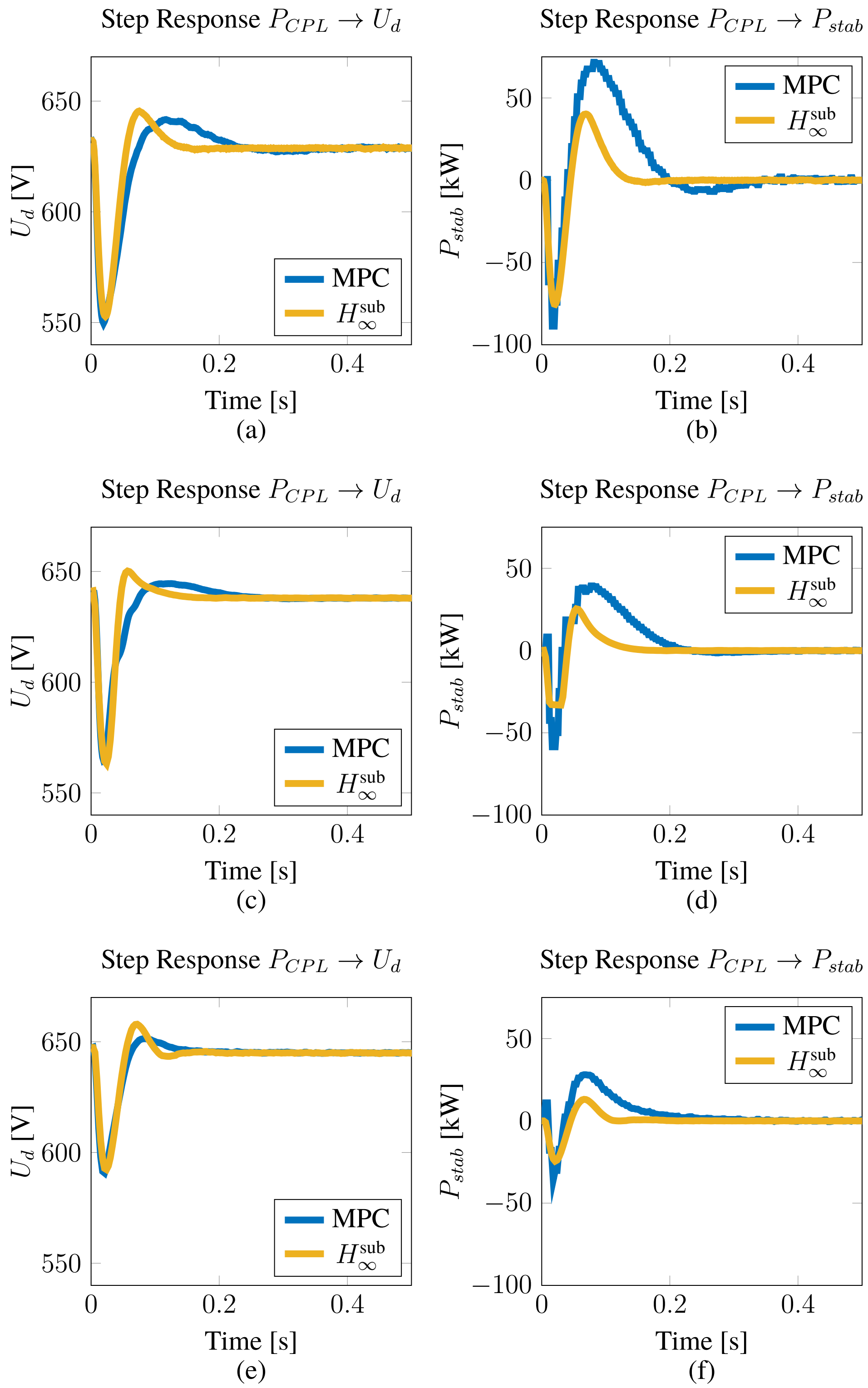


Figure 7.6: Response to a 100 N m change in reference power P_{CPL}. Plots (a) and (b) show response when the operating point is full traction, plots (c) and (d) show the operating point is coasting, and plots (e) and (f) show the response the operating point is full brake. The operating points are according to Table 5.1. The MPC regulator is compared to the H_∞^{sub} regulator.

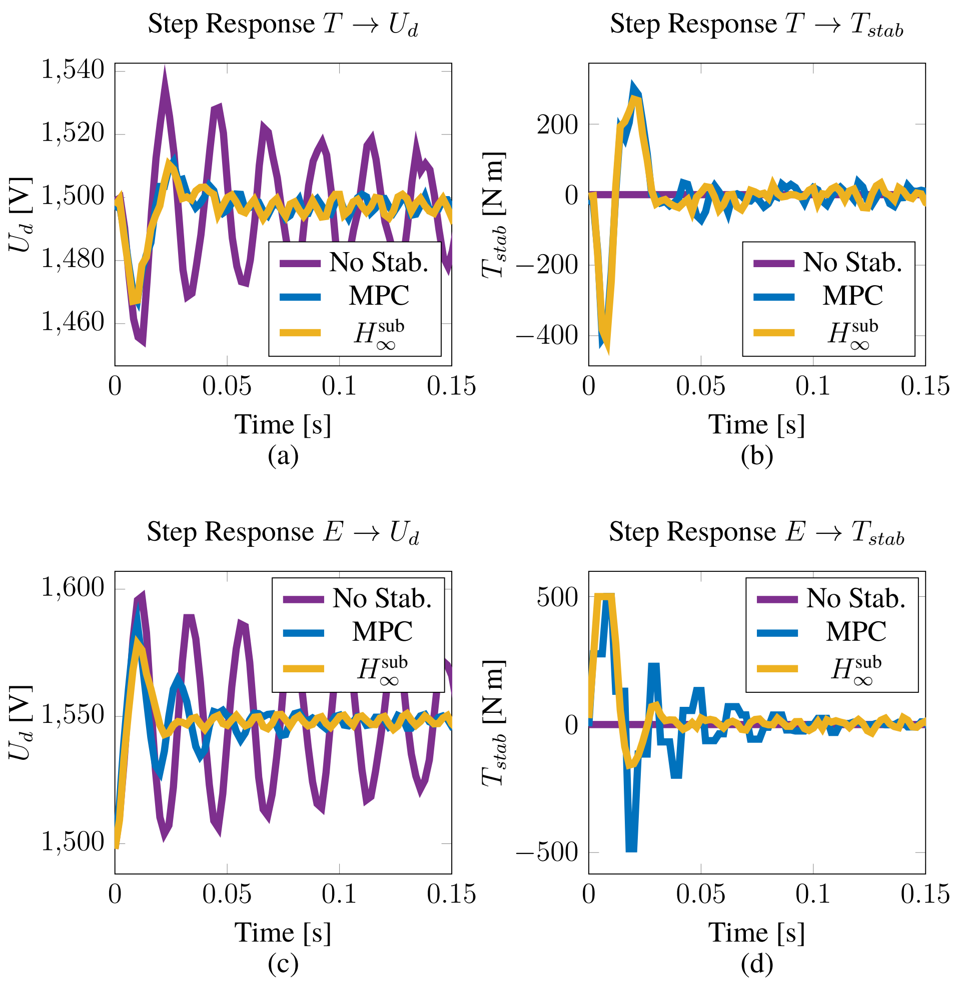
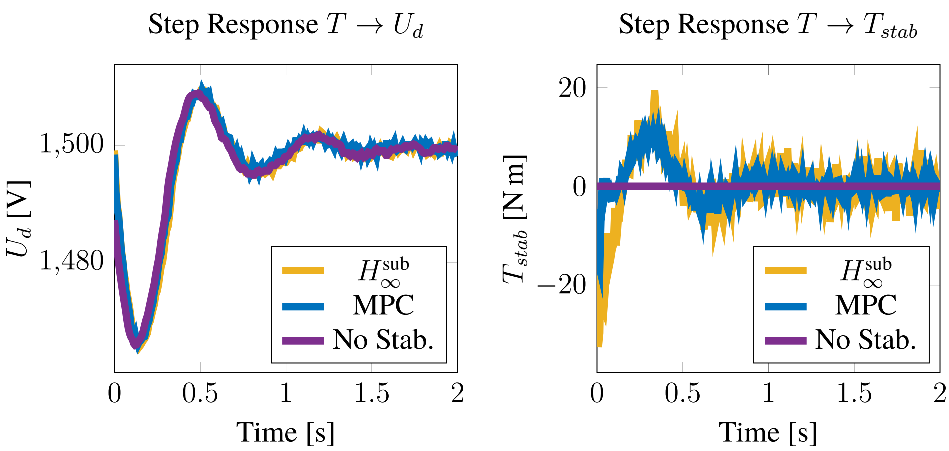


Figure 7.7: HIL step responses while in full traction (reference torque is 1560 N m). The reference velocity of the train is 30 km/h, corresponding to a motor speed of of 140 rad/s. The torque modification is limited to ± 500 N m. Plots (a) and (b) show voltage and torque modification during a 500 N m step in torque reference. Plots (c) and (d) show voltage and torque modification during a 50 V step in line voltage. The MPC regulator is compared to H_∞^{sub} and no stabilization at all.



(a) (b)

Figure 7.8: Step response in power lab while operating at a reference torque of 750 N m corresponding to full traction. The reference velocity of the train was 30 km/h, corresponding to an electrical velocity of about 140 rad/s. The stabilizing torque modification is limited to ± 100 N m. Plots (a) and (b) show voltage response and stabilizing torque modification during a 150 N m step in torque reference. The MPC regulator is compared to H_∞^{sub} and no stabilization at all.

Chapter 8

Discussion

This chapter discusses the results of the tests which were done on the MPC regulator in Chapter 6 and Chapter 7. Section 8.1 discusses results related to the ideal CPL model. These results come from MATLAB/Simulink and were presented in Chapter 6. Then in Section 8.2 we discuss the results of the tests from Chapter 7 which were done in the SIL simulator. Section 8.3 discusses the results regarding real-time implementation of the MPC regulator, which were presented in Chapters 7.2 - 7.3. The discussion of the results are tied back to the performance specifications in Chapter 2.5.

8.1 Stabilization of Ideal CPL

From the results in Chapter 6 we see that the proposed LPV-MPC design is a viable method for stabilization of the CPL and input filter system. This is something which we see both from the Nyquist diagrams of the admittance Y_{DC} and the loop gain L_{DC} in Chapter 6.1, and from the step response results in Chapter 6.2. The magnitude plots in Chapter 6.1.2 and Chapter 6.1.3 show that the LQR, and by reasoning, that also the MPC have a smooth frequency response without any large resonance peaks in the relevant transfer functions. This is achieved despite LQR being analogous with H_2 optimization, which is known to have the tendency of giving large resonance peaks. In the design we have tried to achieve a response which is similar to that of the H_∞^{sub} regulator. Because of the limited degrees of freedom which our regulator has, this was not possible, but we have come fairly close.

The MPC regulator achieves the goal of a well damped system response, and in particular a well damped input filter voltage U_d as we see from the plots of the step response from the excitation signals E and P_{CPL}. The MPC control structure is hence capable of satisfying the performance specifications in Chapter 2.5. It also reflects the way the weights have been chosen in the MPC cost function (2.70), where a larger cost was put on deviations in input filter voltage U_d. This is one of the strengths of the MPC regulator. It is quite intuitive to tune, since it doesn't require much understanding of the frequency domain. Instead we just punish deviations of the relevant signals, in time domain. This is quite easy to grasp, at least when the number of parameters is manageable. Keeping the number of parameters manageable was something which we had in mind throughout the design process. The final LPV-MPC regulator has only two main tuning variables, the relative weight on the input filter voltage U_d and the control input u, as well as the operating point filter parameter ν. This makes it easy to reconfigure the MPC regulator, should design criteria change.

From the simulated step responses in Simulink we see that the way the MPC regulator achieves more damping is by using more of the power modification P_{stab}. This is the case both when the control input is limited and when it is not. These two performance specifications, damping and limited use of control input, are in a way fundamentally at odds, no matter which control method we use. The trade-off between the two are partly affected by the requirements of the application and partly a design choice by the control engineer.

Let us for a while focus on the way the MPC regulator deals with stabilization in the scenario where the control input is limited to negative values. If we look at Figure 6.5 (e) and (f) in Chapter 6.2.1 we see that the MPC regulator, which is aware of the control input constraints, does a lot better than the H_∞^{sub} regulator which simply truncates its control input. The MPC uses more control input in the range where it is allowed compared to what it otherwise would use. The MPC regulator also performs better in this scenario for this reason. Interestingly though, the H_∞ regulator performs better than both the MPC regulator and the H_∞^{sub} regulator. This indicates that good performance in terms of well damped system dynamics in this scenario is not dependent on knowledge of the constraints, even though the MPC outperforms the H_∞^{sub} because of it. Fundamentally what is needed is a fast response and more control input. This is also achieved by the H_∞ regulator. However, the H_∞ regulator has access to the line voltage E, which neither the MPC regulator or the H_∞^{sub} do.

The H_∞ regulator, which has been plotted in red in Figure 6.5 - 6.10 clearly outperforms both the H_∞^{sub} regulator and the MPC regulator and results in a more well damped system. This is true for all simulation scenarios which have been tested in Simulink, even though control input limitations are not part of the regulator design. However, the regulator assumes it is possible to measure the line voltage E. This is the potential of the dc voltage source which feeds the transmission lines of the train. Measuring that signal is technically not possible since it is not located on the train. It is however possible to estimate it, but since the impedance varies depending on how far the train is from the voltage source it is not certain that the estimate will be good enough. These results do however show that, if possible, estimating the line voltage reliably could be very advantageous if utilized in an H_∞ regulator design. Alternatively, if one does not have the signal E available and more damping is desired in constrained scenarios, one could switch to a differently designed regulator in those circumstances; one which uses more power modification. That way well damped system dynamics is achieved in constrained scenarios, while avoiding using too much power modification in unconstrained scenarios, when it is not needed.

Lastly, in Chapter 6.3, we presented results relating to the robustness of the MPC regulator towards modelling errors in the input filter. It was tested by introducing relative errors in the values of the input filter components; in this case the resistance R and the inductance L. The impact of errors in the operating point parameter θ was also tested. We limited our scope to investigating the impact of errors in each parameter on its own. We are aware that this does not capture the entire parameter error space since it could be the case that faults are correlated. From Figure 6.11 (a) and (b) we see that the MPC model is very robust towards errors in the value of the resistance R. The step response is barely affected by changes which are of a factor 10. Changing the inductance L in the model gives a change in the natural frequency ω_0. As long as the admittance Y_{DC} at the natural frequency ω_0 lies close to the positive real axis the dynamics of the system are not affected too much by variations in L. We see in Figure 6.11 (c) and (d) that the system dynamics become less well damped when the inductance L is reduced by a factor 10 in the model, but the system is still stable. The model is more sensitive to variations in the operating point parameter θ than it is to variations in the values of the passive components in the input filter. We see however that introducing relative errors in θ of a factor 2 still gives a well damped system response. We hence conclude that the MPC model is robust towards these types of errors. This is good since we can expect the resistance and the inductance to vary quite a lot during operation as the train is moving along the transmission lines.

Since the design of the MPC regulator has been done for a CPL connected to an input filter, the results are applicable to other system which can be modelled as such, and not only train propulsion systems. As was stated in Chapter

1.3, the stability problem which has been dealt with in this thesis exists in many other systems which include power converters. We therefore hope that these results may be of use in areas beyond those of propulsion systems.

8.2 MPC in the SIL Simulator

From the SIL simulations we can gain understanding of how the MPC regulator behaves when used on a model of a train propulsion system; we are no longer simulating an ideal CPL. In Figure 7.1 in Chapter 7.1.1 we presented a comparison between the system response as simulated for an ideal CPL and for the extended system model from the SIL simulator. The response was to a step in line voltage E, when controlled by the MPC regulator. We see that the response in the SIL simulator corresponds a lot to what we see for an ideal CPL in Simulink. It is however slightly less damped in the SIL simulator when all regulator parameters are the same. This can be attributed to the non-ideal CPL dynamics of the SIL model. Furthermore, there is a mismatch between desired power modification, as computed by the regulator, and realized power modification. This is a result of power losses which are modelled in the SIL simulator, but unaccounted for the the MPC model. This is another cause for the deviation between the ideal Simulink environment and what we see in the SIL simulations. To minimize these deviations, it is important that the regulator design is robust.

One thing which we saw affected the robustness of the MPC regulator was the choice of weights in the cost function. If we look at Figure 7.5 in from Chapter 7.1.4 we see how a larger weight on the input filter voltage U_d gives a more well damped system response. This is especially noticeable in the system response in the second operating point shown in Figure 7.5 (c) and (d). In this operating point the torque dynamics of the system are less ideal. With a larger weight on the input voltage U_d the regulator is more robust towards these system variations. Hence robustness is closely related to a well damped system, as might be expected. The cost for the increased robustness is a larger power modification P_{stab}.

Another thing which we observe from the SIL simulations is the importance of the tuning parameter ν, i.e. the cutoff frequency of the operating point filter. If we look at Figure 7.2 in Chapter 7.1.2 we see that the value of ν affects the damping of the system dynamics a lot. We also again see the balance which exists between power modification P_{stab} and well damped input voltage U_d. Choosing ν too small will result in a very slow settling of the power modification. This is bad for reference following since it limits the bandwidth available for control substantially. If ν instead is chosen to be too close to the natural frequency ω_0 the dynamics become less damped and can even become unstable if ν is too large. This is what was discussed in Chapter 4.3.2, and we see it in practice in Chapter 7.1.2.

We also evaluated the current estimator which was designed in Chapter 4.2. In Chapter 7.1.3 we presented the results which evaluate the estimator. We see from Figure 7.3 that the estimated current is quite noisy compared to the measured signal. This is a result of the way a derivative in the estimator has been approximated using finite difference. This method is sensitive to disturbances. The estimator filters the finite difference derivative, but the output is still noisy because the cut-off frequency of the filter was chosen relatively high. This was done to minimize the amount of phase shift introduced by the filter. In the end, as can be seen from the step response in Figure 7.4, the noise in the estimation does not affect the performance. As previously explained, all the inputs to the MPC are low pass filtered to avoid aliasing. This removes most of the noise from the estimated current. If, however, one should use this estimator in a situation where the output is not filtered before use we suggest that the derivative part of the current is filtered much harder. One such scenario could be implementation of MPC, running at a faster sampling frequency than what we used here.

Lastly, in Chapter 7.1.5 we presented the simulated response to steps in power reference for the SIL system model, regulated by the MPC. These responses are shown in Figure 7.6. The response in input voltage U_d and power modification P_{stab} is shown for the diverse operating conditions of full traction, coasting, and full brake. We hence see that even though designed specifically for stabilization and damping in traction, where the system is unstable, the MPC regulator works as intended and dampens the system response in stable conditions as well. Recall that the undampended system response in stable conditions can still be very oscillatory (see Figure 2.3). This has been achieved using the LPV model, which changes the state space matrices in the MPC problem depending on what the operating point is.

The way the MPC regulator has been tuned, it uses a lot more power modification, P_{stab}, than the H_∞- and H_∞^{sub}-regulators in coasting and braking. As is, the MPC uses constant weights and hence puts the same cost on control input use in stable situations. It could be advantageous to let the weights vary with the operating point parameter θ as well. By increasing the cost on the control input for values of θ where the system is stable, the MPC regulator would likely produce a well damped system response but use less power modification when less is needed.

In the performance specifications we have emphasized the importance of robustness towards modelling errors. One reason why we have done this is because we are aware of the fact that we are neglecting system dynamics, and therefore need that our regulator is robust towards any dynamics which we ignore. The main reason why this has been done is so that we can utilize the power of having a simple model. The second-order model which we use is relatively easy to grasp, and as previously mentioned, the resulting MPC regulator has few tuning parameters. These are advantages when one wants to understand and control a complex system such as a train propulsion system. In a way it is interesting that such a complicated system as a train propulsion system can be modelled in a useful way with such a simple model. An alternative approach, however, would be to try to model some of the dynamics which are neglected in the simple model and cause the system to differ from the ideal CPL. That way the regulator does not need to be as robust, since there is less uncertainty. It would also mean that the regulator could be tuned more towards speed or minimization of control input use. MPC is also quite easy to combine together with more advanced system models since it uses the discrete-time state space system model directly. Once there is a good system model, implementation of the MPC is sort of ”plug-and-play”. Obtaining a good model can be quite difficult however. Another problem with a more advanced model is that it of course will increase the complexity of the MPC optimization problem, making it harder to solve in real time. Real-time implementation aspects are something which we will discuss more in the next subsection.

8.3 Real-Time Implementation

The real-time implementation results from Chapter 7 show that the MPC regulator is a feasible alternative for stabilization of the propulsion system in a train application. It was possible to integrate it into existing Bombardier software and run it on application specific hardware. The MPC optimizer ran on a task level of 5 ms and the results from Chapter 7.4 shows that the execution time of the MPC optimizer was at most about one fifth of the sampling period. The MPC regulator adds a substantial amount of time to the total execution time but leaves plenty of time to spare.

As mentioned in Chapter 5.2.2 there is a disadvantage to running the MPC regulator on such a slow task as 200 Hz, since it lowers the bandwidth of the regulator. If the system which is to be controlled has a high resonance frequency with respect to the available regulator bandwidth, this poses a design challenge. This was the case for the MCX application which was used for the real-time evaluation of the MPC regulator. It would therefore be advantageous from a control perspective to let the MPC operate at a higher sampling frequency. However, as mentioned in Chapter 5.2.2, increasing the sampling frequency increases the complexity of the MPC optimization problem (since the number of prediction samples has to be increased in order for the prediction horizon to cover the same length of time) while also decreasing the available computational time. A similar problem would appear if we would want to stabilize a system which has an input filter with a very low natural frequency. With a low natural frequency, a long prediction horizon, with a large number of time steps, is needed in order for the MPC optimizer to capture enough of the evolution of the system dynamics in the prediction. This will in turn increase the computational complexity. For a system with a lower natural frequency, it might however be possible to lower the sampling frequency, and in turn the regulator bandwidth, without too much of an impact on performance. In the end we see that the conflict between bandwidth and execution speed poses a problem. There are however ways around this problem, which have been out of the scope of this thesis. For example, it is possible let the prediction horizon be different from the control horizon. That way we optimize over a short time horizon, which can be done fast, while still checking that the found solution results in a ”good” predicted future.

Some execution time can be gained by tuning the different solver parameters which were discussed in Chapter 2.9.2. In Chapter 7.1 we experimented with the duality gap tolerance and saw some improvement. However, increasing the sampling frequency of the MPC regulator for our particular problem would probably require different hardware, or a modification of the way the MPC regulator is implemented. For example, one could try explicit MPC, which was discussed in Chapter 2.9.3. With explicit MPC most of the computation is done offline and the online computation can be reduced to a binary search, which can be done very fast. Explicit MPC, however, has the problem that the explicit MPC solution has to be stored. With a system like the one which has been considered in this thesis, where a linear model is parameterized in terms of the operating point, the explicit solution would be different for each operating point. Storing the solution for a wide range of operating points in this manner could easily require a lot of disk memory. On the other hand, the online MPC optimization already requires quite a bit of disk memory. As presented in Chapter 7.5, adding the MPC regulator into Bombardier’s converter control source code increased the file size by around 18 %. The way the optimization algorithm was written, memory use was not taken into account. If that was done, the size of the program could probably be reduced substantially.

Chapter 9

Conclusion

In this chapter we summarize the main findings of the thesis. Then we make an outlook and propose some future work.

9.1 Summary

In this thesis we have designed an LPV-MPC regulator and shown that it is a viable option for stabilizing a CPL connected to a power source via an input filter. The closed loop dynamics, under MPC, can be tuned to reach a quality similar to what can be achieved with classical frequency domain optimization methods, such as H_∞ control. The solution the MPC gives results in a well damped system, with limited control use. It is also robust towards modelling errors. It hence fulfills the performance specifications which were set up. These performance specifications were that the regulator should

1. give well damped system dynamics
2. limit interference with the overall control objective of following the power reference
3. be robust towards modelling errors

Explicit handling of control constraints makes the MPC regulator perform better than the H_∞^{sub} regulator (which has been used as a benchmark), in situations where the control input is constrained, such as when we are operating close to motor current limits and the control input is limited to only negative input signals. However, these improvements are minor. We can learn from the behavior which the MPC exhibits in these situations to improve existing regulator designs. For example, the damping of the H_∞^{sub} regulator could be made dependent on the control constraints. This could improve the system dynamics in those situations.

In this thesis we have also shown that it is feasible to implement MPC in existing Bombardier control software and hardware and run online LPV-MPC in real-time. The results are useful for understanding the demands which online MPC puts on modern propulsion control hardware, should industry adaptation of this control method be desirable. Hence the goals which were set up in Chapter 1.5 have been accomplished.

9.2 Future Work

We suggest the following future investigations on the topic of MPC and converter stabilization.

The system model which has been considered for stabilization controller design in this thesis is that of the CPL coupled with an input filter. Although this model works well in practice for many converter applications, there are important dynamics which are neglected with this simplification. The impact of that was seen in simulations with the MCX train applications, which had dynamics around its natural frequency which deviated a lot from that of a CPL. It would therefore be interesting to investigate the performance of the MPC regulator, would the system model be extended to include some of these dynamics. Would the performance improve, and if so by how much? The real-time implementation details of such a regulator would also be interesting to investigate.

In MATLAB/Simulink simulations we have seen that the H_∞-regulator outperformed both the MPC- and the H_∞^{sub}-regulator in all test cases. The H_∞-regulator differs from the other two in that it requires the line voltage E. This signal, which corresponds to the voltage of dc voltage source which powers the train, cannot be measured directly since it is not located on the train. It would however be interesting to investigate if it is possible to realize a version of the H_∞ regulator which estimates the line voltage E. It would be interesting to know how reliable of an estimate it is possible to get of the line voltage, or if the input voltage to the train will work well enough. Furthermore, it would be useful to know how close to the ideal MATLAB/Simulink results it is possible to get in a real implementation.

In this thesis we have demonstrated the computational limits which exist with online MPC. In a converter application, where computational speed is important, this becomes a big design challenge when it comes to real-time implementation of the MPC regulator. It would therefore be interesting to investigate the implementation of explicit MPC for stabilization of a train propulsion system, to see what the potential advantages or disadvantages would be. For example, how much would an explicit MPC solution gain in terms of computational speed? And in return, what would it cost in terms of memory? The practical aspects of efficiently storing the explicit solution for a wide range of operating points would also be interesting to look at.

In this thesis we have investigated the use of online MPC for stabilization of a converter application which has constraints on the control input. Our last proposal for future work is therefore an investigation into the use of online MPC for stabilization of converter applications which instead have output constraints. Such constraints could for example be related to the quality of the output voltage of the converter. What would the advantages of MPC in such an application be, and what are the design challenges?

Bibliography

[1] Bombardier Transportation. *A London Underground Central line train*. Aug. 2017. URL: https://www.bombardier.com/en/media/newsList/details.BT_20170825.bombardiercom.html.

[2] SCJiang. *Rolling stock of Changzhou Metro Line 1*. Sept. 2019. URL: https://commons.wikimedia.org/wiki/File:Line_1,_Changzhou_Metro.jpg.

[3] Bombardier Transportation. *Rendering of TC 1500 converter module with open side*. 2020.

[4] Bombardier Transportation. *Photo of permanent magnet motor*. 2020.

[5] M. WU and D.D.-C. LU. "Active stabilization methods of electric power systems with constant power loads: a review". In: *Journal of Modern Power Systems and Clean Energy* 2.3 (2014), pp. 233–243. ISSN: 21965625.

[6] Suresh Singh, Aditya R Gautam, and Deepak Fulwani. "Constant power loads and their effects in DC distributed power systems: A review". eng. In: *Renewable and Sustainable Energy Reviews* 72 (2017), pp. 407–421. ISSN: 1364-0321.

[7] Henrik Mosskull. "Constant power load stabilization". eng. In: *Control Engineering Practice* 72 (2018), pp. 114–124. ISSN: 0967-0661.

[8] Xinyun Liu, A.J Forsyth, and A.M Cross. "Negative Input-Resistance Compensator for a Constant Power Load". eng. In: *IEEE Transactions on Industrial Electronics* 54.6 (2007), pp. 3188–3196. ISSN: 0278-0046.

[9] P Magne et al. "Large-Signal Stabilization of a DC-Link Supplying a Constant Power Load Using a Virtual Capacitor: Impact on the Domain of Attraction". eng. In: *IEEE Transactions on Industry Applications* 48.3 (2012), pp. 878–887. ISSN: 0093-9994.

[10] S.D Sudhoff et al. “DC link stabilized field oriented control of electric propulsion systems”. eng. In: *IEEE Transactions on Energy Conversion* 13.1 (1998), pp. 27–33. ISSN: 0885-8969.

[11] Sigurd Skogestad. *Multivariable feedback control : analysis and design*. eng. 2 ed.. Hoboken, NJ: John Wiley, 2005. ISBN: 0-470-01167-X.

[12] Johnc Doyle et al. “State-space solutions to standard H2 and H(infinity) control problems”. eng. In: *IEEE Transactions on Automatic Control* 34 (1989), pp. 831–847. ISSN: 0018-9286. URL: http://search.proquest.com/docview/25488692/.

[13] Jay H. Lee. “Model predictive control: Review of the three decades of development”. In: *International Journal of Control, Automation and Systems* 9 (June 2011), pp. 415–424. URL: https://doi.org/10.1007/s12555-011-0300-6.

[14] Sergio Vazquez et al. “Model Predictive Control for Power Converters and Drives: Advances and Trends”. eng. In: *IEEE Transactions on Industrial Electronics* 64.2 (2017), pp. 935–947. ISSN: 0278-0046.

[15] Basil Kouvaritakis. *Model Predictive Control Classical, Robust and Stochastic*. eng. 1st ed. 2016.. Advanced Textbooks in Control and Signal Processing. 2016. ISBN: 3-319-24853-7.

[16] Yang Wang and S Boyd. “Fast Model Predictive Control Using Online Optimization”. eng. In: *IEEE Transactions on Control Systems Technology* 18.2 (2010), pp. 267–278. ISSN: 1063-6536.

[17] S Mariethoz and M Morari. “Explicit Model-Predictive Control of a PWM Inverter With an LCL Filter”. eng. In: *IEEE Transactions on Industrial Electronics* 56.2 (2009), pp. 389–399. ISSN: 0278-0046.

[18] Ján Drgoňa et al. “Optimal control of a laboratory binary distillation column via regionless explicit MPC”. eng. In: *Computers and Chemical Engineering* 96 (2017), pp. 139–148. ISSN: 0098-1354.

[19] S Kouro et al. “Model Predictive Control-A Simple and Powerful Method to Control Power Converters”. eng. In: *IEEE Transactions on Industrial Electronics* 56.6 (2009), pp. 1826–1838. ISSN: 0278-0046.

[20] J Rodriguez et al. “Predictive Current Control of a Voltage Source Inverter”. eng. In: *IEEE Transactions on Industrial Electronics* 54.1 (2007), pp. 495–503. ISSN: 0278-0046.

[21] P Cortes et al. “Predictive Current Control Strategy With Imposed Load Current Spectrum”. eng. In: *IEEE Transactions on Power Electronics* 23.2 (2008), pp. 612–618. ISSN: 0885-8993.

[22] P Cortes et al. “Predictive Control in Power Electronics and Drives”. eng. In: *IEEE Transactions on Industrial Electronics* 55.12 (2008), pp. 4312–4324. ISSN: 0278-0046.

[23] Tomislav Dragicevic. “Dynamic Stabilization of DC Microgrids With Predictive Control of Point-of-Load Converters”. eng. In: *IEEE Transactions on Power Electronics* 33.12 (2018), pp. 10872–10884. ISSN: 0885-8993.

[24] Stephen Boyd Jacob Mattingley. *CVXGEN: Code Generation for Convex Optimization*. CVXGEN LLC. URL: https://cvxgen.com/docs/index.html. (accessed: 10.03.2020).

[25] Bombardier Transportation. *Photos of EOS Controller Board*. 2020.

[26] Bombardier Transportation. *Diagram of EOS controller and Motor Converter*. 2020.

[27] dSPACE. *Photo of SCALEXIO LabBox*. 2020. URL: https://www.dspace.com/en/pub/home/products/hw/simulator_hardware/scalexio.cfm#144_54760.

Appendix A

Mathematical Definitions

A.1 Singular Value Decomposition

Any complex $n \times m$ matrix A can be factorized using the singular value decomposition (SVD)

$$A = U\Sigma V^H \tag{A.1}$$

where the $n \times n$ matrix U and the $m \times m$ matrix V are unitary, meaning that they fulfill

$$U^H = U^{-1} \tag{A.2}$$

A unitary matrix has eigenvalues (and singular values) with absolute value equal to one. The $n \times m$ matrix Σ is a matrix which contains a diagonal matrix σ_1 with singular values σ_i which are real and non-negative. The matrix Σ is arranged with the the singular values in descending order according to

$$\Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix}, \quad n \geq m \tag{A.3}$$

or

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \end{bmatrix}, \quad m \leq m \tag{A.4}$$

where

$$\Sigma_1 = \text{diag}(\sigma_1, \sigma_2, \cdots, \sigma_l), \quad l = \text{min}(n, m) \tag{A.5}$$

and

$$\overline{\sigma} = \sigma_1 \geq \sigma_2 \cdots \geq \sigma_l = \underline{\sigma} \tag{A.6}$$

The columns of U and V form orthonormal basis vectors for the columns and the rows of A. The singular values σ_i are related to the eigenvalues of $A^H A$ and AA^H through

$$\sigma_i = \sqrt{\lambda_i(A^H A)} = \sqrt{\lambda_i(AA^H)} = \tag{A.7}$$

where λ_i is the ith largest eigenvalue of either A^HA or AA^H.

A.2 Separation Principle

Here follows a proof of the separation principle. Consider a deterministic LTI system on state space form

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{A.8}$$

where x is the state, u is the input signal and y is the output signal. We can design a state feedback

$$u = -Lx \tag{A.9}$$

such that the closed loop dynamics are

$$\dot{x} = (A - BL)x \tag{A.10}$$

If L is chosen such that the eigenvalues of $(A - BL)$ are in the LHP the system is stable.

Suppose that the system isn't fully deterministic. Maybe we cannot measure all the states, or there might be measurement noise or process noise. Let $\hat{x}$ be the estimate of x, and let the new feedback law be

$$u = -L\hat{x} \tag{A.11}$$

We can then design an observer on the form

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{A.12}$$

If we define the error signal as

$$e = x - \hat{x} \tag{A.13}$$

then

$$\dot{e} = (A - KC)e \tag{A.14}$$

$$u = -L(x - e) \tag{A.15}$$

Furthermore, we can write the equations for the closed loop system as

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} (A - BL) & BL \\ 0 & (A - KC) \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} \tag{A.16}$$

Since the matrix is triangular, the eigenvalues of the system is equal to the eigenvalues of the diagonal elements, i.e $(A - BL)$ and $(A - KC)$. This means that the stability is decoupled and that the observer and the feedback can be designed independently.

A.3 Relating Discrete-Time LQR to MPC

Let

$$J(x_0) = \inf_{u_0,u_1,\cdots} \sum_{k=0}^{\infty} x_k^\top Q x_k + u_k^\top R u_k \tag{A.17}$$

be the discrete-time infinite-horizon cost-to-go from x_0. We can rewrite it into an equivalent finite-horizon cost function in the following way

$$\begin{aligned} J(x_0) &= \inf_{u_0,\cdots,u_{N-1}} \sum_{k=0}^{N-1} x_k^\top Q x_k + u_k^\top R u_k + \underbrace{\inf_{u_N,u_{N+1},\cdots} \sum_{k=N}^{N-1} x_k^\top Q x_k + u_k^\top R u_k}_{\infty\text{-horizon cost from state } x_N} \\ &= \inf_{u_0,\cdots,u_{N-1}} \sum_{k=0}^{N-1} x_k^\top Q x_k + u_k^\top R u_k + x_N^\top P x_N \end{aligned} \tag{A.18}$$

where P is the solution to the discrete-time algebraic Riccati equation

$$P = A^\top P A - A^\top P B (R + B^\top P B)^{-1} B^\top P A + Q \tag{A.19}$$

By setting $Q_f = P$ in the MPC cost function (2.70) we will hence get a terminal penalty which reflects the future cost-to-go.

A.4 Definition of Matrices in Compact QP Formulation

In this appendix we give the definition to the matrices in (2.75), which is a compact version of the general QP

$$\begin{aligned} \text{minimize} \quad & \sum_{i=0}^{N-1} \left(\begin{bmatrix} x_i^\top & u_i^\top \end{bmatrix} \begin{bmatrix} Q & S \\ S^\top & R \end{bmatrix} \begin{bmatrix} x_i \\ u_i \end{bmatrix} \right) + x_N^\top Q_f x_N \\ \text{subject to} \quad & x_{i+1} = A x_i + B u_i \qquad\qquad i = 0, 1, \cdots, N-1 \\ & F_x x_i + F_u u_i \le f \qquad\qquad i = 0, 1, \cdots, N-1 \\ & F_f x_N \le f_f, \quad x_0 = x_k \end{aligned} \tag{A.20}$$

of which (2.7) is a subset. We repeat the compact QP here

$$\begin{aligned} \text{minimize} \quad & z^\top H z \\ \text{subject to} \quad & Pz \le h, \quad Cz = b \end{aligned} \tag{A.21}$$

In (A.20) F_x, F_u and f describe the set of linear inequalities constraining x_i and u_i. F_f and f_f describe a separate set of inequalities which only applies to x_N. The definition of the matrices of (A.21) are

$$H = \begin{bmatrix} R & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & Q & S & \cdots & 0 & 0 & 0 \\ 0 & S^\top & R & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Q & S & 0 \\ 0 & 0 & 0 & \cdots & S^\top & R & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & Q_f \end{bmatrix}$$

$$P = \begin{bmatrix} F_u & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & F_x & F_u & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & F_x & F_u & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & F_f \end{bmatrix}$$

$$C = \begin{bmatrix} -B & I & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -A & -B & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & -A & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & I & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & -A & -B & I \end{bmatrix}$$

$$h = \begin{bmatrix} f - F_x x_0 \\ f \\ \vdots \\ f \\ f_f \end{bmatrix}$$

$$b = \begin{bmatrix} Ax_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

TRITA-EECS-EX-2020:666
www.kth.se