DEVELOPMENT OF A ROBUST DRIVER MODEL WITH PARAMETER ADAPTATION

ENTWURF EINES ROBUSTEN FAHRERMODELLS MIT PARAMETER-ADAPTION

Diplomarbeit

von

Norbert Müller

Laboratoire d'Automatique de Grenoble (LAG)
Prof. Dr. L. Dugard, Prof. M. M'Saad, MS R. Ramirez-Mendoza

Institut für Industrielle Informationstechnik (IIIT), Universität Karlsruhe (TH)
Prof. Dr.-Ing. U. Kiencke, Dipl. Ing. R. Majjad

Dezember 1996

Meinen Eltern

und Petra

Abstract

In this work a feasibility study of the robust predictive control for the design of an automatic steering system is presented. The involved vehicle is a full-scale nonlinear model of a BMW520i and the feedback is only on the lateral displacement. The robustness feature is particularly motivated by variations in velocity, tyre-road contact and vehicle mass. The control design is performed using a sensitivity function shaping procedure that heavily borrows from the robust control culture. A feedforward approach using the curvature of the incoming road as preview information, as well as a multi-model adaptation and a suitable self-tuning adaptation have been investigated. Intensive simulations are carried out to emphasise the performance of the proposed automatic steering system.

Zusammenfassung

Die vorliegende Arbeit beschäftigt sich mit der Realisierbarkeit und dem Entwurf eines robusten, automatischen Spurführungssystems für ein Kraftfahrzeug. Motivation für die Robustheit des vorgestellten, prädiktiven Reglers sind unsichere physikalische Parameter wie Fahrgeschwindigkeit, Beladung und Kraftschluß zwischen Reifen und Fahrbahn. Einzige Rückführgröße ist die Querabweichung des Fahrzeugs von der Sollspur. Der Reglerentwurf besteht aus dem „Shaping“ von Empfindlichkeitsfunktionen. Eine Störgrößenaufschaltung unter Verwendung der Kurvenkrümmung als meßbare Störgröße, sowie eine Multi-Modell-Adaption und eine geeignete Self-Tuning Adaption wurden realisiert. Umfangreiche Simulationen unter Verwendung eines komplexen, nichtlinearen Fahrzeugmodells eines BMW520i zeigen die Leistung des vorgestellten Lenksystems.

Acknowledgements

The present work could not have been carried out without the help of many people.

In the first place I wish to thank my supervisors at the Laboratoire d'Automatique de Grenoble, Prof. Dr. L. Dugard, and Prof. M. M'Saad since they contributed the basic ideas of the control aspect and for the formulation of the control problem. Their experience in robust and adaptive control and their kind advices contributed a lot to the present work.

Special thanks go to MS R. Ramirez-Mendoza, I acknowledge countless discussions with him, helping me to climb over all kinds of obstacles during my work. It was not only his experience in control design for automotive applications, but also his modest and honest personality which I greatly appreciated.

Further, I want to thank Prof. Dr.-Ing. U. Kiencke of the University of Karlsruhe as my responsible supervisor in Germany, and Dipl. Ing. R. Majjad, since they always had an open ear to my concerns.

Moreover, my thanks go to Joël Chebassier, who has given his time and effort to help me programming and who enabled me to implement the needed modules of SIMART within this short period of time I had.

Finally, I want to thank Petra, who endured many days of neglect during my absence of this 7 months stay at Grenoble, and who nevertheless supported me - morally and emotionally.

At the end of this final year project I also want to thank my dear parents and my family, they supported me morally, emotionally, and financially, and enabled me to study at the university.

Norbert Müller

Erklärung

Hiermit erkläre ich, daß ich die vorliegende Arbeit selbständig und ohne fremde Hilfe angefertigt habe, und daß ich keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Grenoble, den 2. Dezember 1996

Norbert Müller

Contents

1 INTRODUCTION 3

1.1 ORGANISATION OF THE TEXT 3
1.2 AUTOMATIC STEERING 4

2 VEHICLE DYNAMICS 7

3 PARAMETER ESTIMATION 10

3.1 THE LEAST SQUARES ESTIMATOR 10
3.2 RECURSIVE LEAST SQUARES ESTIMATOR 12
3.3 EXPONENTIAL DATA WEIGHTING 13
3.4 EXPONENTIAL DATA WEIGHTING FOR TIME-VARYING SYSTEMS 14
3.5 NUMERICAL ROBUSTNESS CONSIDERATIONS AND ADAPTATION GAIN REGULARIZATION 14
3.6 NORMALISATION 16
3.7 ADAPTATION FREEZING 17
3.8 INCORPORATION OF PRIOR KNOWLEDGE 19

4 EXPERIMENT DESIGN AND OFF-LINE IDENTIFICATION RESULTS 20

4.1 IDENTIFICATION OF VEHICLE DYNAMICS 20
4.2 INPUT SIGNAL SHAPE 22
4.3 SELECTION OF SAMPLING PERIOD AND FILTER FREQUENCIES 23
4.4 SELECTION OF MODEL STRUCTURE 24
4.4.1 Physical Insights 24
4.4.2 Preliminary Data Analysis 26
4.4.3 Physical Interpretation 27
4.5 MODEL VALIDATION 27
4.5.1 Prediction Error Consideration 28
4.5.2 Spectrum Analysis 28
4.5.3 Pole-Zero Cancellations 29
4.5.4 Simulation 30
4.5.5 Residuals 31
4.6 IDENTIFICATION RESULTS 31

5 DESIGN OF INSENSITIVE CONTROL SYSTEMS 34

5.1 SENSITIVITY FUNCTIONS 34
5.2 ROBUSTNESS MARGINS 37
5.3 STABILITY ROBUSTNESS 39

6 THE GENERALIZED PREDICTIVE CONTROL DESIGN 44

6.1 THE PREDICTIVE CONTROL LAW 44
6.1.1 Plant Model Description and Noise Models 46
6.1.2 The Output Prediction 48
6.1.3 The Recursion of the Diophantine Equation 50
6.1.4 The Control Law 51
6.2 CHOICE OF GPC DESIGN PARAMETERS 52
6.2.1 Choice of Costing Horizons and Weighting Factor 52
6.2.2 Selection of predictor poles 53
6.2.3 Selection of D-Polynomial 54
6.2.4 Selection of Input and Output Filters 54
6.3 IMPLEMENTATION ISSUES AND SHORTCOMINGS 55
6.4 GPC FOR AUTOMATIC STEERING 55
6.4.1 Design Specifications 56
6.4.2 Design Parameters 56
6.4.3 The Robustness Quantifiers 59
6.5 SIMULATION RESULTS 61
6.5.1 Entering the Curved Test Path 61
6.5.2 Side Wind Forces 63
6.5.3 Transition from Manual to Automatic Steering 65

7 FEED-FORWARD 67

8 ADAPTIVE CONTROL 70

8.1 OPEN LOOP ADAPTATION 70
8.1.1 Multi-Model Approach 71
8.1.2 Simulation Results 71
8.2 ADAPTATION IN CLOSED-LOOP: SELF-TUNING CONTROL 73
8.2.1 The Structure of a Self-Tuning Regulator 73
8.2.2 Simulation Results 74
8.2.3 The Role of Signal Normalisation and Adaptation Freezing 76
8.2.4 Supervision 79

9 CONCLUSIONS AND SUGGESTIONS 81

10 LITERATURE 83

APPENDIX A: COMPARISON J-STEP AHEAD PREDICTOR AND SMITH PREDICTOR 87

APPENDIX B: RELATIONSHIP OF GPC WITH OTHER APPROACHES 90

MINIMUM VARIANCE CONTROL 90
DEAD-BEAT CONTROL 90
POLE-ASSIGNMENT CONTROL 91
INFINITE HORIZON LQ CONTROL 92

1 Introduction

The development of advanced control systems in the automotive field, like anti-lock braking and acceleration skid control, like active suspension systems and control of lateral vehicle dynamics, required many years of extensive research in sophisticated control design. Indispensable tools in control design are powerful computer simulators, being cost-efficient, providing reproducible test conditions, and replacing risky driving manoeuvres. The vehicle simulators became more and more complex and can nowadays demonstrate the vehicle dynamics under various driving conditions until the limit of stability [Daiß96, Maj96a, Maj96b].

In order to complete a vehicle simulation, a driver model needs to be considered, or at least a controller that performs equivalent tasks. The development of driver models for vehicle simulators is basically motivated by two facts: firstly, it can be desirable to simulate the vehicle and its dynamics in closed-loop, e.g. with the influence of the driver. This is in contrast to open-loop simulations, where only test signals on the steering wheel are applied to study the vehicle dynamics or where the reaction of the vehicle on specific disturbances as side wind is studied using fixed control, i.e. the steering angle remains constant. Closed-loop manoeuvres with “real” driver models are expected to give further insights into vehicle handling and to enable studying the influence of vehicle parameters on the subjective handling right during the development phase of a vehicle. In this case a driver model is needed, which reflects perfectly the behaviour of a real driver, including his imperfection, but also considering the multitude of information, which is evaluated by a human driver. Research in this field has been done for quite a long time [Horn86, Rei90, Ris91] and becomes currently more and more a field of special interest.

A second task of a steering system, besides perfect driver modelling, can be to carry out specific driving manoeuvres like cornering *with optimal performance*. This is basically important to investigate the performance of implemented control systems like active suspension, anti-lock braking or vehicle dynamics control. In this case not the modelling of a real driver is emphasised, but an optimal behaviour is desired, to follow perfectly the given experimental specifications.

In order to investigate advanced robust control theory, the second approach is followed in this work for automatic steering. The task of steering a vehicle on a given trajectory represents a challenging control problem if variations of velocity, road conditions and vehicle mass are allowed [Ack93a, Ack93b]. The main advantage of this approach is the applicability of the controller not only for simulation purpose, but also as a part of these integrated highway systems that are currently designed for the next century and where automotive steering controller will be of practical interest. In the second subsection of this paragraph, the idea of an integrated highway system and its expected benefits as well as the problem of automatic car steering are introduced.

1.1 Organisation of the Text

The presented steering system has been developed and evaluated using a full-scale, non-linear vehicle simulator which had been developed at the Institute for Industrial Information Techniques of Technical University Karlsruhe to investigate the automotive control [Maj96a, Kie96b]. This vehicle model, implemented under Matlab/Simulink, is described briefly in chapter 2.

In previous studies the classical linear single-track model was used for control design [Ack95, Ack93b, Gul95, Peng90, Ram95]. However, in the present work the vehicle dynamics of the complex, non-linear vehicle simulator have been identified over the domain of possible operating conditions. The underlying Least Squares estimator is described in chapter 3, the estimation procedure and the identification results are explained in chapter 4. The control model identification and its validation was carried out using the advanced control software package SIMART [Hej94, M'Saad94].

When dealing with time-varying systems, basically two approaches for control design are possible: *Robust Control* design provides a controller which stabilises the system for all operating conditions and which meets certain performance specifications, but it does not necessarily work with optimal performance. *Adaptive Control* leads to more complex, non-linear controllers, which adapt their behaviour to the time-varying system [Åst84, Iser87].

An introduction to Robust Control theory is given in chapter 5. Some expressions, commonly used in design and analysis of insensitive control systems, are explained, as well as the standard robustness quantifiers like robustness margins and sensitivity functions.

The control design of the developed robust steering controller has been carried out using an appropriate predictive control approach that has been developed from the Generalized Predictive Control (GPC) [Cla87, Cla89, Soet90] in the spirit of the robust control theory [Lim95, Zho95]. This approach is explained in chapter 6. Its application for automatic steering is explained, and the obtained performance is illustrated for several driving conditions. Besides the standard design specifications, namely maintaining the lateral displacement as small as possible for typical manoeuvres taking into account the actuator constraints and the passenger comfort, suitable shapings of the usual sensitivity functions had to be achieved. The performance specification, the control design, and the stability and performance robustness analysis were carried out using the software package SIMART. The obtained controller was then exported to Matlab/Simulink and the experimental validation was carried out using the complex, non-linear vehicle simulator.

The performance of the robust GPC controller has been improved using a feedforward structure, assuming the curvature of the coming road to be available as *a priori* information. This is explained in chapter 7.

Chapter 8 finally presents the adaptive control approach, an open-loop adaptation as well as a self-tuning regulator have been implemented. For this purpose, the required identification and control modules of the software package SIMART were exported and included into the vehicle simulator under Matlab/Simulink.

Chapter 9 gives a summary and an overall conclusion, as well as an outlook and future perspective.

1.2 Automatic Steering

During the middle 1980s, transportation planners and researchers in Europe, North America, and Japan realised that the rapidly worsening traffic problems could not be solved with conventional technologies. Information technologies need to be applied in a much broader way than ever before, and the integration of travellers, vehicles, and roadway infrastructure into a comprehensive system is necessary. Such applications of information technology are relatively common in air, rail, and marine

transportation systems today. However they are extremely rare in road transportation, despite the dominant role of rubber-tired transportation systems throughout the industrialised world [Shla93].

In 1990 the Intelligent Vehicle Highway Society of America (IVHS) was chartered, to develop a strategic plan for the development and deployment of IVHS in the US. The goals of the IVHS program are to improve the performance of the surface transportation system in a wide variety of dimensions by

- reducing traffic congestion, pollution, and energy use,
- improving safety,
- enhancing mobility of travellers, especially of the elderly and disabled,
- increasing the viability of public transportation, and
- increasing the convenience of travel.

The IVHS program in the US has been subdivided into the following families of technology: Advanced Traffic Management Systems, Advanced Traveller Information Systems, and Advanced Vehicle Control Systems (AVCS). The latter includes not only systems that can take complete longitudinal and lateral control of the movements of a vehicle but also systems that can assist a driver in controlling the vehicle. It subdivides into three separate stages of development:

- driver warning systems, including frontal collision warning, lane departure warning, loss of traction (ice) warning, and driver performance monitoring and drowsiness warning,
- driver control assistance systems, including autonomous intelligent cruise control, collision avoidance (braking and/or steering), and lane holding (steering assistance), and
- fully automated vehicle control systems, such as the Automated Highway Systems (AHS), including automated vehicles on special-purpose lanes, automated vehicles on their own freeway network, autonomous automated vehicles, and automatic parking.

AHS are particularly controversial because of their significant difference from present-day operations. While some observers point out the potentially very large benefits in safety, capacity, and efficiency that AHS could offer, others criticise the technical risks to overcome and the considerable investments that will be needed to realise those benefits [Shla93]. The goal of an AHS design is to significantly increase safety and highway capacity without having to build new roads. The human drivers have no direct control over the vehicles in an AHS as the automated highway system controls all the vehicles [God96].

AHS require a wide range of technologies, including traffic management systems that aid drivers, and co-operative systems that link the two. One of the fundamental parts of AHS are lane following manoeuvres, i.e. automatic steering [Che95].

The main purpose of automatic steering consists in performing a robust tracking of a reference line. This reference line is located in the middle of the lane and may consist of the magnetic field of an electrically supplied wire or permanent magnets in the road [Ack95]. The design of an automatic steering system is a robustness problem in view of large variations in velocity, mass of the vehicle, and friction coefficient

between tyre and road surface. In this work the applicability of the robust predictive control for automatic steering, with and without parameter adaptation, is investigated, using solely feedback of the lateral displacement measurement unlike in the earlier design studies [Ack95, Sie94, Peng90]. The front steering angle is used as the manipulated variable. The involved tracking error is measured by a displacement sensor, which is mounted in the centre of the front end of the vehicle.

Typical reference manoeuvres are

- transition from a straight line into a circle,
- transition from manual to automatic operation,
- side wind forces.

Specifications are given in terms of maximal displacement from the guideline and maximal steering angle and steering angle rate. The involved control problem represents a challenging opportunity to investigate the advanced control techniques.

2 Vehicle dynamics

In this chapter, the non-linear vehicle model, developed at the Institute for Industrial Information Techniques of the Technical University of Karlsruhe, is briefly described. It has been developed in order to investigate the automotive control, previous experimental studies have already demonstrated its suitability and performance [Kie93, Kie94, Kie96]. This realistic vehicle model has been identified over the operating domain in order to obtain a set of linear models, of which a nominal control model has been selected for the robust control design. The complex vehicle model has further been used for experimental evaluation of the robust controller. For a detailed description of the model please see [Daiß96, Maj96a, Maj96b].

The vehicle model consists of several sub-models which can be developed and modified independently. The most important sub-models are chassis, tyres, suspension, road model, steering system, and aerodynamic model, all important nonlinearities are taken into account. A driver model can be included, in the present work a controller for the longitudinal speed was only used. Figure 2.1 shows the basic structure. The data has been adapted to a BMW520i and proved to demonstrate a realistic behaviour during various driving conditions, namely normal to critical situations at the limit of stability [Daiß96, Maj96a].

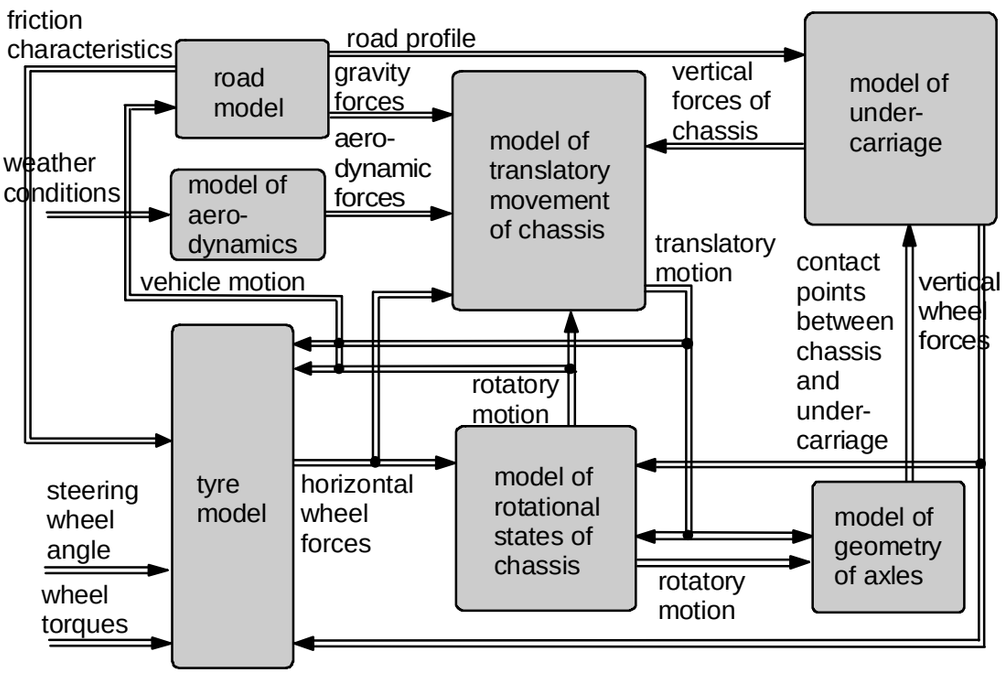


Figure 2.1: Sub-models of the vehicle simulation model

The global input variables for this simulation are the front steering angle, the initial velocity, the weather and road conditions and the road trajectory. This trajectory is composed of different road section for each of them the desired velocity needs to be defined, as well as the road curvature, the wind speed and direction, the weather

condition, e.g. dry road, wet road, snow road, and the road surface, e.g. concrete, asphalt, or cobble-stone.

The Chassis Model

The chassis frame is described by a six degree of freedom model, i.e. three translational and three rotational degrees of freedom, as it is shown in figure 2.2. The latter can be described by three torque equations, namely the equation for the yaw rate, the torque equation round a longitudinal axis (roll movement), and the torque equation round a lateral axis (pitch movement).

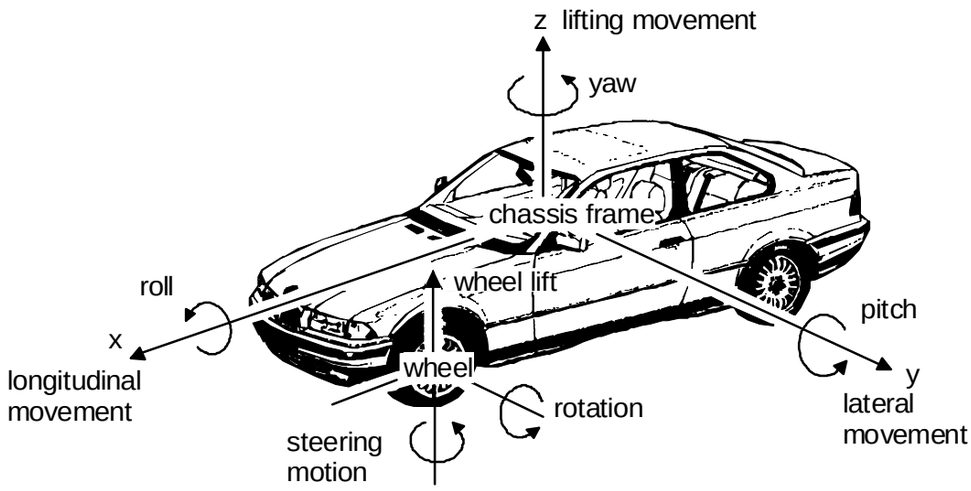


Figure 2.2: Degrees of freedom of the chassis frame and the wheel model

The equations of moments for the translational movements take the longitudinal and lateral wheel forces of the four wheels as well as wind forces, gravity forces and frictional forces into account. The equations of moments consider longitudinal and lateral wheel forces, dynamic load forces due to braking and acceleration, as well as wind influences.

The Tyre Model

The tyre model represents a central part of the vehicle model. A number of factors affect the tyre dynamics and thus the longitudinal and lateral wheel forces: the tyre-road contact characteristic, the velocity of the vehicle, and the wheel load.

The calculation of the wheel slip, wheel slip angle, camber angle, friction coefficient and wheel forces are carried out as described in [Bur93, Daiß96, Maj96a]. The friction characteristic of the road is a non-linear curve with a dependence not only on the surface but also on the vehicle speed v_{ch} and the normal force F_Z between tyre and road [Bur93].

Figure 2.3 shows the characteristics for dry asphalt, for wet asphalt, for snow and for ice.

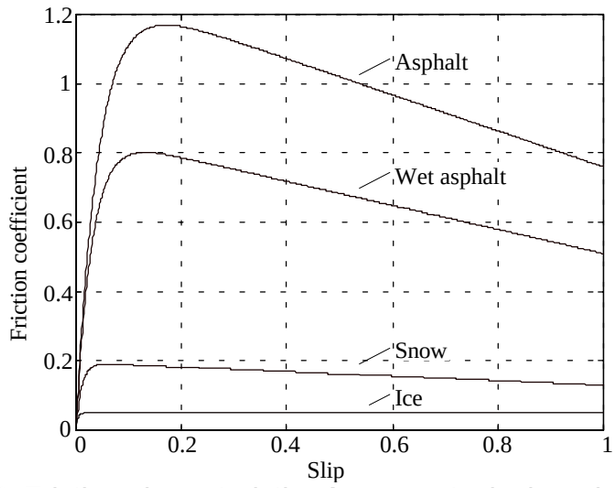


Figure 2.3: Friction characteristics for some typical road conditions

The Suspension Model

The undercarriage and suspension system has been modelled as four independent two-degree-of-freedom quarter-vehicle models with non-linear damper characteristics. All the four suspensions of the vehicle are of the same structure, with different parameters.

Each shock absorber is assumed to be described by a suspension spring and a non-linear damper, both or them are mounted between axle mass and vehicle body. The tyre stiffness is also modelled.

The non-linear damper characteristic is described by a 4^{th} order polynomial and has been obtained by experimental results.

3 Parameter Estimation

Since the applied predictive control design is based on linear control model, the vehicle dynamics have been identified over a domain of possible operating conditions using a recursive least squares estimation algorithm. This represents an ideal case for identification since for each operating condition the plant behaviour remains constant during identification measurements. However, for the proposed adaptive version the vehicle behaviour becomes time-varying since the vehicle dynamics are identified during operation in closed loop. This requires a much more advanced estimation algorithm which performs well in the presence of state disturbances, plant parameter variations and unmodelled dynamics [M'Saad94a, M'Saad94c].

In the following paragraph the applied robust recursive least squares estimation procedure is presented, it includes pre-treatment of data like filtering and normalisation, as well as exponential data weighting, adaptation gain regularization, adaptation freezing and incorporation of prior knowledge. These algorithms are available in the software-package SIMART, which was used for the off-line identification procedure. For the adaptive version, these SIMART-modules have been exported and implemented in Matlab/Simulink.

3.1 The Least Squares Estimator

We consider the following linear discrete time shift operator model

$$A(q^{-1})\, y(t) = B(q^{-1})\, u(t-d-1) \qquad (3.1.1)$$

where

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_{n-1} q^{n-1} + a_n q^{-n} = 1 + q^{-1} A^*(q^{-1})$$

$$B(q^{-1}) = b_0 + \ldots + b_m q^{-m}$$

The order of numerator m and denominator n are known, as well as the time delay d.

The output of the system at the time t can then be expressed as follows:

$$y(t) = -A^*(q^{-1})\, y(t-1) + B(q^{-1})\, u(t-d-1) = -\sum_{i=1}^{n} a_i y(t-i) + \sum_{i=0}^{m} b_i u(t-d-1-i) =$$

$$= -a_1 y(t-1) - \ldots - a_n y(t-n) + b_0 u(t-d-1) + \ldots + b_m u(t-d-1-m)$$

which is the same as

$$y(t) = \phi(t-1)^T \theta \qquad (3.1.2)$$

where

$$\phi(t-1)^T = [-y(t-1), \ldots, -y(t-n), u(t-d-1), \ldots, u(t-d-1-m)]$$

$$\theta^T = [a_1, \ldots, a_n, b_0, \ldots, b_m]$$

$\phi(t-1)^T$ is called *regression vector* or *regressor*.

The least square identification method depends on the ability to rearrange the model which has to be estimated so that the predicted output is describable as a linear function with respect to the parameters, i.e. like equation (3.1.2).

If the estimated model is described as

$$\hat{y}(t) = \phi(t-1)^T \hat{\theta}$$

the *prediction error* between the output of the real model y(t) and the predicted output $\hat{y}(t)$ using the estimated model becomes

$$\varepsilon(t) = y(t) - \phi(t-1)^T \hat{\theta} \qquad (3.1.3)$$

No matter which parameter estimation procedure is selected, the guiding principle is to select $\hat{\theta}$ so that the prediction error $\varepsilon(t)$, t = 1, 2, ..., max(n, m+d) becomes as small as possible. For the least squares algorithm, the following quadratic criterion is to be minimised:

$$J(\hat{\theta}, t) = \sum_{i=1}^{t} [\varepsilon(t)]^2 = \sum_{i=1}^{t} \left[y(i) - \phi(i-1)^T \hat{\theta} \right]^2 \qquad (3.1.4)$$

The result can be calculated by setting the partial derivative to zero:

$$\frac{\partial J(\hat{\theta}, t)}{\partial \hat{\theta}} = -2 \sum_{i=1}^{t} \phi(i-1) \left[y(i) - \phi(i-1)^T \hat{\theta} \right] = 0$$

which can be written as

$$\sum_{i=1}^{t} \phi(i-1)\, \phi(i-1)^T\, \hat{\theta} = \sum_{i=1}^{t} y(i) \phi(i-1)$$

The estimated parameter $\hat{\theta}$ can finally be calculated as

$$\hat{\theta} = \left(\sum_{i=1}^{t} \phi(i-1)\, \phi(i-1)^T \right)^{-1} \sum_{i=1}^{t} y(i)\, \phi(i-1) = F(t) \sum_{i=1}^{t} y(i)\, \phi(i-1)\,, \qquad (3.1.5a)$$

where

$$F(t)^{-1} = \sum_{i=1}^{t} \phi(i-1)\, \phi(i-1)^T \qquad (3.1.5b)$$

Notice that the principle, that the sum of the squares of the differences between the observed and the computed quantities must be minimum, is independent of a stochastic framework. However, a disturbance or error sequence x(t) influences the properties of the estimate $\hat{\theta}$ [Ljung87]. If the system is described by

$$y(t) = \phi(t\text{-}1)^T \theta_0 + x(t)$$

where θ_0 is the “true” parameter, if x(t) can be described by white noise with variance λ, and if $\phi(t)$ is bounded, then the estimated parameter $\hat{\theta}$ is a strongly consistent estimate of θ_0, which means, that

$$\hat{\theta} \rightarrow \theta_0, \qquad \text{as } N \rightarrow \infty.$$

3.2 Recursive Least Squares Estimator

Equation (3.1.5) essentially solves the least squares estimation problem. However, the result can be obtained only after the data acquisition, if additional data arrives, the estimation procedure has to restart from scratch. Due to this, a recursive least squares algorithm has to be derived which can also be used for on-line estimation, this is essential for the proposed adaptive approach.

In a recursive estimation algorithm the estimated parameter vector $\hat{\theta}$ becomes necessarily time-varying, it is why in the following $\hat{\theta}(t)$ will be used instead.

Starting from equation (3.1.5a) we consider the estimated parameters at the time t+1:

$$\hat{\theta}(t+1) = F(t+1)\sum_{i=1}^{t+1} y(i)\,\phi(i-1), \tag{3.2.1}$$

where

$$F(t+1)^{-1} = \sum_{i=1}^{t+1} \phi(i-1)\phi(i-1)^T = F(t)^{-1} + \phi(t)\,\phi(t)^T \tag{3.2.2}$$

and

$$\sum_{i=1}^{t+1} y(i)\,\phi(i-1) = \sum_{i=1}^{t} y(i)\,\phi(i-1) + y(t+1)\,\phi(t) \tag{3.2.3}$$

Since $\sum_{i=1}^{t} y(i)\,\phi(i-1) = F(t)^{-1}\hat{\theta}(t)$, equation (3.2.1) can be rewritten using equation (3.2.3):

$$\hat{\theta}(t+1) = F(t+1)\left(F(t)^{-1}\hat{\theta}(t) + y(t+1)\phi(t) + \left\{\phi(t)\,\phi(t)^T\hat{\theta}(t) - \phi(t)\,\phi(t)^T\hat{\theta}(t)\right\}\right) \tag{3.2.4}$$

$$\Leftrightarrow \quad F(t+1)^{-1}\hat{\theta}(t+1) = F(t)^{-1}\hat{\theta}(t) + \phi(t)\,\phi(t)^T\hat{\theta}(t) + \phi(t)\left[y(t+1) - \hat{\theta}(t)^T\phi(t)\right]$$

Using equation (3.2.2) this can be written as

$$F(t+1)^{-1}\hat{\theta}(t+1) = F(t+1)^{-1}\hat{\theta}(t) + \phi(t)\left[y(t+1) - \hat{\theta}(t)^T\phi(t)\right]$$

which leads to

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t+1)\,\phi(t)\left[y(t+1) - \hat{\theta}(t)^T\phi(t)\right] \tag{3.2.5}$$

In order to obtain a recursive formula for F(t+1) the following matrix inversion lemma is used:

$$\left(F^{-1} + \phi\,\phi^T\right)^{-1} = F - \frac{F\,\phi\,\phi^T F}{1 + \phi^T F\,\phi}$$

With $F(t+1)^{-1} = F(t)^{-1} + \phi(t)\,\phi(t)^T$ (see equation (3.2.2)), which is the inverse of the left hand part of the matrix inversion lemma, the recursive formula for F(t+1) can be written as

$$F(t+1) = F(t) - \frac{F(t)\,\phi(t)\,\phi(t)^T\,F(t)}{1+\phi(t)^T\,F(t)\,\phi(t)} \qquad (3.2.6)$$

Resetting t := t + 1 and with $F(t+1)\,\phi(t) = \frac{F(t)\,\phi(t)}{1+\phi(t)^T F(t)\,\phi(t)}$ (3.2.7)

equations (3.2.5) and (3.2.6) lead to the first recursive version of the least squares estimator:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{F(t-1)\,\phi(t-1)}{1+\phi(t-1)^T F(t-1)\,\phi(t-1)}\left[y(t) - \hat{\theta}(t-1)^T\,\phi(t-1)\right] \qquad (3.2.8)$$

$$F(t) = F(t-1) - \frac{F(t-1)\,\phi(t-1)\,\phi(t-1)^T\,F(t-1)}{1+\phi(t-1)^T\,F(t-1)\,\phi(t-1)} \qquad (3.2.9)$$

where

$F(0) = F_0 = F_0^T > 0$

3.3 Exponential Data Weighting

For this estimation algorithm, the matrix F(t), which is usually called *adaptation gain matrix*, tends to zero for increasing time, which can be easily shown if only one parameter is to be estimated [Lan88]. In this case F(t) and $\phi(t)$ are scalars and equation (3.2.7) results in

$$F(t+1) = \frac{F(t)}{1+\phi(t)^2 F(t)} \le F(t)$$

This implies firstly that the initial adaptation gain F(0) must be greater than zero in order to enable the estimation procedure to work. Secondly it can be seen that with increasing time, and thus with decreasing values of F(t), changes of the parameter vector $\hat{\theta}(t)$ become smaller and smaller. Therefore a forgetting factor f(t) is introduced which assigns less weight to older measurements that are no longer representative for the system [Ljung87]. The update law for F(t) changes then to

$$F(t) = \frac{1}{f(t)}\left[F(t-1) - \frac{F(t-1)\,\phi(t-1)\,\phi(t-1)\,F(t-1)}{1+\phi(t-1)^T\,F(t-1)\,\phi(t-1)}\right] \qquad (3.3.1)$$

The forgetting factor f(t) can be variable in time, it is initialised as follows [M'Saad94c]:

$$f(t) = f_0\,f(t\text{-}1) + (1 - f_0) \qquad (3.3.2)$$

with

$f(0) \le 1, \qquad f_0 < 1$

f_0 and f(0) denote the time constant and the initial value of the forgetting factor sequence. The identification is carried out using an iterative procedure in order to select f_0 and f(0) as well as the initial adaptation gain F(0) so that the initial values of $\hat{\theta}(t=0)$ are rapidly adapted to the true values, but that later measurement values can still influence the estimation result. Appropriate values for F(0) for the case when no initial knowledge about the estimated parameters is available ($\hat{\theta}(0)=\underline{0}$) are large, e.g. 1000, if good initial values for $\hat{\theta}(t)$ are available (e.g. as a result of a previous identification procedure), values of $F(0) \leq 1$ are generally chosen [Lan88]. For the correspondence between measured values and the output of the estimated model, the prediction error $\varepsilon(t)$ is an adequate measure.

3.4 Exponential Data Weighting for Time-varying Systems

For time-varying systems, e.g. in adaptive control, further modifications of the estimation algorithm are needed. This is due to the fact, that in equation (3.3.2) the final value of the forgetting factor sequence $f(t \rightarrow \infty)$ becomes 1 and therefore $F(t \rightarrow \infty) = 0$. This observation motivates two possible modifications of the estimation algorithm. The first possibility is to modify the forgetting factor sequence in that way, that the final value $f(t \rightarrow \infty) < 1$. This leads to the following forgetting factor sequence:

$$f(t) = f_0\, f(t-1) + (1 - f_0)\, f_t \qquad (3.4.1)$$

with

$$f(0) \leq 1, \qquad f_0 < 1, \qquad f_t \leq 1$$

f_t is the final value of the forgetting factor.

The second possible modification is to prevent the adaptation gain matrix F(t) from decreasing to zero. This approach is called *adaptation gain regularization* and is explained in the following chapter.

3.5 Numerical Robustness Considerations and Adaptation Gain Regularization

An advanced estimation algorithm does not update the adaptation gain matrix F(t) as made in equation (3.3.1). F(t) is decomposed into a product of matrices which are updated instead. This improves not only the numerical robustness of the algorithm, but it also allows to implement the already mentioned adaptation gain regularization more easily.

Numerical problems can occur if the changes of the input signal are relatively small, which is usually the case in adaptive control, or if the estimation algorithm is implemented using digital microprocessors with a relatively short word length. Both cases result in badly conditioned matrix equations. In order to work with better-conditioned matrices, it is useful to decompose the adaptation gain matrix F(t) using the so-called UD-factorization. This UD-factorization is generally used as the best form for numerical computations of the adaptation gain matrix [Iser92, Ljung87, M'Saad94c].

The guiding principle is to update not the adaptation gain matrix F(t) which contains the products and covariances of signal values, but to decompose F(t) into a product

of matrices. These matrices contain elements of the same magnitude as the original signal values and are updated instead of F(t).

F(t) can be decomposed like

$$F(t) = U(t)\, D(t)\, U(t)^T$$

with

$$U(t) = \begin{pmatrix} 1 & x & \cdots & x \\ 0 & 1 & x & \vdots \\ \vdots & 0 & \ddots & x \\ 0 & \cdots & 0 & 1 \end{pmatrix}$$

and

$$D(t) = \begin{pmatrix} d_1(t) & 0 & \cdots & 0 \\ 0 & d_2(t) & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & d_i(t) \end{pmatrix}, \qquad i = 1 \ldots n+m+1$$

This factorisation improves not only the numerical robustness, it also allows to easily introduce *regularization*, that is, measures to ensure that the eigenvalues $\lambda_i(t)$ of F(t) stay bounded, at the same time as F(t) remains positive definite [Ljung87]. Indeed a small quotient $\frac{\lambda_{max}}{\lambda_{min}}$ can be considered as quantifier for a good conditioned matrix equation [Iser92].

Without proof, the following statement is true:

$$\det(F(t)) = \prod_{i=1}^{n+m+1} d_i(t) = \prod_{i=1}^{n+m+1} \lambda_i(t)$$

Therefore a lower bound for the eigenvalues $\lambda_i(t)$ of F(t) and thus for the adaptation gain F(t) can be achieved by maintaining the elements of the diagonal matrix D(t) beyond a specified threshold d_0:

$$d_0(t) = d_0 \qquad \text{if } d_0(t) \le d_0$$

This threshold d_0 of the diagonal elements of D(t) is also called the *lower bound* of the adaptation gain regularization [M'Saad94c, M'Saad94]. This introduced lower bound for F(t) can be implicitly expressed by a matrix R(t) in the following modified update law:

$$F(t) = \frac{1}{f(t)}\left[F(t-1) - \frac{F(t-1)\,\phi(t-1)\,\phi(t-1)\,F(t-1)}{1+\phi(t-1)^T F(t-1)\,\phi(t-1)}\right] + R(t) \qquad (3.5.1)$$

Now, the problem of a decreasing adaptation gain matrix F(t) is solved by using an appropriate forgetting factor sequence f(t) and a lower bound of the adaptation gain matrix. One remaining problem with the update algorithm of F(t) in equation (3.5.1)

is, that under low excitation conditions F(t) can grow without bound. In order to ensure that excessively large values are avoided, the trace of F(t) (which is defined as the sum of the diagonal elements) is hold constant whenever it exceeds some threshold. Thus a upper bound λ_{max} is ensured.

More particularly this is implemented as follows:

After the identification process is started, the forgetting factor sequence is calculated as proposed in equation (3.4.1):

$$f(t) = f_0\, f(t-1) + (1 - f_0)\, f_t$$

By selecting appropriate values for f_0, f(0) and F(0), bad initial conditions vanish rapidly. But after this initialisation time, as soon as the relatively large value of F(t) (or more precisely: as soon as the trace of F(t)) has dropped below a specified *trace threshold*, the forgetting factor sequence f(t) is selected so that the trace of F(t) is kept constant:

$$\text{trace}[F(t)] = \text{trace}[F(t-1)]$$

This is achieved by the following, new update law for f(t):

$$f(t) = 1 - \frac{\text{trace}\left[F(t)\,\phi(t)\,\phi(t)^T F(t)\right]}{\left(\eta(t) + \phi(t)^T F(t)\,\phi(t)\right)\ \text{trace}[F(t)]}$$

$\eta(t)$ is a normalisation factor which is explained in the next paragraph.

The shape of the adaptation gain F(t), which can be influenced by f_0, f(0) and F(0) during the initialisation, and later by f_t and the trace threshold, is a trade-off between tracking ability and noise sensitivity. A high adaptation gain means that the algorithm is alert in tracking parameter changes, but at the same time sensitive to disturbances in the data, since the latter are falsely interpreted as signs of parameter changes.

3.6 Normalisation

One mechanism proposed to deal with underinformative data at high signal values is *normalisation* [Bit90, Pra90]. The key idea is to replace the error signal and the regressor in RLS by derived quantities which are guaranteed bounded by dividing by a normalisation signal constructed from past inputs and outputs.

The *normalisation factor* $\eta(t)$ is initialised as follows [M'Saad94c]:

$$\eta(t) = \sigma\, \eta(t-1) + (1 - \sigma)\, \text{Max}([\phi(t)^T \phi(t)], \eta_0), \qquad 0 \le \sigma < 1; \quad 0 < \eta_0; \quad \eta(0) < \infty$$

with the *normalisation dynamics parameter* σ and the *normalisation threshold* η_0.

The regressor becomes then

$$\bar{\phi}(t) = \frac{\phi(t)}{\sqrt{\eta(t)}},$$

and the system output

$$\bar{y}(t) = \frac{y(t)}{\sqrt{\eta(t)}}.$$

Notice that the above normalisation factor is the norm of the input-output data passed through a first order filter $\frac{1}{1-\sigma z^{-1}}$. The normalisation dynamics parameter σ should be chosen so that the unmodelled dynamics modes are outside the normalisation filter bandwidth specified via the polynomial $1 - \sigma z^{-1}$. For instance one can choose σ as follows:

$$P(q^{-1}) = 0 \qquad \text{in } |q| < \sigma$$

where $P(q^{-1})$ is the characteristic polynomial of the closed-loop. This means, σ should be chosen like the largest distance between closed-loop poles and origin.

A reasonable default value for the normalisation threshold η_0 seems to be 1, since then the normalisation only influences the adaptation for high signal values, otherwise $\eta(t)$ is 1.

The equations for the recursive least squares algorithm in equation (3.2.8) and (3.5.1) change as follows:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{F(t-1)\phi(t-1)}{\eta(t) + \phi(t-1)^T F(t-1)\phi(t-1)}\left[y(t) - \hat{\theta}(t-1)^T\phi(t-1)\right] \qquad (3.6.1)$$

$$F(t) = \frac{1}{f(t)}\left[F(t-1) - \frac{F(t-1)\phi(t-1)\phi(t-1)F(t-1)}{\eta(t) + \phi(t-1)^T F(t-1)\phi(t-1)}\right] + R(t) \qquad (3.6.2)$$

3.7 Adaptation Freezing

Another algorithm modification to cope with similar problems is the adaptation freezing. This allows to reject those errors which are too small. Such errors could arise due to normalisation with very large signals or, equally, if the system is not persistently excited [Bit90 , Cla85].

A system is said to be persistently excited if

$$\alpha I \le \sum_{i=k}^{k+m+n-1} \phi(i)\phi(i)^T \le \beta I \text{ , for all } k \qquad (3.7.1)$$

If the matrix $\sum_{i=k}^{k+m+n-1} \phi(i)\phi(i)^T$ should remain above a lower bound αI, then the regressor $\phi(t)$ must have a certain character which insures that this matrix will have only positive eigenvalues. This idea was already explained in chapter 3.5, where the elements $d_i(t)$ of the diagonal matrix D(t) and thus the eigenvalues $\lambda_i(t)$ of F(t) were maintained above a specified threshold d_0.

Therefore the normalised minimum eigenvalue λ_{min} of $\sum_{i=k}^{k+m+n-1} \phi(i)\phi(i)^T$ is a quantifier for a persistently excited system:

$$\zeta(t) = \frac{\min \lambda\left[\sum_{i=k}^{k+m+n} \phi(i-1)\,\phi(i-1)^T\right]}{\sum_{i=k}^{k+m+n} \phi(i-1)^T\,\phi(i-1)} \qquad (3.7.2)$$

This method has been approximated by a faster technique: instead of equation (3.7.2) the following *information measure* $\sigma(t)$ is used for the incoming and past information likelihood:

$$\sigma(t) = \frac{\phi(t-1)^T F(t-1)\,\phi(t-1)}{\eta(t)}$$

The calculation requirements are almost negligible, it is basically a by-product of the estimation algorithm. This quantifier has the unique property, that it measures just the new information content in the signal. Not the real excitation is measured, $\sigma(t)$ measures how much the information in the recent regressors differs from that in the past regressors. This explains why $\sigma(t)$ even diminishes when the system is excited persistently, but with a period sequence: after a while the new data looks like the old data and there is no new information which can improve the estimation. Therefore the estimator may be turned off. [Cla85].

An introduced switching sequence s(t) allows to freeze the parameter adaptation when the available information is not likely to improve the parameter estimation process. More precisely, when the information measure $\sigma(t)$ is smaller than a certain pre-specified *information threshold* σ_0 over a given observation horizon, the incoming information will no longer improve the estimation process. This is illustrated in the following:

$$s(t) = \begin{cases} 0 \quad \text{if } \{s(t-1)=1\} \quad \text{and} \quad \left\{\dfrac{\phi(t-1)^T F(t-1)\,\phi(t-1)}{\eta(t)} < \sigma_0 \quad \text{during} \quad t = t-1,\cdots,t-\text{bfa}\right\} \\ 1 \quad \text{if } \{s(t-1)=0\} \quad \text{and} \quad \left\{\dfrac{\phi(t-1)^T F(t-1)\,\phi(t-1)}{\eta(t)} > \sigma_0 \quad \text{during} \quad t = t-1,\cdots,t-\text{bra}\right\} \\ s(t-1) \quad \text{else} \end{cases}$$

where bfa and bra are pre-specified checking time intervals, which entail a hysteresis characteristic of the switching sequence [M'Saad94c].

Now the final identification algorithm has been obtained:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + s(t)\frac{F(t-1)\Phi(t-1)\left(y(t) - \Phi(t-1)^T\hat{\theta}(t-1)\right)}{\eta(t) + \Phi(t-1)^T F(t-1)\Phi(t-1)}$$

$$F(t) = \frac{1}{f(t)}\left[F(t-1) + s(t)\frac{F(t-1)\Phi(t-1)\Phi(t-1)^T F(t-1)}{\eta(t) + \Phi(t-1)^T F(t-1)\Phi(t-1)}\right] + s(t)R(t)$$

3.8 Incorporation of Prior Knowledge

The estimation result can be improved if any prior knowledge concerning the model structure or the estimated parameters is incorporated into the identification method. This leads to superior rates of convergence for the prediction errors and parameters [Mid90].

If the model is described by

$$y(t) = \phi(t\text{-}1)^T \theta$$

with

$$\hat{\theta} = [\theta_1, \theta_2, \cdots, \theta_m, \theta_{m+1}, \cdots, \theta_n]$$

and if θ_1, θ_2, ..., θ_m are known, then we can define

$$\bar{y}(t) = y(t) - \sum_{i=1}^{m} \phi_i(i-1)\theta_i$$

The estimation is now based on

$$\bar{y}(t) = \bar{\phi}(t-1)^T \bar{\theta}$$

with

$$\bar{\phi} = [\phi_{m+1}, \ldots, \phi_n]^T$$

and

$$\bar{\theta} = [\theta_{m+1}, \ldots, \theta_n]^T$$

where the dimensionality of the problem has been reduced by m. A typical situation is the presence of one or several integrators in the plant model.

4 Experiment Design and Off-line Identification Results

Since for the presented control design a simple linear model of the vehicle is needed, the complex, non-linear vehicle model, which was explained in chapter 2, has been identified over the operating domain. This identification procedure provides an insight into the possible variations of the vehicle dynamics, with a set of identified models, one model for each operating condition. Finally one of these identified models (the *nominal model*) has been selected to design the robust controller.

In the following chapter the practical aspects of the identification procedure, considered for the identification of the vehicle model, are discussed. This includes the selection of an appropriate input signal, the determination of the sampling period, and the choice of the model structure. Various validation procedures are explained and identification results are presented.

4.1 Identification of Vehicle Dynamics

The purpose of the steering controller is to manipulate the steering angle so that the sensor displacement, or lateral displacement from the guideline, remains small. Therefore a transfer function is needed which relates the sensor displacement y_S to the steering angle δ. This steering angle is used as input data for the identification. Figure 4.1 shows the variables of interest.

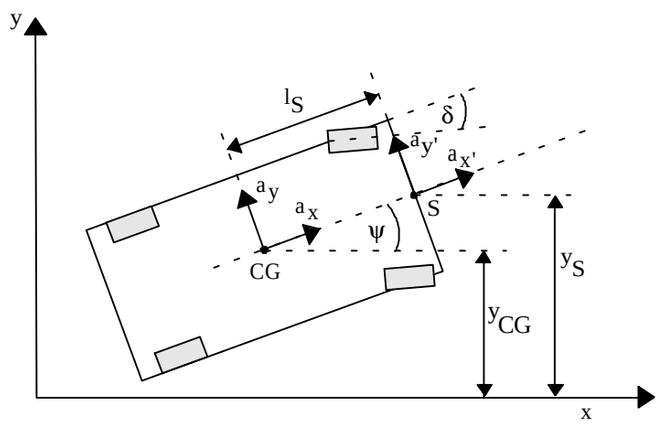


Figure 4.1: Vehicle and pertinent variables

Measurable output signals which are commonly used for identification of vehicle dynamics are the lateral acceleration and the yaw rate. In this approach the lateral acceleration of the vehicle's front end $a'_y(t)$ is used as output data for the identification. This value represents an easily measurable signal since it can be measured by a simple acceleration sensor, furthermore, it is directly related to the second derivative of the sensor displacement $\ddot{y}_S(t)$: if the yaw angle $\psi(t)$ remains small during test measurements, i.e., if the vehicle follows approximately the straight path, then the lateral acceleration at the front end $a'_y(t)$ is approximately equal to $\ddot{y}_S(t)$.

Therefore the identification procedure provides a transfer function which relates the second derivative of the sensor displacement $\ddot{y}_S(t)$ to the steering angle $\delta(t)$. Multiplying this transfer function by a double integrator gives the transfer function which can then be used as control model.

The second derivative of the sensor displacement $\ddot{y}_S(t)$ has been obtained in the vehicle simulator using the lateral acceleration of the vehicle $a_y(t)$ and the second derivative of the yaw angle $\ddot{\psi}(t)$, as follows:

$$\ddot{y}_S(t) = a'_y(t)\cos(\psi(t)) = \left(a_y(t) + l_s\ddot{\psi}(t)\right)\cos(\psi(t)) \approx a_y(t) + l_S\ddot{\psi}(t) \qquad (4.1.1)$$

where l_S denotes the distance between the centre of gravity and the sensor position.

Equation (4.1.1) is valid for a straight path which has been considered for all identification experiments. The assumed approximation holds if the vehicle follows approximately a straight path, which is fulfilled since a *offset-free* binary random sequence is used.

Figure 4.2 shows for one particular driving condition the driven path. This driving condition corresponds to a velocity of 30m/s on a dry road with an average mass and will further be called *nominal* condition. Figure 4.2 explains, why the second derivative of the sensor displacement $\ddot{y}_S(t)$ is used as identified value instead of the sensor displacement $y_S(t)$: if the proposed steering system is implemented into a test vehicle, it is very difficult to measure sensor displacements of more than approximately 20 cm.

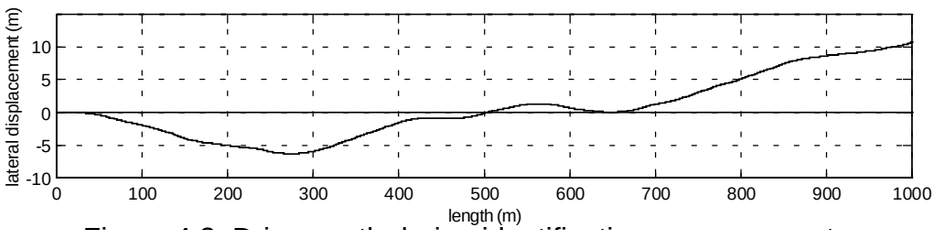


Figure 4.2: Driven path during identification measurements

Figure 4.3 shows the steering angle $\delta(t)$ at the wheels as the input signal, the identified value $a'_y(t)$ and the yaw angle $\psi(t)$. The latter justifies that the approximation, assumed in equation (4.1.1), is satisfied: the values remain always smaller than 5 degrees.

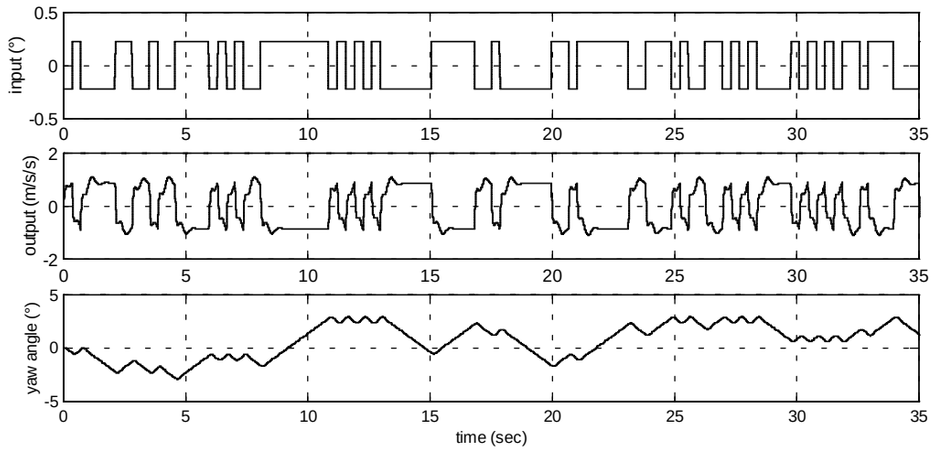


Figure 4.3: Identification input and output signals and yaw angle

For the proposed system, the sensor requirements for identification purpose is a single accelerometer to measure the lateral acceleration $a_y(t)$ at the front end of the vehicle.

4.2 Input Signal Shape

A variety of input signals is possible for estimation purpose, e.g. impulse, step, sine, etc., however pseudorandom binary signals (PRBS) are particularly useful due to the highest power spectral density for limited signal amplitudes [Kro91, Iser92].

For PRBS, the following guidelines have to be considered:

- The steeper the slopes of the input signal, the more high frequencies are induced.
- The level of the binary input sequence should be as large as possible, without exceeding the linear region of the plant.

The levels of the offset-free binary random sequences, used to identify the vehicle dynamics, are determined so, that the maximum values of the lateral acceleration $a_{y,\,max}$ do not exceed approximately $2\,\frac{m}{s^2}$ on a dry road. For this acceleration value the vehicle dynamics are assumed to be describable by linear models [Mit90]. For wet conditions and on snow roads the levels of the input sequences are reduced accordingly.

- The longer the minimal time interval between two slopes, the more low frequencies are induced and the better is the estimation of the static gain of the linear model.

The focus in the identification is on the frequency region below approximately 5 Hz and on the static value of the estimated transfer function. This is due to the fact that the steering commands of an average driver contain frequencies between 0 and 1.5 Hz [Mit90], assuming that the actuator capabilities exceed these of an average driver. The minimum time interval between two signal slopes are chosen accordingly.

Figure 4.4 shows the lateral acceleration of the vehicle $a_y(t)$, again for the nominal condition, i.e. a velocity of 30 m/s on a dry road with an average mass. The corresponding steering angle at the wheels has already been shown in figure 4.3.

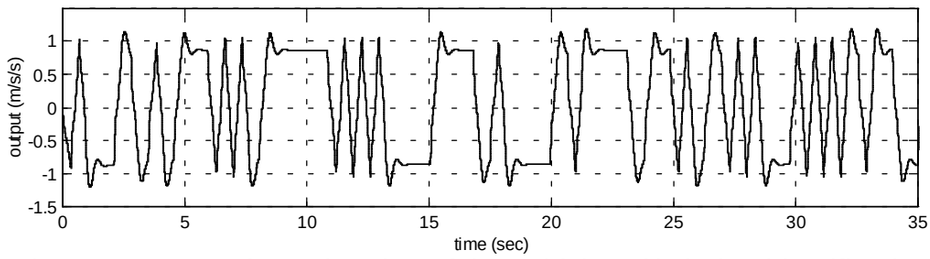


Figure 4.4: Lateral acceleration of the vehicle $a_y(t)$ during identification measurements

4.3 Selection of Sampling Period and Filter Frequencies

The selection of the sampling period is of fundamental importance for implementation purpose since most digital controllers only work well if the sampling period is chosen appropriately [Cla87, Ljung87, Iser92, Kro91]. Various aspects have to be kept in mind. In case of very large sampling periods the precision of the obtained model will decrease and the ability to describe the dynamic behaviour of the plant will be poor. Unnecessarily high sampling on the other hand makes the implementation complex and expensive since the hardware requirements will increase, in addition the variance of the data set and the sensibility to measurement noise will increase; moreover numerical problems occur since the model poles are shifted close to 1. Thus the selection of the sampling period will be a trade-off between noise reduction and relevance for the dynamics.

Since the obtained model will be used for control purpose, the sampling period for which the estimated model is built, should be the same as for the control application. In case of the automatic steering problem a high sampled system will detect an entered curve earlier. A very fast sampled model on the other hand will often be non-minimum in phase [Ljung87, Åst84] and a system with dead time has to be modelled with delay of many sampling periods which may cause problems for the control design.

The following guidelines are given [Åst84, Cla87, Kro91, Lan88, Ljung87]:

- The sampling frequency should be chosen in accordance with the desired bandwidth of the closed loop. Values between 6 and 25 times the bandwidth of the closed loop are appropriate, a sampling frequency that is about ten times the bandwidth of the system should be a good choice in most cases.
- If the settling time of the plant is considered, appropriate values for the sampling period are approximately 1/5 to 1/15 of the settling-time of the plant.
- Aström and Wittenmark [Åst84] consider the settling time of the closed-loop system and recommend a sampling period which is 10-20 times smaller than the settling time of the closed-loop system.

The desired bandwidth of the closed loop is chosen to 5 Hz (see chapter 4.1), the settling time of the closed-loop system (consider figure 6.6) is between 3 sec (30 m/s on dry road) and 15 sec (5 m/s on dry road), the settling time of the plant is between 0.5 sec and 1.5 sec (see figure 4.4). Therefore a sampling period of 10 ms seems to be high, but reasonable, it was also used in previous studies [Gul94, Peng90]. Notice that in the trade-off between noise reduction and relevance of the dynamics, the latter was emphasised in order to obtain a good tracking performance.

Data filtering is of fundamental importance for system identification. The purpose is to eliminate high and low frequency disturbances. Due to the sampling of the signals an analogue low pass pre-sampling filter is required to avoid aliasing effects [Iser92]. The cut-off frequency is set to values smaller than the Nyquist frequency.

Of special importance are low frequency disturbances, e.g. drift signals and offsets, possibly of periodic character, since they can considerably deteriorate the estimation results. The approach, which gives in general the best results in eliminating these disturbances, consists in filtering the very low frequencies using a digital high pass filter. Notice that the cut-off frequency should be below the smallest frequency induced by the binary random sequence used to excite the system. Notice also, that the plant input and the plant output signals have to be filtered in the same way before they are used in the estimator, in order not to falsify the transfer function to be estimated.

For the identification of the vehicle dynamics a digital band pass filter is used, the low and high cut-off frequencies are chosen to f_{low} = 0.0005 Hz and f_{high} = 25 Hz.

4.4 Selection of Model Structure

After the measured signals, as data for the identification algorithm, have been obtained, the next step is to select the model structure and model order. The choice of an appropriate model structure is most crucial for a successful identification application. This choice must be based both on insights and knowledge about the system to be identified, on an understanding of the identification procedure, and on the collected data set [Ljung87].

4.4.1 Physical Insights

Physical insight will often tell which range of model order and time delay should be considered, nevertheless, it will be an iterative procedure to answer the question ‘which is the simplest model which represents the behaviour of the plant good enough?’.

In order to get a general knowledge about the complexity of the physical model, the physical transfer function between sensor displacement and steering wheel using the classical single-track model is derived.

As already shown, the second derivative of the sensor displacement can be calculated as follows:

$$\ddot{y}_S(t) \approx a_y(t) + l_S\ddot{\psi}(t)$$

The pulse transfer function which relates $\ddot{y}_S(t)$ to the steering angle δ is therefore

$$\frac{\ddot{y}_S(s)}{\delta(s)} \approx \frac{a_y(s)}{\delta(s)} + l_S \cdot \frac{\ddot{\psi}(s)}{\delta(s)} = \frac{a_y(s)}{\delta(s)} + l_S \cdot s \cdot \frac{\dot{\psi}(s)}{\delta(s)}$$

The transfer functions $\frac{a_y(s)}{\delta(s)}$ and $\frac{\dot{\psi}(s)}{\delta(s)}$ are derived in [Mit90] using the classical single-track model:

$$\frac{a_y(s)}{\delta(s)} = \left(\frac{a_y}{\delta}\right)_{Stat} \frac{1 + T_{z1}s + T_{z2}s^2}{1 + \frac{2\sigma_f}{\nu^2}s + \frac{1}{\nu^2}s^2} = V_y \frac{1 + a_1 s + a_2 s^2}{1 + b_1 s + b_2 s^2} \qquad (4.4.1)$$

with

$T_{z1} = l_H / v$,

$T_{z2} = J_z / c_{\alpha H}\, l$, and

$$\left(\frac{a_y}{\delta}\right)_{Stat} = \frac{1}{l} \frac{v^2}{1 + \left(\frac{v}{v_{Ch\delta}}\right)^2}.$$

and

$$\frac{\dot{\psi}(s)}{\delta(s)} = \left(\frac{\dot{\psi}}{\delta}\right)_{Stat} \frac{1 + T_{z3}s}{1 + \frac{2\sigma_f}{\nu^2}s + \frac{1}{\nu^2}s^2} = V_\psi \frac{1 + c_1 s}{1 + b_1 s + b_2 s^2} \qquad (4.4.2)$$

with

$T_{z3} = (m\, v\, l_v) / (c_{\alpha H}\, l)$, and

$$\left(\frac{\dot{\psi}}{\delta}\right)_{Stat} = \frac{1}{l} \frac{v}{1 + \left(\frac{v}{v_{Ch\delta}}\right)^2}$$

l_H, l_V, and l denote the distance between centre of gravity and rear axle, between centre of gravity and front axle, and between front and rear axle, respectively. J_z, v, and $v_{Ch\delta}$ represent the moment of inertia of the vehicle around the vertical axis, the velocity of the vehicle, and the characteristic velocity of the vehicle. The latter is a function of the cornering stiffness of the front and the rear wheels $c_{\alpha H}$ and $c_{\alpha V}$, of l_H, l_V, and l, and of the vehicle mass m:

$$v_{Ch\delta} = \sqrt{\frac{c_{\alpha H} c_{\alpha V} l^2}{m \cdot \left| c_{\alpha H} l_H - c_{\alpha V} l_V \right|}}$$

The decay factor σ_f and the undamped natural angular frequency ν of the characteristic equation of the vehicle are defined as

$$\sigma_f = \frac{m(c_{\alpha V} l_v^2 + c_{\alpha H} l_H^2) + J_z (c_{\alpha V} + c_{\alpha H})}{2 J_z m v}$$

The pulse transfer function which relates $\ddot{y}_S(t)$ to the steering angle δ is therefore

$$\frac{\ddot{y}_S(s)}{\delta(s)} \approx \frac{a_y(s)}{\delta(s)} + l_S \cdot \frac{\ddot{\psi}(s)}{\delta(s)} = \frac{a_y(s)}{\delta(s)} + l_S \cdot s \cdot \frac{\dot{\psi}(s)}{\delta(s)}$$

The transfer functions $\frac{a_y(s)}{\delta(s)}$ and $\frac{\dot{\psi}(s)}{\delta(s)}$ are derived in [Mit90] using the classical single-track model:

$$\frac{a_y(s)}{\delta(s)} = \left(\frac{a_y}{\delta}\right)_{Stat} \frac{1 + T_{z1}s + T_{z2}s^2}{1 + \frac{2\sigma_f}{\nu^2}s + \frac{1}{\nu^2}s^2} = V_y \frac{1 + a_1 s + a_2 s^2}{1 + b_1 s + b_2 s^2} \qquad (4.4.1)$$

with

$T_{z1} = l_H / v$,

$T_{z2} = J_z / c_{\alpha H}\, l$, and

$$\left(\frac{a_y}{\delta}\right)_{Stat} = \frac{1}{l} \frac{v^2}{1 + \left(\frac{v}{v_{Ch\delta}}\right)^2}.$$

and

$$\frac{\dot{\psi}(s)}{\delta(s)} = \left(\frac{\dot{\psi}}{\delta}\right)_{Stat} \frac{1 + T_{z3}s}{1 + \frac{2\sigma_f}{\nu^2}s + \frac{1}{\nu^2}s^2} = V_\psi \frac{1 + c_1 s}{1 + b_1 s + b_2 s^2} \qquad (4.4.2)$$

with

$T_{z3} = (m\, v\, l_v) / (c_{\alpha H}\, l)$, and

$$\left(\frac{\dot{\psi}}{\delta}\right)_{Stat} = \frac{1}{l} \frac{v}{1 + \left(\frac{v}{v_{Ch\delta}}\right)^2}$$

l_H, l_V, and l denote the distance between centre of gravity and rear axle, between centre of gravity and front axle, and between front and rear axle, respectively. J_z, v, and $v_{Ch\delta}$ represent the moment of inertia of the vehicle around the vertical axis, the velocity of the vehicle, and the characteristic velocity of the vehicle. The latter is a function of the cornering stiffness of the front and the rear wheels $c_{\alpha H}$ and $c_{\alpha V}$, of l_H, l_V, and l, and of the vehicle mass m:

$$v_{Ch\delta} = \sqrt{\frac{c_{\alpha H} c_{\alpha V} l^2}{m \cdot \left| c_{\alpha H} l_H - c_{\alpha V} l_V \right|}}$$

The decay factor σ_f and the undamped natural angular frequency ν of the characteristic equation of the vehicle are defined as

$$\sigma_f = \frac{m(c_{\alpha V} l_v^2 + c_{\alpha H} l_H^2) + J_z (c_{\alpha V} + c_{\alpha H})}{2 J_z m v}$$

and $\nu_f^2 = \dfrac{c_{\alpha V} c_{\alpha H} l^2 + mv^2 (c_{\alpha H} l_H - c_{\alpha V} l_V)}{J_z m v^2}$

Of particular importance is that the combination of equation (4.4.1) and equation (4.4.2) leads to a model of second order, as shown below:

$$\frac{\ddot{y}_S(s)}{\delta(s)} = \frac{a_y(s)}{\delta(s)} + l_S s \frac{\psi(s)}{\delta(s)} = \frac{V_y + (V_y a_1 + l_S V_\psi)s + (V_y a_2 + l_S V_\psi c_1)s^2}{1 + b_1 s + b_2 s^2} = V_y \frac{1 + \alpha_1 s + \alpha_2 s^2}{1 + b_1 s + b_2 s^2}$$

Furthermore, by considering the parameters of equations (4.4.1) and (4.4.2) it can already be assumed, that the variation of the velocity will have the major impact on the vehicle dynamics.

4.4.2 Preliminary Data Analysis

Besides the physical insights in the plant, an analysis of the experimental frequency response can give valuable information about the resonance peaks and the high-frequency roll-off and phase shift. This gives additional hints as to what model orders will be required to give an adequate description of the dynamics of the plant.

The experimental frequency response of the vehicle is determined using an appropriate FFT procedure, i.e. $G(j\omega) = \dfrac{\text{FFT}\{y(t)\}}{\text{FFT}\{u(t)\}}$. In figure 4.5 the experimental frequency responses for various vehicle speeds and an average mass on a dry road are shown.

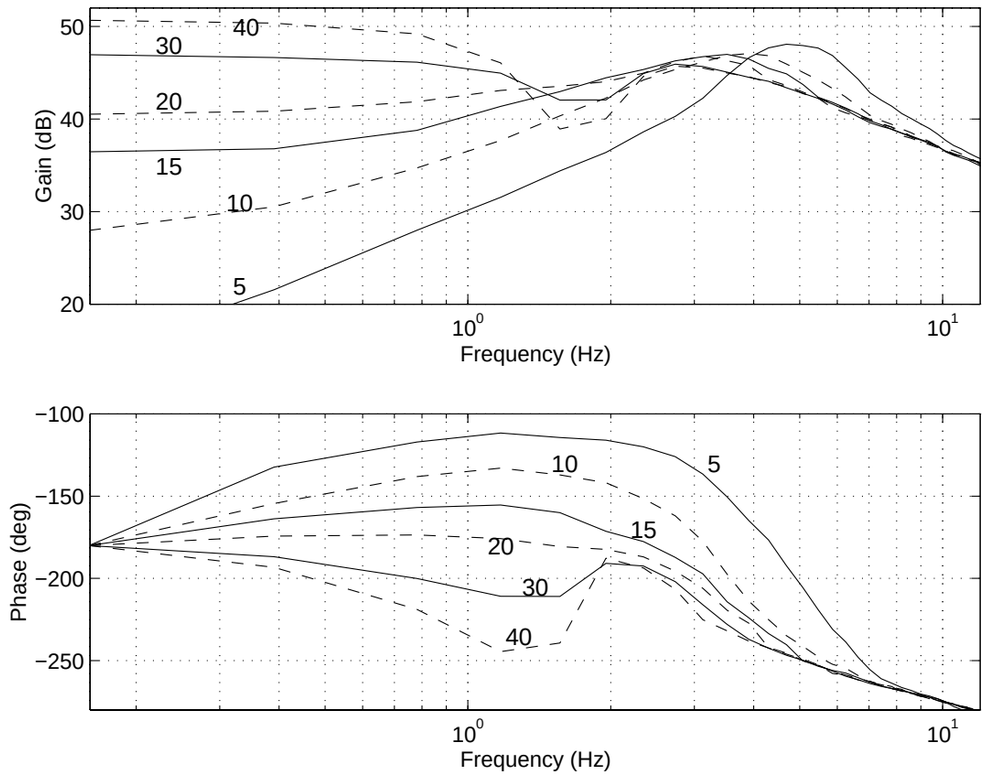


Figure 4.5: Experimental frequency responses for various vehicle speeds

The roll-off of -20 dB indicates a model of at least 2nd order, however it is obvious, that due to an additional low-frequency mode which occurs for velocities above 20 m/s, models of at least 3rd order will be required to describe the dynamics of the vehicle for these higher speeds.

4.4.3 Physical Interpretation

In this chapter, a physical interpretation for the appearance of an additional mode above a certain velocity, which was observed in the previous chapter, is derived. This interpretation is equivalent to the question, why the single-track model, which gives a transfer function of 2nd order, cannot describe the dynamics of the test vehicle satisfactorily. Therefore the simplifications, assumed for the single-track model, are recalled in the following.

- The centre of gravity is assumed to be on road level. This assumption neglects an influence of a centrifugal force on the wheel loads between inner and outer wheels during cornering.
- The roll motion of the vehicle is neglected. Doing so, and with the first assumption, the two wheels of one axle can be considered as one wheel (*single-track* model).
- The wheel characteristics are linearized, i.e. the side force of a wheel is a product of the cornering stiffness and the wheel slip angle.
- Longitudinal wheel forces are neglected.
- The velocity is assumed to be constant.
- The influence of aerodynamic side forces is neglected.

The latter assumption usually holds for velocities below 30 m/s. At higher speeds, even without side wind, aerodynamic side forces have to be considered. Due to the vehicle slip angle, air drag occurs not only on the front, but also on the side face of the vehicle which changes the characteristic equation of the vehicle $1+\frac{2\sigma_f}{\nu^2}s+\frac{1}{\nu^2}s^2$, i.e. the decay factor σ_f and the undamped natural angular frequency ν.

As shown in chapter 4.4.2 at least some of the mentioned simplifications don’t hold for velocities above 20 m/s for the vehicle simulation model of the considered BMW520i.

4.5 Model Validation

Once a model structure has been chosen, the identification procedure provides a particular model in this structure. This model may be the best available one, but the crucial question is whether it is good enough for the intended purpose. Testing if a given model is appropriate is known as *model validation*. It is worth to point out, that the selection of the model structure and the model validation are carried out in parallel and iteratively. Once the selected model has been validated, one investigates if a model of reduced order can also meet the given requirements.

Various validation techniques, in the time and frequency domain, will be described in this chapter.

4.5.1 Prediction Error Consideration

Before the identification procedure is started, besides the model structure, the design parameters for the estimation algorithm have to be selected. As described in chapter 3, in case of an off-line identification these parameters are basically the time constant f_0 and the initial value f(0) of the forgetting factor sequence, and the initial adaptation gain F(0).

Since for the least squares algorithm the quadratic criterion

$$J(\hat{\theta}, t) = \sum_{i=1}^{t} [\varepsilon(t)]^2$$

is to be minimised (see equation (3.1.4)), this cost function can be considered in order to select the best estimation parameters f_0, f(0), and F(0). This will be an iterative procedure, adapting the initial values of $\hat{\theta}(t = 0)$ rapidly to the true values, without neglecting the influence of later measurement values on the estimation result (see chapter 3.3).

Notice, that this procedure is carried out in order to extract the best available model values for the selected model structure. The question, if this model structure and the obtained model values can satisfy the expected correspondence with the true plant will be examined in the following paragraphs.

4.5.2 Spectrum Analysis

In chapter 4.4.2 an experimental frequency response was considered for preliminary data analysis. Such a frequency response can now be used as a reference in order to validate the identified model in the frequency domain. A good correspondence between the identified model and the real plant will entail a good correspondence between the frequency response of the identified model and the experimental frequency response. The validation result for a second and a third order model for the nominal condition is shown in figure 4.6.

Notice that the crucial question is to select the simplest model which is good enough. If the model order can be reduced without affecting the input-output properties very much, then the original model was unnecessarily complex [Ljung87]. However, for the nominal condition a reduction of the model order from 3rd to 2nd order is not acceptable as it can be seen in figure 4.6.

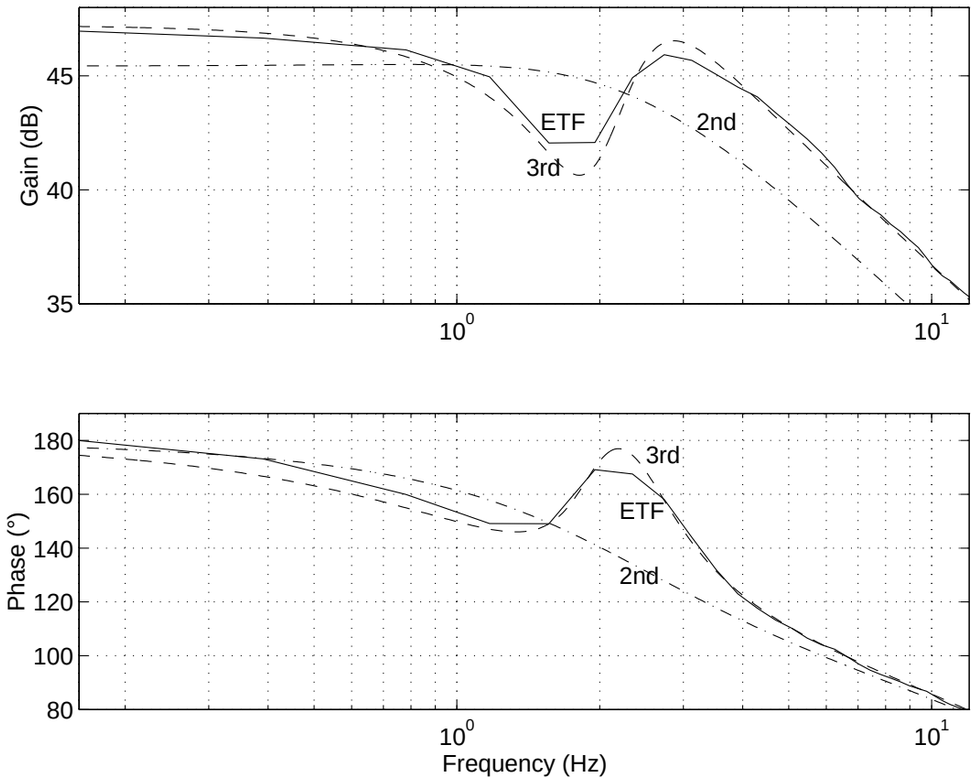


Figure 4.6: Validation result in the frequency domain for a 2nd and a 3rd order model for the nominal condition

4.5.3 Pole-Zero Cancellations

An identified model which is of higher order than the plant will contain additional poles and zeros which can be cancelled. This property can be used to determine the model order. Figure 4.7 shows the poles (crosses) and zeros (circles) of a 3rd order model which corresponds to a condition at a velocity of 15 m/s on a wet road with an average mass. Considering figure 4.5 it can be assumed, that for this velocity a 2nd order model should be sufficient. This expectation is confirmed which can be seen in the pole-zero cancellation in figure 4.7. Notice also the badly damped model poles, which complicate the controller design. In addition to these badly damped poles, two poles on the unit circle are later added, in order to obtain the control model (see chapter 4.1).

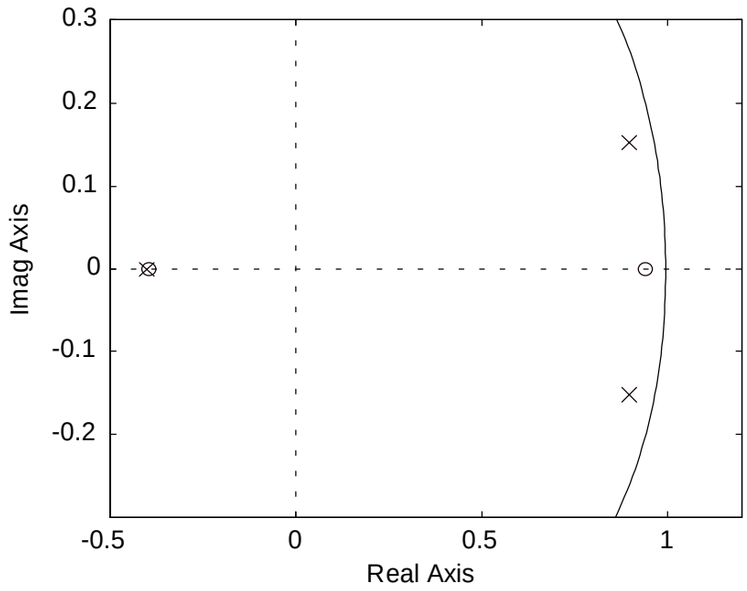


Figure 4.7: Poles and zeros of an overestimated model of 3rd order for a condition of 15 m/s

4.5.4 Simulation

An obvious validation method is to compare the output signal of the estimated model with the measured output of the plant in the time domain. Figure 4.8 shows the validation result for the nominal condition and a 3rd order model. The PRBS input sequence and the plant output for this identification experiment were already shown in figure 4.3, now this output signal is compared with the output of the estimated model. This validation method becomes particularly interesting if the output of the estimated model is compared with the plant output using a different data set of the same condition (*cross-validation*).

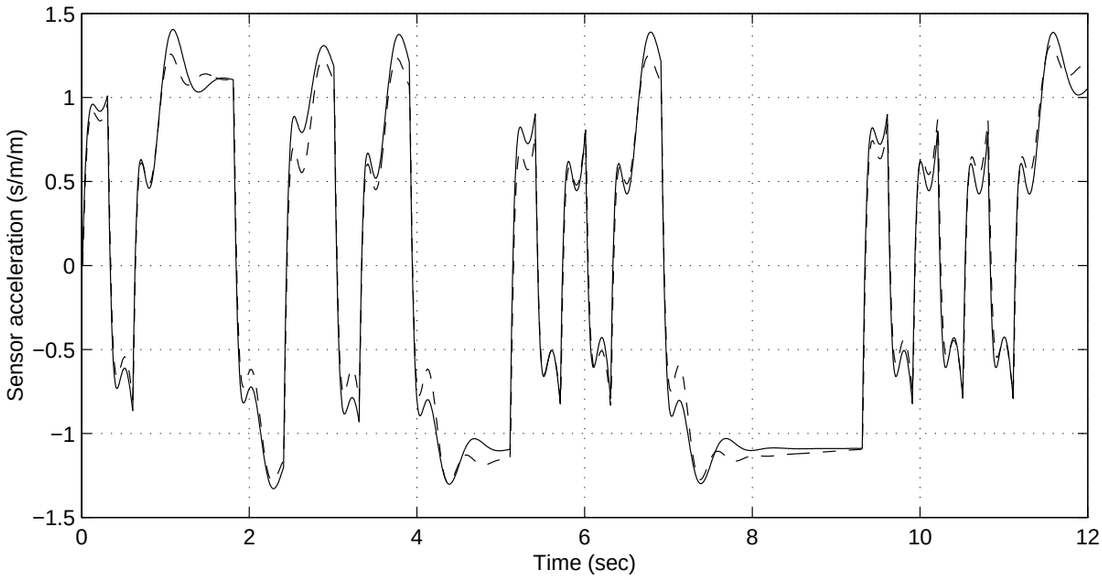


Figure 4.8: Measured plant output (solid) in comparison to output of identified model (dashed)

4.5.5 Residuals

The Least Squares Estimation method is based on the assumption that the noise is Gaussian. This assumption can be verified by evaluating properties of the resulting prediction errors.

A linear model can be described by

$$y(t) = \phi(t\text{-}1)^T \theta + x(t)$$

where for LS methods, x(t) is assumed to be white noise.

If the estimated model is described by

$$\hat{y}(t) = \phi(t-1)^T \hat{\theta}$$

then the prediction errors (*residuals*, or “leftovers”) become

$$\varepsilon(t) = y(t) - \phi(t-1)^T \hat{\theta}$$

which has to be white noise if the model structure is chosen correctly, if the identification method is appropriate for the chosen model structure, and if the noise is white.

The typical whiteness test is to determine the covariance estimate

$$\hat{R}(i) = \frac{1}{N} \sum_{t=1}^{N} \varepsilon(t)\,\varepsilon(t-i)$$

where N is the number of sampled data points.

For practical purpose one considers a normalised value

$$\hat{R}_N(i) = \frac{\hat{R}(i)}{\hat{R}(0)},\ i = 1, 2, ..., n_A, ...$$

which is in case of perfect white noise

$$\hat{R}_N(0) = 1; \qquad \hat{R}_N(1) = 0,\ \text{for } i \geq 1$$

In practice a model can be considered to be valid if the following criterion holds [Lan88]:

$$\hat{R}_N(0) = 1; \qquad \left|\hat{R}_N(i)\right| \leq 0.15 \cdots 0.17$$

4.6 Identification Results

As already mentioned, the dynamics of the vehicle have been identified over a domain of operating conditions. Variations in velocity, road tire contact and vehicle mass were considered. The velocity varied between 5 m/s (18 km/h) and 30 m/s (108 km/h), the considered variations in mass correspond to a vehicle loaded only with the driver, up to a vehicle loaded with 5 persons and an additional load of 100 kg, the considered road-tire contact characteristics were conditions on dry, wet and snow road. As a matter of fact, the maximum speed needs to be adapted to the road condition, therefore the considered top speed on snow was 18 m/s (65 km/h).

The identified models show that the crucial variations are those of the velocity and road-tire contact. This was already assumed in chapter 4.4.1. The dynamics of the vehicle are not very sensitive to variations in mass, which was already observed by previous studies [Ram95]. Throughout the following experimental evaluation, an average mass corresponding to the mass of the vehicle loaded with the driver and one passenger has been considered. Therefore a two-dimensional operating domain is considered, shown in figure 4.9.

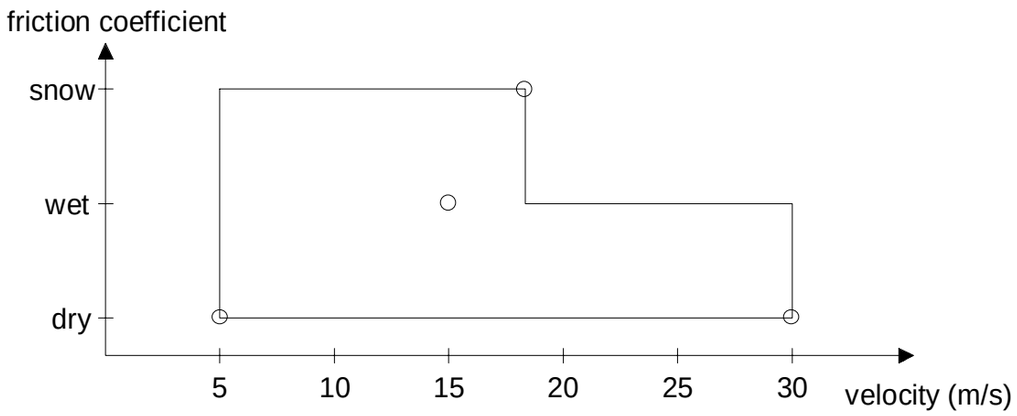


Figure 4.9: Operating domain

As already mentioned, above a velocity of approximately 20 m/s an additional low-frequency mode appears which requires a model of 3rd order for these conditions. 2nd order models are used to approximate conditions of a velocity of 20 m/s or less.

The Bode plots of the identified models obtained for various velocities on a dry road are shown in figure 4.10. The considerable changes of the plant behaviour, dependent on the velocity, can be seen. For high velocities (30 m/s) an additional mode occurs, for low speed (5 m/s) a shift of the main mode can be observed. The static gain varies between 20 and 47 dB. The considerable changes of the phase give an additional hint on the required robustness of the designed controller.

Figure 4.11 shows the Bode plots of the models for a constant speed of 15 m/s on a dry, a wet and on snow road. The fact that with some care even a novice driver can handle a vehicle on a wet road can be explained with the relatively small differences in the frequency responses between a dry and a wet road. However the differing frequency response on a snow road confirms the particularity of vehicle handling on snow.

This identification procedure provides a set of pulse transfer functions corresponding to the many different operating conditions. For control purpose one transfer function has to be selected for which a robust controller is developed. This selection is described in chapter 6.4.2. The robustness of this controller is finally investigated in the complex, non-linear vehicle simulator for all considered operating conditions.

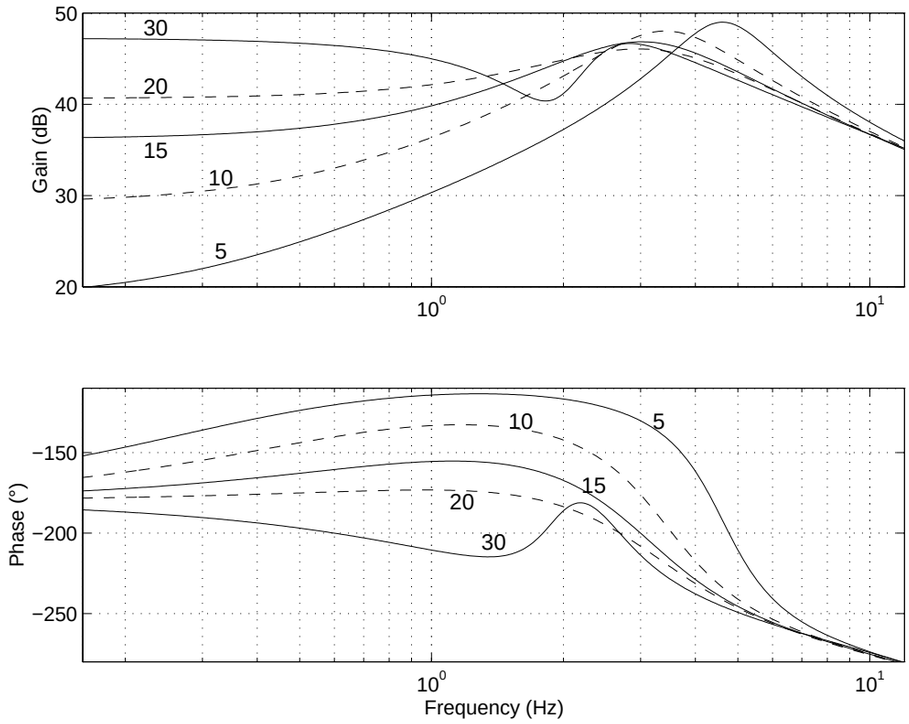


Figure 4.10: Frequency responses of identified models on a dry road for varying speed

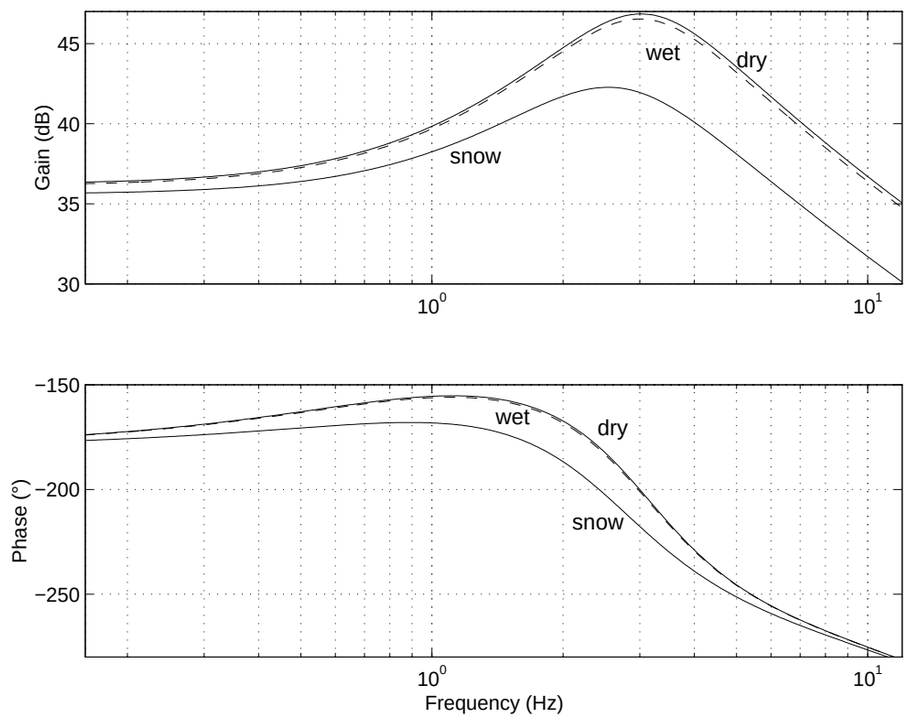


Figure 4.11: Frequency responses of identified models for a constant speed and varying road-tire contact

5 Design of Insensitive Control Systems

This chapter contains background information about expressions and notions which are commonly used in the analysis and design of insensitive control systems.

In the discussion of control systems usually a distinction is made between two kinds of adverse effects: *disturbances* and *perturbations* [Kwa85]. *Disturbance* is an external signal that influences the plant and whose behaviour cannot be controlled. In the formulation of the control problem of automatic steering the radius of the road is considered as disturbance. *Perturbation* is a transient or permanent change in the dynamic and/or static properties of the plant. In automatic steering, the variations in velocity, road condition and mass are considered as perturbations.

The primary task in the design of robust control systems is to ensure stability for all possible perturbations. This property is called *stability robustness*. *Performance robustness* presumes stability robustness and indicates that certain performance specifications are not violated under all possible perturbations.

These performance specifications and the robustness performance of a control system are usually expressed and quantified using robustness margins and sensitivity functions, which is explained in the following.

5.1 Sensitivity Functions

The sensitivity functions play a crucial role in the robustness analysis of the closed loop system with respect to modelling errors. These functions are “shaped” in order to assure “nominal performances” for the rejection of the disturbances of the closed loop system in presence of plant perturbations [Lan94]. They are used as suitable quantifiers for both nominal performance and stability robustness [Lim95, Zho95].

Usually four sensitivity functions are considered. The most important one is called *output sensitivity function* $S(z^{-1})$, in the literature it is often simply called the “sensitivity function”. It determines how the output disturbance $p_y(t)$ is transmitted to the control system output y(t). It can be derived considering the following control system:

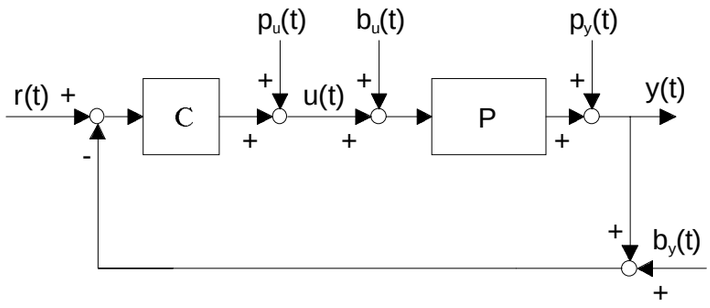


Figure 5.1: Closed loop system with disturbances and measurement noises

r(t) and u(t) denote respectively the reference sequence and the plant input, $C(z^{-1})$ and $P(z^{-1})$ represent the controller and the plant transfer functions. $p_u(t)$ and $p_y(t)$ are input and output disturbances and $b_u(t)$ and $b_y(t)$ are input and output noise measurements.

For the automatic steering problem the reference sequence r(t) is zero (if no lane change is considered), the input u(t) represents the steering angle at the wheels which is manipulated to track the guideline, y(t) denotes the output of the plant which is the tracking error between sensor and guideline, the input disturbances $p_u(t)$ and input noise measurements $b_u(t)$ could be detrimental influences of the steering actuator on the steering angle. The influence of a curved road on the sensor displacement is considered as output disturbance $p_y(t)$, the output noise measurement $b_y(t)$ takes the influence of a noisy displacement sensor signal into account.

Since the sensitivity function $S(z^{-1})$ determines how the output disturbance $p_y(t)$ is transmitted to the control system output y(t), the transfer function is given by

$$S(z^{-1}) = \frac{1}{1 + P(z^{-1})C(z^{-1})} = \frac{1}{1 + L} \tag{5.1.1}$$

This transfer function explains why classical design techniques concentrate on making the loop gain L = PC large. A large loop gain L will result in a small sensitivity function $S(z^{-1})$, i.e. output disturbances are well damped, however, this will also lead to very unstable systems, and moreover, large loop gains may result in unacceptably large sensitivity to measurement noise [Kwa85], which will be explained later.

However, having $S(z^{-1})$ small reduces the closed loop sensitivity to disturbances and plant variations. This motivates the minimisation of the peak value $\|S(z^{-1})\|_\infty$ of the sensitivity function, as it is considered in the H_∞ approach. Nevertheless, the minimisation of the maximum peak value $\|S(z^{-1})\|_\infty$ as it stands is not a useful design tool, it does not show how small $S(z^{-1})$ is at low frequencies, which is of paramount importance for the control system. This indicates that a sensitivity function *shaping* procedure is required. A typical shape of the sensitivity function is shown in the figure 5.2, where also the peak value $\|S(z^{-1})\|_\infty$ is indicated. T_s denotes the sampling period of the system.

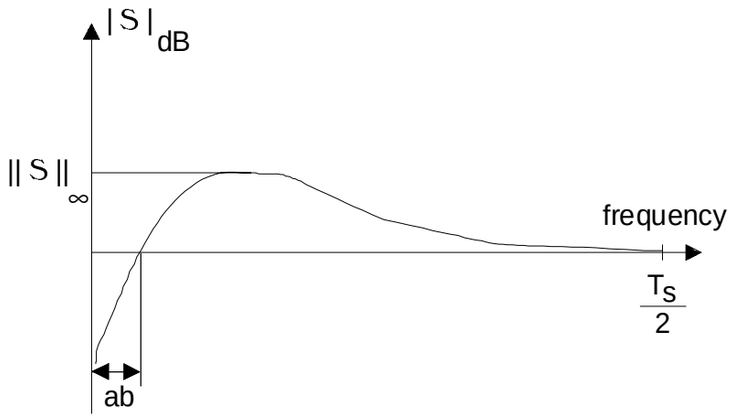


Figure 5.2: Typical shape of the output sensitivity function

The gain of the sensitivity function can be interpreted as the amplification or attenuation factor of the output disturbances. At frequencies where the gain of $S(z^{-1})$ is smaller that 1 (< 0 dB) the disturbances are attenuated. This attenuation is desired, but it cannot be obtained over all frequencies since for discrete systems which are open loop stable, the integral of the sensitivity function as it is shown in figure 5.2 is equal to zero [Sun88]. Therefore the sensitivity function is shaped in order to keep it

small in an important frequency range, and to allow larger values where the attenuation is less important.

For the sensitivity shaping procedure the following guidelines are given:

- in the low frequency region an attenuation of disturbances is of particular importance. The upper limit of this attenuated frequency region, which is called *attenuation bandwidth* ab and which is also shown in figure 5.2, should be as large as possible.
- The maximum value of the sensitivity function should not exceed 6 dB,
- an amplification of disturbances at higher frequencies than the bandwidth of the closed-loop is generally allowed, except in specific applications where it can be useful to locally attenuate one or more specific frequencies.
- An amplification at high frequencies, i.e. close to the sampling frequency, needs to be avoided.

The *complementary sensitivity function* $\mathrm{T}(z^{-1})$ or *noise sensitivity function* characterises the sensitivity to measurement noise**.** Its transfer function is

$$\mathrm{T}(z^{-1}) = \frac{\mathrm{P}(z^{-1})\,\mathrm{C}(z^{-1})}{1+\mathrm{P}(z^{-1})\,\mathrm{C}(z^{-1})}$$

Having $\mathrm{T}(z^{-1})$ small reduces the closed loop sensitivity to measurement errors. The appellation “complementary sensitivity function” is motivated by the fact that

$$\mathrm{S} + \mathrm{T} = 1,$$

which shows, that both sensitivity functions cannot be simultaneously small. A compromise between sensitivity to measurement noise and sensitivity to output disturbances has to be achieved.

Beside the sensitivity function S and the complementary sensitivity function T, two further transfer functions should be considered: the product of the plant P by the sensitivity function S represents the transfer function between the input disturbances $p_u(t)$ and the output y(t), the product of the compensator C and the sensitivity function S describes the influence of the output disturbance $p_y(t)$ and output measurement noise $b_y(t)$ on the input. The latter transfer function is often denoted by *U*.

For good performance robustness, all these sensitivity functions should be kept small in appropriate frequency ranges since they emphasise the effects of the output and input disturbances and measurement noises on the input-output behaviour of the closed loop system. This is illustrated in figure 5.3 and figure 5.4.

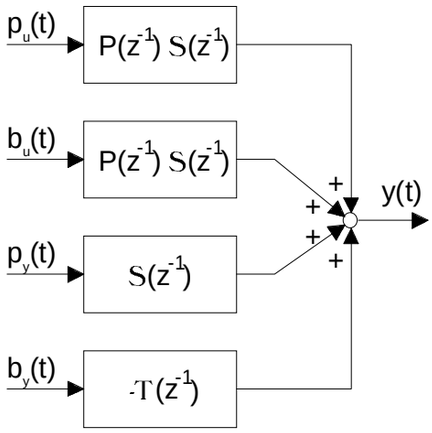


Figure 5.3: Nominal output performance

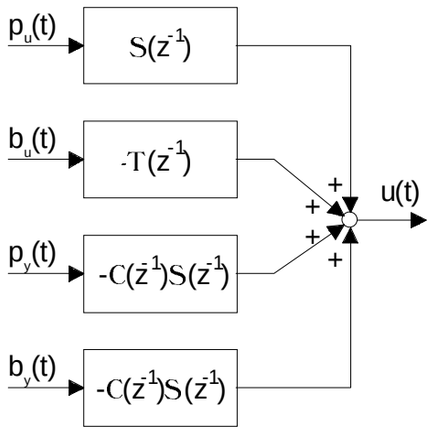


Figure 5.4: Nominal input performance

5.2 Robustness Margins

In the present work four indicators are used to express the robustness of the design in terms of the minimal distance with respect to the critical point [-1, j0] in the Nyquist plane. These indicators are the commonly used gain and phase margins ΔG and $\Delta\phi$, as well as the modulus margin ΔM and the delay margin $\Delta\tau$. Figure 5.5 shows the margins in the Nyquist plane.

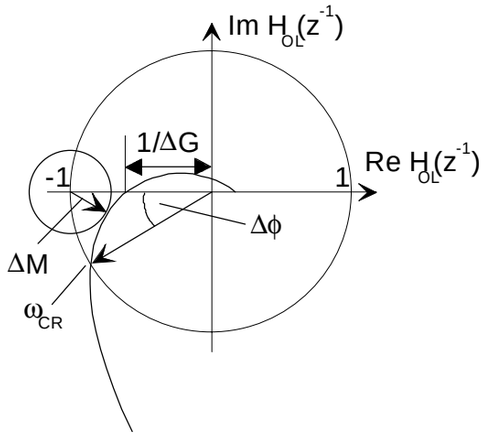


Figure 5.5: Nyquist plane with robustness margins

Gain margin ΔG and phase margin Δϕ

The gain and phase margins indicate the stability margins in the Nyquist plane as it can be seen in figure 5.5. They are commonly used in analysis of robust control systems, however, they describe the distance between Nyquist curve and the critical point [-1; j0] just for two particular directions. Thus they represent indicators which are only valid for the proximity of the critical point.

Therefore in the following the modulus margin is considered, which describes the minimum distance between Nyquist curve and the critical point *over all frequencies*.

Modulus Margin ΔM

The modulus margin is the minimal distance between the critical point [-1; j0] and the Nyquist plot of the open loop transfer function $H_{OL}(z^{-1}) = C(z^{-1})\,P(z^{-1})$, where $C(z^{-1})$ and $P(z^{-1})$ denote the transfer functions of the controller and the plant model as shown in figure 5.1. This can be written as

$$\Delta M = |1+H_{OL}(e^{-j\omega})|_{min} = |S^{-1}(e^{-j\omega})|_{min} = 1 / \| S \|_{\infty} = (|S(e^{-j\omega})|_{max})^{-1}$$

This shows the relation between the sensitivity function of the closed loop $S(z^{-1})$ and the modulus margin ΔM. It implies that a reduction of $|S(e^{-j\omega})|_{max}$ will increase the modulus margin. Since $|S(e^{-j\omega})|_{max}$ is nothing but the H_∞ norm of the output sensitivity function, the modulus margin directly corresponds to the H_∞ norm. Minimisation of the H_∞ norm of $S(e^{-j\omega})$ will maximise the modulus margin.

To obtain the modulus margin it is therefore sufficient to simply plot the frequency characteristics of the modulus (gain) of the sensitivity function in dB. In this case:

$$\Delta M\ dB = (|S(e^{-j\omega})|_{max})^{-1}\ dB = -\,|S(e^{-j\omega})|_{max}\ dB$$

Delay margin (Δτ)

The delay margin is the maximal additional delay which will be tolerated in the open loop system without causing the instability of the closed loop system. Thus, it con-

tains additional information which is not expressed in the modulus margin. It is defined as

$$\Delta\tau = \min_i \left\{ \frac{\Delta\phi_i}{\omega_{CR_i}} \right\}$$

where ω_{cr} denotes the cross-over frequency. The index i indicates the case when the Nyquist curve intersects the unit circle at several cross-over frequencies ω_{CR_i} .

Typical values of the robustness margins for a “robust” controller design are:

- Modulus margin: $\Delta M \geq 0.5$ (-6 dB),
- Delay margin: $\Delta\tau \geq T_s$ where T_s denotes the sampling period,
- Gain margin: $\Delta G \geq 2$ (6 dB),
- Phase margin: $30° \leq \Delta\phi \leq 60°$.

Notice, that the robustness margins can not replace the analysis of the sensitivity functions, since the latter provide information concerning system performance at *each* frequency point.

5.3 Stability Robustness

Control systems are designed using mathematical models that are only approximations of a real plant. The modelling of a physical system is therefore only complete when the modelling errors have been quantified. This can be done in terms of bounds or probability distributions of some type.

The input-output behaviour of a plant is described by a nominal model $P_n(z^{-1})$ and its uncertainty $\Delta(z^{-1})$. This plant-model mismatch may be represented in several ways, in dependence of the *a priori* knowledge about the plant and the main control design features.

The direct additive form is shown in figure 5.6.

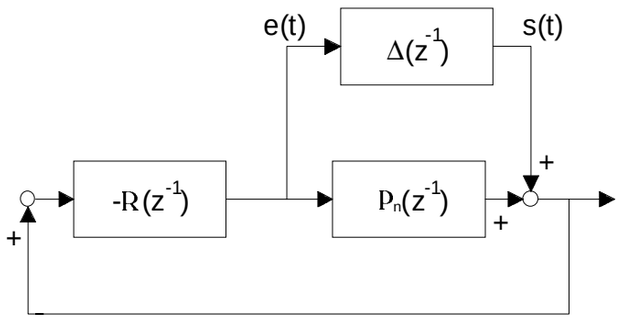


Figure 5.6: Direct additive form

For this form the plant can be described by

$$P(z^{-1}) = P_n(z^{-1}) + \Delta(z^{-1})$$

and e(t) can be described by

$$e(z^{-1}) = -R(z^{-1}) [P_n(z^{-1}) e(z^{-1}) + s(z^{-1})]$$
$$= -[1 + R(z^{-1}) P_n(z^{-1})]^{-1} R(z^{-1}) s(z^{-1})$$
$$= -R(z^{-1}) S(z^{-1}) s(z^{-1})$$

For the inverse additive form, shown in figure 5.7, the plant becomes

$$P(z^{-1}) = [1 + P_n(z^{-1}) \Delta(z^{-1})]^{-1} P_n(z^{-1})$$

and

$$e(z^{-1}) = P_n(z^{-1}) [-R(z^{-1}) e(z^{-1}) - s(z^{-1})]$$
$$= -[1 + R(z^{-1}) P_n(z^{-1})]^{-1} P_n(z^{-1}) s(z^{-1})$$
$$= -S(z^{-1}) P_n(z^{-1}) s(z^{-1})$$

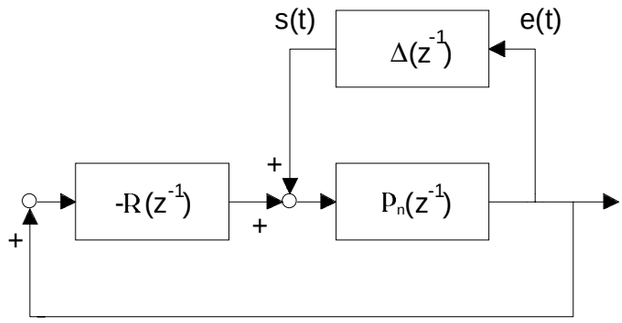


Figure 5.7: Inverse additive form

The additive forms are generally used to represent modelling errors which are independent of the frequency band.

The model error may also be represented in the direct multiplicative form:

A.) at the plant input:

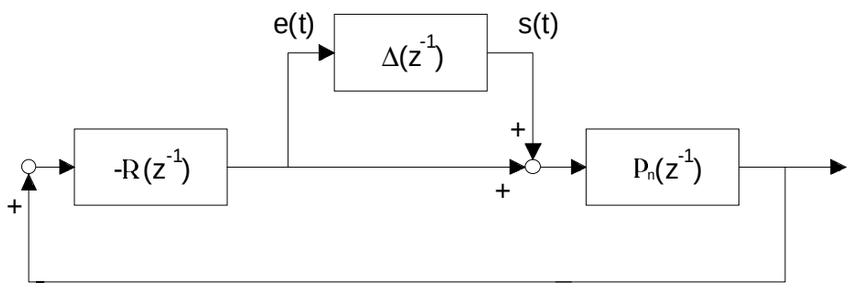


Figure 5.8: Direct multiplicative form at the plant input

B.) at the plant output:

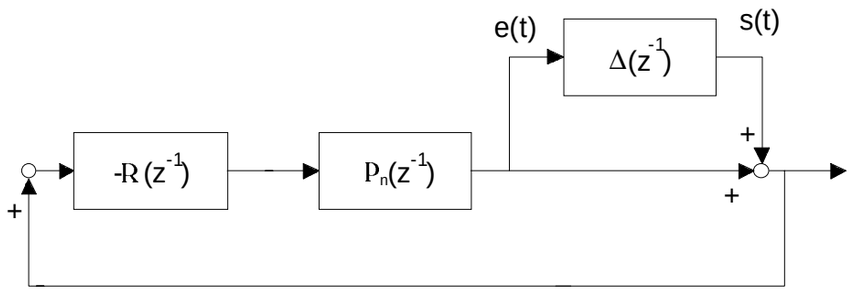


Figure 5.9: Direct multiplicative form at the plant output

For both forms the plant becomes

$$P(z^{-1}) = [1 + \Delta(z^{-1})]\, P_n(z^{-1})$$

and

$$e(z^{-1}) = -R(z^{-1})\, P_n(z^{-1})\, [e(z^{-1}) + s(z^{-1})]$$

$$= -[1 + R(z^{-1})\, P_n(z^{-1})]^{-1}\, R(z^{-1})\, P_n(z^{-1})\, s(z^{-1})$$

$$= -T(z^{-1})\, s(z^{-1})$$

The direct multiplicative forms are more appropriate to represent the modelling errors involved in the actuators or the sensors.

The model error for the inverse multiplicative form can be expressed for

A.) at the plant input:

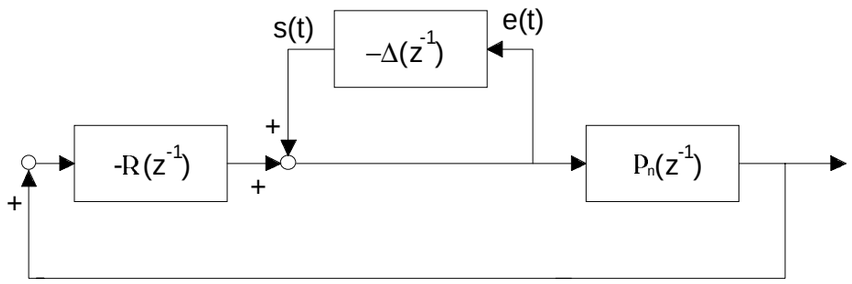


Figure 5.10: Inverse multiplicative form at the plant input

B.) at the plant output:

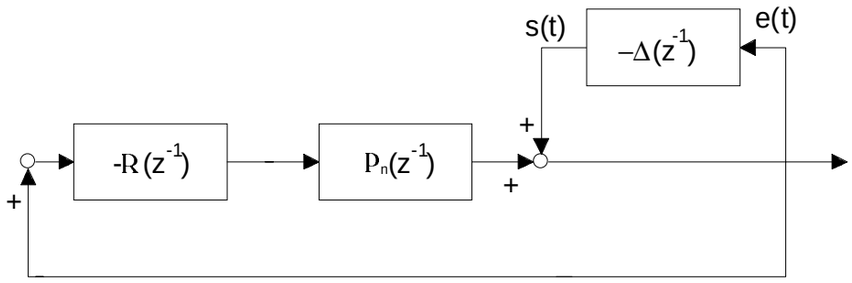


Figure 5.11: Inverse multiplicative form at the plant output

the plant becomes

$$P(z^{-1}) = [1 + \Delta(z^{-1})]^{-1} P_n(z^{-1})$$

and

$$\begin{aligned} e(z^{-1}) &= - s(z^{-1}) - R(z^{-1}) P_n(z^{-1}) e(z^{-1}) \\ &= - [1 + R(z^{-1}) P_n(z^{-1})]^{-1} s(z^{-1}) \\ &= - S(z^{-1}) s(z^{-1}) \end{aligned}$$

The inverse multiplicative forms allow to deal with the parametric errors.

In the following the conditions for the stability robustness for the different uncertainties are given. Therefore the previous block diagram are represented in a unified form shown in figure 5.12.

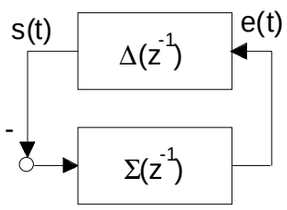


Figure 5.12: Unified form for system uncertainty

By the small gain theory (see for instance [Des75]) a sufficient condition for the closed-loop system of figure 5.12 to be stable is that the norm $||\Sigma(z^{-1}) \cdot \Delta(z^{-1})||$ of the loop be less than 1 [Kwa93]. By the inequality

$$||\Sigma(z^{-1}) \cdot \Delta(z^{-1})|| \leq ||\Sigma(z^{-1})|| \cdot ||\Delta(z^{-1})||$$

this is guaranteed if $||\Sigma(z^{-1})|| \cdot ||\Delta(z^{-1})|| < 1$. If the perturbation is characterised by $||\Delta(z^{-1})||_\infty \leq 1$, which is simple yet very general, then the sufficient and necessary condition for the perturbed system to be stable is

$$||\Sigma(z^{-1})||_\infty < 1$$

Table 5.1 gives the expressions of $\Sigma(z^{-1})$ for the corresponding plant-model mismatch representation [M'Saad96].

Plant-model Mismatch Representation	Sensitivity Function $\Sigma(z^{-1})$
Direct additive form	$R(z^{-1})\, S(z^{-1})$
Inverse additive form	$S(z^{-1})\, P_n(z^{-1})$
Direct multiplicative form	$T(z^{-1})$
Inverse multiplicative form	$S(z^{-1})$

6 The Generalized Predictive Control Design

The design of a stable “general purpose” controller has been a challenging problem over the previous decade, which has to cover a variety of process properties and which has to be applicable to:

- a non-minimum-phase plant: most continuous-time transfer functions have discrete-time zeros outside the unit circle when sampled at a fast enough rate,
- an open-loop unstable plant or plant with badly-damped poles,
- a plant with variable or unknown dead-time,
- a plant with unknown order: pole-placement and LQG self-tuners perform badly if the order of the plant is overestimated because of pole/zero cancellations in the identified model, unless special precautions are taken.

The Generalized Predictive Control (GPC) is effective with plants which are non-minimum-phase, open-loop unstable, with variable dead-time, and whose models are overparameterized by the estimation scheme without special precautions being taken [Cla87].

In the first paragraph (chapter 6.1) the predictive control law is derived, starting from the Minimum Variance controller, which is considered as the origin of GPC. The reader who is more interested in practical information can jump over chapter 6.1.1, 6.1.2 and 6.1.3, where the plant model and the noise model is described, and the output prediction and its solution by recursion of the Diophantine equation is explained. In chapter 6.1.4 the general structure of the obtained GPC controller is shown.

In the second part (chapter 6.2) the choice of the GPC design parameters is explained for the general case, in chapter 6.3 the application of GPC for automatic steering is presented and the design parameters are given, in the last part (chapter 6.4) simulation results of the robust GPC controller are shown for several driving conditions.

6.1 The Predictive Control Law

The roots of GPC can be found in the well-known Minimum Variance Controller, where for a given linear input-output model the variance of the tracking error is minimised:

$$J_{MV} = E\left\{\left(y(t+d) - r(t+d)\right)^2\right\}$$

y(t), r(t) and d denote, as usual, the system output, the reference sequence, and the assumed value of the plant’s dead-time, respectively. In this approach a d-step-ahead predictor is used. This control strategy suffers from the fact that is works only for minimum phase systems, i.e. for systems with stable plant zeros.

The Generalized Minimum-Variance (GMV) approach proposed by Clarke and Gawthrop in 1975 [Cla75] minimises the tracking error *and* a weighted control signal u(t):

$$J_{GMV} = E\left\{(y(t+d) - r(t+d))^2 + \lambda u^2(t)\right\} \qquad (6.1.1)$$

Further modifications consider a filtered control signal $D(q^{-1})\,u(t)$

$$J_{GMV} = E\left\{(y(t+d) - r(t+d))^2 + \lambda\left[D(q^{-1})u(t)\right]^2\right\} \qquad (6.1.2)$$

Selecting $D(q^{-1})$ as the differencing operator $1 - q^{-1}$ ensures offset-free performance in the case of step-like disturbances for plant models without an integrator. Notice that in this case not the control signal u(t) is considered in the criterion, but the incremental input $D(q^{-1})\,u(t)$.

GMV showed to be very robust, but sensitive to varying dead-time unless λ is large, which results in poor control performance. Therefore an additional extension was made by Clarke, Mohtadi and Tuffs in 1987 when the following GPC criterion was proposed:

$$J_{GPC} = E\left\{\sum_{j=sh}^{ph}\left[y(t+j) - r(t+j)\right]^2 + \lambda\sum_{j=0}^{ch}\left[D(q^{-1})u(t+j-1)\right]^2\right\} \qquad (6.1.3)$$

with $D(q^{-1})\,u(t+i) = 0$ for i = ch, ..., ph-d-1.

This minimisation predicts the output error over a future observation horizon and produces a *future* control sequence u(t+j) so that the predicted output of the process y(t+j) is close to the desired process output r(t+j) and that after a given *control horizon* ch no additional changes of the control sequence u(t+j) are needed. The observation horizons of the predicted output error are called *starting prediction horizon* or simply *starting horizon* sh and *prediction horizon* or *maximum output horizon* ph.

This reveals the real power of the GPC approach: instead of allowing the future control outputs u(t) to be “free”, the control increments are assumed to be zero beyond the control horizon ch < ph,

i.e. $\Delta u(t+i)=0$, $ch \le i < ph-d-1$.

For example, if ch = 1 only one control change (i.e. $\Delta u(t)$) is considered, after which the controls u(t+i) are all taken to be equal to u(t). Notice that, of the produced control sequence u(t+j), $0 \le j \le ch - 1$ only u(t) is actually applied. At time t+1 a new minimisation problem is solved. This implementation is called *Receding Horizon Control*. The criterion itself represents a finite horizon (N_2) criterion.

The control law is carried out by the following steps, which are performed at each sample-instant t:

1.) the future set-point sequence r(t+j) is calculated or, if it is not known in advance, it is assumed to be constant and equal to the current set-point r(t),

2.) the prediction model of equation (6.1.13) is used to generate a set of predicted outputs $\hat{y}(t+j|t)$,

3.) an appropriate quadratic cost function of the future errors and controls is minimised, assuming that after a specified control horizon further increments in control are zero, to provide a suggested sequence of future controls u(t+j),

4.) the first element u(t) of the control output is applied to the plant and the data vectors are shifted so that the calculation can be repeated at the next sample instant.

Finally, the basic key ideas of GPC are summed up:

- instead of a CARMA model, a CARIMA model is assumed, (see chapter 6.1.1)
- a long-range prediction over a finite horizon greater than the dead-time of the plant is used, (the long range prediction is explained in chapter 6.1.2)
- the prediction is performed by recursion of the Diophantine equation, (see chapter 6.1.3)
- weighted increments are considered in the cost-function, (see equation (6.1.3) for $D(q^{-1}) = 1 - q^{-1} = \Delta$)
- a control horizon is given after which control increments are taken to be zero.

6.1.1 Plant Model Description and Noise Models

In the following paragraph we assume the system description to be

$$y(t) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(t) + x(t) \qquad (6.1.4)$$

where

$$A(q^{-1}) = 1 + a_1q^{-1} + ... + a_{n-1}q^{n-1} + a_nq^{-n} = 1 + q^{-1}\,A^*(q^{-1})$$

$$B(q^{-1}) = b_0 + ... + b_mq^{-m}$$

u(t) is the control input, y(t) is the measured variable or output, and x(t) is a disturbance term.

Noise or disturbance models are usually described using MA (moving-average), AR (autoregressive) or ARMA (autoregressive moving-average) models [Kam92].

The MA model

is described by

$$x(t) = C(q^{-1})\,\gamma(t)$$

the AR model

is described by

$$x(t) = \frac{1}{A(q^{-1})}\gamma(t)$$

and the ARMA model

can be described by

$$x(t) = \frac{C(q^{-1})}{A(q^{-1})}\gamma(t)$$

where $C(q^{-1}) = 1 + c_1 q^{-1} + ... + c_{nc} q^{-nc}$,

$A(q^{-1}) = 1 + a_1 q^{-1} + ... + a_n q^{-n}$, and

$\gamma(t)$ is an uncorrelated random sequence with zero mean values or white noise.

The influence of noise *and* the controller on the process can be described by a CARMA model (Controlled Auto-Regressive and Moving-Average), which is equivalent to a ARMAX model (Auto-Regressive Moving-Average eXogenous input), or by a CARIMA or ARIMAX model (integrated moving-average). The *controlled* part or the *exogenous input* (or external input) represents the plant input u(t) in our case [Lan88].

The CARMA (ARMAX) model

If x(t) is considered as ARMA model we obtain for the process output:

$$y(t) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{A(q^{-1})}\gamma(t)$$

which is equal to

$$A(q^{-1})\, y(t) = B(q^{-1})\, u(t-1-d) + C(q^{-1})\, \gamma(t) \qquad (6.1.5)$$

The CARIMA (ARIMAX) model

Although the CARMA model has been widely used, it is inappropriate for

- step disturbances, and
- Brownian motion (found in plants relying on energy balance).

Therefore the following CARIMA (ARIMAX) noise model is preferred:

$$x(t) = \frac{C(q^{-1})}{A(q^{-1})D(q^{-1})}\gamma(t)$$

where the polynomial $D(q^{-1})$ with $D(1) = 0$ ensures integral action in the controller in order to provide offset-free performance. In the simplest case $D(q^{-1})$ is the differencing operator $1 - q^{-1}$.

This model can then be written as:

$$y(t) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{A(q^{-1})D(q^{-1})}\gamma(t) \qquad (6.1.6)$$

$$\Leftrightarrow \quad A(q^{-1})\, D(q^{-1})\, y(t) = B(q^{-1})\, D(q^{-1})\, u(t-1-d) + C(q^{-1})\, \gamma(t) \qquad (6.1.7)$$

Notice that in the disturbance model and in the plant model are the same denominator polynomial $A(q^{-1})$ is no restriction of the general case. This is shown, if different polynomials are used:

$$y(t) = \frac{q^{-d}B_1(q^{-1})}{A_1(q^{-1})}u(t) + \frac{C_2(q^{-1})}{A_2(q^{-1})}\gamma(t)$$

This is equal to

$$y(t) = \frac{q^{-d}B_1(q^{-1})A_2(q^{-1})}{A_1(q^{-1})A_2(q^{-1})}u(t) + \frac{C_2(q^{-1})A_1(q^{-1})}{A_1(q^{-1})A_2(q^{-1})}\gamma(t)$$

which can be written as

$$y(t) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{A(q^{-1})}\gamma(t)$$

6.1.2 The Output Prediction

Before deriving the predictor equations let us sum up which information is available at time t:

- the output y(t-i) for $i \geq 0$, which is the present and the past output signal of the plant,
- the input u(t-i) for $i \geq 1$, which is the past input signal of the plant,
- the present and the future inputs u(t+i) for $0 \geq i \geq j-d-1$, which have been determined by the control law, and
- the present and the past output disturbance $\gamma(t-i)$ for $i \geq 0$

This information will be used in the following, to predict the future outputs of the plant.

The design of the long-range prediction is based on the following two Diophantine equations:

$$C(q^{-1}) = A(q^{-1})\,D(q^{-1})\,E_j(q^{-1}) + q^{-j}\,F_j(q^{-1}) \qquad (6.1.8)$$

$$E_j(q^{-1})\,B(q^{-1}) = C(q^{-1})\,G_{j-d}(q^{-1}) + q^{-j+d}\,H_{j-d}(q^{-1}) \qquad (6.1.9)$$

with

$$E_j(q^{-1}) = e_0 + e_1\,q^{-1} + \ldots + e_{j-1}\,q^{-j+1}$$

$$F_j(q^{-1}) = f_0 + f_1\,q^{-1} + \ldots + f_{nf}\,q^{-nf}$$

$$G_{j-d}(q^{-1}) = g_0 + g_1\,q^{-1} + \ldots + g_{j-d-1}\,q^{-j+d+1}$$

$$H_{j-d}(q^{-1}) = h_0 + h_1\,q^{-1} + \ldots + h_{nh}\,q^{-nh}$$

where

$nf = \max(n_A + n_D - 1, n_C - j)$ and

$nh = \max(n_C, n_B + d) - 1$

The calculation of the uniquely defined polynomials $E_j(q^{-1})$, $F_j(q^{-1})$, $G_{j-d}(q^{-1})$ and $H_{j-d}(q^{-1})$ will be explained later, of importance right now is, that $E_j(q^{-1})$ and $F_j(q^{-1})$ can be obtained for given $A(q^{-1})$, $C(q^{-1})$, and the prediction interval j, and that $G_{j-d}(q^{-1})$ and $H_{j-d}(q^{-1})$ can be obtained for given $B(q^{-1})$, $C(q^{-1})$ and the prediction interval j.

Equation (6.1.7) can be multiplied by $E_j(q^{-1})\ q^j$ in order to obtain

$$E_j(q^{-1})\ A(q^{-1})\ D(q^{-1})\ y(t+j) = E_j(q^{-1})\ B(q^{-1})\ D(q^{-1})\ u(t+j-1-d) + E_j(q^{-1})\ C(q^{-1})\ \gamma(t+j) \tag{6.1.10}$$

In this equation $E_j(q^{-1})\ A(q^{-1})\ D(q^{-1})$ can be replaced by using equation (6.1.8):

$$C(q^{-1})\ y(t+j) = E_j(q^{-1})\ B(q^{-1})\ D(q^{-1})\ u(t+j-1-d) + F_j(q^{-1})\ y(t) + E_j(q^{-1})\ C(q^{-1})\ \gamma(t+j) \tag{6.1.11}$$

The polynomial $E_j(q^{-1})\ B(q^{-1})\ D(q^{-1})\ u(t+j-1-d)$ contains one part which can be measured at time t and one part which is to be determined by the control law. Therefore equation (6.1.11) can be decomposed using equation (6.1.10):

$$C(q^{-1})\ y(t+j) = C(q^{-1})\ G_{j-d}(q^{-1})\ D(q^{-1})\ u(t+j-1-d) + H_{j-d}(q^{-1})\ D(q^{-1})\ u(t-1) + \\ + F_j(q^{-1})\ y(t) + E_j(q^{-1})\ C(q^{-1})\ \gamma(t+j)$$

$\Leftrightarrow$

$$y(t+j) = G_{j-d}(q^{-1})D(q^{-1})u(t+j-1-d) + \frac{H_{j-d}(q^{-1})D(q^{-1})}{C(q^{-1})}u(t-1) + \frac{F_j(q^{-1})}{C(q^{-1})}y(t) + E_j(q^{-1})\gamma(t+j) \tag{6.1.12}$$

This equation reveals the fact, that the future output y(t+j) of the plant is composed of three parts:

- $G_{j-d}(q^{-1})\ D(q^{-1})\ u(t+j-1-d)$ depends on future control values which have to be determined by the control law,
- $\frac{H_{j-d}(q^{-1})D(q^{-1})}{C(q^{-1})}u(t-1) + \frac{F_j(q^{-1})}{C(q^{-1})}y(t)$ contains values which are already available as measurable data of the past, and
- $E_j(q^{-1})\ \gamma(t+j)$ is the future plant disturbance which is unpredictable at time t.

Since the future plant disturbance is assumed to be white noise with zero mean, the minimum variance (MV) j-step-ahead predictor can be simply obtained by setting this unpredictable part to zero:

$$\hat{y}(t+j\,|\,t) = G_{j-d}(q^{-1})D(q^{-1})u(t+j-1-d) + \frac{H_{j-d}(q^{-1})D(q^{-1})}{C(q^{-1})}u(t-1) + \frac{F_j(q^{-1})}{C(q^{-1})}y(t) \tag{6.1.13}$$

Remarks:

- The prediction error can be calculated as
  $y(t+j) - \hat{y}(t+j \mid t) = E_j(q^{-1})\gamma(t+j) = \gamma(t+j) + e_1\gamma(t+j-1) + \cdots + e_{j-1}\gamma(t+1)$.
- The predictor dynamics are determined by the polynomial $C(q^{-1})$. This dynamics can be obtained by a steady-state Kalman filter in the sense of optimal prediction, but it can also be chosen by the user as specifying the observer dynamics. In this case $C(q^{-1})$ is denoted as $T(q^{-1})$. The appropriate selection or $T(q^{-1})$ is described in chapter 6.2.2 .

6.1.3 The Recursion of the Diophantine Equation

The most effective way to implement long-range prediction is to use recursion of the Diophantine equation [M'Saad96, Cla85, Soet90]. This recursion is explained in the following.

We consider equation (6.1.8):

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_j(q^{-1}) + q^{-j}\, F_j(q^{-1}) \qquad (6.1.14)$$

The recursive solution of this equation consists in calculating $E_j(q^{-1})$ and $F_j(q^{-1})$ by using $E_{j-1}(q^{-1})$ and $F_{j-1}(q^{-1})$. Both, $E_j(q^{-1})$ and $F_j(q^{-1})$, as well as $E_{j-1}(q^{-1})$ and $F_{j-1}(q^{-1})$ represent solutions of (6.1.14):

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_j(q^{-1}) + q^{-j}\, F_j(q^{-1}) \qquad (6.1.15)$$

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_{j+1}(q^{-1}) + q^{-j-1}\, F_{j+1}(q^{-1}) \qquad (6.1.16)$$

with

$E_j(q^{-1}) = e_{j,0} + e_{j,1}\, q^{-1} + \ldots + e_{j,j-1}\, q^{-j+1}$, and

$E_{j+1}(q^{-1}) = e_{j+1,0} + e_{j+1,1}\, q^{-1} + \ldots + e_{j+1,j}\, q^{-j}$

$F_j(q^{-1}) = f_{j,0} + f_{j,1}\, q^{-1} + \ldots + f_{j,nf}\, q^{-nf}$

$F_{j+1}(q^{-1}) = f_{j+1,0} + f_{j+1,1}\, q^{-1} + \ldots + f_{j+1,nf}\, q^{-nf}$

where

$nf = \max(n_C - j,\ n_A + d - 1)$

Subtraction of equation (6.1.15) by (6.1.16) leads to

$$0 = A(q^{-1})\, D(q^{-1})\, [E_{j+1}(q^{-1}) - E_j(q^{-1})] + q^{-j}\, [q^{-1}\, F_{j+1}(q^{-1}) - F_j(q^{-1})] \qquad (6.1.17)$$

Since $n_{E_{j+1}} = j$ and $n_{E_j} = j - 1$, the following holds:

$$E_{j+1}(q^{-1}) - E_j(q^{-1}) = e_{j+1,j}\, q^{-j} + E'(q^{-1}) \qquad (6.1.18)$$

where

$E'(q^{-1})$ is a polynomial of degree i - 1 and $e_{j+1,j}$ is the j^{th} coefficient of E_{j+1}.

Equation (6.1.17) can now be written as

$$0 = A(q^{-1})\, D(q^{-1})\, E'(q^{-1}) + q^{-j}\, [q^{-1}\, F_{j+1}(q^{-1}) - F_j(q^{-1}) + A(q^{-1})\, D(q^{-1})\, e_{j+1,j}]$$

Remarks:

- The prediction error can be calculated as
  $y(t+j) - \hat{y}(t+j \mid t) = E_j(q^{-1})\gamma(t+j) = \gamma(t+j) + e_1\gamma(t+j-1) + \cdots + e_{j-1}\gamma(t+1)$.
- The predictor dynamics are determined by the polynomial $C(q^{-1})$. This dynamics can be obtained by a steady-state Kalman filter in the sense of optimal prediction, but it can also be chosen by the user as specifying the observer dynamics. In this case $C(q^{-1})$ is denoted as $T(q^{-1})$. The appropriate selection or $T(q^{-1})$ is described in chapter 6.2.2 .

6.1.3 The Recursion of the Diophantine Equation

The most effective way to implement long-range prediction is to use recursion of the Diophantine equation [M'Saad96, Cla85, Soet90]. This recursion is explained in the following.

We consider equation (6.1.8):

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_j(q^{-1}) + q^{-j}\, F_j(q^{-1}) \qquad (6.1.14)$$

The recursive solution of this equation consists in calculating $E_j(q^{-1})$ and $F_j(q^{-1})$ by using $E_{j-1}(q^{-1})$ and $F_{j-1}(q^{-1})$. Both, $E_j(q^{-1})$ and $F_j(q^{-1})$, as well as $E_{j-1}(q^{-1})$ and $F_{j-1}(q^{-1})$ represent solutions of (6.1.14):

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_j(q^{-1}) + q^{-j}\, F_j(q^{-1}) \qquad (6.1.15)$$

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_{j+1}(q^{-1}) + q^{-j-1}\, F_{j+1}(q^{-1}) \qquad (6.1.16)$$

with

$E_j(q^{-1}) = e_{j,0} + e_{j,1}\, q^{-1} + \ldots + e_{j,j-1}\, q^{-j+1}$, and

$E_{j+1}(q^{-1}) = e_{j+1,0} + e_{j+1,1}\, q^{-1} + \ldots + e_{j+1,j}\, q^{-j}$

$F_j(q^{-1}) = f_{j,0} + f_{j,1}\, q^{-1} + \ldots + f_{j,nf}\, q^{-nf}$

$F_{j+1}(q^{-1}) = f_{j+1,0} + f_{j+1,1}\, q^{-1} + \ldots + f_{j+1,nf}\, q^{-nf}$

where

$nf = \max(n_C - j,\ n_A + d - 1)$

Subtraction of equation (6.1.15) by (6.1.16) leads to

$$0 = A(q^{-1})\, D(q^{-1})\, [E_{j+1}(q^{-1}) - E_j(q^{-1})] + q^{-j}\, [q^{-1}\, F_{j+1}(q^{-1}) - F_j(q^{-1})] \qquad (6.1.17)$$

Since $n_{E_{j+1}} = j$ and $n_{E_j} = j - 1$, the following holds:

$$E_{j+1}(q^{-1}) - E_j(q^{-1}) = e_{j+1,j}\, q^{-j} + E'(q^{-1}) \qquad (6.1.18)$$

where

$E'(q^{-1})$ is a polynomial of degree i - 1 and $e_{j+1,j}$ is the j^{th} coefficient of E_{j+1}.

Equation (6.1.17) can now be written as

$$0 = A(q^{-1})\, D(q^{-1})\, E'(q^{-1}) + q^{-j}\, [q^{-1}\, F_{j+1}(q^{-1}) - F_j(q^{-1}) + A(q^{-1})\, D(q^{-1})\, e_{j+1,j}]$$

with the following solution:

$E'(q^{-1}) = 0$, and

$$q^{-1} F_{j+1}(q^{-1}) = F_j(q^{-1}) - A(q^{-1}) D(q^{-1}) e_{j+1,\,j} \qquad (6.1.19)$$

Since the polynomial $A(q^{-1}) D(q^{-1}) = A'(q^{-1})$ has a unit leading element, equation (6.1.19) leads to

$$e_{j+1,\,j} = f_{j,\,0} \text{ , and} \qquad (6.1.20a)$$

$$f_{j+1,\,j} = f_{j,\,j+1} - a'_{j+1}\, e_{j+1,\,j} \text{ , for j = 0 to the degree of } F_{j+1}. \qquad (6.1.20b)$$

The polynomial E_{j+1} is now given by

$$E_{j+1}(q^{-1}) = E_j(q^{-1}) + q^{-j}\, e_{j+1,\,j} \qquad (6.1.21)$$

In order to initialise the iterations the solution for j = 1 is required. Equation (6.1.17) for j = 1 leads to:

$$E_1 = c_0/a'_0 \qquad (6.1.22a)$$

$$F_1(q^{-1}) = q\,(C(q^{-1}) - A'(q^{-1})\, c_0/a'_0) \qquad (6.1.22b)$$

Since the plant polynomials $A(q^{-1})$ and $B(q^{-1})$ are given, and if one solution E_1 and F_1 is known (equation (6.1.22)), then equation (6.1.20) can be used to obtain $F_{j+1}(q^{-1})$, equation (6.1.21) is used to obtain $E_{j+1}(q^{-1})$, and so on.

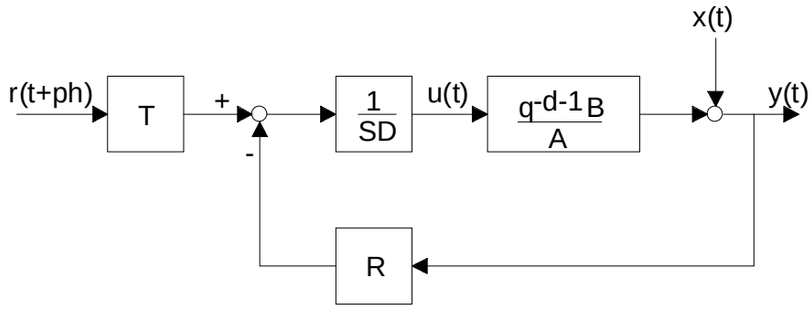
6.1.4 The Control Law

The derivation of the predictive control law can be read in [M'Saad96, Soet90], here only the result is given:

The predictive control law can be implemented using the usual RST-approach:

$$S(q^{-1})\, D(q^{-1})\, u(t) + R(q^{-1})\, y(t) = T(q^{-1})\, r(t+ph)$$

which results in a structure like in the following block diagram:



$S(q^{-1})$ is a function of the prediction horizon ph, of the starting horizon sh, of the disturbance polynomial $C(q^{-1})$, of $H_{j-d}(q^{-1})$ and $G_{j-d}(q^{-1})$ (see chapter 6.1.2),

$R(q^{-1})$ is a function of ph, $F_j(q^{-1})$ and $G_{j-d}(q^{-1})$, and

$T(q^{-1})$ is a function of ph, of $C(q^{-1})$, and of $G_{j-d}(q^{-1})$.

6.2 Choice of GPC Design Parameters

The GPC criterion depends upon the specification of several design parameters, which are explained in the following chapter. Special importance has to be paid on the selection of the three main design parameters, ch, ph, and the weighting factor λ.

6.2.1 Choice of Costing Horizons and Weighting Factor

Simulation exercises on a variety of plant models, including stable, unstable and non-minimum-phase processes with variable dead-time, have shown how the costing horizons sh, ph and ch should best be selected [Cla87, Cla89, Soet90].

The *starting horizon* sh

The starting prediction horizon sh can often be taken as 1; if it is known *a priori* that the dead-time of the plant is at least d sample-intervals, then sh can be chosen as d or more, to minimise computations. If d is not known or is variable, then sh can be set to 1 with no loss of stability and the degree of $B(q^{-1})$ increased to encompass all possible values of d.

The *prediction horizon* or *maximum output horizon* ph

In general the prediction horizon ph is selected to consider all the plant model response which is significantly affected by the current control. In discrete-time this implies that ph exceeds the degree of $B(q^{-1})$ as then all states contribute to the cost. In practice, however, a rather larger value of ph is suggested, ph is set to approximate the rise-time of the plant. Increasing ph improves the robustness of the closed-loop system and simultaneously makes the system respond slower to set point changes. If the process is badly damped or unstable, ph can be chosen to approximate 3 times the rise-time of the plant [Soet90].

ph depends also on the sampling period. A short sampling period results in a large prediction horizon. However, if, as a role of thumb, the sampling period is chosen 10-20 times smaller than the settling time of the closed-loop system's step response, typical values for ph are 10-20 [Soet90].

The *control horizon* ch

The control horizon is an important design parameter. It determines the degrees-of-freedom in future control increments. For a simple plant, e.g. open-loop stable though with possible dead-time and non-minimum-phasedness, a value of ch of 1 gives generally acceptable control. Increasing ch makes the control and the corresponding output response more active until a stage is reached where any further increase in ch makes little difference. An increased value of ch is more appropriate for complex systems where it is found that good control is achieved when ch is at least equal to the number of unstable or badly-damped poles. The smaller ch, the slower the closed-loop system's step response and the larger its robustness. If the process is stable but has badly situated zeros, then ch must be chosen $\leq n_A + 1$. If the process is badly damped or unstable, ch should be chosen > 1 [Soet90].

The weighting factor λ

GPC can be used with a non-minimum-phase plant even if λ is zero. In this case the optimal control law tries to remove the process dynamics using an inverse plant model in the controller. Non-minimum-phase plant stability requires a non-zero value of λ. The easiest choice for λ is zero, with increasing values of λ the robustness of

the closed-loop system improves, but it also becomes slower. Notice that robustness can also be improved using a frequency weighting filter in the output as it is described in chapter 6.2.4. This method is usually preferable[Cla89].

The above discussion implies that GPC can be considered in two ways. A large class of plant models in many industrial applications can be stabilised with default values of sh =1, ch = 1, and ph equal to the plant rise-time. For high-performance applications such as the control of coupled oscillators with modal modes, a higher value for ch is recommended [Cla87]. Choosing ch = n_A has shown to yield a compromise between robustness and performance [Soet90].

6.2.2 Selection of predictor poles

The disturbance term

$$x(t) = \frac{C(q^{-1})}{A(q^{-1})D(q^{-1})}\gamma(t)$$

in equation (6.1.6) can be interpreted in different ways.

If $\gamma(t)$ is a sequence of independent random variables and $C(q^{-1})$ has been estimated by an identification of the plant disturbance, then a predictor based on this model will be asymptotically optimal (minimum-variance). This assumption is commonly used in the self-tuning literature.

However, the polynomials $C(q^{-1})$ and $D(q^{-1})$ are usually difficult to estimate why fixed polynomials are commonly used instead [Åst84]. In this case $C(q^{-1})$ is usually denoted as $T(q^{-1})$. The predictions will then not be optimal but a well selected observer polynomial $T(q^{-1})$ can improve both the robustness to unmodelled dynamics disturbances and the sensitivity to high frequency noise.

Basically two choices are considered in practice [Soet90]:

1.) $T(q^{-1}) = \hat{A}(1 - \mu q^{-1})$

with $0 \le \mu \le 1$,

where $\hat{A}(q^{-1})$ denotes the denominator polynomial of the estimated model. This means, that μ can be considered as a damping factor and the poles of the control model are moved towards the origin. An increasing value of μ leads usually to increasing robustness of the process.

When $\mu = 0$ (hence, $T = \hat{A}$) a compromise between robustness and regulator performance is obtained. However, this choice for T cannot be used if the model has badly damped poles.

2.) $T(q^{-1}) = (1-\mu q^{-1})^{n_A}$

with $0 \le \mu \le 1$.

This leads mostly to a maximum robustness for a particular value for μ (usually $0.6 \le \mu \le 0.9$) and it can also be used with unstable processes. Increasing μ makes the closed-loop system respond more slowly to set point changes.

The default value suggested by Clarke is $T(q^{-1}) = (1 - 0.8 \cdot q^{-1})^{n_A}$ which lead to acceptable results in most cases [Soet90].

The $T(q^{-1})$ polynomial can also be used to decrease the variance of the controller output if the controller is too active due to noise acting on the system. Increasing values of μ make the controller output less active and the noise rejection is better, however, load changes are rejected more slowly. Thus the robustness of the system is improved.

6.2.3 Selection of D-Polynomial

As the load disturbances generally exhibit stepwise behaviour, a natural choice of the polynomial $D(q^{-1})$ is a simple differentiator, i.e. $D(q^{-1}) = 1 - q^{-1} = \Delta$.

$D(q^{-1}) = \Delta^{\beta}$ with $\beta > 1$ can be possible in order to prevent steady-state errors, but in general, β must be chosen as small as possible in order to avoid stability problems [Soet90]. Other choices, like $D(q^{-1}) = 1 - 2 \cos \alpha\, q^{-1} + q^{-2}$ allow to eliminate periodic disturbances of frequency $f = \alpha/2\pi$ [M'Saad96].

6.2.4 Selection of Input and Output Filters

The linear quadratic cost function of the GPC criterion in equation (6.1.3) can be modified if instead of the controller output u(t) a filtered value $u_f(t)$ is considered with

$$u_f(t) = \frac{W_{un}(q^{-1})}{W_{ud}(q^{-1})} u(t)$$

The following criterion function is obtained:

$$J_{GPC} = E\left\{ \sum_{j=sh}^{ph} \left[y(t+j) - r(t+j) \right]^2 + \lambda \sum_{j=0}^{ch} \left[D(q^{-1}) u_f(t+j-1) \right]^2 \right\} \qquad (6.2.1)$$

This controller output weighting filter $W_u(q^{-1})$ can attenuate or enhance certain frequencies in u(t). This can be used, for instance, if the process to be controlled is a mechanical system which has one (or more) resonance frequencies. In that particular case one can design the filter $W_u(q^{-1})$ so that these frequencies are more weighted than other frequencies. When one frequency in the controller output is to be attenuated, a notch filter can be designed and used as $W_u(q^{-1})$. $W_u(q^{-1})$ can also be used for attenuating high frequencies in the controller output. This can improve robustness against measurement noise in the output.

The output y(t), which is the same as the controller input, can be weighted equivalently by the following weighting filter:

$$y_f(t) = \frac{W_{yn}(q^{-1})}{W_{yd}(q^{-1})} y(t)$$

The cost function changes analogously. $W_y(q^{-1})$ can be used to penalise a large overshoot in case of set-point and load changes, and in special cases it can also be interpreted as "approximate inverse closed-loop model" [Cla87]. It has in general the same influence on the regulator behaviour and the robustness of the system as the polynomial $T(q^{-1})$ [Soet90].

6.3 Implementation Issues and Shortcomings

In order to complete the discussion of generalized predictive control the hardware requirements and some shortcomings of GPC should be mentioned.

Implementation Issues

GPC is a particularly simple computational solution of an optimal control design which does not require the solution of a Riccati equation, unlike state-space formulated optimal control problems. It is simple to derive and to implement in a computer, however, calculation time increases with an increasing control horizon. Since a small sampling period in relation to the settling time of the process entails large prediction horizons and a large controller order (if the process contains time delay), this can cause numerical problems.

The developed, non-adaptive GPC controller was of 9^{th} order and therefore very simple to implement into a microcontroller. But also the adaptive version has already successfully been implemented in 386 IBM-PC 33 MHz and micro controller HP 1000 at the Laboratoire d'Automatique de Grenoble [Dug87, M'Saad94a].

Shortcomings

The large applicability of GPC has already been pointed out (see introduction of chapter 6), the convincing performance will be shown in chapter 6.5, however, certain prerequisites are needed.

It should be kept in mind, that the selection of the design parameters, namely the starting horizon sh, the prediction horizon ph, the control horizon ch, the predictor polynomial $T(q^{-1})$ and the frequency weighting filters $W(q^{-1})$, offer a powerful tool and a large applicability, but the selection of these parameters represents an iterative and hence time-consuming procedure. A solid background in robust control theory is required, especially if the system is badly damped, if it contains one integrators or more (as it is the case in automatic steering), or if it is unstable or has unstable zeros.

The design parameters should be chosen bearing in mind the plant-model mismatch insensitivity and implementation simplicity requirements, but they also influence the tracking performance. Therefore a Partial State Reference Model Control (PSRMC) approach is recommended for tracking problems, as it was proposed by M'Saad and Sanchez 1992 [M'Saad92]. Notice that this can also be regarded as a considerable advantage, since then the tracking and regulating performance can be specified independently.

Furthermore an *a priori* knowledge about the uncertainty of the plant is needed. This implies that a previous study and a good physical insight is required, a set of identification experiments for different operating conditions is recommended. The selection of an appropriate nominal control model is an additional task, which is usually not obvious, but an iterative procedure.

6.4 GPC for Automatic Steering

Until now the structure of GPC was explained and general recommendations were given concerning the selection of design parameters. In this chapter the GPC design for the automatic steering problem is described. The design specifications and the design parameters of the robust controller are given, as well as the corresponding

robustness quantifiers, and finally simulation results with the complex, non-linear vehicle model are presented.

The performance of the robust controller, i.e. the sensitivity functions as robustness quantifiers and simulation results of the automatic steering system, is shown for the nominal situation and three mismatch conditions, namely

- FC: a fast condition corresponding to a speed of 30 m/s on a dry road (solid lines), this represents the nominal condition,
- OC: an operating condition corresponding to a speed of 18 m/s on a snow road (dashed lines),
- AC: an average condition corresponding to a speed of 15 m/s on a wet road (dash-dotted lines),
- SC: a slow condition corresponding to a speed of 5 m/s on a dry road (dotted lines).

6.4.1 Design Specifications

The design specifications for the automatic steering system have been adapted from those considered in earlier design studies [Ack95, Gul95, Smi78]. They were basically given in terms of actuator constraints and passenger comfort considerations as follows:

- The steering angle and its rate should satisfy $|\delta(t)| \le 40$ deg and $|\dot{\delta}(t)| \le 45$ deg/s.
- The displacement from the guideline must not exceed 15 cm in transient behaviour and 2 cm in steady state behaviour.
- The lateral acceleration should satisfy $|a_y(t)| \le \frac{v(t)^2}{R(t)} \pm 0.1g$ where v(t), R(t) and g denote the velocity, the curve radius and the gravity constant, respectively.

6.4.2 Design Parameters

In chapter 4.6 the considered operating domain of the automatic steering system was shown, as well as the identification results. The identification of the vehicle dynamics over this operating domain provides a set of pulse transfer functions of which a nominal control model has to be selected. This selection plays an important role in robust control design to satisfy the performance for the set of plants.

There are several ways to choose a nominal model [Ver95], one of them is to select an average model; another way is to select the worst case model, arguing that if the controller works well for this condition, the performance for any other condition will be even better. Notice that a general solution for selecting the best nominal model can not be given, and moreover, that the performance for all the operating domain can not be guaranteed only from the nominal behaviour.

Simulation results have shown that the identified model corresponding to 30 m/s on a dry road with an average mass, which represents a worst case model, is the best robust control design model. Indeed, the velocity has shown to be relatively crucial for control system robustness considerations. At this point it is worth to mention that satisfying a worst case, i.e. a high speed model, does not necessarily solve the

robustness problem: it is possible to design controllers, which work perfectly for velocities above 30 m/s, however for low speed, e.g. 5 m/s they show unacceptable oscillating behaviour.

The transfer function of the robust control design model is given by

$$P(z^{-1}) = \frac{-0.00391698 + 0.00759141 z^{-1} - 0.00373038 z^{-2}}{1 - 4.78466 z^{-1} + 9.1727 z^{-2} - 8.807713 z^{-3} + 4.235966 z^{-4} - 0.816293 z^{-5}}$$

The design parameters involved in the control objective are

sh = 1,

ch = 3,

ph = 60, and

λ = 30.

The frequency weighting filter in the controller output (which is the plant input) is selected to

$$W_u(z^{-1}) = \frac{1 - 0.4 z^{-1}}{1 + 185 z^{-1} + 0.855 z^{-2}}$$

which improves the roll-off of the Regulator*Sensitivity-function, which is shown in figure 6.4.

A frequency weighting of the plant output $W_y(t)$ is not used, i.e. $W_y(t) = 1$. Notice that the selection of appropriate frequency weighting filters is not only a time-consuming, iterative procedure, it increases also the order of the controller polynomials. It is therefore desirable, although not always possible, to achieve the required performance without frequency weighting filters.

In order to determine the polynomial $D(q^{-1})$, the block diagram of the closed-loop system of the automatic steering problem is considered, as shown in figure 6.1. $Be(q^{-1})$ and $Ae(q^{-1})$ denote the estimated nominator and denominator polynomial, which have to be multiplied by a double integrator in order to obtain the transfer function of the vehicle. u(t) denotes the plant input, which is the steering angle, and y(t) denotes the sensor displacement. The disturbance x(t) on the output can be modelled if the steady state value of the lateral acceleration during cornering is considered:

$$a_y = \frac{v^2}{R},$$

where v is the velocity of the vehicle and R is the radius of the road trajectory. The consideration of a double integrator leads to the transfer function for the output disturbance x(t) to the road curvature $\kappa(t)$:

$$x(s) = \frac{v^2}{s^2}\frac{1}{R(s)} = \frac{v^2}{s^2}\kappa(s)$$

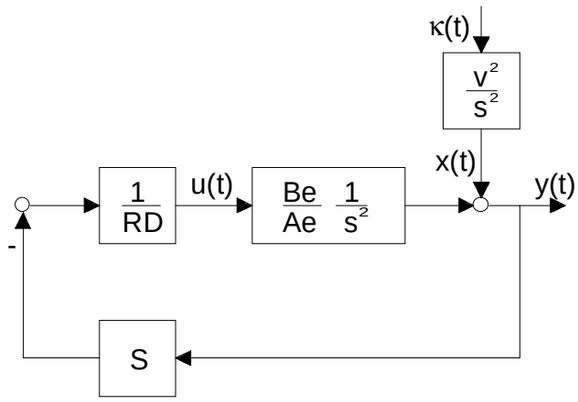


Figure 6.1: Block diagram for automatic steering with output disturbance model

The polynomial D in this block diagram has to be determined in order to obtain off-set-free performance for the closed loop. Therefore the transfer function between the road curvature κ(t) and the output y(t) in continuous time is derived:

$$Y(s) = X(s) + \frac{Be}{Ae \cdot s^2} U(s)$$

with

$$U(s) = -\frac{S}{RD} Y(s)$$

and

$$X(s) = \frac{v^2}{s^2} \kappa(s)$$

This leads to

$$Y(s) = \frac{v^2}{s^2} \kappa(s) - \frac{Be}{Ae \cdot s^2} \frac{S}{RD} Y(s) = \frac{Ae \cdot RD \cdot v^2}{Ae \cdot RD \cdot s^2 + Be \cdot S} \kappa(s)$$

The static value of y(t) for a step-wise disturbance $\kappa(s) = \frac{1}{s} \kappa_0$ can be calculated as

$$\lim_{t \to \infty} y(t) = \lim_{s \to 0} Y(s) \cdot s = \frac{Ae \cdot RD \cdot v^2}{Ae \cdot RD \cdot s^2 + Be \cdot S} \cdot \frac{1}{s} \kappa_0 \cdot s = \frac{Ae \cdot RD \cdot v^2}{Ae \cdot RD \cdot s^2 + Be \cdot S} \kappa_0$$

which is different from zero for D ≠ s. This implies that, in case of automatic steering, the D polynomial in discrete time should be chosen as a simple differentiator $D(q^{-1}) = 1 - q^{-1}$ in order to ensure offset-free performance.

The predictor poles of the $T(q^{-1})$ polynomial have been obtained by damping the poles of the control model. They were set to 0.5, 0.6, 0.7, 0.8, 0.9, and 0.95.

6.4.3 The Robustness Quantifiers

The sensitivity functions as described in chapter 5.1.1 are shaped using an iterative procedure. Figure 6.2 shows the output sensitivity functions for the particular four driving conditions. It can be seen that, for the high speed condition (solid), for the snow condition (dashed) and for the average condition (dash-dotted) good sensitivity functions have been achieved (less than 6 dB), only the low speed condition (dotted) shows two high peak values, one of them in the very low frequency region, the other at about 6 Hz.

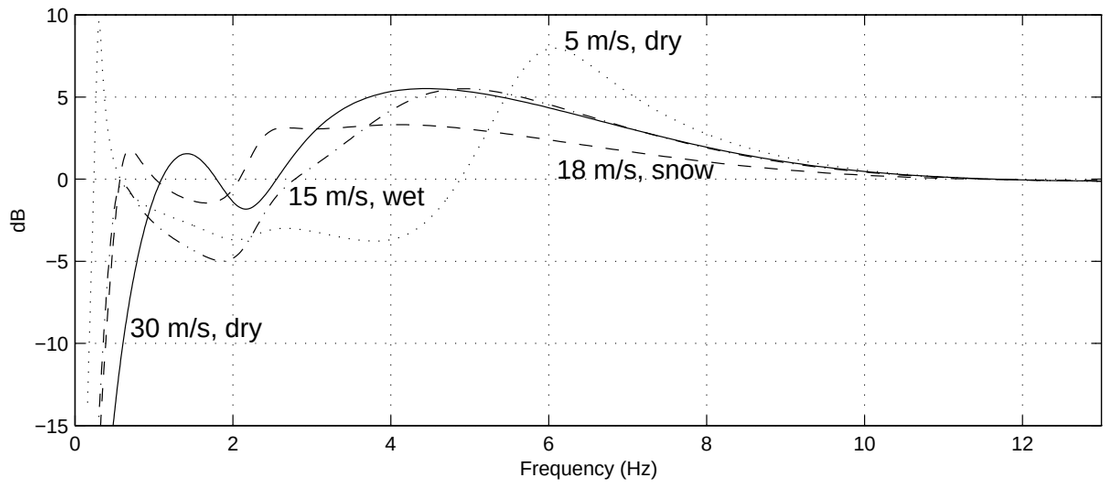


Figure 6.2: Sensitivity functions

Figure 6.3 shows the complementary sensitivity functions for the four presented operating conditions. In chapter 5 it was mentioned, that this sensitivity function characterises the sensitivity to measurement noise, why it is also called “noise sensitivity function”. More specifically, it describes the influence of measurement noise in the input ($b_u(t)$) and output ($b_y(t)$) on the plant input u(t) and the plant output y(t), respectively. It can be seen in figure 6.3, that for all operating conditions, a good attenuation in the high frequency region, which is important in this case, has been achieved.

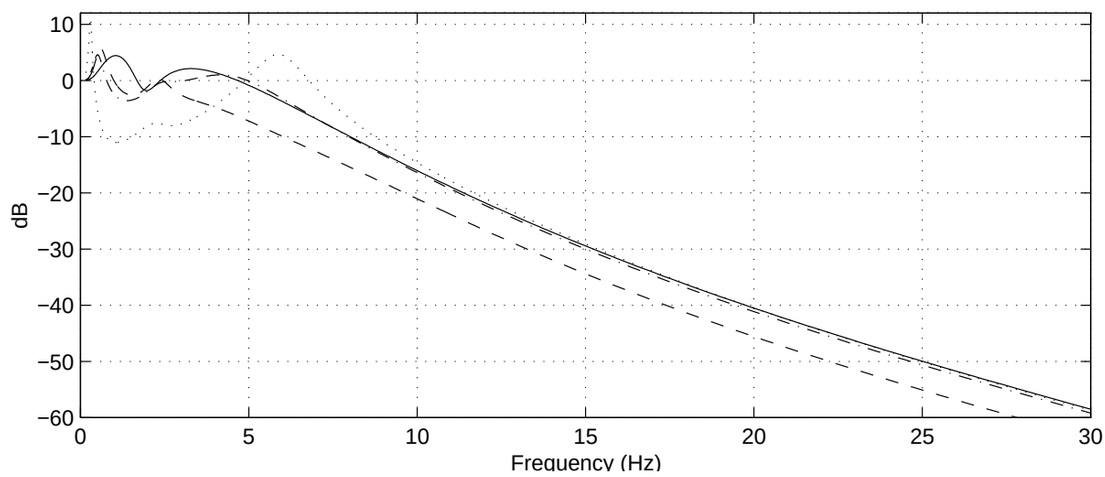


Figure 6.3: Complementary sensitivity functions

The sensitivity function “Regulator*Output Sensitivity” describes the influence of the output disturbance and output measurement noise on the input, as it can be seen in

figure 5.4. In figure 6.4 it can be seen that, in the frequency band between 3 Hz and 32 Hz, the disturbances and the measurement noise are amplified by the controller. This indicates possible difficulties if the controller is actually implemented in a test vehicle: measurement noise of the displacement sensor induces oscillations of the plant input, which is the steering angle. An oscillating steering angle entails oscillations of the lateral acceleration, which degrades passenger ride comfort. This problem was already reported from previous studies [Peng90]. Therefore a further improvement by modification of the frequency weighting filters may be required.

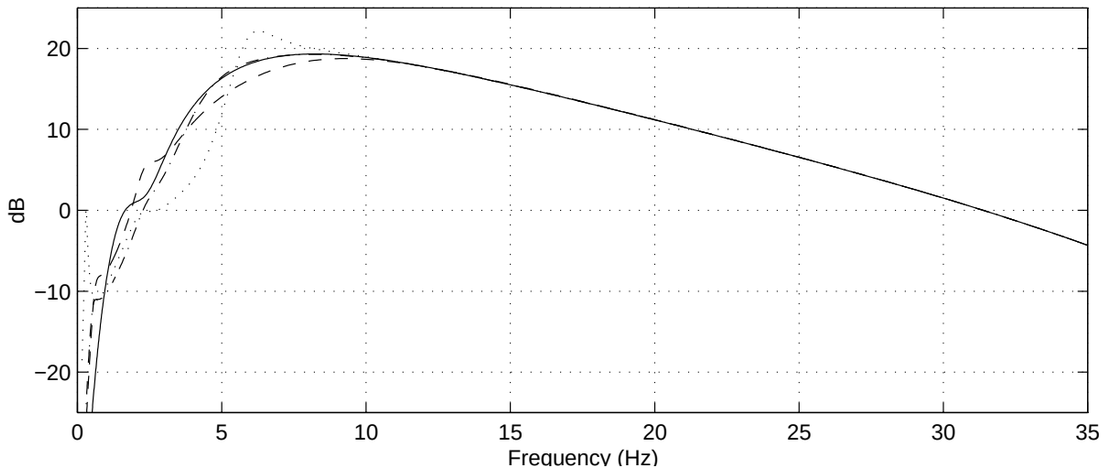


Figure 6.4: Regulator*Sensitivity functions

Figure 6.5 shows the sensitivity functions “Plant*Output Sensitivity”, which indicate how detrimental actuator effects influence the plant output. It can be seen that for all conditions a good attenuation in the important high frequency region has been obtained.

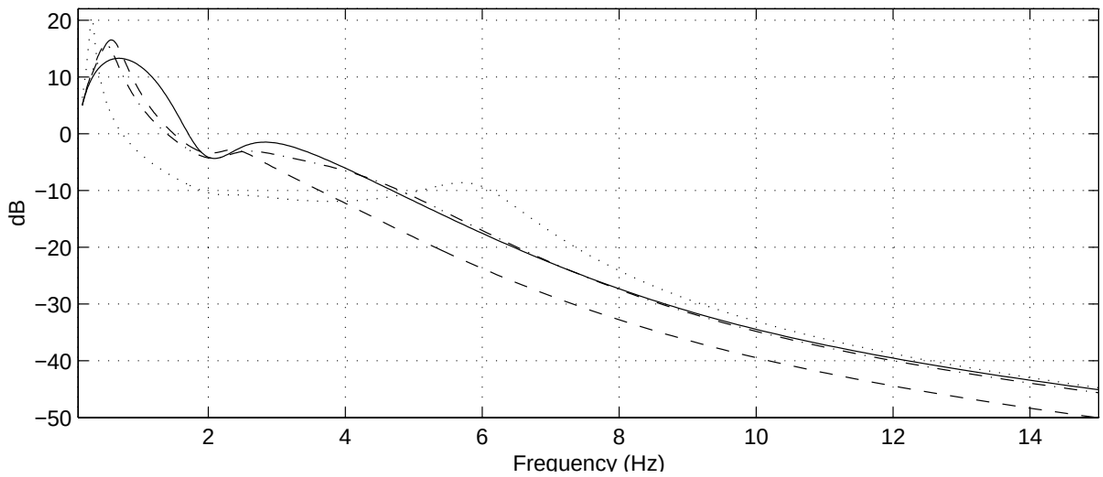


Figure 6.5: Plant model*Sensitivity functions

Table 6.1 compares the modulus margins for the different driving conditions. It can be seen, that for all except the slow condition (SC) the values remain smaller than the desired -6 dB. The modulus margin as well as the attenuation bandwidths can also be seen in the preceding figures, notice, that the plots of the sensitivity functions provide a much more detailed information. However, the information of the

delay margins is not contained in the sensitivity function plots, why it is also shown in table 6.1.

Conditions	Modulus Margins (dB)	Delay Margins (sec)	Attenuation Band-widths (Hz)
FC	-5.51	0.59	1.10
OC	-3.32	1.71	0.57
AC	-5.50	0.61	2.80
SC	-9.61	0.34	0.24

Table 6.1: Robustness margins and attenuation bandwidths

6.5 Simulation Results

Three reference driving manoeuvres have been considered to evaluate the performance of the automatic steering system:

- a transition from a straight line into a circular one with a radius of 400 m,
- a wind gust disturbance at 20 m/s in lateral direction, and
- a transition from manual mode to automatic steering.

6.5.1 Entering the Curved Test Path

Figures 6.6, 6.7, 6.8, and 6.9 show the lateral sensor displacements, steering angles, steering angle rates and lateral accelerations for entering the curved test path. As already mentioned, the

- *solid* lines correspond to *30 m/s* on a *dry* road, the
- *dashed* lines correspond to *18 m/s* on a *snow* road,
- *dash-dotted* correspond to *15 m/s* on a *wet* road, and the
- *dotted* lines correspond to *5 m/s* on a *dry* road.

The different steady state values of the steering angles in figure 6.7 are due to the understeering behaviour of the BMW520i. Notice that the actuator constraints as well as the passenger comfort specifications have been satisfied with the proposed robust controller.

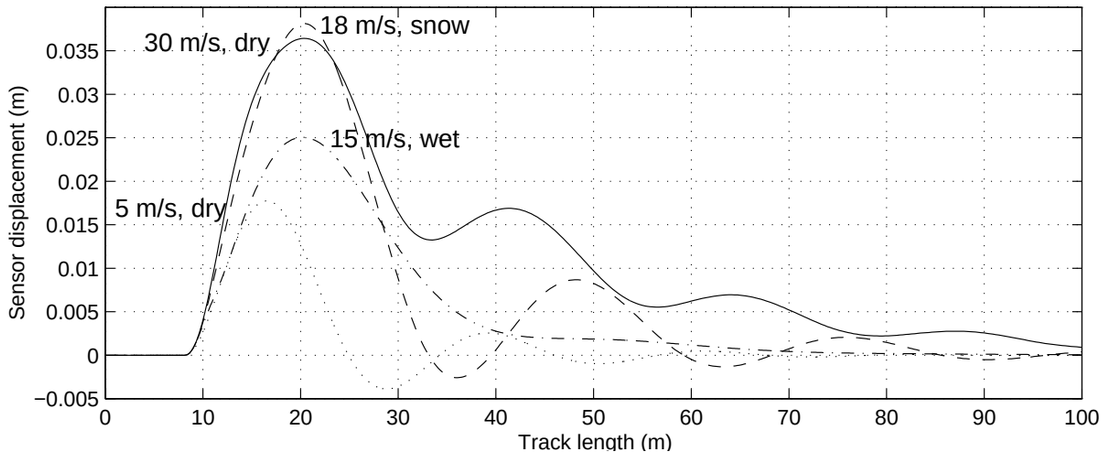


Figure 6.6: Lateral displacements for entering the curved path

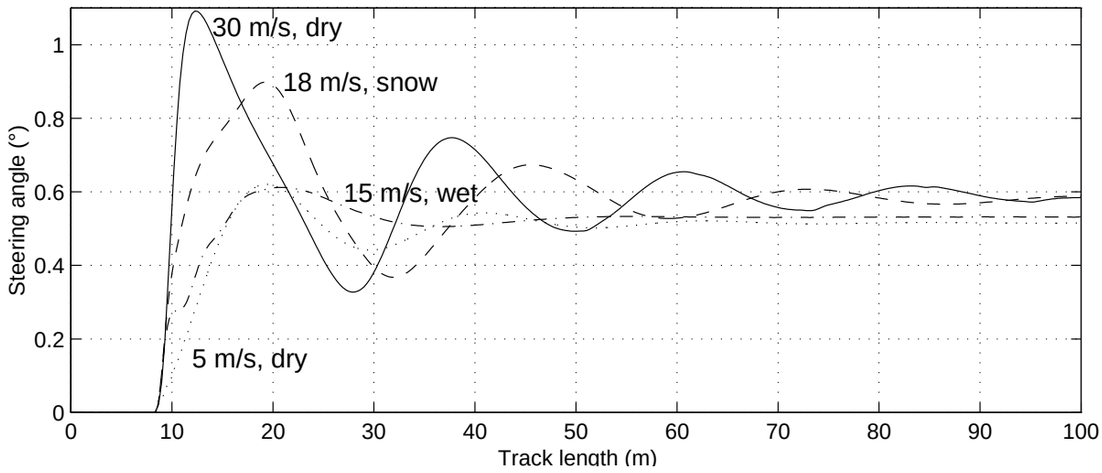


Figure 6.7: Steering angles for entering the curved path

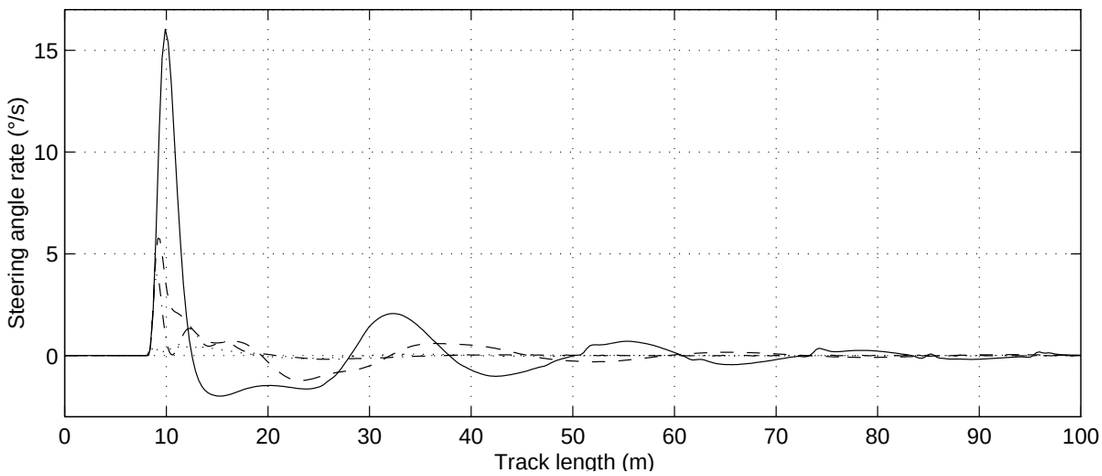


Figure 6.8: Steering angle rates for entering the curved path

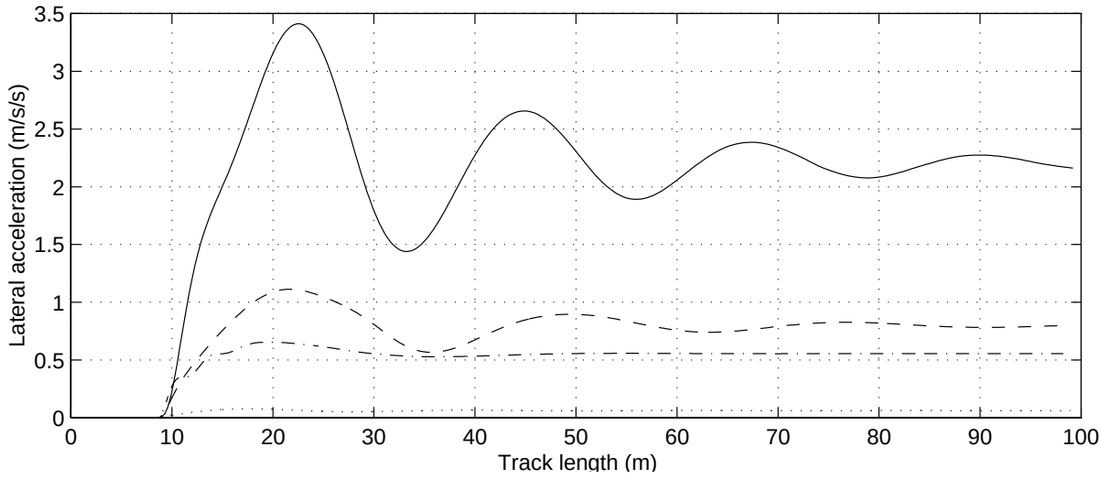


Figure 6.9: Lateral accelerations for entering the curved path

6.5.2 Side Wind Forces

Figures 6.10, 6.11, 6.12 and 6.13 show the lateral sensor displacements, steering angles, steering angle rates and lateral accelerations of the robust controller in presence of a heavy step-like wind gust of 20 m/s in lateral direction. Such a condition can occur when leaving a tunnel or behind a bridge, when the vehicle is suddenly hidden by a lateral wind gust. Notice that the lateral displacement is not significantly affected, however in the low speed case (dotted lines) an oscillating behaviour of the steering angle deteriorates the passenger comfort, which can be seen in figure 6.13.

The frequency of this oscillation is approximately 6 Hz, which corresponds to the peak of the sensitivity function for this condition, which was shown in figure 6.2 (dotted line). This underlines the qualification of the sensitivity functions as performance and robustness quantifier.

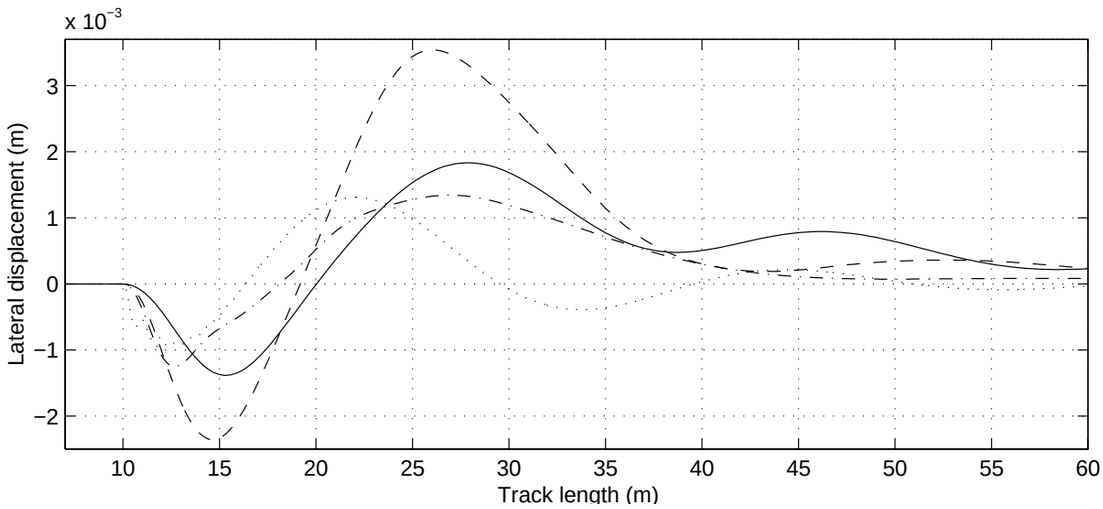


Figure 6.10: Lateral displacements for wind disturbance

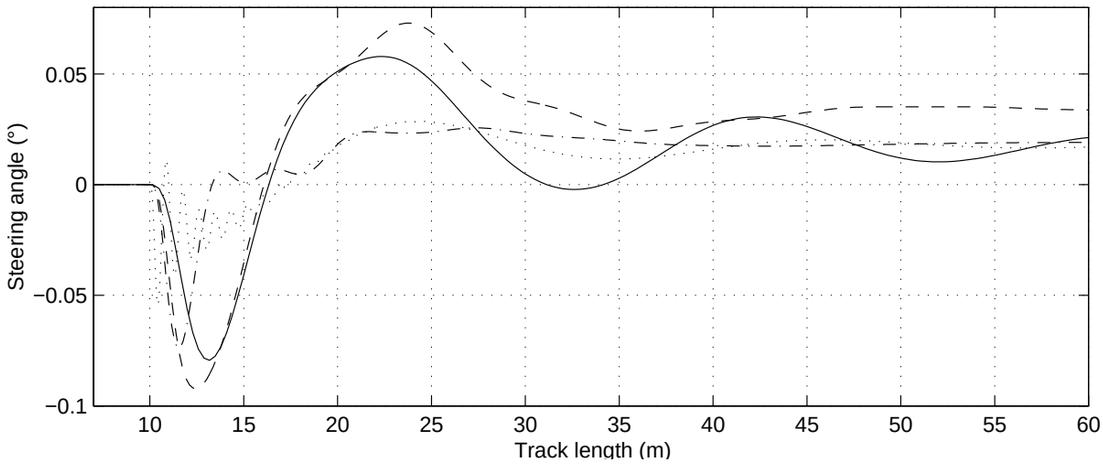


Figure 6.11: Steering angles for wind disturbance

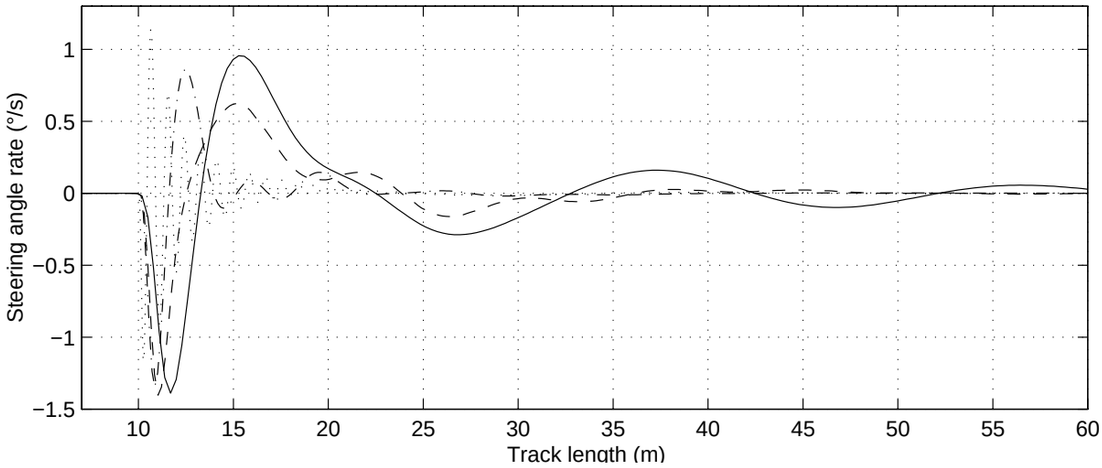


Figure 6.12: Steering angle rates for wind disturbance

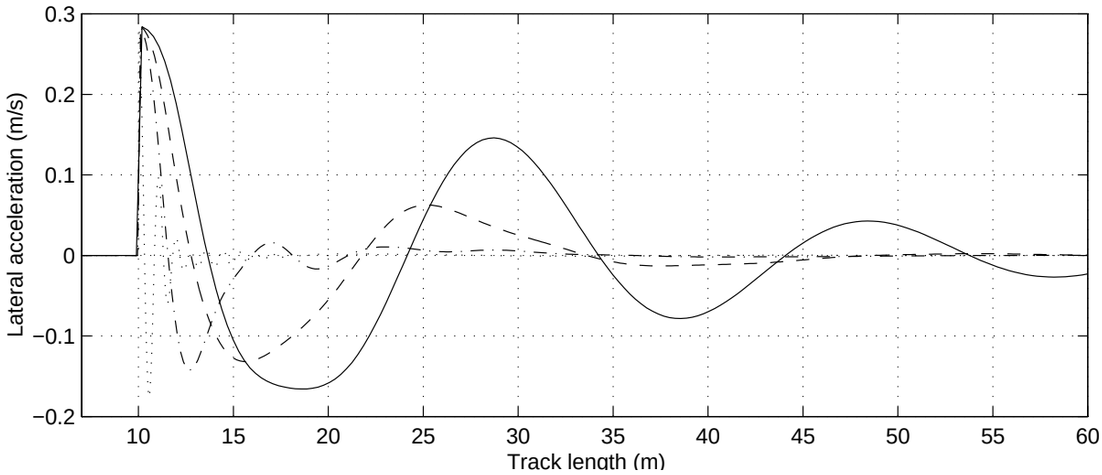


Figure 6.13: Lateral accelerations for wind disturbance

6.5.3 Transition from Manual to Automatic Steering

Figures 6.14, 6.15, 6.16 and 6.17 show the lateral sensor displacements, steering angles, steering angle rates and lateral accelerations of the robust controller for a transition from manual to automatic mode. The disturbances are well rejected, however the steering angle rate saturates.

This seems to be a typical behaviour of a linear controller [Hunt96], however, this problem can be treated by transforming this regulation problem into a tracking problem: during manual mode the present sensor displacement can be imposed as a desired reference value; during transition from manual to automatic mode a suitable tracking sequence can be imposed, which leads to a smooth transition to a zero value of the sensor displacement.

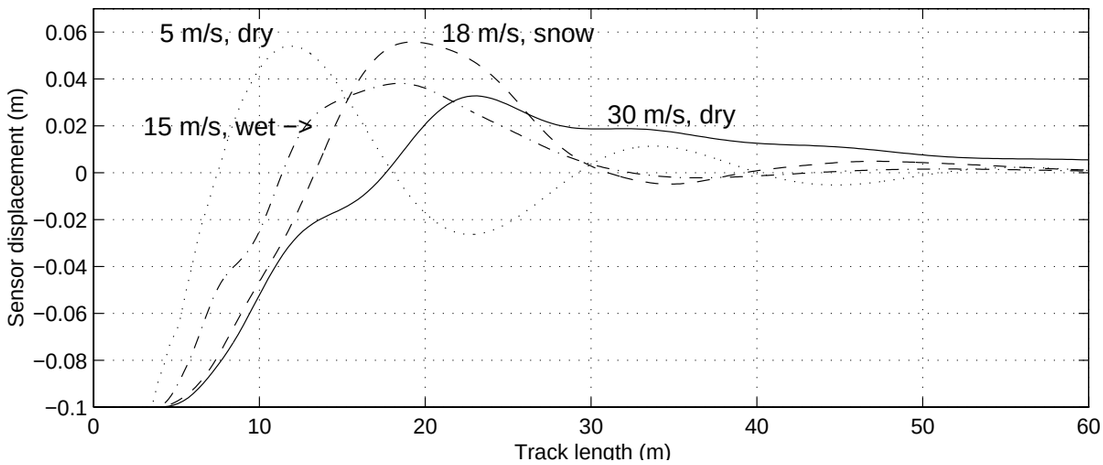


Figure 6.14: Lateral displacements for transition from manual to automatic steering

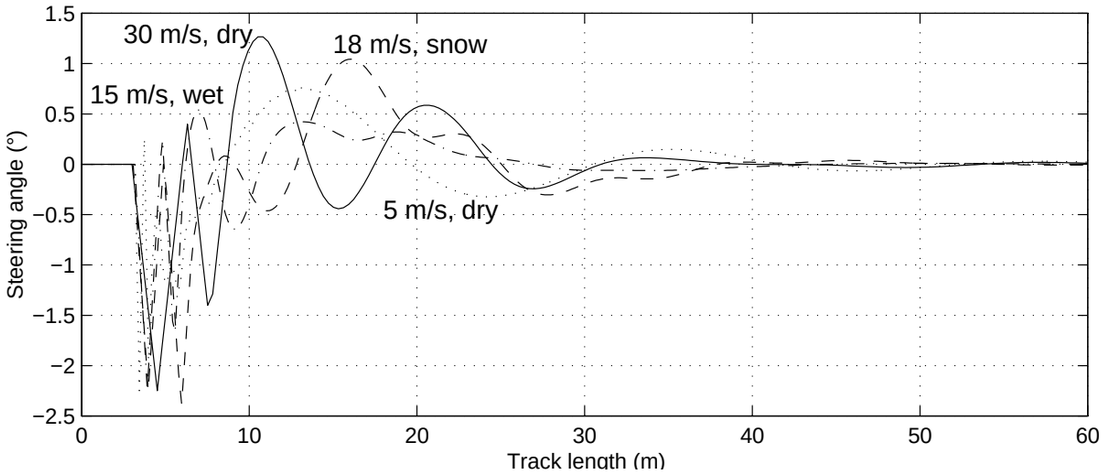


Figure 6.15: Steering angles for transition from manual to automatic steering

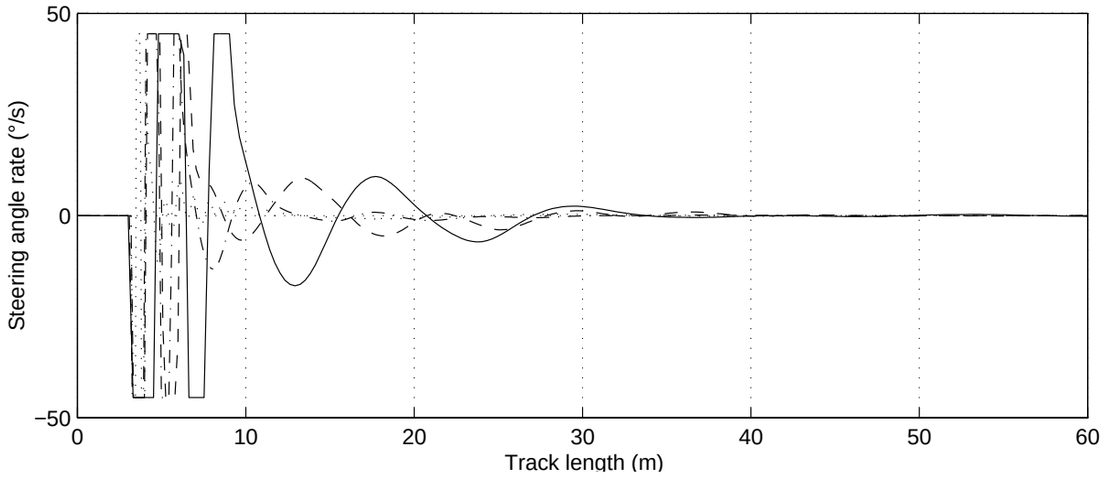


Figure 6.16: Steering angle rates for transition from manual to automatic steering

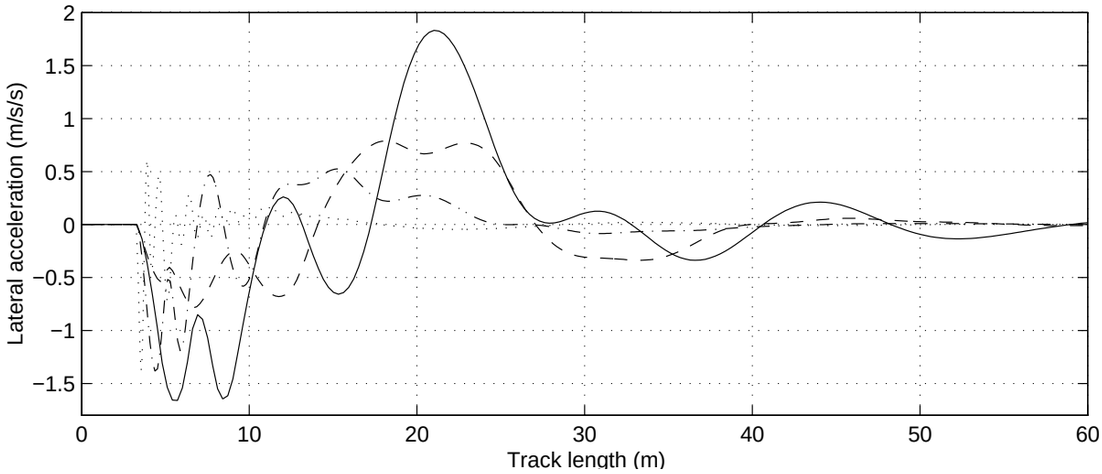


Figure 6.17: Lateral accelerations for transition from manual to automatic steering

7 Feed-forward

From previous studies it is known, that the behaviour of a human driver during line-following manoeuvres is basically of a feed-forward characteristic [Horn86]. When entering a curve, a driver turns immediately the steering wheel (feed-forward behaviour), according the curve and based on experience. Only a relatively small percentage of his reaction is based on feed-back information, e.g. the actual lateral displacement from a desired trajectory.

From a control point of view, it is generally recommended to compensate disturbances which are measurable by using a feed-forward structure, and to compensate by a feed-back law only those disturbances, which are not measurable [Iser87].

This motivates the improvement of the controller performance using the curvature of the coming road as preview information. This preview information is certainly available if the steering system is used in a vehicle simulation, if the steering system is implemented in a test vehicle this information can be obtained by a visual sensor. In future Integrated Highway Systems (IHS) the coming road curvature could be binary encoded into the road infrastructure.

The feed-forward controller can generate the steady-state or preview steering command corresponding to the curvature $\kappa(t) = \frac{1}{R(t)}$ of the road, where R(t) denotes the curve radius. The resulting structure is shown in figure 7.1.

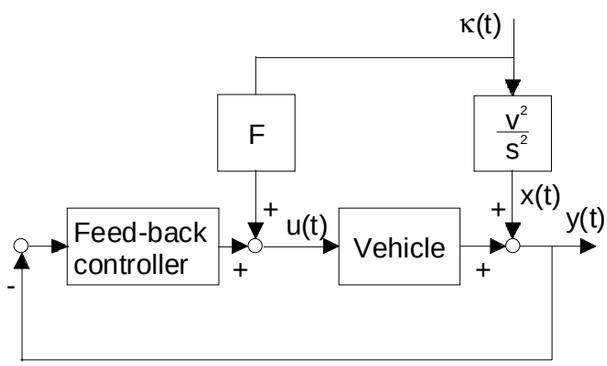


Figure 7.1: Block diagram for automatic steering with output disturbance model and feed-forward controller

$\kappa(t)$ denotes the road curvature, F represents the transfer function of the feed-forward controller, u(t), y(t) and x(t) denote respectively the steering angle, the lateral sensor displacement and the output disturbance. The output disturbance model is the square of the velocity v, multiplied by a double integrator, which was derived in chapter 6.4.2.

The steady-state steering angle δ_{static} can be as calculated for the linear single-track model as

$$\delta_{static} = \left(l + \frac{m v^2 \left(C_r l_r - C_f l_f \right)}{l C_r C_f} \right) \frac{1}{R} = F \frac{1}{R} = F \kappa \qquad (7.1.1)$$

which shows that the transfer function F is related to the velocity v and the front and rear cornering stiffnesses C_f and C_r [Mit90]. A feed-forward controller which is based on this equation suffers from the fact that it needs not only to measure the velocity v of the vehicle, which can be easily done, it requires also an estimation of the cornering stiffness C_f and C_r as it was pointed out in [Peng90].

In the presented study, it will be shown that, even with a fixed controller gain F, a considerable performance improvement can be obtained. In figure 6.7 the steering angles for entering a curve with radius 400 m are shown. It can be seen, that the steady-state values δ_{static} vary between approximately 0.5° and 0.6°. In radians this corresponds to 0.0087 and 0.0105, respectively. Since $F = \delta_{static} \cdot R = \delta_{static} \cdot 400$ m, a gain F of 3.5 has been used.

The simulation results are shown in figure 7.2. Notice that a reduction of the maximum overshoot of at least 50 % had been obtained (for 5 m/s on dry road), for a condition of 18 m/s on a snow road the overshoot decreased to a value of 18 % (from 3.8 cm to 0.7 cm).

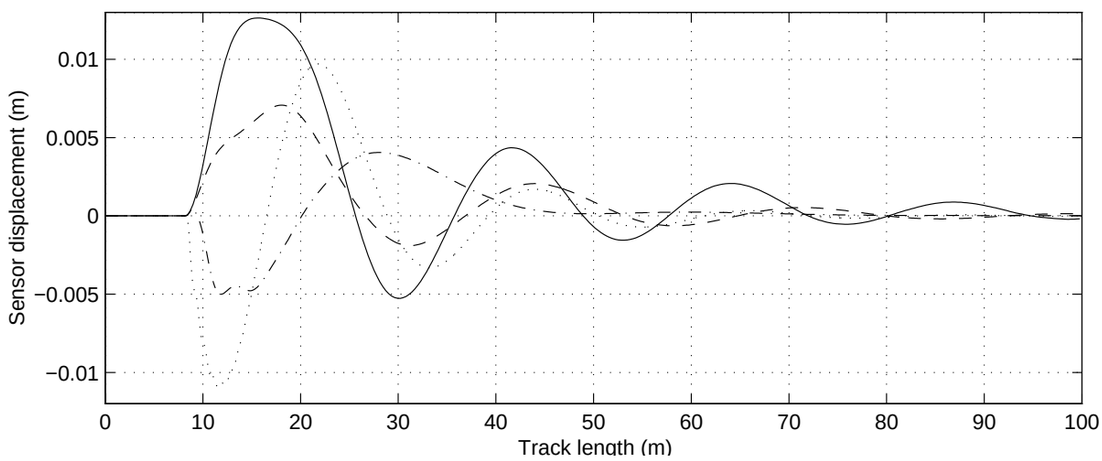


Figure 7.2: Lateral displacements

It can also be seen that the sensor displacement for low speed conditions changes sign and starts with a negative overshoot. This is due to the fact that the steering angle at the wheels is manipulated by the feed-forward controller not when the front wheels enter the curve, but when the sensor, which is mounted at the front end of the vehicle, enters the curve.

Figure 7.3 compares the robust controller with and without feed-forward for the nominal condition (v = 30 m/s on a dry road) for entering the test curve. The dotted line in the lower diagram shows the contribution of the feed-forward part to the steering angle.

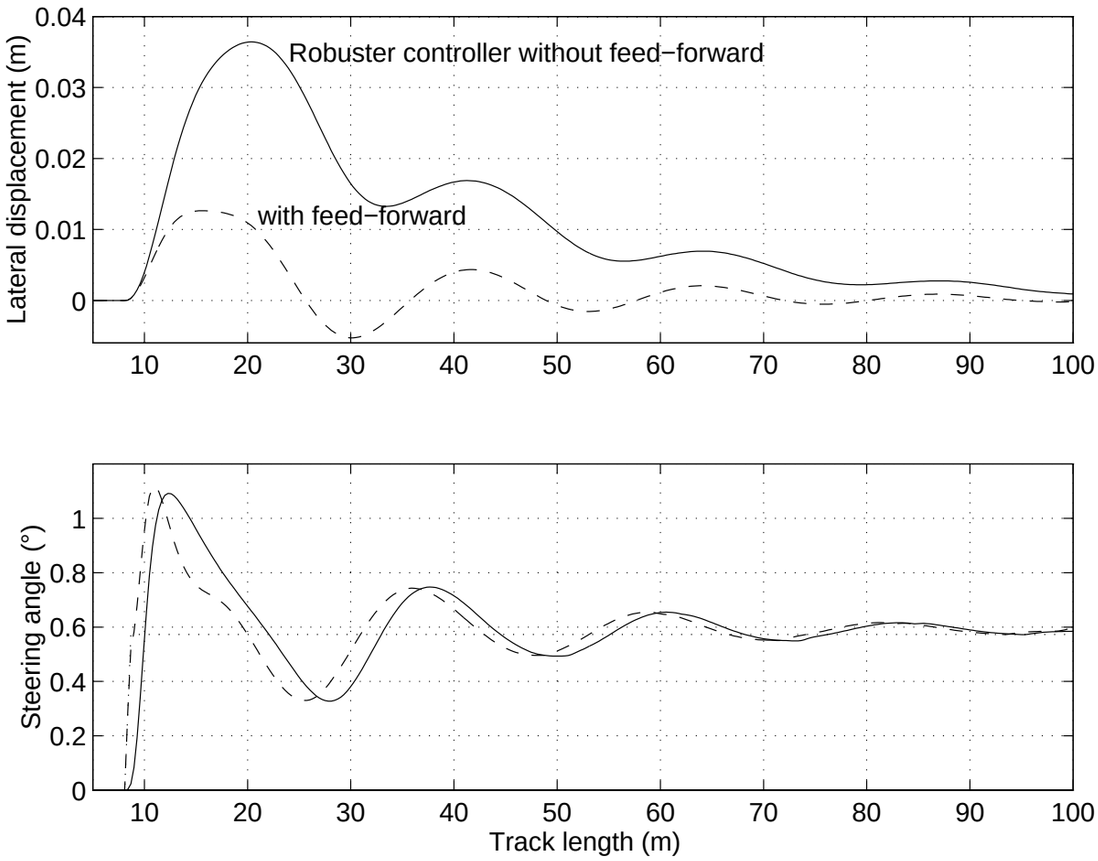


Figure 7.3: Lateral displacements for v = 30 m/s on a dry road for entering the test curve, with and without feed-forward

8 Adaptive Control

When robust control design does not allow to obtain a single linear controller which gives acceptable performances for all operating conditions because the dynamic characteristics of the plant vary too much, one has to consider the “adaptation” of the controller.

Basically there are two types of adaptive controllers: *Direct Adaptive Controllers*, which are adapted without an explicit identification of the process, and *Indirect Adaptive Controllers*. For the automatic steering problem both types of adaptation methods have been implemented: a direct adaptive controller, more particularly an *open-loop adaptation*, and an indirect adaptive controller which is also called a *self-tuning* controller. Both methods are explained in the following and simulation results are presented.

8.1 Open loop Adaptation

For adaptive controllers a variety of adaptation methods have been developed. The simplest case represents the *open loop adaptation*, which can be used if the changing behaviour of the process can be measured using appropriate signals, and if it is known, in which way the controller has to be modified. This approach is also called *gain scheduling* because the system was originally used to accommodate changes only in process gain [Åst84, Iser87]. The structure of this adaptation is shown in figure 8.1.

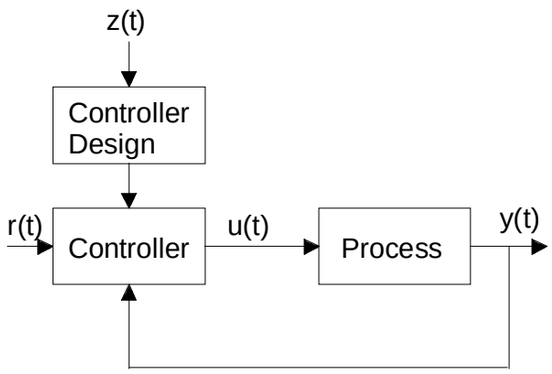


Figure 8.1: Block diagram for open loop adaptation

r(t) and z(t) denote respectively the reference sequence and a measurable signal, the latter serves as an indicator for the plant characteristics. Distinctive for open loop adaptation is that no feedback of control loop signals on the controller design is used. This represents also a drawback, since no feedback loop can compensate for an incorrect schedule. Gain scheduling can thus be viewed as an extension of feed-forward compensation. An advantage of a feed-forward controller is that the parameters can be changed very quickly in response to process changes [Åst84], provided that the process changes can be easily measured. Therefore additional sensors are required. This is in contrast to an adaptation in closed loop, where no additional sensors are needed, but additional calculations have to be carried out instead.

8 Adaptive Control

When robust control design does not allow to obtain a single linear controller which gives acceptable performances for all operating conditions because the dynamic characteristics of the plant vary too much, one has to consider the “adaptation” of the controller.

Basically there are two types of adaptive controllers: *Direct Adaptive Controllers*, which are adapted without an explicit identification of the process, and *Indirect Adaptive Controllers*. For the automatic steering problem both types of adaptation methods have been implemented: a direct adaptive controller, more particularly an *open-loop adaptation*, and an indirect adaptive controller which is also called a *self-tuning* controller. Both methods are explained in the following and simulation results are presented.

8.1 Open loop Adaptation

For adaptive controllers a variety of adaptation methods have been developed. The simplest case represents the *open loop adaptation*, which can be used if the changing behaviour of the process can be measured using appropriate signals, and if it is known, in which way the controller has to be modified. This approach is also called *gain scheduling* because the system was originally used to accommodate changes only in process gain [Åst84, Iser87]. The structure of this adaptation is shown in figure 8.1.

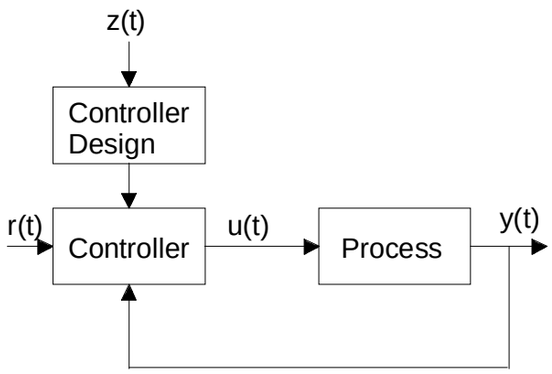


Figure 8.1: Block diagram for open loop adaptation

r(t) and z(t) denote respectively the reference sequence and a measurable signal, the latter serves as an indicator for the plant characteristics. Distinctive for open loop adaptation is that no feedback of control loop signals on the controller design is used. This represents also a drawback, since no feedback loop can compensate for an incorrect schedule. Gain scheduling can thus be viewed as an extension of feed-forward compensation. An advantage of a feed-forward controller is that the parameters can be changed very quickly in response to process changes [Åst84], provided that the process changes can be easily measured. Therefore additional sensors are required. This is in contrast to an adaptation in closed loop, where no additional sensors are needed, but additional calculations have to be carried out instead.

8.1.1 Multi-Model Approach

The controller design in open-loop adaptation can be replaced by a controller selection, i.e. the operating domain is divided in a number of operating intervals for each of them a corresponding controller will be used. This is also called *multi-model approach*. In the case of automatic steering a suitable indicator signal z(t) for the behaviour of the vehicle is the velocity. An open loop adaptation can be simply carried out by switching between different controllers according to the velocity of the vehicle. In general this kind of adaptation is a very useful technique for reducing the effects of parameter variations and of considerable practical importance.

8.1.2 Simulation Results

Figure 8.2 shows the simulation result for entering the curved test path at a velocity of 5 m/s on a dry road. The dotted line shows the simulation result of the robust controller which was already shown in chapter 6. The solid line shows the simulation result of a controller, which has been obtained using the same control parameters but a control model, which corresponds this actual condition, i.e. 5 m/s on a dry road.

The simulation result can be considerably improved, if instead the robust controller, a controller is used, which was developed just for the corresponding operating condition. In figure 8.3 it can be seen that the problems concerning oscillations of the steering angle for this conditions are solved.

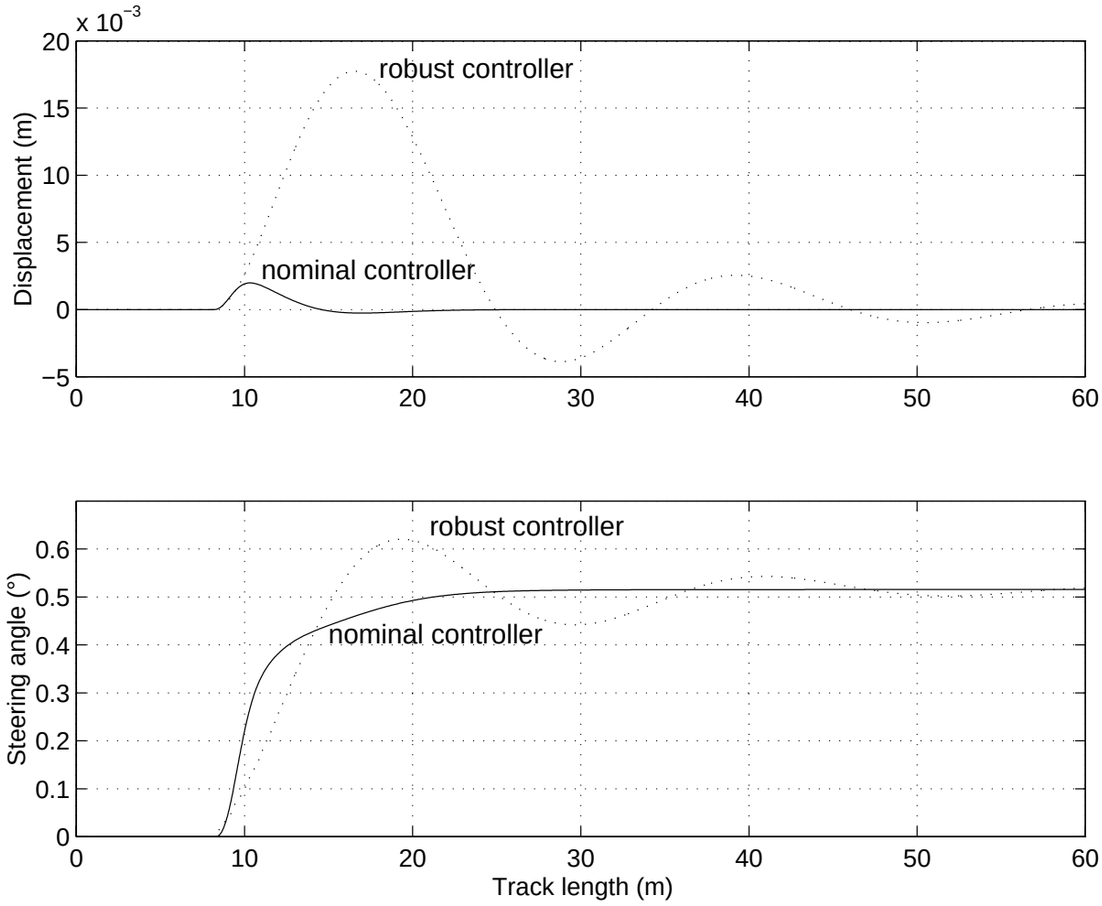


Figure 8.2: Simulation result: Robust and nominal controller for entering the test curve: sensor displacements and steering angles

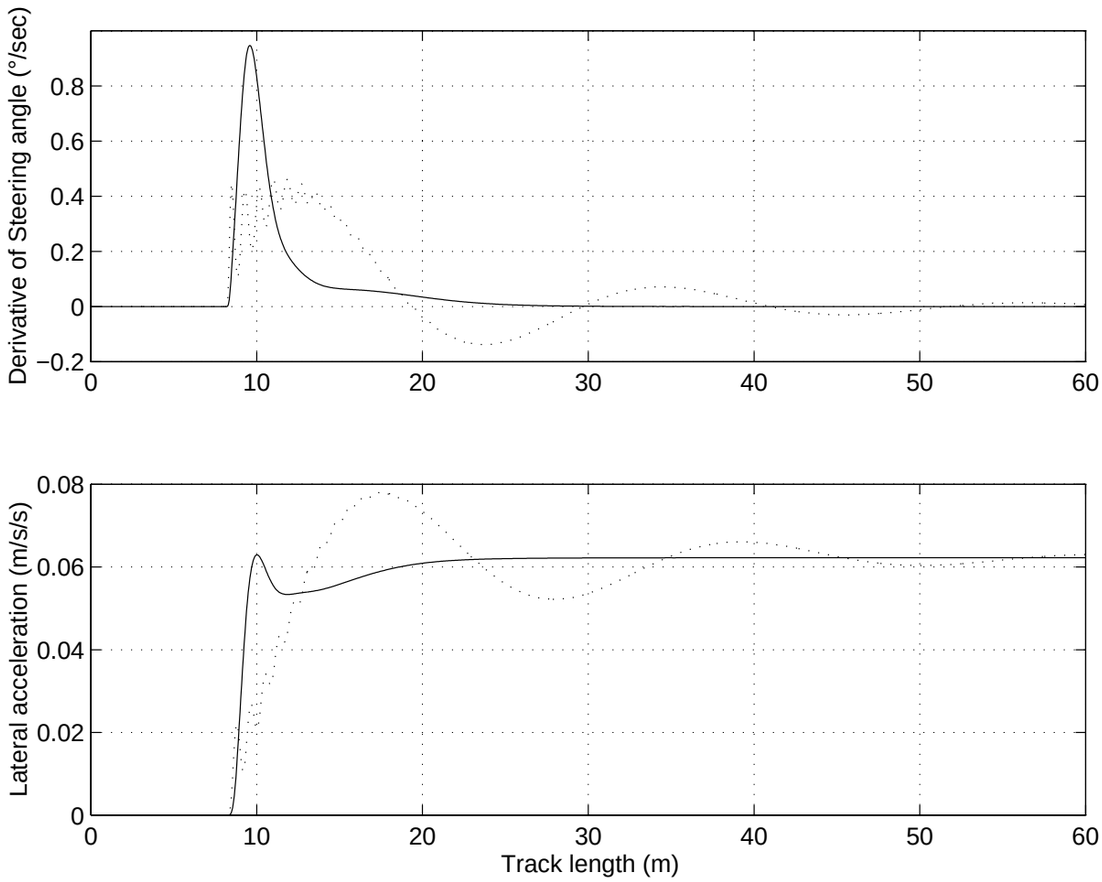


Figure 8.3: Simulation result: Robust and nominal controllers for entering the test curve: steering angle rates and lateral accelerations

Due to the decreasing robustness requirements for each local regulator the global operating domain can be increased. The maximal velocity for which a regulator has been obtained is 37 m/s on dry and wet roads. However, this regulator showed unacceptable oscillating behaviour for very low frequencies. Therefore for a suitable multi-model approach at least 2 regulators are required, 3 regulators are sufficient.

Notice, that during transition from one regulator to another the stability of the system can not be theoretically guaranteed. This requires a cautious switching and that in the boundary region between two controller operating domains both regulators are sufficiently stable. A hysteresis characteristic can reduce the transitions when the system operates in the cross-over area.

As already mentioned, this adaptation is of practical importance, its implementation does not pose particular problems. Therefore no further simulation results are shown in order to continue to the more challenging self-tuning control.

8.2 Adaptation in Closed-Loop: Self-Tuning Control

In the following chapter the structure of the implemented self-tuning regulator is explained, simulation results are presented, and the effect of signal normalisation and adaptation freezing as it was explained in chapter 3 is shown. The latter features of the estimation procedure are of fundamental importance for the applicability of a self-tuning controller for real-life control problems.

8.2.1 The Structure of a Self-Tuning Regulator

The so-called Self-Tuning Regulator (STR) represents an indirect adaptive controller since it identifies the parameters of the process explicitly. It is therefore also called *Model Identification Adaptive System* (MIAS). A block diagram of this controller is shown in figure 8.4. It combines control design and recursive estimation method.

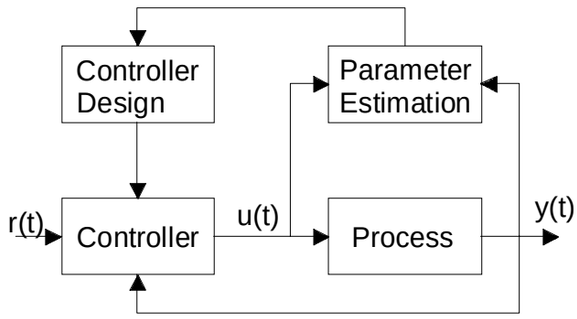


Figure 8.4: Block diagram of a self-tuning regulator

The system can be thought of as being composed of two loops. The inner loop consists of the process and an ordinary linear feedback regulator. The parameters of the regulator are adjusted on-line by the outer loop, which is composed of a recursive parameter estimator and a design calculation. The estimated model is used to redesign the controller during operation, assuming that the model is identical to the process. This is called the *certainty equivalence principle*.

This kind of algorithm is very attractive since it offers the capability of automatic initial tuning of a control system or of re-tuning of the system if the plant parameters change subsequently. Furthermore it is very flexible with respect to the design method. Virtually any design technique can be accommodated. Self-tuners based on phase and amplitude margins, pole-placement, minimum-variance control, and LQG-control have been considered. Many different parameter-estimation schemes may be used, e.g. stochastic approximation, least squares, extended and generalised least squares, instrumental variables, extended Kalman filtering, and maximum-likelihood [Åst84]. It can conveniently be implemented using microprocessors.

A drawback of adaptive control in general is that the regulator will be more complex than constant-gain regulators, the closed-loop systems becomes non-linear and time-varying [Åst84]. A critical point of indirect adaptive controllers is the robustness of the underlying linear control law concerning inaccuracy of the identified plant model. This accuracy has a double origin: firstly, even under ideal conditions and excitations the parameterized model can usually referred to as undermodelling. Secondly, the estimated parameters do not coincide with the best possible parame-

ters within the chosen model structure, either because they have not yet converged, or because of an insufficiently excited plant. The explicit self-tuner converges only if the parameter estimates converge. This requires that the model structure used in the estimation is correct and that the input signal is sufficiently rich in frequencies. Since in real-life applications this is not necessarily true, a sophisticated estimation algorithm is required.

In the automatic steering problem the reference sequence r(t) is always zero since the vehicle should track the guideline without deviation. This represents an extremely challenging application for the self-tuning regulator, since the system is only excited by an external output disturbance, which can be neither influenced, nor is it known in advance.

In the presented adaptive version of the automatic steering controller, the estimation algorithm contains not only features like filtering, forgetting factor sequences and incorporation of prior knowledge, but also adaptation gain regularisation, signal normalisation and adaptation freezing, as explained in chapter 3. In order to avoid pole-zero-cancellations at lower frequencies (see chapter 4.4.2) the estimated model is only of 4^{th} order. Since the double integrator characteristic of the plant is incorporated as prior knowledge, this corresponds to a off-line estimated model of 2^{nd} order.

The fact, that for velocities above 20 m/s a control model of 5^{th} order had to be used to describe the vehicle dynamics, explains, why the proposed adaptive controller just works well below this velocity. The limit for stability was 28 m/s, however, above 20 m/s the control performance deteriorated.

The applied controller design was basically the same GPC controller as described in chapter 6, only one observer pole has been cancelled due to the reduced order of the estimated model. The on-line controller design as well as the estimation in closed-loop was carried out using the corresponding modules of the software-package SIMART, which were exported and implemented in Matlab/Simulink.

8.2.2 Simulation Results

Before considering the simulation results we should keep in mind that the estimated parameters are only adapted when the system is sufficiently excited. For the automatic steering system this is usually the case when entering a curve or a straight line, but only at the beginning of each section, when there are sufficiently changes of the sensor displacement and the steering angle. When the sensor displacement remains zero and no changes of sensor displacement and steering angle occur, the parameter adaptation is frozen.

Figure 8.5 shows a simulation result of the adaptive predictive controller. The two upper diagrams show the experimental specification in terms of road trajectory and velocity. The initial velocity of the vehicle is 5 m/s. The vehicle model of the estimator, which is initially used to update the GPC controller, corresponds to a velocity of 20 m/s and has been obtained by a previous off-line identification procedure.

After 5 meters the vehicle enters the first curve of a radius of 400 m. The maximum overshoot of the sensor displacement is approximately 2 centimetres. Now the wrong initial model parameters are adapted to the parameters corresponding to this low speed condition, and after 30 meters, when entering the straight path, the vehicle follows the desired path with almost zero deviation. The reduction of the maximum

overshoot when entering the second curve after 65 meters shows that the adaptation performs adequately.

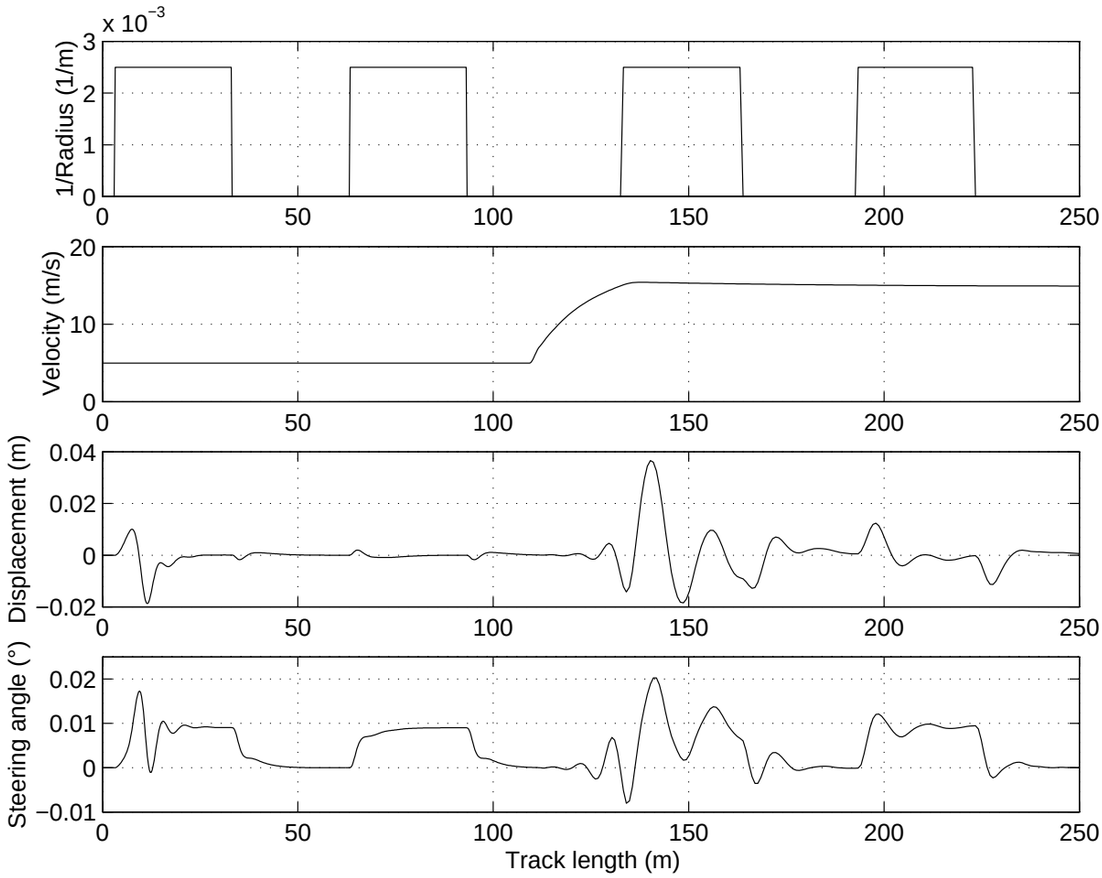


Figure 8.5: Simulation result of adaptive GPC controller for a changing operating condition (increasing speed)

After 110 meters the vehicle in figure 8.5 accelerates on the straight, which is shown in the first and the second diagrams. Since on the straight line the vehicle dynamics are usually not sufficiently excited, the estimated parameters are frozen and the controller is not adapted to the changing velocity. In addition, the road conditions are changed from a dry to a wet road. Therefore, when entering the third curve after 130 meters, there is a considerable overshoot of 4 centimetres. However, this overshoot represents an excitation of the system, the parameters adapt to this condition, and after 190 meters, when for the same condition the curved section is entered again, the overshoot decreases considerably and a remarkable control performance is obtained.

If this result is compared with the robust controller (see figure 6.6, the dotted (5 m/s on dry road) and dash-dotted (15 m/s on wet road) lines) it can be seen that the overshoot in the first curve (2 cm) is greater than the overshoot of the robust controller for this condition (1.8 cm). However, after the parameter have been adapted the sensor displacement of the adaptive controller remains almost zero and the control performance of the adaptive controller is by far better.

The same observation can be made for the second condition, where the initial overshoot of the adaptive controller is almost 4 cm in comparison to the robust controller, where the overshoot is 2.5 cm. However, after the adaptation, when the vehicle enters the second curved section, the overshoot of the adaptive controller decreases to approximately 1 cm, which exceeds the performance of the robust controller considerably.

Figure 8.6 shows an equivalent simulation result, in this case the vehicle slows down after the second curve. During the first sections, the road conditions correspond to a wet road, after the deceleration the vehicle enters a dry road. For both, the initial and the final condition the adaptation performed adequately and the control performance improves between the first and the second curve of each condition.

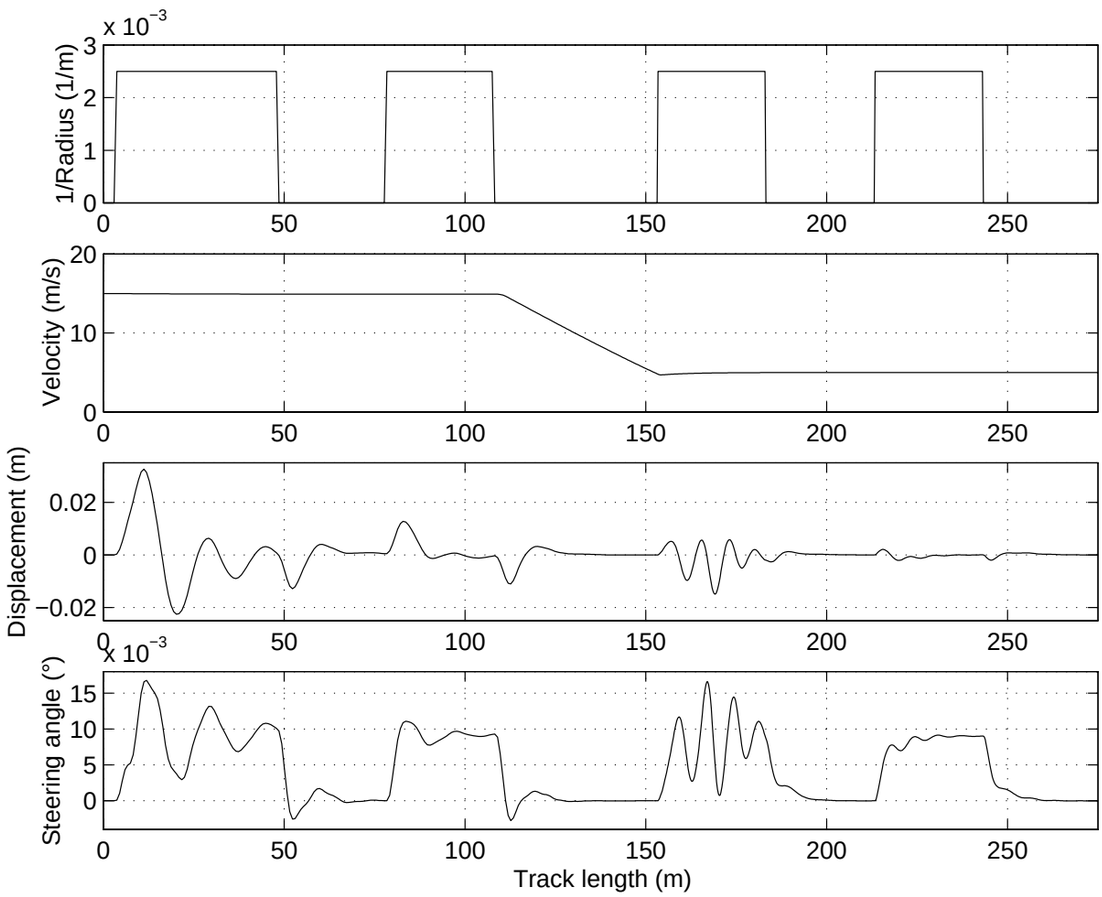


Figure 8.6: Simulation result of adaptive GPC controller for a changing operating condition (decreasing speed)

8.2.3 The Role of Signal Normalisation and Adaptation Freezing

As already pointed out, the normalisation of identification signals represents an effective mechanism to deal with underinformative data at high signal values. Figure 8.7 shows the sensor displacement and the steering angle as output and input of the

estimation algorithm for the same driving manoeuvre that was shown in figure 8.5. The third diagram shows the normalisation factor η(t).

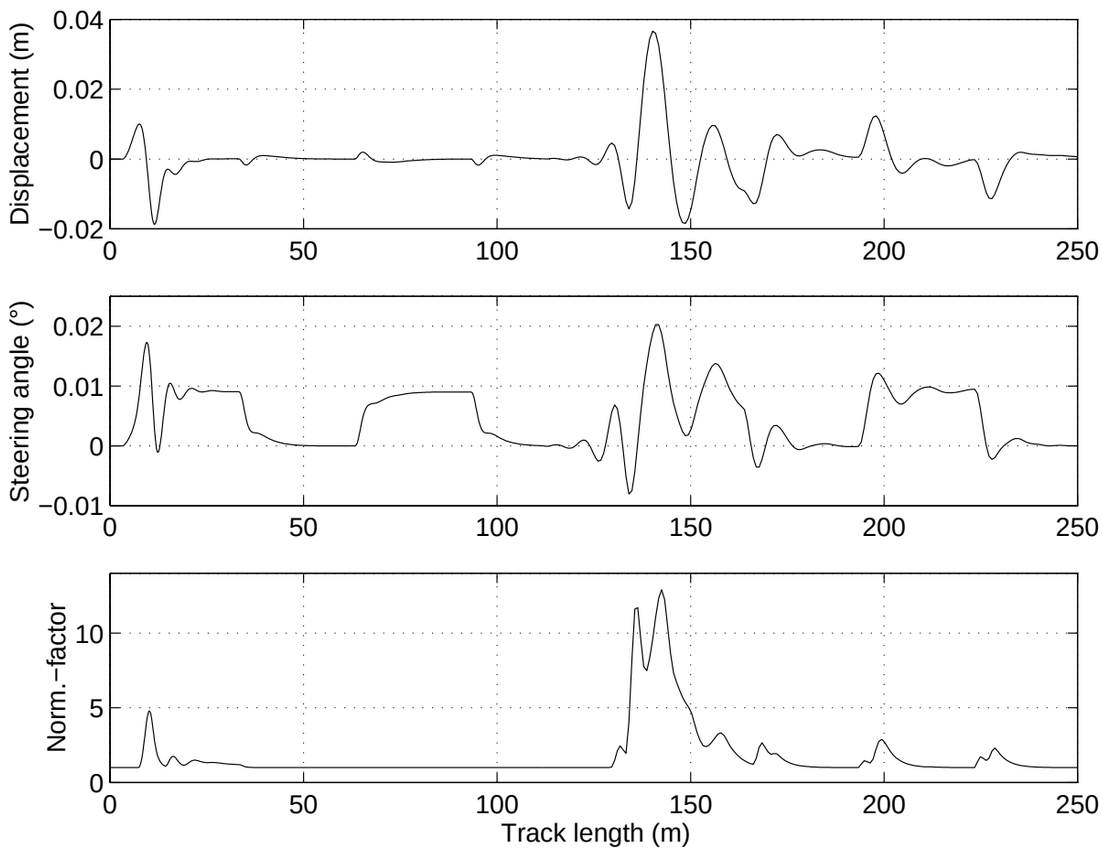


Figure 8.7: Identification output and input signals and normalisation factor η(t)

The normalisation dynamics parameter σ is set to 0.95, which corresponds to the largest distance between closed-loop poles and origin, the normalisation threshold η_0 is set to 1 for the first simulation (figure 8.5) and to 0.1 for the second simulation (figure 8.6), which was the result of an iterative procedure. The nomenclature corresponds to the parameters which were used in chapter 3.6.

Figure 8.8 shows the influence of adaptation freezing, which allows to reject errors due to normalisation with very large signals, or, if the system is not persistently excited [Bit90, Cla85]. The first diagram shows the sensor displacement, the second diagram shows the information measure σ(t), which was explained in chapter 3.7. The freezing signal s(t) is shown in the third diagram, the last diagram shows the adaptation of one particular parameter of the estimated model, which is the first parameter of the numerator polynomial Be. Notice that the obtained estimated parameters depend on the identification signal and its frequency content.

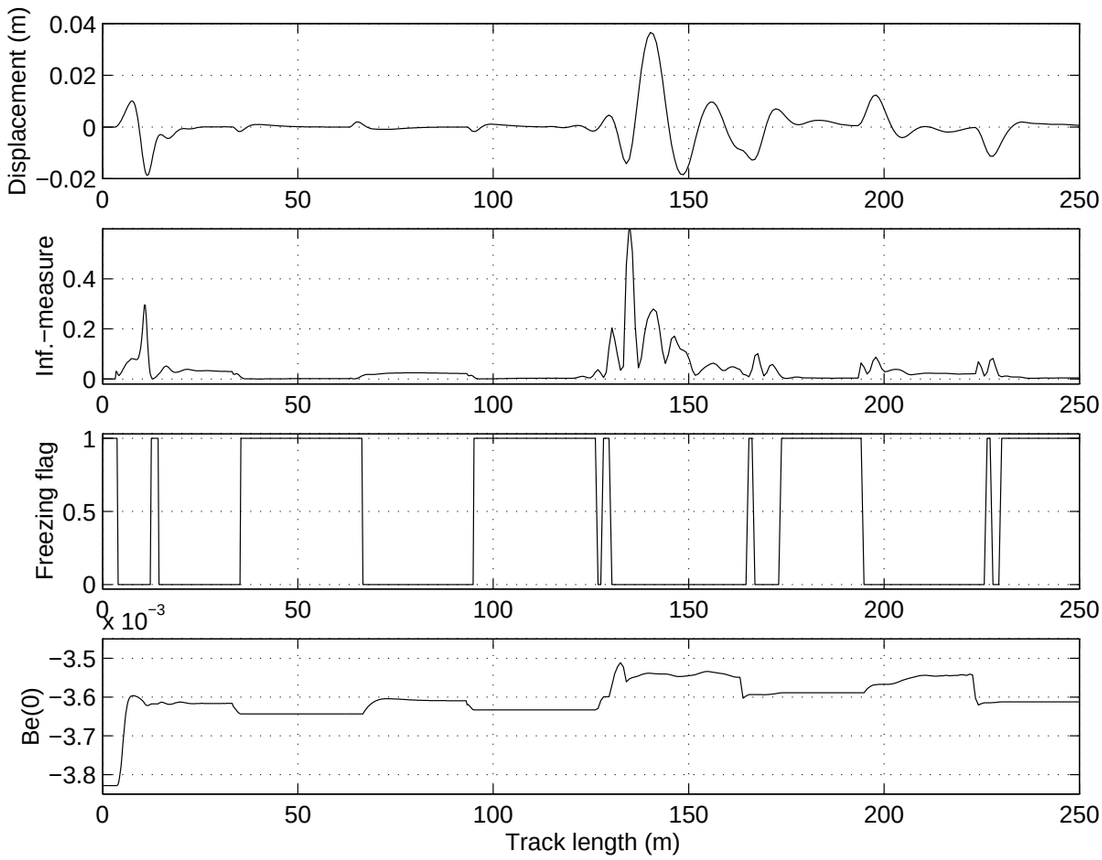


Figure 8.8: Influence of adaptation freezing

For the first simulation (figure 8.5) the information threshold σ_0 is set to 0.01, for the second simulation (figure 8.6) it is set to 0.001. The checking time intervals bfa and bra are set to 1 and 10. This was the result of an iterative procedure.

A forgetting factor sequence did not improve the adaptation result, therefore the corresponding values were set to f(0) = 1, f_t = 1, f_0 = 0.99, the initial adaptation gain F(0) was set to 1, since no initial parameter tuning was used, but initial estimated parameters for $\hat{\theta}(t)$.

A further modification of the estimation algorithm, not mentioned up to now, consists in an external normalisation of the identification signals. This is motivated by the fact, that numerical problems can be avoided if the filtered input and filtered output signal are of the same level and of signal values of approximately 1. This is illustrated in figure 8.9.

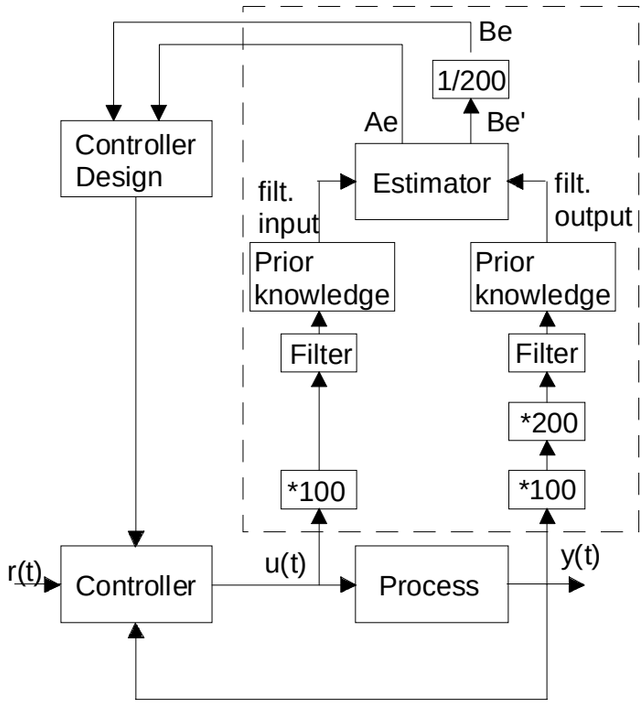


Figure 8.9: External normalisation of input and output signals

The multiplication of the output signal y(t) by 200 provides filtered input and output signals of the same magnitude, since the denominator polynomial Ae has to be monic, i.e. the first element is one, this multiplication is compensated by dividing the numerator polynomial Be by 200. The multiplication of input and output signal by 100 provides filtered input and output signals levels of approximately 1; since input and output signals are multiplied no compensation is needed. This external normalisation improves the estimation result considerably.

8.2.4 Supervision

Practical experience with adaptive controllers as well as stability and convergence analysis shows, that adaptive control performs well, if all prerequisites are fulfilled. However, this cannot be guaranteed in real-life applications. Therefore a supervisor is strongly recommended. Notice that in the presented simulations of the adaptive controller two identification parameters, namely the information threshold and the normalisation threshold, had to be adapted, which should be carried out by a supervisor. The block diagram of the resulting adaptive controller is shown in figure 8.10.

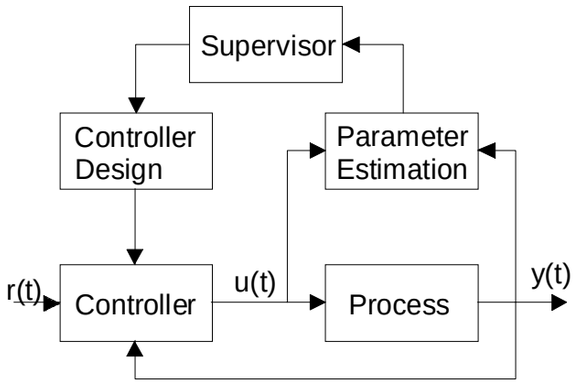


Figure 8.10: Block diagram of a self-tuning regulator with supervisory loop

The supervisor allows to monitor the estimation procedure, the controller design, and the closed-loop system behaviour.

The estimation procedure can suffer from low excited identification signals, from an incorrect model structure like model order and time delay, from an incorrect sampling period, and from inappropriate estimation parameters like adaptation gain, information measure threshold, normalisation threshold or the forgetting factor sequence. The supervisor can check pole-zero cancellations in order to determine the model order, it can reset the estimation parameters or the adaptation gain matrix, it can freeze the estimation, it can determine a suitable forgetting factor sequence or it can introduce perturbation signals to ensure that the control signal is sufficiently rich in frequencies [Iser87]. By storing process inputs and outputs, it is also possible to estimate models with different sampling periods, different orders, and different structures [Åst84].

The supervisor can also monitor the controller design, in order to avoid inappropriate design parameters or sampling periods. Therefore model poles and zeros can be calculated before controller design, and an estimated plant model admissibility test can be carried out. This admissibility test depends on the considered control design, for the GPC approach the characteristic polynomial of the closed loop has to be Hurwitz. It is nonetheless worth mentioning that the estimated plant model admissibility is generally satisfied in practice [M'Saad94c].

The closed-loop system behaviour, like actuator saturation, oscillating behaviour, large overshoots or increasing signal values, can be monitored in order to detect imminent instability. If this occurs in spite of the supervision of estimation procedure and controller design, then the adaptive controller can be replaced by a robust back-up regulator [Iser87].

Notice that up to now no supervisory loop has been implemented, this leads to the assumption, that there is considerable potential for increasing the performance of the presented adaptive controller.

9 Conclusions and Suggestions

The motivation of this work was to investigate the applicability of a robust predictive control approach with and without a suitable parameter adaptation for the design of an automatic steering system.

In a first step the vehicle dynamics have been identified over an important operating domain, unlike in all previous studies, where the usual single-track model was used. This was of fundamental importance in order to get a good insight into the possible model uncertainty. It also showed, that the usual single track model of 2^{nd} order could not describe the vehicle dynamics of the simulated BMW520i for velocities above 20 m/s.

The robust control design allowed to overcome the restrictions of automatic steering control for passenger cars to speed below highway speed, i.e. velocities below 20 m/s, which was reported in all previous approaches. Even without gain scheduling techniques and with feedback only on the lateral displacement unlike in the available studies where an additional feedback on the yaw rate is commonly used, the robust controller meets the design specifications up to a velocity of 30 m/s. The standard safety and passenger comfort specifications have been achieved, three fundamental design features of the proposed control approach are worth to be pointed out, namely offset-free performance, stability robustness and implementation simplicity [Mül96].

Notice that a great attention should be paid to the identification of the control design model as well as the choice of the design parameters. To this end, a comprehensive iterative procedure has been developed using the CACSD package SIMART.

The performance of the robust controller has been improved by a feed-forward controller, which generates the steady-state steering command when the vehicle is in a curved section. Even with a constant proportional controller for all conditions, considerable reductions of the maximum overshoot to values between 50 % and 18 % have been achieved.

Two adaptive approaches have been presented, an open-loop adaptation, more particularly a multi-model approach, and an indirect adaptation with the structure of a self-tuning regulator.

The multi-model approach is of practical importance and improves the robustness by dividing the operating domain into a number of operating intervals for each of them a corresponding controller will be used. This allows to extend the operating domain to velocities up to 37 m/s and to improve control performance for low speed.

The automatic steering problem represents a challenging application for the self-tuning regulator, since the system is only excited by an external output disturbance, which can be neither influenced, nor is it known in advance. It has been shown that, by selecting appropriate estimation parameters, for all driving manoeuvres the control performance can be improved with respect to the robust controller. The prerequisites are that the changes of the operating conditions are slower than the adaptation of the controller, which is generally satisfied in practice, and that the identification and design parameters are selected appropriately, therefore a supervision system is recommended and further research is required.

Suggestions for further work

The dynamics of the steering actuator have a considerable influence on the performance of an automatic steering system. Therefore they should already be considered in the development phase.

As far as the line following properties are concerned the proposed predictive controller showed to be robust to measurement noise, however, the ride comfort degrades, as explained in chapter 6. In order to improve the robustness of the control system to measurement noise in the output, gain scheduling techniques could be applied. Since in robust control design always a compromise between sensitivity to measurement noise and sensitivity to output disturbances has to be achieved, by the use of gain scheduling techniques one can attach more importance to measurement noise without loosing sensitivity to output disturbances.

In order to simplify the control design of the presented predictive controller a general methodology for the selection of the design parameters needs to be developed. Especially the use of the frequency weighting filters in the input and output signals remains an iterative and time-consuming procedure.

The proposed feed-forward controller is of a constant gain, assuming, that the steering angle depends only on the curve radius. The performance of the feed-forward controller can be improved, if the influence of the velocity and of the road-tyre contact characteristics are considered. This could be done by measuring the velocity, the lateral acceleration and the yaw rate and by estimating the cornering stiffness [Peng90].

The implemented indirect adaptive controller should be enhanced by a supervisory loop in order to monitor the estimation procedure, the controller design, and the closed-loop system behaviour.

10 Literature

[Ack93a] Ackermann, J.: *Robuste Regelung*. Springer Verlag, 1993.

[Ack93b] Ackermann, J.; W. Sienel; R. Steinhauser: *Robust Automatic Steering of a Bus*. Proc. of the ECC, Groningen, NL, pp. 1534-1539, 1993.

[Ack95] Ackermann, J.; J. Guldner; W. Sienel; R. Steinhauser; V.I. Utkin: *Linear and Nonlinear Controller Design for Robust Automatic Steering*. IEEE Transactions on Control Systems Technology, Vol. 3, No. 1, pp. 132-143, March 1995.

[Åst84] Åström, K.; B. Wittenmark: *Computer Controlled Systems, Theory and Design*. Prentice-Hall, Englewood Cliffs, 1984.

[Bit90] Bitmead, R.R.; M. Gevers; V. Wertz: *Adaptive Optimal Control, The Thinking Man's GPC.* Prentice Hall, Sydney, 1990.

[Bur93] Burckhardt, M.: *Fahrwerktechnik: Radschlupfregelsysteme.* Vogel Fachbuch, 1993.

[Che95] Chee, W; M. Tomizuka; S. Patwerdhan; W.-B. Zhang: *Experimental Study of Lane Change Manoeuvre for AHS Applications*. Proceedings of the ACC, Seattle, Washington, June 1995.

[Cla75] Clarke, D.W.; P.J. Gawthrop: *Self-tuning controller*. Proc. IEE, 122, pp. 633-640, 1975.

[Cla85] Clary, J.P.; G.F. Franklin: *A Variable-Dimension Self-Tuning Controller.* Proceedings of the ACC, Boston, June 1985.

[Cla87] Clarke, D.W.; C. Mohtadi; P.S. Tuffs: *Generalized Predictive Control*. Part I and II, Automatica, Vol. 23, No. 2, pp. 137-160, 1987.

[Cla89] Clarke, D.W.; C. Mohtadi: *Properties of Generalized Predictive Control*. Automatica, Vol. 25, No. 6, pp. 859-875, 1989.

[Daiß96] Daiß, A.: *Beobachtung fahrdynamischer Zustände und Verbesserung einer ABS- und Fahrdynamikregelung*. Düsseldorf: Fortschrittsberichte VDI-Verlag, Reihe 12, 1996.

[Des75] Desoer, C.A.; M. Vidyasagar: *Feedback Systems: Input-Output Properties*. Academic Press, New York, 1975.

[Dug87] Dugard, L.; M. M'Saad; M. Duque: *Adaptive LQ Control of a Flexible Transmission System*. ICAR 87, Versailles, France, 1987.

[God96] Godbole, D.N.; F.H. Eskafi; P.P. Varaiya: *Automated Highway Systems*. 13th Triennial World Congress, IFAC, San Francisco, 1996.

[Gul94] Gulder, J.; V.I. Utkin; J. Ackermann: *A Sliding Mode Control Approach to Automatic Car Steering*. Proceedings of the ACC, Baltimore, Maryland, June 1994.

[Gul95] Gulder, J.; V.I. Utkin; J. Ackermann; T. Bünte: *Sliding Mode Control for Active Steering of Cars*. IFAC-Workshop on Advanced Automotive Control, Ascona, Switzerland, 1995.

[Hej94] Hejda, I.; M. M'Saad: *The CACSD Package SIMART: Description and Case Study*. Proc. of the IEEE/IFAC Symposium on Computer-Aided Control System Design, pp. 461-466, 1994.

[Horn86] Horn, A.: *Fahrer - Fahrzeug - Kurvenfahrt auf trockener Straße*. Dissertation TU Braunschweig, 1986.

[Hunt96] Hunt, K.J.; R. Haas, J.C. Kalkkuhl: *Local Controller Network for Autonomous Vehicle Steering.* 13th Triennial World Congress, IFAC, San Francisco, 1996.

[Iser87] Isermann, R.: *Digitale Regelsysteme.* Band 1 u. 2, 2. Aufl., Springer-Verlag Berlin Heidelberg, 1987.

[Iser92] Isermann, R.: *Identifikation dynamischer Systeme.* Band 1 u. 2, 2. Aufl., Springer Verlag, Berlin Heidelberg 1992.

[Kam92] Kammeyer, K.D.; K. Kroschel: *Digitale Signalverarbeitung.* 2. Aufl., B.G. Teubner Stuttgart, 1992.

[Kie93] Kiencke, U: *Realtime Estimation of Adhesion Characteristic between Tyres and Road*. IFAC World Congress, Sydney, 1993.

[Kie94] Kiencke, U.; A. Daiß: *Estimation Tyre Friction for enhanced ABS-Systems*. AVEG-Congress, Tokyo, 1994.

[Kie96] Kiencke, U.; A. Daiß: *Observation of the Lateral Vehicle Dynamics*. 13th World Congress of IFAC, San Francisco, 1996.

[Kro91] Kronmüller, H.: *Digitale Signalverarbeitung*. Springer Verlag, Berlin Heidelberg, 1991.

[Kwa85] Kwakernaak, H.: *Uncertainty Models and the Design of Robust Control Systems.* Proc. of an International Seminar organised by DFVLR, Bonn, May, 1985.

[Kwa93] Kwakernaak, H.: *Robust Control and H_∞-Optimisation - Tutorial Paper*. Automatica, Vol. 29, No. 2, pp. 255-273, 1993.

[Lan88] Landau, I.D.: *Identification et Commande des Systèmes.* Hermès, Paris, 1988.

[Lan94] Landau, I.D.; A. Voda; D. Rey: *Robust Digital Control of Flexible Transmissions using the Combined Pole Placement / Sensitivity Function Shaping Method*, Proceedings of the ACC, Baltimore, Maryland, 1994.

[Lim95] Limebeer, D; M. Green: *Robust Linear Control*. Prentice Hall, Englewood Cliffs, 1995.

[Ljung87] Ljung, L: *System Identification, Theory for the User*. Prentice Hall PTR, Upper Saddle River, New Jersey, 1987.

[Maj96a] Majjad, R.; U. Kiencke: *Modelling and Simulation of Non-linear Vehicle Dynamics.* MTNS-96 St. Louis, Missouri, June 1996.

[Maj96b] Majjad, R.; U. Kiencke*: Modularer Ansatz zur Modellierung der Kraftfahrzeugdynamik*. VDI-Tagung: Elektronik im Kraftfahrzeug, Baden-Baden, September 1996.

[Mid90] Middleton, R.H.; G.C. Goodwin: *Digital Control and Estimation, A Unified Approach*. Prentice Hall, Englewood Cliffs, 1990.

[Mit90] Mitschke, M.*: Dynamik der Kraftfahrzeuge, Band C: Fahrverhalten*. Springer Verlag, 1990.

[Mül96] Müller, N.; R. Majjad; U. Kiencke; R. Ramirez-Mendoza; M. M'Saad; L. Dugard: *Robust Predictive Control Design for Automatic Steering*. In 'Annual Meeting IAR 96', Karlsruhe, Germany, 1996.

[M'Saad92] M'Saad, M.; G. Sanchez: *Partial State Reference Model Adaptive Control of Multivariable Systems*. Automatica, Vol. 28, pp. 1189-1192, 1992.

[M'Saad94] M'Saad, M.: *SIMART: un Logiciel pour la Commande Avancée de Procédés Industriels*. Proc. de la conférence 2AO 94, Groupe ESIEE Paris Noisy-Le-Grand, 1994.

[M'Saad94a] M'Saad, M.; I. Hejda: *Adaptive Control of a Flexible Transmission System*. Control Eng. Practice, Vol. 2, No. 4, pp. 629-639, 1994.

[M'Saad94c] M'Saad, M.: *Commande Adaptative de Procédés*. Ecole d'été d'Automatique de Grenoble: Commande et Filtrage Optimaux, Grenoble, 1994.

[M'Saad96] M'Saad, M.; J. Chebassier: *Commande Optimisée des Systèmes: Commande Optimale*. Diderot, 1996.

[Peng90] Peng, H., M. Tomizuka: *Lateral Control for Highway Automation*, Proc. of the ACC, 1990.

[Pra90] Praly, L.: *Almost exact Modelling Assumption in Adaptive Control*. Int. Journal of Control, Vol. 51, pp. 643-668. 1990.

[Ram95] Ramirez-Mendoza, R.; M. M'Saad; L. Dugard; R. Majjad; U. Kiencke: *Linear Quadratic Gaussian Control Design for Robust Automatic Steering*. In 'Annual Meeting IAR 95', Grenoble, France, 1995.

[Rei90] Reichelt, W.: *Ein adaptives Fahrermodell zur Bewertung der Fahrdynamik von Pkw in kritischen Situationen*. Dissertation, TU Braunschweig, 1990.

[Ris91] Risse, H.-J.: *Das Fahrerverhalten bei normaler Fahrzeugführung*. Düsseldorf: Fortschrittsberichte VDI-Verlag, Reihe 12, Nr. 160, 1991.

[Shla93] Shladover, S.E.: *Research and Development Needs for Advanced Vehicle Control Systems*. Partners for Advanced Transit and Highways (PATH), University of California, Berkeley, IEEE Micro pp. 11-18, 1993.

[Sie94] Sienel, W.; J. Ackermann: *Automatic Steering of Vehicles with Reference Angular Velocity Feedback*. Proceedings of the ACC, Baltimore, Maryland, 1994.

[Smi78] Smith, C.C.; D.Y. McGehee; A.J. Healey: *The Prediction of Passenger Riding Comfort from Acceleration Data*. Trans. ASME J. Dyn. Syst., Meas. Control, Vol. 100, pp. 34-41, 1978.

[Soet90] Soeterboek, R.: *Predictive Control, A Unified Approach*. Dissertation, Technical University Delft, 1990.

[Sun88] Sung, H.K.; S. Hara: *Properties of Sensitivity and Complementary Sensitivity Function in SISO Digital Systems*. International Journal of Control, Vol. 48, pp. 2429-2439, 1988.

[Ver95] Verde, C.; J. Flores: *Nominal Model Selection for Robust Control Design*. Proc. of ACC, Seattle, Washington, 1995.

[Zho95] Zhou, K.; J.C. Doyle; K. Glover: *Robust and Optimal Control*, Prentice Hall, Englewood Cliffs, 1995.

11 : Comparison between j-step ahead predictor and Smith predictor

In this section a comparison is made between a predictor which is based on the iteration of a Diophantine equation, and the Smith predictor.

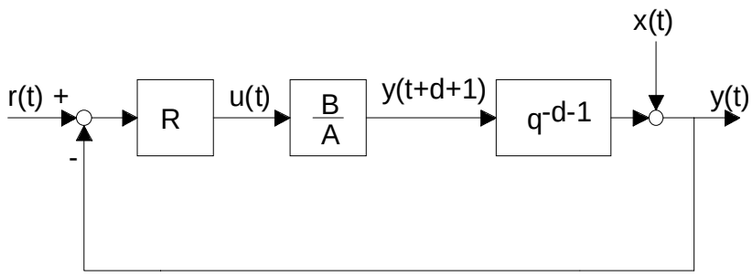


Figure A.1: Block diagram for feedback of y(t)

If a closed-loop system is considered like in figure A.1, the transfer function between reference sequence r(t) and output y(t) is given by

$$y(t) = \frac{q^{-d-1}BR}{A + q^{-d-1}BR} r(t) \tag{A.1}$$

where R denotes the transfer function of the controller.

Due to the time delay of the process this systems becomes easily unstable why the following closed-loop structure is preferred:

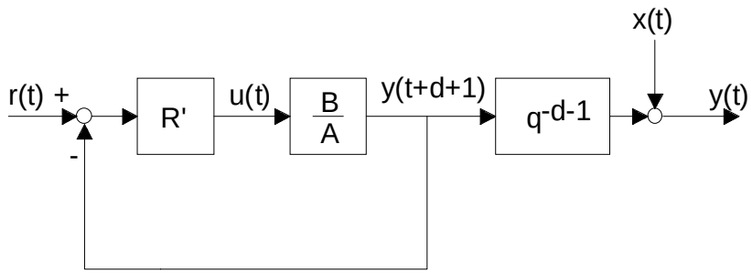


Figure A.2: Block diagram for feedback of y(t+d+1)

Now the input-output transfer function is given by

$$y(t) = \frac{q^{-d-1}BR'}{A + BR'} r(t) \tag{A.2}$$

A comparison between equations (A.1) and (A.2) leads to

$$R = \frac{R'}{1 + R'\frac{B}{A}(1 - q^{-d-1})}$$

which is shown in figure A.3:

Appendix A : Comparison between j-step ahead predictor and Smith predictor

In this section a comparison is made between a predictor which is based on the iteration of a Diophantine equation, and the Smith predictor.

If a closed-loop system is considered like in figure A.1, the transfer function between reference sequence r(t) and output y(t) is given by

$$y(t) = \frac{q^{-d-1}BR}{A + q^{-d-1}BR} r(t) \tag{A.1}$$

where R denotes the transfer function of the controller.

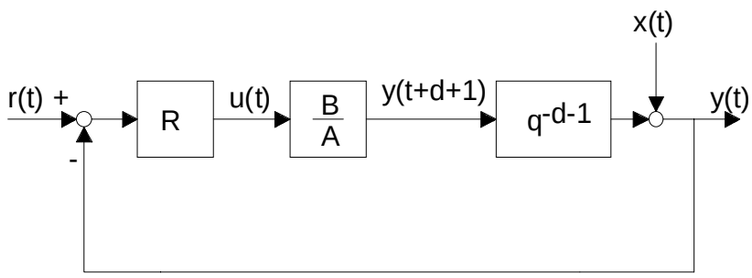


Figure A.1: Block diagram for feedback of y(t)

Due to the time delay of the process this systems becomes easily unstable why the following closed-loop structure is preferred:

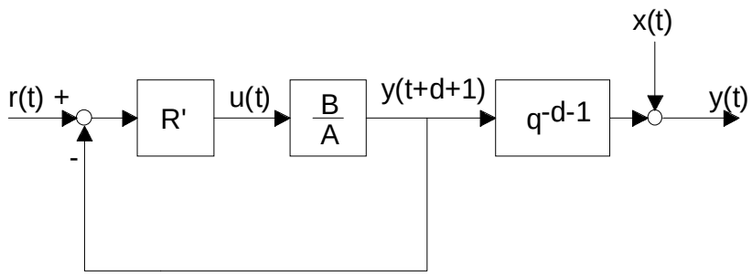


Figure A.2: Block diagram for feedback of y(t+d+1)

Now the input-output transfer function is given by

$$y(t) = \frac{q^{-d-1}BR'}{A + BR'} r(t) \tag{A.2}$$

A comparison between equations (A.1) and (A.2) leads to

$$R = \frac{R'}{1 + R'\frac{B}{A}(1 - q^{-d-1})}$$

which is shown in figure A.3:

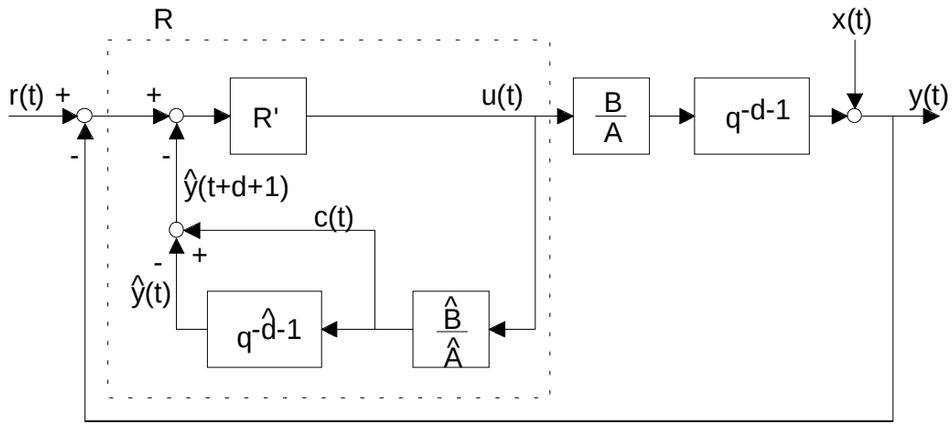


Figure A.3: Block diagram for feedback of $\hat{y}(t+d+1)$ using Smith predictor

This block diagram shows that the predicted output of the process $\hat{y}(t+d+1)$ is calculated using a model of the process, which is indicated by the symbol ^ . Now the formula for the *Smith predictor* is given:

$$\hat{y}(t+d+1) = \frac{\hat{B}}{\hat{A}}u(t) + \left[y(t) - q^{-d}\frac{\hat{B}}{\hat{A}}u(t-1) \right] \tag{A.3}$$

Notice that if the process is correctly estimated and there are no disturbances, then the correction signal c(t) is zero. Thus, in this case y(t) is equal to $\hat{y}(t)$.

In order to compare the Smith predictor with the MV j-step-ahead predictor derived in chapter 6.1.2, equation (6.1.11) is considered:

$$C(q^{-1})\, y(t+j) = E_j(q^{-1})\, B(q^{-1})\, D(q^{-1})\, u(t+j-1-d) + F_j(q^{-1})\, y(t) + E_j(q^{-1})\, C(q^{-1})\, \gamma(t+j)$$

With $\gamma(t+j) = 0$ (MV predictor), with $C(q^{-1}) = T(q^{-1})$, and with $j = d + 1$ this leads to

$$\hat{y}(t+d+1) = \frac{E_{d+1}\hat{B}\hat{D}}{T}u(t) + \frac{F_{d+1}}{T}y(t) \tag{A.4}$$

Equation (6.1.8)

$$C(q^{-1}) = A(q^{-1})\, D(q^{-1})\, E_j(q^{-1}) + q^{-j}\, F_j(q^{-1})$$

can be multiplied by $\hat{B}$, this leads, with $j = d + 1$ and $C(q^{-1}) = T(q^{-1})$, to

$$E_{d+1}\hat{B} = \frac{T\hat{B}}{\hat{A}\hat{D}} - q^{-d-1}\frac{F_{d+1}\hat{B}}{\hat{A}\hat{D}} \tag{A.5}$$

Using equation (A.5) equation (A.4) can be written as

$$\hat{y}(t+d+1) = \frac{\hat{B}}{\hat{A}}u(t) + \frac{F_{1+d}}{T}\left[y(t) - q^{-d}\frac{\hat{B}}{\hat{A}}u(t-1)\right] \tag{A.6}$$

which is visualised in figure A.4.

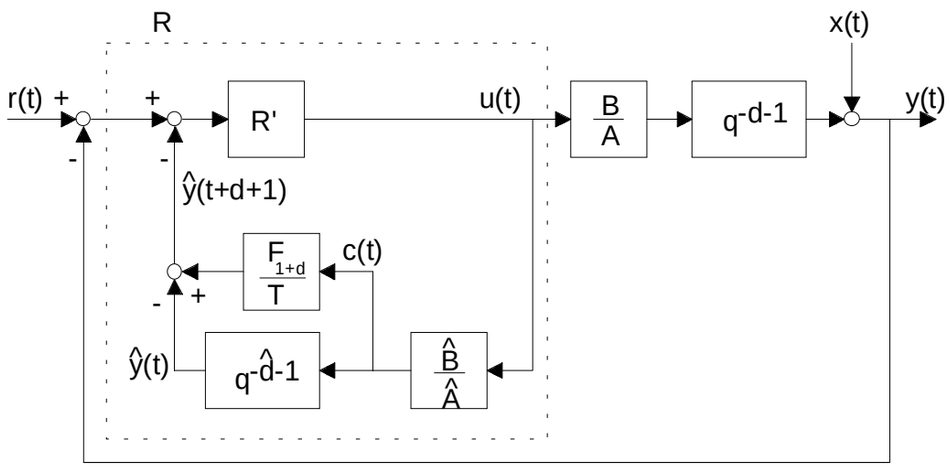


Figure A.4: Block diagram for feedback of $\hat{y}$ (t+d+1) using MV d+1-step-ahead predictor

This shows that the correction term c(t) is filtered with $\frac{F_{1+d}}{T}$, and that the MV d+1-step-ahead predictor takes, unlike the Smith predictor, disturbances into account. It corresponds to the Smith predictor for

$$F_{1+d}(q^{-1}) = T(q^{-1})$$

Now equation (A.5) becomes

$$E_{d+1} = \frac{T}{\hat{A}\hat{D}} - q^{-d-1}\frac{T}{\hat{A}\hat{D}} = \frac{T}{\hat{A}\hat{D}}\left(1 - q^{-d-1}\right)$$

For T = $\hat{A}$ this equation can be written as

$$\hat{D}E_{d+1} = 1 - q^{-d-1}$$

If $\hat{D}$ is assumed to be a simple differencing operator $\Delta = 1 - q^{-1}$ the polynomial E_{d+1} is given by

$$E_{d+1}(q^{-1}) = 1 + q^{-1} + \ldots + q^{-d}$$

This shows that both predictors are identical if T = $\hat{A}$ and $\hat{D} = \Delta$, hence, if the output of the process is disturbed by step disturbances or Brownian motion.

Appendix B : Relationship of GPC with other Approaches

If the control parameters sh, ph, ch and λ are chosen appropriately, the GPC can be interpreted as infinite-stage LQ controller, as dead-beat, or as pole-assignment controller [Cla89, M'Saad96, Soet90]. The special settings are given in the following chapter.

Minimum Variance Control

The Minimum Variance (MV) controller was already explained in chapter 6.1 since it represents an ancestor of GPC. It can be obtained by setting

ph = ch = d, and

$\lambda = 0$.

This minimises the variance of the tracking error for a given linear input-output model:

$$J_{MV} = E\left\{\left(y(t+d) - r(t+d)\right)^2\right\}$$

y(t), r(t) and d denote, as usual, the system output, the reference sequence, and the assumed value of the plant's dead-time, respectively. In this approach a d-step-ahead predictor is used. Notice, that this control strategy works only for minimum phase systems.

The Generalized Minimum-Variance (GMV) controller can be obtained by setting

ph = ch = d, and

$\lambda \neq 0$,

then the cost function

$$J_{GMV} = E\left\{\left(y(t+d) - r(t+d)\right)^2 + \lambda\left[D(q^{-1})u(t)\right]^2\right\}$$

is minimised. Notice, that GMV showed to be very robust, but sensitive to varying dead-time unless λ is large, which results in poor control performance.

Dead-beat control

A stable dead-beat controller can be obtained by selecting the control parameters as

$sh = n_B + d + 1$,

$ph \geq n_A + n_B + d + 1$,

$ch = n_A + 1$, and

$\lambda = 0$

n_A denotes the order of the plant denominator and n_B the order of the plant nominator.

Dead-beat controller are characterised by a limited settling time for system output y(t) *and* control signal u(t). After this settling time the responses of y(t) and u(t) on e.g. a step input have to be constant.

Dead-beat control places all the poles of the closed loop at the origin while retaining the open-loop zeros. This is obtained by cancelling the original poles of the process. If the control model does not perfectly correspond to the plant, which is generally the case, the pole-cancellation is only approximately performed which entails problems if the plant poles are outside the unit circle. Therefor dead-bead control is only recommended for stable processes without integrating behaviour, that means, when the poles are sufficiently inside the unit circle.

Dead-beat controller usually demand relatively high control signals, nevertheless, they are very simple to derive and to implement in a computer which is particularly appreciated in adaptive control, when control synthesis is performed on-line. [Iser87].

Pole-Assignment Control

Since dead-beat control often requires excessive control signals a strategy placing poles in better locations in generally preferred. An algorithm to move the closed-loop poles to better locations can be easily incorporated into the GPC method. This can be done by pre-filtering the measured output y(t) and the control input u(t) by a polynomial $P(q^{-1})$ with unit gain and which roots are at the desired closed-loop poles [Soet90]. The control parameters are selected like

$sh = n_B + d + 1,$

$ph \geq n_A + n_B + d + 1,$

$ch = n_A + 1,$ and

$\lambda = 0$

With the auxiliary signals

$\psi(t) = P(q^{-1})\, y(t)$ and

$\nu(t) = P(q^{-1})\, u(t)$

the plant model becomes

$$A(q^{-1})\, D(q^{-1})\, \psi(t) = B(q^{-1})\, D(q^{-1})\, \nu(t-1)$$

and the standard pole-placement closed-loop response is

$$y(t) = \frac{B(q^{-1})}{B(1)P(q^{-1})} w(t)$$

Thus, the predictive controller can be regarded as a pole-placement controller which places the closed-loop poles by the minimisation of a criterion function.

In spite of the fact, that for special settings GPC can be interpreted as pole-placement controller one should consider that pole-placement strategies are very sensitive to model-order changes. GPC in contrast seems to be unaffected if the plant model is overparameterized.

Infinite Horizon LQ Control

GPC can be interpreted as an infinite horizon LQ controller for the following settings:

$ph \rightarrow \infty$,

$ch \rightarrow \infty$, with $ph = ch$,

$sh = 1$,

$\lambda > 0$.

Note that this is just a theoretical interpretation, for large values of ch the usual GPC computation becomes impractical and instead methods based on the recursive solution of the equivalent Riccati equation are preferred [M'Saad96].