Improvement of efficiency in identification and model predictive control of industrial processes

A flexible linear parametrization approach

Edwin T. van Donkelaar

Improvement of efficiency in identification and model predictive control of industrial processes

A flexible linear parametrization approach

Improvement of efficiency in identification and model predictive control of industrial processes

A flexible linear parametrization approach

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.ir. K.F. Wakker,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 21 november 2000 om 16.00 uur
door

Edwin Teunis VAN DONKELAAR

werktuigkundig ingenieur
geboren te Ede

Dit proefschrift is goedgekeurd door de promotoren:
Prof.ir. O.H. Bosgra
Prof.dr.ir. P.M.J. Van den Hof

Samenstelling Promotiecommissie:

Rector Magnificus,	voorzitter
Prof.ir. O.H. Bosgra,	Technische Universiteit Delft, promotor
Prof.dr.ir. P.M.J. Van den Hof,	Technische Universiteit Delft, promotor
Prof.dr.ir. A. Backx,	Technische Universiteit Eindhoven
Prof.dr. J. Bokor,	Hungarian Academy of Science
Prof.dr.ir. T.H.J.J. van der Hagen,	Technische Universiteit Delft
Prof.dr.ir. P.H.A.M. Verhaegen,	Universiteit Twente
Prof.dr.ir. A.A. Stoorvogel,	Technische Universiteit Delft

ISBN 90-370-0184-x

Summary

Both in model predictive control and system identification an optimization problem has to be solved. In model predictive control an optimal input signal over a certain future horizon is calculated on-line, while in system identification an optimal model is required. In both optimization problems it is important to choose the degrees of freedom wisely. The choice of the degrees of freedom is denoted with parametrization, i.e. input parametrization in predictive control and model parametrization in identification. For quadratic cost functions, which are common in both areas, and a linear parametrization the optimization problems are convex: a quadratic programming problem for predictive control and a linear least squares problem for identification. There is a wide variety of linear parametrizations that can be utilized. In this thesis it is investigate to what extend flexibility in the linear parametrization can contribute to the improvement of model predictive control and system identification techniques.

In prediction error system identification a nonconvex optimization problem has to be solved to find an optimal model for the system dynamics and the disturbance dynamics. This nonconvex optimization problem, which originates from the nonlinear parametrization of the model, is intractable for identification of industrial processes which often have a large number of inputs and outputs. Alternatively, a linear parametrization provides a convex optimization problem but can suffer from overdimensioning, i.e. many free parameters are required for a certain accuracy which increases the variance of the estimate. In the thesis it is investigated how the optimal prediction error model, i.e. the result of a nonconvex optimization, can be approximated in terms of system-based orthogonal basis functions, e.g. Laguerre, two-parameter Kautz and generalized basis functions, such that a low number of parameters is utilized to obtain a model with a small bias and variance contribution. Also the computational load must be limited due to the large scale of the optimization problems originating in industrial system identification.

System based orthogonal basis functions can be utilized as building blocks for multivariable model structures that exploit prior knowledge about system dynamics and structure. The more accurate prior knowledge is utilized, the higher the accuracy of the identified model and the lower the optimization complexity. Identification of the system transfer amounts to solving a linear least squares problem. Various model structures can be built that contain both the system dynamics and disturbance dynamics. A model structure is proposed in which the system and disturbance model is parametrized independently and that is bilinear in the parameter vector. Identification with this model structure is analyzed and an alternative optimization procedure is proposed which is shown to converge to the global optimum under mild conditions.

Dependent on the amount and accuracy of prior knowledge, the identified models are overdimensioned. With an iterative scheme of high order identification with system based

orthogonal basis functions and model reduction the number of free parameters can be reduced systematically. In this way the optimal estimate, that is equal to the global minimum of the nonconvex optimization problem mentioned above, can be approximated with increased accuracy. A thorough convergence analysis of this iterative scheme is still lacking, but simulations show promising results for this identification approach.

In model predictive control a quadratic programming problem is solved on-line to find an optimal input trajectory that satisfies all process constraints like restrictions on the input and output signals. Model predictive control is arguably the only control technology that deals with process constraints in a systematic way. This on-line optimization is time consuming and therefore this technology is mainly applied to systems with a relatively large sampling time, such as petrochemical and chemical processes. Due to the large computational load, model predictive control is not readily applicable to fast systems, e.g. consumer electronics, airplanes, cars and telecommunication networks, although limitations of these systems can be reached during operation and should therefore be accounted for in a systematic way.

The computational load of model predictive control can be decreased, apart from the increased computational speed of computer hardware and software, by a choice of a low number of degrees of freedom in the input trajectory because the number of free parameters in the optimization has a large influence on the computational load. The choice of the parametrization must be such that high control performance is obtained although the number of free parameters is low.

In this thesis two approaches are proposed to choose the input parametrization in model predictive control. The first utilizes the observation that all unconstrained optimal input trajectories are contained in a subspace with dimension equal to the model order. For an infinite prediction horizon this subspace is generated by a linear dynamical system. To further reduce the complexity, this linear dynamical system can be reduced with standard model reduction algorithms that are tailored to the performance criterion, like e.g. LQG balanced reduction and frequency weighted balanced reduction with the appropriate weighting. With this approach the unconstrained performance loss can be quantified by the indicators of the model quality provided by the reduction algorithm. The system based parametrization provides an efficient parametrization for unconstrained and mildly constrained situations but is not tailored to situations where constraints play a major role. Therefore, a second approach is developed that utilizes prior knowledge of the active constraints to construct a parametrization. The approach can be regarded as an extension to quadratic programming that selects a low dimensional subset of the full parameter space which contains the optimal solution. This can improve the computational speed considerably.

The developed approach is applied to a simulation example of an industrial type system, which illustrates the efficiency improvement that can be obtained.

Voorwoord

Zo, eindelijk is het dan zo ver. Mijn promotie zit erop. Eigenlijk beschouw ik dit niet als de promotie van mij alleen. Er zijn veel mensen bij betrokken geweest die zeker zo essentieel waren in de verwezenlijking van dit onderzoek als ikzelf. Deze wil ik hier graag bedanken.
Allereerst wil ik mijn promotor Okko Bosgra bedanken. Met zijn brede kennis van het vakgebied wist hij in het begin van het project met enkele rake zinsneden een visie te verwoorden die mijn verdere onderzoek richting heeft gegeven. Ook in latere fasen wist hij steeds weer nieuw licht te werpen op mijn onderzoek.
Ook Paul Van den Hof ben ik erg dankbaar. Met zijn enthousiasme voor het vakgebied, scherpzinnigheid en virtuositeit aan het bord heeft hij heel wat vage ideeën meer helderheid gegeven. Toen het schrijfproces niet wilde vlotten heeft hij veel tijd vrijgemaakt om mij te helpen.
Ton Backx heeft met zijn gedrevenheid en kennis van de praktijk een belangrijke bijdrage geleverd aan deze promotie. Zijn niet aflatende enthousiasme heeft me gesteund in de tijd dat ik wat minder tevreden was met mijn eigen werk. Ook Jobert Ludlage heeft, terwijl hij zijn sigaartje rookte, me inhoudelijk geholpen en gesteund.
Ik heb het voorrecht gehad dat drie studenten een steentje bij wilden dragen aan het project. Daniël Blom, Christiaan Houben en Martijn Leskens, bedankt voor jullie inzet. Ik heb veel van jullie geleerd. De contacten met de afstudeerders bij Bosch, Kamal Ahmed en Hilko Hakvoort, die Ton van der Weiden en ik samen hebben begeleid, vormden een leuke ervaring en een welkome afleiding.
Ik wil iedereen in de sectie Systeem- en Regeltechniek bedanken voor de hulp die ik gekregen heb en de prettige sfeer. Ook iedereen in de vakgroep Modelgebaseerd Meten en Regelen van de subfaculteit Technische Natuurkunde wil ik van harte bedanken voor de prettige ontvangst die zij me gegeven hebben. Jullie hebben mijn laatste maanden in Delft echt veraangenaamd. Speciaal wil ik Thomas de Hoog en Peter Heuberger, mijn projectgenoten, bedanken voor de prettige samenwerking.
Mijn dank gaat tevens uit naar de Stichting Technische Wetenschap (STW). De Stichting heeft mijn werk en conferentiebezoek gesponsord en daarvoor ben ik hen dankbaar.
Natuurlijk wil ik graag mijn ouders en zus bedanken voor het feit dat ze, ondanks dat ze vermoedelijk geen idee hadden waar ik in Delft zo druk mee bezig was, vertrouwen in mij gehad hebben en met me meegeleefd hebben.
Tot slot wil ik mijn vriendin Annemieke bedanken voor de liefde, steun en begrip die ik van haar ontvangen heb.

Edwin van Donkelaar
Breda, oktober 2000

Contents

Summary i

Voorwoord iii

1 Introduction 1
- 1.1 Motivation and background . . . 1
- 1.2 Constrained control of industrial processes . . . 2
  - 1.2.1 Industrial practice . . . 2
  - 1.2.2 Model predictive control . . . 3
  - 1.2.3 Possibilities and limitations . . . 5
- 1.3 Process modeling . . . 6
  - 1.3.1 Industrial practice . . . 6
  - 1.3.2 System identification . . . 7
  - 1.3.3 Possibilities and limitations . . . 8
- 1.4 Optimization in control and identification . . . 9
- 1.5 Problem formulation . . . 10
- 1.6 Solution strategy . . . 11
- 1.7 Short outline of the thesis . . . 12

2 Control and identification: preliminaries and parametrization approaches 13
- 2.1 Introduction . . . 13
- 2.2 Concepts, notations, definitions . . . 13
  - 2.2.1 Systems and signals . . . 13
  - 2.2.2 Control . . . 20
- 2.3 Optimization . . . 23
  - 2.3.1 Mathematical programming . . . 23
  - 2.3.2 Least squares problem . . . 29
  - 2.3.3 Quadratic programming . . . 29
- 2.4 Model predictive control . . . 30
  - 2.4.1 Historic overview . . . 30
  - 2.4.2 Optimal control . . . 32

2.4.3 The basic principle of model predictive control . . . 34
2.4.4 Computational issues . . . 37
2.4.5 Stability issues . . . 42
2.4.6 Performance issues . . . 43
2.4.7 Parametrization issues . . . 44
2.4.8 Computational complexity . . . 47
2.5 System identification . . . 48
2.5.1 Historic overview . . . 48
2.5.2 Elements of system identification . . . 49
2.5.3 Convergence and consistency . . . 54
2.5.4 Bias and variance . . . 54
2.5.5 Parametrization issues . . . 56
2.5.6 Computational complexity . . . 56
2.6 Model predictive control and system identification: a linear parametrization approach . . . 57
2.6.1 Motivation . . . 57
2.6.2 Linear parametrization in identification . . . 58
2.6.3 Linear parametrization in model predictive control . . . 58
2.6.4 Finding a suitable linear parametrization . . . 59
2.7 Detailed problem formulation and outline . . . 61

3 System identification with system-based basis functions 65
3.1 Introduction . . . 65
3.2 Scalar identification with orthogonal basis functions . . . 68
3.2.1 Construction of an orthogonal system-based basis . . . 68
3.2.2 Parameter estimation . . . 72
3.2.3 Asymptotic bias . . . 73
3.2.4 Asymptotic variance . . . 75
3.3 Multivariable model structures . . . 76
3.3.1 Constructing of a complete orthogonal basis . . . 76
3.3.2 Parameter estimation . . . 82
3.3.3 Asymptotic bias . . . 82
3.3.4 Asymptotic variance . . . 83
3.4 Identification of system and disturbance dynamics . . . 84
3.4.1 Generalized ORTFIR structures . . . 84
3.4.2 The basic principle of ORT-GLS . . . 86
3.4.3 Convergence properties of ORT-GLS . . . 87
3.4.4 Practical implementation . . . 88
3.5 Design issues . . . 90

3.5.1 Motivation . . . 90
3.5.2 Finite data aspects . . . 91
3.5.3 Selection of poles . . . 97
3.5.4 Estimation of initial conditions . . . 98
3.5.5 A bilinear model structure . . . 99
3.5.6 Constrained estimation . . . 100
3.5.7 Combination of data sets in system identification . . . 102
3.6 Simulation example . . . 108
3.7 Discussion . . . 110

4 Model reduction and basis improvement 113
4.1 Introduction . . . 113
4.2 Model reduction for high order ORTFIR estimates . . . 114
4.2.1 Model reduction techniques . . . 114
4.2.2 Model reduction of estimated high order models . . . 115
4.2.3 An asymptotic approach to ORTFIR model reduction . . . 117
4.2.4 A nonasymptotic approach . . . 119
4.3 Iterative basis improvement . . . 121
4.3.1 The basic principle . . . 121
4.3.2 An alternative nonlinear optimization strategy . . . 121
4.3.3 An alternative nonlinear optimization strategy: Box-Jenkins case . 124
4.3.4 A motivating example . . . 126
4.3.5 Discussion . . . 129
4.4 Simulation example . . . 131
4.5 Discussion . . . 134

5 Model predictive control with system-based input parametrization 137
5.1 Introduction . . . 137
5.2 Finding a suitable input parametrization . . . 138
5.3 Model predictive control with input parametrization . . . 141
5.3.1 Infinite horizon model predictive control . . . 141
5.3.2 Nominal stability under constraints . . . 143
5.4 Systematic reduction of the complexity . . . 144
5.4.1 The basic principle . . . 144
5.4.2 LQG balanced reduction . . . 145
5.4.3 Frequency weighted balanced reduction . . . 146
5.5 Improving the constrained performance . . . 148
5.5.1 Motivation . . . 148
5.5.2 Parametrization embedding . . . 148
5.5.3 Parametrization scheduling . . . 150

5.6 Simulation example . . . 154
5.7 Discussion . . . 159

6 Constrained model predictive control with on-line input parametrization 162
6.1 Introduction . . . 162
6.2 Selection of degrees of freedom in quadratic programming . . . 163
6.3 Algorithmic properties . . . 168
6.4 Model predictive control with on-line input parametrization . . . 168
6.5 Simulation example . . . 170
6.6 Discussion . . . 174

7 Identification, optimization and control of a cracking unit 176
7.1 Introduction . . . 176
7.2 Overall strategy . . . 177
7.3 The process . . . 178
7.4 System identification . . . 181
7.4.1 Experiments . . . 181
7.4.2 Parametric identification and validation . . . 182
7.5 Static optimization . . . 192
7.6 Model predictive control . . . 196
7.6.1 Introduction . . . 196
7.6.2 MPC with system-based input parametrization . . . 198
7.6.3 MPC with input parametrization scheduling . . . 200
7.7 Discussion . . . 201

8 Conclusions and recommendations 204
8.1 Review . . . 204
8.2 Contributions of this research . . . 205
8.3 Recommendations for future research . . . 206

Appendix A Proofs for chapter 3 208

List of symbols 214

Bibliography 231

Samenvatting (Dutch summary) 232

Curriculum vitae 234

Chapter 1

Introduction

1.1 Motivation and background

We live in a society in which the needs of the people are insatiable. The demands put on consumer products and products for the industrial markets are ever increasing. The highest quality is demanded at the lowest possible price. Many companies are competing to fulfill the day to day needs of both consumers and industries. This has led to the development of highly complex industrial processes that manufacture these consumer and industrial products. Examples are chemical and petrochemical processes encountered in e.g. oil refineries, discrete manufacturing processes encountered in e.g. food production and aluminium and steel industry, and transportation systems encountered in e.g. aircraft and car industry.

The first requirement put on these systems is that the safety of the people working on or near the site of operation is guaranteed. Secondly, the production must by kept at a constant level and of a quality that is consistent with the needs of the clients. Last but certainly not least, it is important for the companies to produce in a cost efficient way to remain competitive and viable.

These requirements must be met within the limitations and restrictions that any system exhibits. These restrictions come, first of all, from equipment constraints, such as e.g. the finite stroke of valves and the finite cooling capacity. A second restriction comes form the fact that the impact of our industrial production on the environment must be limited to ensure a livable future for generations to come. Governments in many countries have issued strict regulations regarding environmental pollution that must be respected by industrial companies. Also the safety requirement can be viewed as a restriction imposed on the system, e.g. a maximum temperature of a reactor or a maximum pressure in gas tubes.

The demand for optimal profitability leads to the desire to operate the processes such that maximal profit is obtained with the installation in which usually a very large amount of

money is invested. This leads to the desire, assuming there is a sufficient demand for the product that is produced, to drive the system to the limits of its capabilities. If the process unit is driven to its true potential, it is inevitable that one or many of the constraints mentioned above will be active during operation. If not, the process unit is overdimensioned and henceforth a lot of capital is wasted.
The complexity of the processes that are present and being built nowadays expresses itself first of all in their large scale. A large number of process variables can be manipulated to obtain a process behaviour that is satisfactory. For the process behaviour to be satisfactory a large number of variables have to remain at a certain target value or at least within certain limits. These variables and a large number of other relevant variables are measured on the basis of which the manipulation of the process takes place. This large amount of information must be processed in order to come up with an adequate operating strategy. Secondly, the dynamics of the processes can be very complex encompassing both slow and fast dynamics. The dynamics of processes exhibit relevant phenomena that take place at a time scale ranging from milliseconds to several hours. For example pressure and flow phenomena are very fast while temperature balance in a reactor is obtained over a period of several hours and sometimes even days.
Due to the large complexity of the processes, automation of the operation of industrial processes is indispensable these days. Because of the fast development of computer technology, it is possible to use complicated control strategies to increase the plants safety and profitability. Due to the fast development in computer hardware and software, in control engineering more and more advanced control strategies can be applied to improve process operation that were not applicable before due to their complexity. The increased need of the industry to operate in a more efficient yet flexible way in combination with the increased possibilities for control design, makes control engineering a thriving field of research.

1.2 Constrained control of industrial processes

1.2.1 Industrial practice

In many industrial processes restrictions of the system are dealt with by operating the process in such a way that activation of constraints is avoided. By the overdesign of systems one aims to prevent the system from coming close to the limitations of the equipment. With respect to safety constraints or constraints coming from environmental regulations, large safety margins are taken into account to prevent excursions of the system into the forbidden region. With this operating procedure a large amount of the potential of the capital invested in the industrial infrastructure is left unused.
In the chemical and petrochemical industry, where the competition is arguably fiercer than

anywhere else, this possibility to increase the profitability of the capital invested in the equipment is recognized and technology is developed to deal with multivariable control of constrained systems in a somewhat systematic way. This technology is now known under the generic name model predictive control (MPC). Industrial practice of MPC started in the late seventies and early eighties with the industrial work of Cutler [24][22][23] and Richalet [132][130]. At present it is successful in the chemical and petrochemical industry with a long list of successful applications [41]. By now numerous model predictive control algorithms are commercially available [127].
In the chemical and petrochemical industry MPC is a so-called advanced control system (ACS) that runs at a sampling time of one or several minutes. It is used on top of a comprehensive number of relatively simple control loops that run at a sampling time in the order of seconds. These elementary control loops are usually implemented in a distributed control system (DCS) and consist of PID type of controllers, split range etc. These controllers are considered to control the relatively fast phenomena of the plant, say on a second to second basis, without regarding the constraints.
Due to the slow sampling frequency at which MPC is running in current practice, the dynamic range, in which the controller is active, is relatively small.

1.2.2 Model predictive control

Model predictive control is a class of automatic control algorithms that have the following properties.

- At each time instant a performance cost function, that reflects the main process requirement, is optimized on-line to obtain the most suitable control action.
- In this cost function the expected future behaviour of the plant is used. This estimated behaviour is obtained with the use of an explicit computer model of the system dynamics.
- The optimization problem is solved at each time instant anew to incorporate the new information that is obtained in the form of a new measurement of the process variables. The recursive nature is known as the moving or receding horizon principle.

In figure 1.1 this principle is graphically depicted. Model predictive control is able to deal with constraints due to the fact that on-line an optimization problem is solved over future input variables and predicted output variables. The restrictions on the plant inputs and outputs can be added to this optimization problem as a set of inequality constraints.
The basic idea of MPC existed in the literature since the early 60s [46] while industrial practice of MPC started in the late seventies and early eighties with the industrial work of Cutler and Richalet. By now it is a thriving field of research in which many results

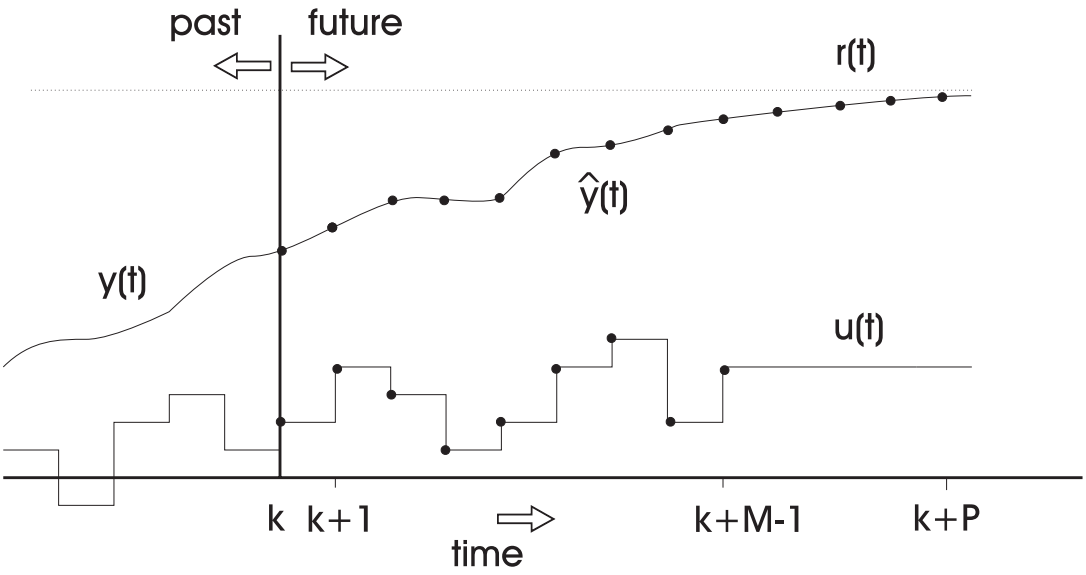

Fig. 1.1: *General principle of the moving horizon approach in model predictive control. At time instant k an optimal input $u(t)$ over a time horizon P is calculated using the measured output $y(t)$, the output prediction $\hat{y}(t)$ and the reference $r(t)$.*

have been obtained and are still being realized and a good cooperation between industry and academia takes place. This is also reflected in the words of Richalet

> *In the last 10 years ... the acacemic community devoted more attention to model based predictive control. This indicates, it is to be noted with some satisfaction, that theory and practice supported each other quite efficiently in this methodology. Progress, which has consequently been steady, is now accelerating* [130]

Progress has been made on several issues. In the early eighties an attempt is made to understand the structure of the control formulation in terms of internal model control [43][44][45]. This failed to give insight in the structure of MPC but led to important observations with respect to controller tuning. The real breakthrough in understanding the structure was the work of Keerthi and Gilbert [67] who showed that model predictive control could be viewed as a way to obtain constrained optimal control while avoiding to solve the notoriously difficult Hamilton-Jacobi-Bellman partial differential equation. They also showed that by letting the prediction horizon tend to infinity, all favorable properties of optimal control like stability for all settings of the tuning parameters are preserved. However, an infinite horizon implied that a solution of an infinite dimensional optimization problem has to be solved on-line which is intractable with the current state of optimization theory. Progress was made in this respect by Rawlings and co-workers who showed how an infinite horizon receding horizon controller can be implemented using only a finite dimensional optimization problem [128][95][140]. Their approach relied on a

finite dimensional parametrization of the infinite dimensional optimization problem. This also led to valuable insight into the stability properties of model predictive control that can now be regarded as a solved problem. Main directions of current research are now robustness properties [69] and model predictive control of constrained nonlinear systems [129]. In both directions considerable number of algorithms are suggested but no consensus is obtained with respect to the best way to deal with these issues.

1.2.3 Possibilities and limitations

Part of the success of MPC can be explained by the fact that two important requirements for industrial process control are met: it is applicable to multivariable systems, and constraints can be incorporated in the control problem.
An important limitation of MPC is caused by the high on-line computational load. The type of optimization problem that has to be solved on-line is, dependent on the cost function and the applied model, a linear, quadratic or nonlinear programming problem for which a vast mathematical theory and a wide range of algorithms exist. Due to the scale of industrial processes and the desire for high performance, the optimization problems that have to be solved on-line are large. The solution must be obtained in a limited amount of time because it is implemented in a receding horizon fashion which implies that as soon as the next measurement becomes available, a new input must be calculated on the basis of this new information.
The high computational complexity has restricted the use of this technology to relatively slow systems encountered in the chemical and petrochemical industry. There are, several reasons why it is desirable to decrease the sampling time that is achievable with this technology. In industries where inherently faster systems are used than in the chemical and petrochemical the motivation is obvious. Higher sampling rates are necessary for fast systems such as high performance mechanical systems, like robot arms and high precision positioning systems, and aerospace systems, such as airplanes and satellites. For these applications the current generation model predictive control is not feasible due to the high computational load while efficient constraint handling can have important benefits in these industrial sectors.
But also for the chemical and petrochemical industry an increase in the speed of model predictive control technology can have important benefits. The basic control layer implemented in the DCS consist of a multitude of control loops that have, almost organically, evolved over time and are expensive to maintain due to the large amount of expertise needed to understand the underlying control structure. It would simplify the design and maintenance of the automatic controller if model predictive control, which is inherently multivariable and has a relatively transparent and intuitive structure, could be used on top of a much leaner basic control layer.

The current process operation where no constraint handling occurs at a small sample time but only at a larger sample time is understandable from a practical point of view but from a more theoretical view it is questionable. For unconstrained systems, both linear and nonlinear, singular perturbation theory [68] provides a firm theoretical basis for separating time scales. For constrained systems this theory is not applicable. Constraints are a system property that is not related to a certain time scale. Therefore the current practice of controlling a system at a high sampling frequency while discarding the constraints and dealing with the process constraints at a higher level running at a lower sampling frequency is unnatural. The reason why constraints are incorporated in a controller running at a low sampling frequency is purely practical but has no theoretical basis. The reason is that with the current state of technology it is not yet possible to run model predictive control at a high sampling frequency due to the large complexity of the on-line computation. If MPC is fast enough to run at the higher sampling rate, this unnatural situation can be avoided.

1.3 Process modeling

1.3.1 Industrial practice

In industry computer models are nowadays indispensable. Models are being used in a wide variety of applications. Large, complex and detailed models are used in process design, process operation (production planning, real time optimization both static and dynamic) and process management (training of process operators on simulators, information management, production scheduling) [6]. In model predictive control and other model based control design strategies more compact and simple models are applied.
There are various ways to obtain a mathematical model of a system. First of all theoretical relations between the different process variables can be connected to obtain a, so called, *first principles model*. These models are characterized by their large complexity and are expensive in terms of man-hours needed to develop such a model. These models are able to capture the process behaviour over a wide range of operation and can therefore be fruitfully used in applications where small computation times are not crucial. Although these models are used more and more in on-line applications like real time optimization and scheduling, with the current state of technology for fast applications more compact models are required.
Another way to model a system is modeling on the basis of experimental data using *system identification* techniques [81]. With this technique a process model is obtained that is able to reproduce the process data and therefore gives an accurate description of the local behaviour of the system.
Many successful applications of identification techniques on industrial process are reported

in literature [178]. The identified models are used in control design but also for the adjustment of free parameters in first principle models, fault detection and process monitoring. This wide range of applications of identified models has spurred the development of industrial identification software packages [75][177].

1.3.2 System identification

In *parametric* system identification the following steps are taken to come from a system to a model of this system [81].

1. An appropriate experiment is designed and executed on the system such that those aspects that are deemed to be relevant for the model are excited.

2. A set of candidate models or model structure has to be chosen which consists usually of a dynamic model that connects the excited inputs with the measured outputs and contains unknown parameters or free variables on various locations inside the model.

3. To determine the best model in the set some criterion function is chosen that measures the distance between model predictions and the process measurements as a function of the free variables. By some mathematical optimization procedure this cost function is minimized to find optimal parameter values.

4. The last step is to assess whether the model is "good enough" for its purpose. This is the model validation step.

An important class of parametric identification approaches is *prediction error identification* [81]. This is an identification approach that has strong relations to maximum likelihood estimation [145], with which statistically optimal models are obtained for data satisfying specific assumptions.
In prediction error identification there are a number of developments that improve and widen the class of applications that can be handled with this technology. An important question that is addressed recently is what identification strategy provides models that are most suitable for control design [48][153]. A result in this respect is the observation that for linear feedback control, closed loop experiments are optimal [60]. The standard open loop identification methods are usually not suitable for identification of accurate models on the basis of closed loop data [153]. This has spurred the developments closed loop identification methods like e.g. the two stage method [152], coprime factor identification [154] and tailor-made parametrization [32]. Also a number of alternatives for prediction error identification are available like frequency domain identification [120][121] and the recently developed subspace identification [117][157][158]. An important open problem is the systematic use of prior knowledge about the system. The question is how the physical knowledge of the system can be used to define model structures that are computationally

attractive and can be applied in on-line applications, while the computation of the unknown dynamics and parameters is still tractable. Some progress is made on this so-called grey box identification [80], but a systematic approach is lacking.

1.3.3 Possibilities and limitations

The specific properties of the processes encountered in industry make it difficult to obtain an accurate model of the behaviour of the system. First of all the systems are usually large. The large scale stems from the high number of inputs and outputs of system but also from the complex dynamics that are modeled. These circumstances give rise to need to solve large optimization problems to find accurate models. Secondly, the systems are usually sensitive for directional changes in the inputs. In system theoretical terms this is denoted as a *bad conditioning in space*. Thirdly, the processes exhibit dynamical phenomena that take place at totally different time-scales: very fast phenomena and slow phenomena have to be accounted for. This is denoted with a *stiff system* or a system with *bad conditioning in time*. If mathematical calculations, e.g. simulation and optimization, are performed for these systems numerical problems are likely to occur.
Another important property of industrial systems from an identification point of view is the fact that the freedom to experiment with the system is limited. Usually the system is in operation during the experiments, therefore it is desirable that these experiments do not disturb the system too much because this results in lower production and hence loss of money while for an accurate model it is desirable to excite the system as much as possible. Therefore the industrial data is less informative than experiments performed in a research environment.
If prediction error identification techniques are used for large scale industrial processes, usually a large scale nonconvex optimization problem must be solved to find the model that is optimal in terms of the chosen criterion. For this problem there exist standard optimization algorithms to solve it. However, due to the large scale of the problem the tools are slow and can get stuck in, so called, bad local minima such that there is no guarantee that a sufficiently accurate model is found. In industrial identification packages nonconvex optimization is circumvented by the use of techniques that require convex optimization [75][177]. Convexity of the optimization problem is generally obtained with the loss of optimality in terms of the prediction error or maximum likelihood criterion. Consequently there is room for improvement of parametric identification technology towards more efficient tools to obtain accurate model for large scale systems.

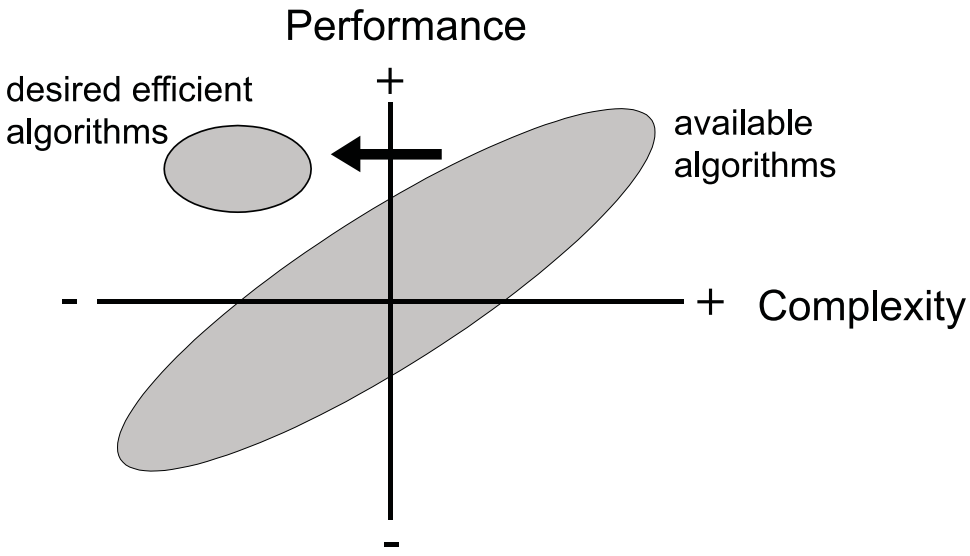

Fig. 1.2: *Graphical representation of the notion of efficient algorithms. Efficiency implies a high ratio between performance and computational complexity.*

1.4 Optimization in control and identification

In control and system identification of industrial processes optimization problems have to be solved. Optimization is used in control design to find an input (trajectory) for the plant or a controller transfer function that is optimal in some predefined sense. In system identification a mathematical description or model of the system is to be found that mimics the data optimally in some prechosen sense. The nature of the systems encountered in industry is such that the optimization problems are large scale and possibly badly conditioned.

In the previous sections it is argued that an increase in efficiency of optimization is desired to improve model predictive control and system identification technology. Efficiency in this context refers to the property of an algorithm that with relatively low complexity a high level of performance is obtained. This concept of efficiency is graphically depicted in figure 1.2. The large grey area signifies the class of available algorithms, for both model predictive control and system identification. High constrained control performance and accuracy of identified models can be obtained with the available technology, but at the expense of a high computational complexity.

Several developments provide an increased efficiency, i.e. make the large grey area in figure 1.2 to shift to the left. First, the efficiency of model predictive control and system identification algorithms is increased by hardware improvements which are happening at an enormous pace e.g. increased processor speed and the use of parallel processor hardware [4]. Also developments in theory and practice of mathematical optimization are proceeding at a considerable pace e.g. interior point methods and the exploitation of

sparsity in the variables [169]. However, just sit and wait for the hardware and optimization software to become fast enough to deal with constrained control and identification of large scale industrial systems is probably not an appropriate choice. The increase in efficiency needed to apply model predictive control to (very) fast sampled systems is so large that it justifies parallel tracks of research to obtain this.
It is desirable to have available a procedure to systematically influence the trade-off between performance and complexity of a model predictive control or system identification algorithm. Consider the situation where e.g. a high performance model predictive control algorithm attains the desired level of performance but the computational complexity is too high for real time implementation. In that case a systematic procedure to reduce the computational complexity of the algorithm such that the performance deterioration is small compared to the decrease in computational load. The influence of such a systematic reduction procedure is graphically represented in figure 1.2 by the horizontal arrow. The question is how this can be achieved.

1.5 Problem formulation

In the previous sections it has been made plausible that new technology is desired to apply model predictive control and system identification to large scale industrial systems that are subjected to constraints. More efficient algorithms are required as well as algorithms in which the computational load can be traded off against the performance. This leads to the following objective that is pursued in this thesis.

> *Investigate the possibilities to improve the efficiency of the optimization occurring in industrial model predictive control and system identification, by influencing the trade-off between performance and computational load.*

The aim of this research is to develop an approach to model predictive control and system identification such that a clear trade-off can be made between performance and complexity. Algorithms are desired that attain a high level of performance with relatively low complexity. Also the user must be able to decrease the complexity, if the application requires this, but at the expense of only a small loss in performance. In this thesis the focus is on industrial systems. From all the properties that these processes possess, special attention is given to

- the multivariable nature of the processes,
- the multi time scale property and

- the process constraints.

These properties have to be regarded, obviously among other aspects, as essential to any industrial system.

1.6 Solution strategy

Several strategies can be followed to obtain an improvement in efficiency of optimization. An obvious strategy is the improvement of the optimization algorithms. The desired flexibility in influencing the trade-off between performance and computational load comes from the possibility to terminate the iteration before optimality is obtained. In this direction considerable progress is made in the optimization community. In this thesis an approach is developed that is parallel to the improvement of optimization algorithms.
The strategy that is further explored in this thesis is the following. Choose the *degrees of freedom* carefully such that only a small number of free parameters is needed to obtain accurate control profiles and models. The way in which the degrees of freedom are chosen in the optimization problem is denoted with parametrization. Efficiency in this context refers to the property of a parametrization that with a relatively low number of free variables a high level of performance is obtained. The desired flexibility to trade off performance and complexity must come from the property of the parametrization that with increased complexity an increase in performance is obtained. The question is how to choose the location of the free variables such that the desired level of efficiency and flexibility is attained.
Some important aspects of the algorithms to be developed are the following. First, nonconvex optimization should be avoided as this gives rise to several local minima in which the optimization can get stuck. Instead, in the developed strategies preferably convex optimization is utilized as this guarantees a unique global minimum that can be calculated fast. This is in line with the current practice in industrial identification [177][75] and also with a common viewpoint in control theory [14] that convex optimization is to be preferred over nonconvex optimization. Secondly, the algorithms must possess handles that give the user the opportunity to trade off the quality of the solution against computational load. For certain applications, e.g. on-line applications like model predictive control, a large emphasis is on computation time rather than on solution accuracy. For other applications, e.g. off-line applications like system identification, the emphasis is on an accurate solution while the computation time must not be excessive. It is therefore useful to have an optimization strategy of which the performance can be traded off against the complexity such that the algorithm can be tailored to the specific application at hand. In this thesis the opportunities are explored to increase the efficiency of optimization, that occur in model predictive control and system identification problems, using a systematic choice of the location of the free variables in the optimization.

1.7 Short outline of the thesis

In chapter 2 some concepts of control, identification, model reduction and optimization theory will be discussed. Also the currently available theory on identification and model predictive control is discussed. The approach developed in this research is discussed conceptually and the chapter is ended with a more detailed outline of the thesis. Thereafter, in part II the identification approach is described. Next, in part III a model predictive control strategy is presented which reasons along similar lines as the identification strategy. Next, in part IV an application is presented of the developed tools on a simulation model of a large scale petrochemical process. In part V the thesis is ended with conclusions and recommendations for further research.

Chapter 2

Control and identification: preliminaries and parametrization approaches

2.1 Introduction

In this chapter the framework is described. Definitions are given of the most important concepts in control, system identification, model reduction and optimization. A brief overview is given of the field of model predictive control and system identification. In these sections special attention is given to issues concerning choice of degrees of freedom in the optimization, i.e. parametrization. Lastly, the problem formulation is made more precise.

2.2 Concepts, notations, definitions

2.2.1 Systems and signals

In many fields of science, e.g. economics, psychology, biology, mathematics, the concept of system is introduced to denote the object of study. In the sciences that deal with physical systems, such as the area of systems and control, it is used to denote the physical process, or subprocesses thereof that is to be modeled or controlled. To distinguish between the system and the surrounding a *system boundary* is defined. These boundaries are chosen regarding the goal that is pursued with the definition. In this thesis, a *system* or *plant* is a part of the real world that is separated from its surroundings by the definition of *system boundaries*. Under this definition a system is clearly something that is living in the real world. However, the word system is also used to denote a mathematical model of a physical system, which is common in mathematics.
The system and its surroundings are coupled through *signals* which are process variables that originate in the surroundings or the system. Signals that originate in the surroundings

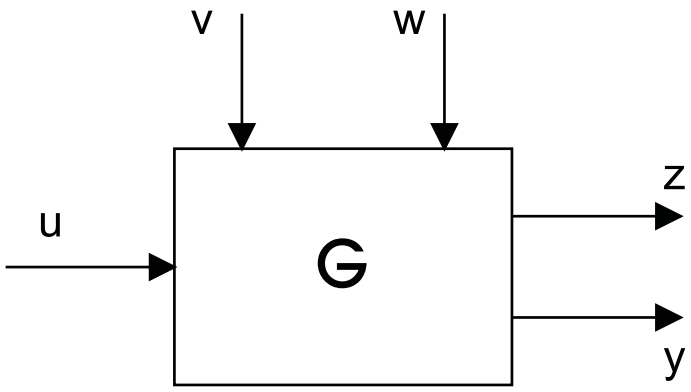

Fig. 2.1: *System description with u the manipulated input, v the unmeasured disturbance, w measured disturbance, y the measured output and z the performance output.*

and influence the system by exerting an action upon the system by passing across the system boundary are called *input variables*. Those variables that originate in the system and which pass outward across the system boundary such that they become available to the surroundings and are regarded relevant are denoted *output variables*. All variables that occur in the system and that are not included in the set of input and output variables are called *internal variables*.
An input is a signal or a vector of signals that drives the operation of a system. A system can have two types of inputs:

1. The controlled input denoted with u can be manipulated. A measurement of this signal is assumed to be available at all time.

2. The uncontrolled input or disturbance, denoted with d, cannot be manipulated. This signal is divided in a part w that can and a part v that cannot be measured.

An output is a signal or a vector of signals that displays the operation of a system. A system can have two types of outputs:

1. The measured output, denoted y, is available for control.

2. The monitored output, denoted z, embodies the qualities to be controlled.

From the definitions above it is clear that we regard systems as operators that map inputs into outputs. In the context of this thesis, this is a feasible standing point as we assume that the inputs that can be used to manipulate the system are predetermined as well as the measured variables. If there is no clear cause-effect between the signals, a less specialized viewpoint is more appropriate. This is e.g. the behavioral approach of Willems and co-workers [166][163][164][165].

Systems and their representations

In this thesis a system is an operator that maps one signal into another, denoted as

$$y = \mathcal{G}.u$$

where u, y are signals and $\mathcal{G}$ denotes the system. We will make a distinction between the continuous time and discrete time. In the former the signals $u(t)$ and $y(t)$ are functions of time $t \in \mathbb{R}$ and in the latter of $t \in \mathbb{Z}$.
A finite dimensional linear time invariant (FDLTI) discrete time system is defined by

$$y(t) = G(q)u(t) \tag{2.1}$$

where the shift operator is defined as $qx(t) = x(t+1)$ and the back shift operator as $q^{-1}x(t) = x(t-1)$ and G rational. Due to the analogy between the shift operator q and the z-transform of the shift operator, the argument q will be replaced by z frequently. A FDLTI discrete time system is stable if it is analytic on and outside the unit circle in the complex plane. This implies that its poles all lie within the unit circle.
For a proper transfer function $G(z)$, an alternative representation by means of a state space realization can be used. A state space representation of a FDLTI discrete dynamical system is given by a set of difference equations

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \quad x(0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is the state vector that represents the internal variables with initial value x_0, $u(t) \in \mathbb{R}^{n_u}, y(t) \in \mathbb{R}^{n_y}$ are the input and output vector, A, B, C, D are the real valued state space matrices of appropriate dimensions that specify the system dynamics. The state space description is an alternative representation of the transfer function description (2.1) by taking $x(0) = 0$ and

$$G(z) = D + C(zI - A)^{-1}B.$$

Another way to represent a system is the Markov parameter description by taking a Laurent series expansion around $z = \infty$ of (2.1). This yields

$$G(z) = \sum_{i=0}^{\infty} M_i z^{-i}$$

where the Markov parameters can expressed in terms of the state space matrices by

$$M_0 = D, \; M_i = CA^{i-1}B.$$

If the state space system is stable, then the controllability Gramian P and the observability Gramian Q can be defined as

$$P = \sum_{i=0}^{\infty} A^i B B^T A^{T\,i} \qquad\qquad Q = \sum_{i=0}^{\infty} A^{T\,i} C^T C A^i$$

The Gramians satisfy the following discrete time Lyapunov equations

$$APA^T + BB^T = P \qquad\qquad A^TQA + C^TC = Q.$$

A state space description of the type (2.2) is controllable if and only if $P > 0$ and observable if and only if $Q > 0$.
State space representations are not unique. Given a state space description that describes the desired input output behaviour, then any coordinate transformation $x = Tz$ with a nonsingular square matrix T yields the same input output behaviour but with a different state vector. The transformed state space matrices are $\{T^{-1}AT, T^{-1}B, CT^{-1}, D\}$. Note that the Markov parameters of the transformed system are the same as those of the original realization. Hence, the Markov parameter system description specifies the system uniquely.
A specific similarity transformation that is frequently utilized, is the balancing transformation [92]. The definition of a balanced state space description is given next.

Definition 2.2.1 *A state space description $\{A, B, C, D\}$ of a stable system is denoted balanced if*

$$P = Q = \Sigma$$

where Σ is a positive semi-definite diagonal matrix.

Given a state space description of a stable system, it is always possible to find a similarity transformation such that the transformed state space description is balanced. Such a transformation is denoted with balancing transformation. In [92] a numerical procedure is given. Other specific realization are the input balanced realization with the property $P = I, Q = \Sigma^2$ and the output balanced realization with the property $P = \Sigma^2, Q = I$.
In this thesis the notion of an all-pass function plays a role. The properties of this transfer function are the following.

Definition 2.2.2 *A discrete time $n_y \times n_u$ transfer function $E(z)$ is called*
right all-pass if $E(z)E^T(z^{-1}) = I$ for $n_y \leq n_u$,
left all-pass if $E^T(z^{-1})E(z) = I$ for $n_u \leq n_y$,
all-pass if $E^T(z^{-1})E(z) = E(z)E^T(z^{-1}) = I$ for $n_u = n_y$.

If the transfer function $E(z)$ is also stable, it is called an inner transfer function. A balanced realization of a square inner function has the remarkable property that

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^T \begin{bmatrix} A & B \\ C & D \end{bmatrix} = I$$

implying that the state space matrices constitute an orthogonal or unitary matrix [135].

Spaces of signals and systems

Many different mathematical spaces can be defined in which signals and systems live and in which specific operation on these objects are defined [70]. Examples of vector spaces are a metric space on which, very loosely speaking, only a distance measure or metric is defined; a normed space, which is a vector space with a metric defined by a norm; a Banach space which is a normed space that is a complete metric space. In this thesis the signals and systems all live in Hilbert spaces.

Definition 2.2.3 [70] *A* Hilbert space X *is a complete inner product space. Here an inner product on* X *is a mapping of* $X \times X$ *into the scalar field* $\mathbb{R}$ *or* $\mathbb{C}$*. The inner product is written*

$$\langle x, y \rangle$$

where x, y *are vectors in* X*. An inner product defines a norm on* X *given by*

$$\|x\| = \sqrt{\langle x, x \rangle}$$

and a metric on X *given by*

$$d(x, y) = \|x - y\| = \sqrt{\langle x - y, x - y \rangle}$$

The Hilbert spaces that play a crucial role in this thesis are, apart from the Euclidian space $\mathbb{R}^n$, are the spaces l_2 and H_2 that will be defined after the definition of some commonly used norms.

Norms of vectors, signals and systems

It is desirable to have some sort of measure of "size" for signals and systems. For this purpose the mathematical concept of a *norm* can be used. An important class of norms are the p-norms.
Discrete signals are nothing more than vectors which can be both finite and infinite dimensional according to the length of the signal. Therefore vector norms defined below can be used to measure the size of both vectors and signals.

Definition 2.2.4 *Norms of vectors in* $\mathbb{C}^n$*. Let a vector* $x = (x_1, \ldots, x_n) \in \mathbb{C}^n$ *be given. Then for* $1 \leq p \leq \infty$ *the p-norm is defined as*

$$\|x\|_p = \begin{cases} \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p} & \text{for } 1 \leq p < \infty \\ \max_{i=1,\ldots,n} |x_i| & \text{for } p = \infty \end{cases} \tag{2.2}$$

An important special case is the familiar *Euclidean norm* $\|x\|_2$. For infinite dimensional vectors the Euclidean norm is also denoted with l_2-norm

The square of the 2-norm $\|x\|_2^2$ is also called the energy of the signal, and the ∞-norm $\|x\|_\infty$ is also called the amplitude or peak value. The signal space l_2 is a space equipped with the 2-norm and the corresponding inner product and is defined as follows.

Definition 2.2.5 [70] *The space l_2 consists of square summable sequences with support in $\mathbb{Z}^+$ equipped with the inner product $\langle f, g\rangle = \sum_{k=0}^{\infty} f(k)g^*(k)$.*

A sequence is said to be square summable if it satisfies

$$\sum_{k=0}^{\infty} \|f(k)\|^2 < \infty$$

In engineering terms such sequences are said to have finite energy.
Norms can also be defined for LTI systems. The system norms that are most widely used in control engineering are the H_2- and H_∞-norm.

Definition 2.2.6 *(H_2-norm). Let $G(z)$ be a stable LTI discrete time system, then the 2-norm is defined as* [175]

$$\begin{aligned} \|G\|_2 &:= \sqrt{\int_{-\pi}^{\pi} trace\{G^*(e^{i\omega})G(e^{i\omega})\}d\omega} \\ &= \sqrt{\sum_{i=0}^{\infty} trace\{M_i^* M_i\}} \end{aligned}$$

where M_i denote the pulse response coefficients or Markov parameters of G.

The Hilbert space for functions H_2 is defined as follows.

Definition 2.2.7 [70] *The space H_2 consists complex valued functions that are analytic outside the unit disc and are squared integrable over the unit circle T, equipped with the inner product*

$$\langle F, G\rangle = \frac{1}{2\pi}\int_{-\pi}^{\pi} trace\left\{F(e^{i\omega})G^*(e^{i\omega})\right\} d\omega$$

This Hilbert space is also denoted with H_2. Note that this space contains only stable systems due to the requirement that the functions are analytic outside the unit disc. A specific subclass of H_2 is $\mathbb{R}H_2$ which contains all real rational functions in H_2.
Another widely used norm is the ∞-norm.

Definition 2.2.8 *(H_∞-norm). Let $G(z)$ be an LTI discrete time system, then the ∞-norm is defined as* [175]

$$\|G\|_\infty = \sup_{\omega\in[-\pi,\pi]} \bar{\sigma}(G(e^{i\omega}))$$

where $\bar{\sigma}$ denotes maximal singular value.

These system norms play a crucial role in modern linear control design as will be explained later.

Bases and orthogonality

In any vector space the concept of linear independence plays an important role. A necessary and sufficient condition for linear independence is given in the following definition.

Definition 2.2.9 [84] *A set of vectors $x_1, x_2, \ldots, x_n$ in a vector space is linear independent if the expression $\sum_{k=1}^{n} \alpha_k x_k = 0$ implies $\alpha_k = 0$ for all $k = 1, 2, \ldots, n$.*

An important consequence is that a vector expressed as a linear combination of linearly independent vectors can be expressed in only one way. This leads to the definition of a basis.

Definition 2.2.10 [84] *A set S of linearly independent vectors is said to be a basis for the space X if S generates X. A vector space having a finite basis is said to be* finite dimensional*, all other vector spaces are said to be* infinite dimensional*.*

An orthogonal basis is only defined in inner product spaces and consists of basis vectors that are mutually perpendicular, i.e. $\langle x_i, x_j \rangle = 0$ for all $i \neq j$. An orthonormal basis has the additional property that $\langle x_i, x_j \rangle = 1$ for $i = j$.

Some stochastic notions

A common assumption in identification is that the disturbance is a realization of a stationary stochastic process. A stochastic process is said to be stationary if the mean and the covariance function is independent of time t. These so-called moments can be used to characterize a stationary stochastic processes and realizations thereof. The mean is denoted by $\mu_x = E\{x\}$ and the covariance function by $R_x(\tau) = Ex(t)x^T(t-\tau)$. These two properties specify the so called second order properties of a stationary stochastic process. When the input signal of a system is regarded as deterministic and the output disturbance is a stationary process the output is not a stationary process. To deal with this hybrid situation an important property of signal in system identification is quasi-stationarity [81].

Definition 2.2.11 *Consider a signal $\{x(t)\}$. This signal is called quasi-stationary if it satisfies the following conditions.*

1. $Ex(t) = \mu_x(t), |\mu_x(t)| \leq C, \forall t$
2. $Ex(t)x^T(r) = R_x(k, r), |R_x(t, r)| \leq C$ *with*

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} R_x(t, t-\tau) = R_x(\tau), \forall \tau$$

For dealing with quasi-stationary signals, use will be made of a generalized expectation operator, denoted by $\bar{E}$ defined by

$$\bar{E}y(t) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} Ey(t),$$

with related (cross-)covariance functions

$$R_{yu}(\tau) := \bar{E}y(t)u^T(t-\tau)$$

and (cross-)power spectral densities

$$\Phi_{yu}(e^{i\omega}) := \sum_{\tau=-\infty}^{\infty} R_{yu}(\tau)e^{-i\omega\tau}.$$

These notions play an important role in system identification.

2.2.2 Control

In control theory methods and approaches are developed to influence a system such that it behaves in a desirable manner. The simplest way to influence the system's behaviour is to apply a signal at the input of the system such that the output behaves in a certain prespecified way. This can be done either without measured information about the output, resulting in *open-loop* or *feedforward* control, or with measurement of the system output leading to *closed-loop* or *feedback* control.
Usually it is not trivial to formulate precisely what we want with the control system. Or the description of the requirements is such that it is difficult to use it as a basis for the control design. Usually in control theory three desirable aspects of the control system are pursued:

1. stability,
2. performance,
3. robustness.

These aspects are further elaborated in the sequel of this section.

Stability

The first requirement of any system is stability, i.e. the signals must remain bounded at all times. Stability can be defined in many ways. We use the following definitions for stability and asymptotic stability [7]. Consider the discrete time state space equation (possibly nonlinear and time-varying)

$$x(t+1) = f(x(t), t) \tag{2.3}$$

and let $x_0(t), x(t)$ be solutions of (2.3) when the initial conditions are $x_0(t_0), x(t_0)$ respectively.

Definition 2.2.12 Stability. *The solution $x_0(t)$ is* stable *if for a given $\varepsilon > 0$, there exists a $\delta(\varepsilon, t_0) > 0$ such that all solutions with $\|x(t_0) - x_0(t_0)\| < \delta$ are such that $\|x(t) - x_0(t)\| < \varepsilon$ for all $t \geq t_0$.*

Definition 2.2.13 Asymptotic stability. *The solution $x_0(t)$ is* asymptotically stable *if it is stable and if $\|x(t) - x_0(t)\| \to 0$ when $t \to \infty$ provided that $\|x(t_0) - x_0(t_0)\|$ is small enough.*

Note that stability is defined for a particular solution and not for the system. Various methods exist to test stability points of a system. An important method that plays a role in this thesis is Lyapunov's method to assess stability .
Lyapunov's second method is originally developed to determine stability of nonlinear dynamic systems. It is developed for differential equations but similar results can be obtained for difference equations. This version is given here [7].

Definition 2.2.14 *A functional $V(x)$ is a Lyapunov function for the system*

$$x(t+1) = f(x(t)), \quad f(0) = 0$$

if

1. *$V(x)$ is continuous in x and $V(0) = 0$.*
2. *$V(x) > 0$ for $x \neq 0$.*
3. *$\Delta V(x) := V(f(x)) - V(x) < 0$ for $x \neq 0$.*

The last condition implies that the dynamics of the system are such that the value of the function V is decreasing for each new time instant until it reaches the origin. It thus seems reasonable that the existence of such a function ensures asymptotic stability of the system around the origin. This is made more precise in the next theorem.

Theorem 2.2.15 *Lyapunov stability theorem. The solution $x(t) = 0$ is asymptotically stable if there exists a Lyapunov function to the system $x(t+1) = f(x(t))$. Further if*

$$0 < \varphi(\|x\|) < V(x)$$

where $\varphi(\|x\|) \to \infty$ as $x \to \infty$, then the solution is asymptotically stable for all initial conditions.

Hence, if we are able to find a function with this properties that is sufficient for stability of the system. Usually it is a difficult task for general nonlinear systems to find such a function but for model predictive control it can be constructed in an elegant way as will be discussed in the sequel of this thesis.

Also for linear systems of the form $x(t+1) = Ax(t)$ a Lyapunov function can easily be found. The positive definite quadratic function is defined by

$$V(x) = x^T X x$$

with a specific matrix X that can be calculated. The derivative of this function along solutions of the linear autonomous system is given by

$$\Delta V(x) = x^T [A^T X A - X] x$$

The matrix

$$A^T X A - X = -Q$$

should be negative definite to guarantee that the origin is an asymptotic stable equilibrium point of the linear system. For a given positive definite matrix Q, a matrix X can be calculated such that $V(x)$ is a Lyapunov function.

Performance

The second requirement is that the controlled system performs better than the uncontrolled system. The definition of the desired behaviour of the control system is not always obvious. It is a mix of, usually subjective and sometimes conflicting, goals as safety, speed, accuracy, reliability, economic profit etc. Performance has two branches:

- The quality with which the effect of disturbances can be suppressed. This is denoted with the *regulator behaviour* of the control system.
- The quality with which the control system is able to follow a predeterminded signal r. This is denoted with *servo behaviour* of the control system.

Performance can be measured in several ways. A way to measure performance for linear control systems that has become popular in the control community since the start of the development of robust control theory [172][35] is to use a system norm or a weighted version thereof to asses the closed loop transfer from one signal to another. For this purpose the generalized plant is a powerful concept in the design of feedback controllers. Two important performance measures in this class are the H_2 and H_∞ performance measures. Let the transfer from the unmeasured disturbance w to the performance output z in figure 2.1 be given by G_{wz}. Then a performance measure that is relevant for disturbance attenuation can be defined by

$$\|G_{wz}\|_2$$

which is the $\mathcal{H}_2$ norm. A deterministic interpretation of this is that it signifies the 2-norm of the pulse response of the system. A probabilistic interpretation is that is equal to the root mean square (rms) value of the output of the system G in response to white

noise excitation of w. This performance measure can be minimized to obtain a feedback controller that is H_2-optimal. This problem can be solved with e.g. state space techniques [35].
If the focus is on robustness properties rather than disturbance attenuation the following performance measure can be defined. Let the uncertainty in the model be represented by an uncertain dynamical transfer function Δ that has an ∞-norm bounded by γ and connects the output z with in input w. Now if we can find a controller that connects the measured output y and the controlled input u such that the infinity norm transfer from w to z is smaller than $1/\gamma$ then the controlled system is robust for this class of uncertainties according to the small gain theorem [175]. The performance measure applied here can be defined by

$$\|G_{zw}\|_\infty$$

This norm has many interpretations. First, it is the worst-case steady-state gain for sinusoidal inputs at any frequency. Furthermore, it is the induced 2-norm in the time-domain [28]

$$\|G(z)\|_\infty = \max_{w \neq 0} \frac{\|z\|_2}{\|w\|_2} = max_{\|w\|_2=1} \|z\|_2$$

The H_∞-norm is also equal to the induced power (rms) norm and also has an interpretation as an induced norm in terms of the expected values of stochastic signals. Especially in robust control this performance measure is widely used [175].
Also a combination of several performance specifications can be made such that a so-called multi-objective performance measure is used [138]. This is a way to deal with the several conflicting specifications that a control system sometimes has to satisfy.
All these interpretations and the availability of good numerical tools to synthesize feedback controllers that are optimal in terms of these norms have made them useful in engineering applications.

Robustness
A third requirement on the control system is robustness, i.e. the capability of the control system to deal with changes of the system that can occur e.g. due to wear, ageing or a mismatch between the system and the model on the basis of which the controller is designed.

2.3 Optimization

2.3.1 Mathematical programming

In control a signal or system has to be searched for that is optimal in some sense while in identification a model has to be searched for that is optimal in some sense. Clearly, opti-

mization theory plays an important role in both fields, control and system identification. In this section a brief review is given of some basic concepts of optimization theory that are used in this thesis.
In optimization theory two classes of problems can be distinguished:

1. mathematical programming problems. In this class only algebraic relations occur.
2. dynamic optimization problems. In this class apart from algebraic relations also differential equations occur.

The first branch will be briefly discussed in the sequel of this section whereas the second branch is discussed in the section on optimal control. In this thesis two convex optimization problems obtain special attention: the least squares problem and the quadratic programming problem; both are given here in more detail.
A fairly general formulation of a mathematical programming problem is the following.

$$\min_{x \in \mathbb{R}^n} f(x) \qquad \text{subject to} \qquad \begin{aligned} h(x) &= 0 \\ g(x) &\leq 0 \end{aligned} \tag{2.4}$$

where the objective function $f(x)$ is at least twice differentiable, the set of equality and inequality constraints $h(x) = (h_1(x), \ldots, h_m(x))$ $g(x) = (g_1(x), \ldots, g_p(x))$ are vector valued function where the functions h_i, g_i are assumed to be continuous and twice differentiable. The gradient of a function and set of functions is denoted as

$$\partial_x f(x) = [\frac{\partial f}{\partial x_1} \cdots \frac{\partial f}{\partial x_n}] \text{ and } \partial_x h(x) = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_n} \end{bmatrix}$$

respectively. Note that the gradient is defined as a row vector.
Several strategies are available to solve this type of problem (see [85][40][97]). We will not give an overview of the algorithms but give some basic results of optimization theory that play a role in this thesis.
The first question that has to be addressed is when is a x^* optimal? Therefore the first important notion that is addressed is that of optimality.

Optimality conditions

An important notion that is explained first is the notion of feasible point and feasible direction. A feasible point x is for unconstrained optimization simply a point in $\mathbb{R}^n$ and for constrained optimization a point that satisfies all constraints. All points that are feasible are denoted with the feasible region. A vector $d \in \mathbb{R}^n$ is a feasible direction at

x if there is an $\bar{\alpha} > 0$ such that all points $\{x + \alpha d \mid 0 \leq \alpha \leq \bar{\alpha}\}$ are feasible points. With respect to the optimization problem stated above, two types of optimal point can be distinguished: local and global minima.

Definition 2.3.1 *A point $x^* \in \mathbb{R}^n$ is said to be*

- *a local minimum of f if there is an ε such that $f(x) \geq f(x^*)$ for all feasible $|x - x^*| < \varepsilon$;*
- *a global minimum of f if $f(x) \geq f(x^*)$ for all feasible x.*

Hence, a global optimum is to be preferred over a local one as no better value of the cost function can be obtained.
The conditions for unconstrained, equality constrained and inequality constrained optimality are given next. The first order necessary condition of unconstrained optimality is given in the next proposition [85].

Proposition 2.3.2 *(First order necessary condition). If x^* is a local minimum point of f, then for any $d \in \mathbb{R}^n$ that is a feasible direction we have $\partial_x f(x^*)d \geq 0$.*

This reduces to the well known requirement that the gradient is zero, $\partial f(x^*) = 0$ if x^* is an interior point of the feasible region.
The second order necessary conditions are as follows.

Proposition 2.3.3 *(Second order necessary condition). If x^* is a local minimum of f, then for any $d \in \mathbb{R}^n$ that is a feasible direction we have*
i) $\partial_x f(x^*)d \geq 0$.
ii) if $\partial_x f(x^*)d = 0$, then $d^T \partial_x^2 f(x^*)d \geq 0$.

This reduces to the well-known requirement of the second derivative to be positive (semi)definite if x^* is a point in the interior of the feasible region.
These conditions have a straightforward extension if equality constraints are present in the problem [85]. This is possible due to the definition of a Lagrangian.

Theorem 2.3.4 *(First order necessary conditions for equality constrained optimization) Let x^* be a local minimum of f satisfying the constraints $h(x) = 0$. Then there is a $\lambda \in \mathbb{R}^m$ such that*

$$\partial f(x^*) + \lambda^T \partial h(x^*) = 0$$

The vector λ is denoted as the *Lagrange multiplier* and

$$l(x, \lambda) := f(x) + \lambda^T h(x) \tag{2.5}$$

as the *Lagrangian* associated with the constrained problem. The necessary conditions can then be expressed in the form

$$\partial_x l(x^*, \lambda) = 0 \tag{2.6}$$

$$\partial_\lambda (x^*, \lambda) = 0, \tag{2.7}$$

the second of these being simply a restatement of the constraints. The second order conditions are given next, where the following notation is adopted for the second derivative of a function

$$F(x) = \left[\frac{\partial^2 f(x)}{\partial x_i \partial x_j}\right]$$

The notation for the second derivative of vectors of functions $h(x)$ is given by the tensor $H(x)$ that always occurs in a product with a vector $\lambda^T = [\lambda_1 \dots \lambda_m]$ such that the tensor is reduced to the matrix

$$\lambda^T H(x) = \sum_{i=1}^{m} \lambda_i H_i(x)$$

where the second derivative of the consecutive functions is given by $H_i(x) = \left[\frac{\partial^2 h_i(x)}{\partial x_i \partial x_j}\right]$. This definition also applies to $g(x)$.

Theorem 2.3.5 *(Second-order necessary conditions for equality constrained optimization) Suppose that x^* is a local minimum of f satisfying $h(x) = 0$. Then there is a $\lambda \in \mathbb{R}^m$ such that*

$$\partial f(x^*) + \lambda \partial h(x^*) = 0$$

If we denote by M the tangent plane $M = \{y | \partial_x h(x^*) y = 0\}$, then the matrix

$$L(x^*) = F(x^*) + \lambda^T H(x^*) \tag{2.8}$$

is positive semidefinite on M, that is $y^T L(x^*) y \geq 0$ for all $y \in M$.

Due to the definition of the Lagrangian the unconstrained and equality constrained conditions are practically the same. For inequality constraints the necessary conditions become much more complicated. The first order conditions, also denoted more commonly with Kuhn-Tucker or Karush-Kuhn-Tucker Conditions, are as follows.

Theorem 2.3.6 *(First order necessary conditions for inequality constrained optimization) Let x^* be a local minimum of f satisfying the constraints $h(x) = 0, g(x) \leq 0$. Then there is a vector $\lambda \in \mathbb{R}^m$ and a vector $\mu \in \mathbb{R}^p$ with $\mu \geq 0$ such that*

$$\partial_x f(x^*) + \lambda^T \partial_x h(x^*) + \mu^T \partial_x g(x^*) = 0 \tag{2.9}$$

$$\mu^T g(x^*) = 0 \tag{2.10}$$

The condition (2.10) is responsible for the fact that the generalization of the unconstrained conditions to the inequality constrained conditions is not as straightforward as it was for the equality constrained case. The second-order necessary conditions are as follows.

Theorem 2.3.7 *(Second-order necessary conditions for inequality constrained optimization) If x^* is a relative minimum point for the problem minimizing f subject to the constraints $h(x) = 0, g(x) \leq 0$, then there is a $\lambda \in \mathbb{R}^m, \mu \in \mathbb{R}^p, \mu \geq 0$ such that*

$$L(x^*) = F(x^*) + \lambda H(x^*) + \mu^T G(x^*)$$

is positive semidefinite on the tangent subspace of the active constraints at x^* which is the space M extended with the tangent subspace of the active inequality constraints.

The second order necessary conditions for this case are an obvious generalization of the previous results.
Summarizing it can be said that the optimality conditions for unconstrained and equality constrained are practically equal. Therefore the complexity of algorithms for these cases are generally in the same order. For inequality constrained problems the first order conditions are much more complicated than for the other classes of problems, which expresses itself in the high complexity and computational load of algorithms.

Convexity
Another important notion in optimization theory is convexity. Convexity is defined for sets as well as functions [85].

Definition 2.3.8 *(Convex sets). A set $\mathcal{S}$ in a linear vector space is said to be convex if*

$$\{x_1, x_1 \in \mathcal{S}\} \Rightarrow \{x := \alpha x_1 + (1-\alpha)x_2 \in \mathcal{S} \text{ for all } \alpha \in [0,1]\}$$

An important example is a set determined by a number of linear inequalities.

Definition 2.3.9 *(Convex function) A function $f : \mathcal{S} \to \mathbb{R}$ is called convex if*

1. *$\mathcal{S}$ is convex and*
2. *for all $x_1, x_2 \in \mathcal{S}$ and $\alpha \in (0,1)$ there holds that*

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2) \tag{2.11}$$

f is called strictly convex if the inequality in (2.11) is strict for $x_1 \neq x_2$.

If a convex function is optimized over a convex set then the optimization problem is denoted with *convex*. Convexity of the optimization problem has the favorable property that a local minimum is also a global minimum. In nonconvex optimization there can be many local minimima that are arbitrarily far away from the global optimum. In some cases one is satisfied with a good local minimum but this is merely because the computational cost to find a better local minimum is disproportionately large compared to the gain in solution accuracy.

Duality

An important concept of optimization theory that plays a role in this thesis is the concept of *duality*. Dual methods are based on the viewpoint that it is the Lagrange multipliers λ that are the fundamental unknowns associated with constrained problems; once these multipliers are known determination of the solution point is simple (at least in some cases). Associated to each convex constrained optimization problem there is the dual function defined as follows for equality constraints [85][40].

Definition 2.3.10 *(Dual function) Given the problem of minimizing $f(x)$ subject to $h(x) = 0$. Let the pair x^*, λ^* be the solution. Near λ^* the* dual function ϕ *is defined by*

$$\phi(\lambda) = \min_x [f(x) + \lambda^T h(x)]$$

So the dual function is a function of the Lagrange multipliers. The optimality conditions for this transformed problem are as follows.

Theorem 2.3.11 *(Local duality theorem) Suppose the equality constrained optimization problem has a solution at x^* with corresponding value r^* and Lagrange multiplier λ^*. Suppose that the corresponding Hessian of the Lagrangian $L^* = L(x^*)$ in* (2.8) *is positive definite. Then the dual problem*

$$\textit{maximize } \phi(\lambda)$$

has a solution at λ^* with corresponding value r^* and x^* as the point corresponding to λ^* in the definition of ϕ.

To deal with inequality constraints only a minor modification of the theorem is required. It suffices to redefine the dual function to

$$\phi(\lambda, \mu) = \min_x [f(x) + \lambda^T h(x) + \mu^T g(x)], \quad \mu \geq 0$$

From the duality theorem it is clear that the optimum of the problem can be obtained both from maximization of the dual function and from minimization of the primal function. So, the theory of duality provides an alternative way to solve a constrained optimization problem. Several algorithms use this theory such as e.g. augmented Lagrangian or multiplier methods [85] and primal-dual interior point methods [169].

2.3.2 Least squares problem

The (full rank) least squares (LS) problem is defined as follows [51]

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 \tag{2.12}$$

with $A \in \mathbb{R}^{p \times n}$ where $p \geq n$ and A full column rank. This problem is more tractable than other p-norm minimizations for two reasons:

- $f(x) = \frac{1}{2}\|Ax - b\|_2^2$ is a differentiable function of x and so the minimizers satisfy the gradient equation $\partial_x f(x) = 0$ which is an easily constructed symmetric linear system which is positive definite if A has full column rank.
- The 2-norm is preserved under orthogonal transformation. This means that we can seek an orthogonal Q satisfying $Q^T Q = I$ such that the equivalent problem if minimizing $\|(Q^T A)x - (Q^T b)\|_2$ is "easy" to solve.

The closed form expression for the optimal solution of (2.12) is given by

$$x^* = (A^T A)^{-1} A^T b$$

where $(A^T A)^{-1} A^T$ is also denoted as the pseudo-inverse of A. Rather than calculating the optimal solution from the closed form expression, more accurate and faster ways are available to solve the full rank least squares problem. In [51] some algorithms are discussed such as the method of normal equations and the use of QR factorization. If A nearly looses column rank the least squares problem is denoted with badly conditioned. In [51] also algorithms are discussed to deal with this situation.

2.3.3 Quadratic programming

Another convex optimization problem that plays an important role in the sequel of this thesis is the *quadratic programming problem*. This is a problem of the form (2.4) where the cost function $f(x)$ is quadratic and the equality and inequality constraints $h(x), g(x)$ are linear. In this thesis no equality constraints $h(x)$ are present in the quadratic programming problems.

$$\begin{aligned} \text{minimize} \quad & \min_{x \in \mathbb{R}} \tfrac{1}{2} x^T Q x + c^T x \\ \text{subject to} \quad & Ax \leq b \end{aligned} \tag{2.13}$$

with $Q > 0$. Sometimes it is favorable from a computational point of view to describe this problem as a constrained least squares problem [11]. Several algorithms are available to solve this problem but due to the inequalities the solution strategies are iterative in

nature and are therefore not as fast as e.g. algorithms to solve the least squares problem. The dual of the quadratic program is an optimization over the Lagrange multiplier λ and is given by [84]

$$\min_{\lambda}\{\tfrac{1}{2}\lambda^T P\lambda + d^T\lambda\} \text{ subject to } \lambda \geq 0 \tag{2.14}$$

with $P = AQ^{-1}A^T$ and $d = AQ^{-1}c + b$. This can be easily seen from definition 2.3.10. For the specific case of the quadratic program (2.13) the dual function is given by

$$\phi(\lambda) = \tfrac{1}{2}\min_x x^T Qx + c^T x + \lambda^T(Ax - b) = \min_x \tfrac{1}{2}x^T Qx + (c^T + \lambda^T A)x - \lambda^T b \tag{2.15}$$

The minimizing argument is given by $x^* = -Q^{-1}(c + A^T\lambda)$. If this is substituted in (2.15) the dual quadratic programming problem (2.14) is obtained. The dual is again a quadratic program but with the dimension of the Lagrange multiplier p and with a nonnegativity constraint instead of a set of general linear constraints.

2.4 Model predictive control

2.4.1 Historic overview

Model predictive control refers to a family of control strategies that has the following properties.

- An on-line constrained optimization of a cost function is performed to find an optimal input trajectory for some horizon in the future,
- A prediction of the future behaviour of the plant is used to specify the cost function,
- the receding (or moving) horizon principle is used to find the optimal input for the next time instant, meaning that the first sample of the optimal input trajectory is used and the whole procedure is repeated for the next time instant.

As the start of linear model predictive control usually the industrial work of Cutler (Dynamic Matrix Control or DMC) [24] and Richalet (Model Algorithmic Control or MAC) [132]. Their optimizing control strategies had and still have considerable success in the industry. Due to the large computational load stemming from the constrained optimization that has to be performed on-line, the applications were restricted to slow plants mainly in the petro-chemical industry. After this commercial success academia started to analyze why this type of control performed so well. This has led to a vast literature on this topic. First, the structure of the strategy was analyzed such that a theoretical analysis of its properties were possible. One example is the work by Garcia and Morari [43][44][45] who tried to analyze model predictive control algorithms by the development of internal model

control (IMC) [93]. This led to insight in tuning for stability and robustness but failed to provide a framework for the analysis of the constrained case.
Next, stability properties were analyzed. From the work on internal model control it became clear that for certain choices of the tuning parameters the closed loop could become unstable. This was true even for stable plants with no constraints. This observation has spurred the research to a model predictive control algorithm that provides constrained closed-loop stability for all values of the tuning parameters. A landmark paper in this respect is the work by Rawlings and Muske [128] who provided such an algorithm by realizing that under appropriate conditions the cost function that is optimized each time instant can be used as Lyapunov function for the closed loop system. These conditions are satisfied when the states are accounted for in the cost function over an infinite horizon. In section 2.4.5 the mathematical details are given on how this can be done with finite dimensional optimization. This has resulted in the development of a great number of so called infinite horizon model predictive control algorithms.
The work of Rawlings and coworkers has, to a great extent, contributed to the insight that model predictive control is strongly related to classical LQ control only with the possibility to incorporate constraints. In [142] it is shown how constrained LQ optimal control can be obtained while still performing finite dimensional optimization problems.
Research directed towards increased closed-loop performance of model predictive controllers is restricted to the unconstrained case. This is possibly because no compact tools are available to measure the performance of a nonlinear system. These tools are necessary here because MPC is a nonlinear control law even if the model and plant are linear. The aspect of performance has not received the attention is deserves. This is probably due to the fact that only relatively slow systems are controlled with MPC with moderate dynamic performance specifications.
Parallel to the developments of stability properties of MPC the robustness properties were analyzed. This has resulted in many different strategies to induce robustness for specific classes of systems. At this point, different than with respect to stability, no consensus is obtained on the best strategy to follow. Still, new approaches are developed [171][16][47] [69].
Considerable research effort is devoted to model predictive control for nonlinear systems [1]. This has resulted in many algorithms for specific subclasses of the extremely rich class of nonlinear systems, such as Wiener and Hammerstein systems [118], feedback linearizable systems [17][101] and more general nonlinear systems [52][156].
Relatively little research effort is directed to improvement of the efficiency of model predictive control. One approach followed is the use of new optimization strategies that are tailored to the specific optimization problem solved in model predictive control [168]. Other research is directed to dispose of the on-line optimization by precalculation of the non-linear control law that is obtained with model predictive control [9][88][125]. A third

way to improve efficiency in MPC is to decrease the number of degrees of freedom in the optimization problem while keeping the loss in performance as small as possible, such as pursued in predictive functional control (PFC,[131]) and [137] where only one degree of freedom is used.
This last approach to efficiency improvement in MPC is especially of importance for this thesis. Several linear parametrizations have been proposed in the literature, which will be described in detail in this section. Also the more general role of parametrization is explained and an analysis is made of the requirement that should be imposed on a parametrization. But first an overview of some of the theory behind model predictive control is given.

2.4.2 Optimal control

The optimal control problem and the search for an answer to this problem has a long history. A fairly general formulation of the optimal control problem, which is a dynamic optimization problem, is the following.

$$\begin{aligned} & minV(u(t),x(t)), t \in [t_0, t_f] \\ \text{subject to} \quad & \dot{x} = f(x(t),u(t)), x(t_0) = x_0 \\ & h(x(t),u(t)) = 0 \\ & g(x(t),u(t)) \leq 0 \end{aligned}$$

where the input is constrained to be in some set $u(t) \in \mathcal{U}$. The input and state $u(t), x(t)$ are vector valued functions of time t This is a minimization of a cost function subject to a set of differential and algebraic constraint and some additional equality and inequality constraints.
To solve this problem basically two possible solution strategies are available.

1. Dynamic programming. This strategy boils down to finding a solution to the Hamilton Jacobi Bellman partial differential equation. Which is, for problems with a large number of states and inputs, a very large computational problem due to Bellman's curse of dimensionality. A consequence of this approach that the optimal input is a function of the state. In this respect the Hamilton Jacobi Belman approach represents a closed-loop optimal strategy.

2. Classical Variational Approach. This strategy boils down to finding necessary conditions for optimality using the Euler-Lagrange equations. A general formulation of the necessary conditions for optimality is provided by Pontryagin Minimum Principle.

A historic overview of the developments in this field from the time of Johann Bernouilli until the present time is given in [146]. In [100] a review is given of these two solution strategies and a comparison between the two is made.
The dynamic programming approach has provided some important results in the specific case of unconstrained linear systems with a quadratic cost function. Kalman [64] showed that the HJB partial differential equation has a solution for this case and provided a method to compute this optimum mathematically using the Riccati difference equation (RDE) for the finite horizon case and the algebraic Riccati equation (ARE) for the infinite horizon case. This is the solution to the well known linear quadratic regulator (LQR) problem. For the discrete time case the LQR problem and its solution are given next.

Problem 2.4.1 Discrete time LQR problem

$$\min_u \sum_{t=t_0}^{T-1} x^T(t)Q_1x(t) + u^T(t)Q_2u(t) + x^T(T)Q_0x(T)$$

with $Q_0, Q_1, Q_2 > 0$ *subject to:* $x(t+1) = Ax(t) + Bu(t)$, *with* $x(t_0)$ *given.*

The solution for finite T is the time-varying state feedback

$$u(t) = -\left[B^TX(t+1)B + Q_2\right]^{-1} B^TX(t+1)Ax(t) \tag{2.16}$$

with $X(t)$ the solution to the Riccati difference equation RDE running backwards in time

$$X(t) = A\left[X(t+1) - X(t+1)B[B^TX(t+1)B + Q_2]^{-1}B^TX(t+1)\right]A + Q_1, X(T) = Q_0 \tag{2.17}$$

The solution for infinite T is a static state feedback

$$u(t) = -\left[B^TXB + Q_2\right]^{-1} B^TXAx(t) \tag{2.18}$$

where X is the solution to the discrete algebraic Riccati equation (DARE)

$$X = A\left[X - XB[B^TXB + Q_2]^{-1}B^TX\right]A + Q_1$$

An important property of the infinite time LQ optimal state feedback is that it is a stabilizing state feedback for all tuning parameters $Q_1, Q_2 > 0$. For a thorough description of linear quadratic optimal control the reader is referred to [72][3]. There also extensions to e.g. tracking systems and disturbance feedforward are given provided with the mathematical proofs.

2.4.3 The basic principle of model predictive control

Continuous time optimal control needs the solution of a function optimization problem. This is an infinite dimensional optimization problem which is in many cases intractable. The main idea of model predictive control is to restrict the set of possible inputs such that only a finite dimensional optimization problem has to be solved. This is done by using a discrete time model and a finite horizon. Instead of calculating a closed form state or output feedback as was possible for the LQR problem, the receding horizon principle is used. This implies that the optimal input trajectory is calculated at each time instant on the basis of the last measurement of the output. In this way the optimal control solution can be utilized in a feedback strategy also if no closed form expression exists for the optimal feedback controller.
Model predictive control is applied in many settings in which e.g. the system description and the cost function differ. The description of model predictive control in this section is restricted to a linear model subject to linear input and state constraints and a quadratic cost function. Some generalizations are given at the end of the section.
In figure 1.1 the general principle of model predictive control is graphically depicted. The aim is to control the following constrained system.

Linear model predictive control
Let the system to be controlled be given by the linear state space description

$$\begin{aligned} x(t+1) &= A_0 x(t) + B_0 u(t), \\ y(t) &= C_0 x(t) \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^{n_u}$ the input vector at time t. The pair (A_0, B_0) is controllable and (C_0, A_0) is observable. The system is subject to linear inequality constraints on the input, input move $\Delta u(k) = u(k) - u(k-1)$ and state

$$\begin{aligned} K_u u(t) &\leq b_u, \\ K_{\Delta u} \Delta u(t) &\leq b_{\Delta u} \\ K_x x(t) &\leq b_x. \end{aligned}$$

The aim is to control this system such that the input signal and the states satisfy the constraints for all time t.
Model predictive control solves this constrained control problem in the following way. At each time instant t_0 the linear quadratic cost function

$$J(u(\cdot)) = \sum_{t=0}^{P-1} \{\hat{x}^T(t) Q_1 \hat{x}(t) + u^T(t) Q_2 u(t)\} + \hat{x}^T(P) Q_0 \hat{x}(P). \tag{2.19}$$

is minimized over a finite prediction horizon P subject to the constraints

$$\begin{aligned} K_u u(t) &\leq b_u, \\ K_{\Delta u}\Delta u(t) &\leq b_{\Delta u} \\ K_x \hat{x}(t) &\leq b_x \end{aligned}$$

for $t = 0, 1, \ldots, P$. In (2.19) for simplicity of notation and without loss of generality the time axis is shifted such that $t_0 = 0$. The state is predicted with a state space model

$$\begin{aligned} \hat{x}(t+1) &= A\hat{x}(t) + Bu(t),\ x(0) = x_0 \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{2.20}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^{n_u}$ the input vector at time t. The pair (A, B) is controllable and (C, A) is observable. Finally, x_0 is the state at the current time instant t_0. The current state vector x_0 can be either measured (full information case) or estimated with the use of an observer (partial information case). Let the optimal solution over the time horizon P be given by

$$u^* = \{u^*(0), u^*(1), \ldots, u^*(P)\}.$$

The first sample of this sequence is used as input for the current time instant t_0, i.e.[1]

$$u(t_0) = u^*(0)$$

These steps are repeated to obtain an optimal input for the next time instant $t_0 + 1$ using the state measurement $x(t_0 + 1)$ and with a time horizon that is shifted one sample into the future. This principle is denoted with moving or receding horizon principle.
Model predictive control in this setting can be regarded as an approximation of the infinite horizon LQ problem 2.4.1 for systems with constraints. For the unconstrained LQ problem a closed form expression for the linear time varying state feedback is present and is given in (2.18). Such a closed form expression is not available for the constrained case. In model predictive control the finite horizon LQ problem is solved on-line to obtain a feedback controller. This is, however, not equivalent with the feedback controller that would result from the solution of the Hamilton-Jacobi-Bellman equation but can be viewed is a moving horizon approximation thereof [67]. In [140] it shown how the infinite horizon LQ controller for constrained systems can be obtained by choosing an appropriate prediction horizon P and terminal state weighting matrix Q_0. This will be discussed in more detail in the section on parametrization issues.

[1]This can only be applied if the computational time is small compared to the sample time, otherwise a computational delay must be introduced, i.e. $u(t_0 + 1) = u^*(0)$

Model structures

In model predictive control algorithms several model structures are used to predict the future behaviour of the system that is used in the calculation of the cost function.

- convolutional models like finite impulse and step response models, [24][132][42][53][91][90][171][171],
- transfer function descriptions like AR(MA)X models [20][21][19][18] [136],
- state space models [128][76][73][78][87][133][174][173].

Although the descriptions differ in the way they describe the system transfer and the disturbance transfer, they can be rewritten into one another.

Cost functions

The most widely used cost function is a quadratic performance specification given in (2.23). This performance criterion can also be extended to a contain a tracking specification

$$J = \sum_{t=0}^{P-1} \{(r(t) - y(t))^T Q_1 (r(t) - y(t)) + u(t)^T Q_2 u(t)\} + z^T(P) Q_0 z(P) \tag{2.21}$$

error and the input signal. Performance spec can also be obtained if $r = 0$ and the working point of the system is shifted with r. In that case the cost function is equal to the one used in the classical LQ problem.

Apart from the 2-norm performance criterion also model predictive controllers with other objective functions have been developed. A number of MPC algorithms apply a l_1-norm performance criterion. For example in [94] the following objective function is chosen

$$J = \sum_{t=1}^{P} q(t) \|(y(t) - r(t))\|_1$$

subject to the constraints, where $q(t)$ are weighting factors. The solution of the minimization, which can be obtained with linear programming, is always at the intersection of constraints, which can be a drawback in practice. Another possibility is a l_∞-norm performance criterion. For example in [15] the following objective function is chosen

$$J = \max_t q(t) \|(y(t) - r(t))\|_\infty$$

subject to the constraints, where $q(t)$ are weighting factors. This can be calculated with linear programming. This objective function is specifically suitable for processes where peaks in the deviation of the plant output and desired output are to be avoided [93]. Another possibility is a multiobjective performance criterion. This is a combination of cost functions as specified above. This can be a suitable way to find a satisfactory trade-off

between several (conflicting) performance requirements.

Soft constraints

State and output constraints may lead to infeasibility of the optimization problem. For example, an output disturbance may push the output out the feasible region such that no feasible input trajectory is able to bring in back in the constraint region. In that case the hard output constraints can no longer be maintained. In practice the constraints are relaxed or softened [174].

A possible way to proceed is to discard the output constraints that cause the infeasibility. A more subtle way that tries to bring the output back in the feasible region is the use of a slack variable ε. The slack variable is introduced to relax the constraints that cause the infeasiblity according to $Ax \leq b + \varepsilon$. Additionally, a term $\varepsilon^T Q_3 \varepsilon$ is added to the cost function where $Q_3 > 0$ is some positive definite weighting matrix. The slack variables are treated as free variables and are optimized such that on the one hand the infeasible constraints are relaxed and on the other hand the constraint violation is minimized. The optimization problem remains a quadratic programming problem because the new variables ε are introduced quadratically in the cost function and linearly in the constraints.

Tuning parameters

Model predictive control has several tuning variables that can be used to influence the performance of the controller. The most important tuning variables are the following. First, the weighting matrices Q_0, Q_1, Q_2 can be used to tune the controller. Secondly, the prediction horizon P. Next, in the partial information case the applied observer has a large influence on the properties of the controller. Also the chosen cost function influence the controller dynamics considerably. These tuning variables are standard and in [77] a detailed discussion is given about their influence and how they can be chosen to obtain a robust model predictive controller.

Another way to influence the controller dynamics rigorously is to change the input parameterization. The input parametrization specifies the way in which free variables are located in the input trajectory. In the setup of model predictive control the input trajectory is parametrized freely over the prediction horizon P. There is a large flexibility in the choice of the location of the free variables that influences the performance of the controller. This is an important subject in this thesis that is treated separately in section 2.4.7.

2.4.4 Computational issues

In this part the optimization problem that will be referred to as the MPC problem is specified. Let the system be given by the state-space description (2.19). The aim is to

control this system while satisfying constraints on the input and state variables

$$\begin{aligned} K_u u(t) &\leq k_u \text{ for all } t \\ K_x x(t, x_0) &\leq k_x \text{ for all } t \end{aligned} \tag{2.22}$$

where $K_u \in \mathbb{R}^{n_u \times n_u}, K_x \in \mathbb{R}^{n \times n}$ are the matrices that specify the input and state constraints. These matrices can have a different row dimension than specified, can be time-varying and input moves can be constrained but they are specified in this way for convenience of notation.

The most commonly used cost function in model predictive control is the quadratic cost function given by

$$J(u(\cdot)) = \sum_{t=0}^{P-1} \{x^T(t) Q_1 x(t) + u^T(t) Q_2 u(t)\} + x^T(P) Q_0 x(P). \tag{2.23}$$

with the weighting matrices $Q_1, Q_2 \geq 0$ and the pair $\{Q_1, A\}$ is detectable. Using the state space description of the model (2.20) this optimization can easily be recast as a vector optimization problem in terms of the vector $U^T = [u^T(0)\ u^T(1) \dots u^T(P-1)]$. This vector optimization problem is given by the quadratic programming problem

$$\min_U \{U^T H U + 2 x_0^T g^T U\} \tag{2.24}$$

subject to $K_U U \leq k_U, K_X U \leq k_X(x_0)$.

The Hessian is given by $H = R + G^T Q G$ and $g^T = H_x^T Q G$ where

$$G = \begin{bmatrix} 0 & & \cdots & 0 \\ B & 0 & \ddots & \\ AB & B & \ddots & \vdots \\ \vdots & \ddots & \ddots & \\ & & & 0 \\ A^{P-1}B & A^{P-2}B & \cdots & AB \quad B \end{bmatrix} \in \mathbb{R}^{(P+1)n \times (P+1)n_u},\ H_x = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^P \end{bmatrix} \in \mathbb{R}^{(P+1)n \times n} \tag{2.25}$$

$$Q = diag(Q_1, \dots, Q_1, Q_0) \in \mathbb{R}^{(P+1)n \times (P+1)n} \text{ and } R = diag(Q_2, \dots, Q_2) \in \mathbb{R}^{P n_u \times P n_u} \tag{2.26}$$

$$\begin{aligned} K_U &= diag(K_u, \dots, K_u) \in \mathbb{R}^{n_{cu} \times P n_u} \text{ and } k_U = \begin{bmatrix} k_u^T & \dots & k_u^T \end{bmatrix}^T \in \mathbb{R}^{n_{cu}} \\ K_X &= diag(K_x, \dots, K_x) G \in \mathbb{R}^{n_{cx} \times P n_u} \text{ and } k_X(x_0) = \begin{bmatrix} k_x^T & \dots & k_x^T \end{bmatrix}^T - K_X H_x x_0 \in \mathbb{R}^{n_{cx}} \end{aligned}$$

This can be easily seen by realizing that the stacked state vector $X^T = [x^T(0)\ x^T(1) \dots x^T(P-1)]$ can be written as $X = GU + H_x x_0$ using the state space equations (2.20).

Model predictive control in this setting amounts to a nonlinear state feedback which is continuous and piecewise affine [9]

$$u(t) = f(x(t)) = F_{\mathcal{A}}x(t) + b_{\mathcal{A}} \tag{2.27}$$

where the state feedback vector $F_{\mathcal{A}}$ and the bias term $b_{\mathcal{A}}$ is dependent on the set of constraints that is active. That this is true can easily be seen from the first order necessary conditions for optimality. Let the active (input) constraints at the optimal solution of (2.24) be given by $LU = l$. The first order necessary conditions for optimality are given by the Lagrangian system

$$\begin{bmatrix} H & L^T \\ L & 0 \end{bmatrix} \begin{bmatrix} U \\ \lambda \end{bmatrix} = \begin{bmatrix} -gx_0 \\ l \end{bmatrix}$$

Clearly, the optimal input trajectory can be described by $U = \phi x_0 + \varphi$ which is affine in x_0 for any set of active constraints. For output constraints an additional affine term in x_0 occurs due to the affine dependency of the constraint on x_0 but the optimal input can still be described as (2.27). In [9] this observation is used to design a model predictive control algorithm in which all affine state feedback relations are computed a prior to construct the nonlinear state feedback. This nonlinear state feedback is implemented instead of solving the optimization problem on-line. In this way all computational load is shifted of-line at the cost of a decreased flexibility in on-line tuning.
In the unconstrained case model predictive control is equivalent to a linear state feedback

$$u(t) = [I\ 0 \dots 0]H^{-1}g = Fx(t).$$

The state feedback F in this equation is equivalent to the first sample of the LQ optimal linear time varying state feedback (2.16) resulting from the Riccati difference equation (2.17). Hence, the unconstrained finite horizon LQ optimal state feedback can be obtained either by solving the Riccati equation or solving the quadratic optimization problem (2.24).
Clearly, the structure of the version of MPC utilized in this thesis is a nonlinear state feedback controller where the state feedback is calculated on-line by the solution of the quadratic program. Several useful extensions are possible some of which are discussed next.

Output feedback
In the case of full information, all system states are measured, model predictive control is a a nonlinear state feedback. In the case of partial information, only a measurement of the output is available, a state observer is needed to reconstruct the states from the output measurement. In that case linear model predictive control has the structure of an

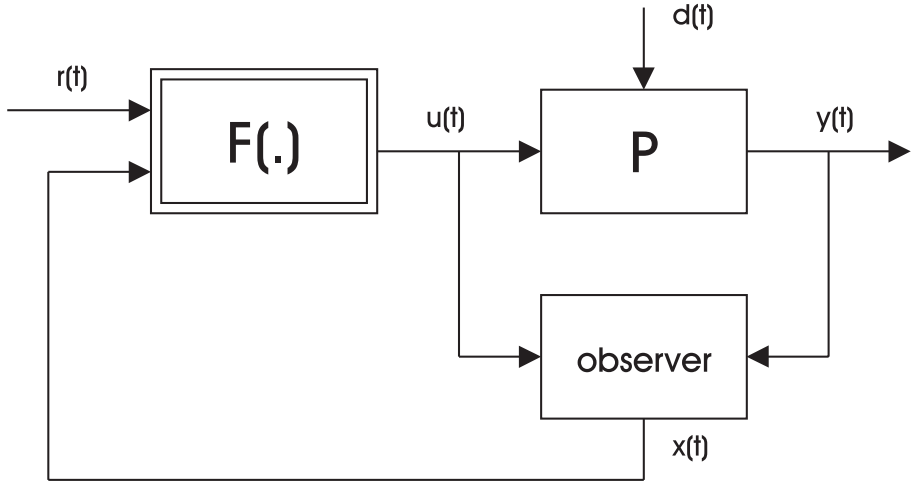

Fig. 2.2: *Model predictive control as nonlinear state feedback controller.*

observer based nonlinear state feedback. A graphical representation of this structure is given in figure 2.2.
Basically two types of observers can be used in model predictive control. The filter type and the predictor type observer [10]. The predictor type observer is given by

$$\hat{x}(t+1) = A\hat{x}(t) + Bu(t) + K(y(t) - C\hat{x}(t))$$

If no constraints are active the LTI controller that results is given by

$$\begin{aligned} \hat{x}(t+1) &= (A - KC - BF)\hat{x}(t) + Ky(t) \\ u(t) &= F\hat{x}(t) \end{aligned}$$

where F is the linear state feedback that results from the on-line optimization. The controller is strictly proper, i.e. contains a delay, because the output measurement at this time instant $y(t)$ is applied to correct the state prediction at the next time instant $\hat{x}(t)$. In the filter type observer no delay arises because the measurement at time t is used to correct the state prediction at t. The filter type observer is given by

$$\begin{aligned} \hat{x}(t) &= A\hat{x}(t-1) + Bu(t-1) + K(y(t) - C[A\hat{x}(t-1) + Bu(t-1)]) \\ &= (I - KC)A\hat{x}(t-1) + (I - KC)Bu(t-1) + Ky(t) \end{aligned} \tag{2.28}$$

The transfer function description of this observer is given by

$$\begin{aligned} x(t) &= \left(I - (I - KC)Aq^{-1}\right)^{-1} \begin{bmatrix} (I - KC)Bq^{-1} & K \end{bmatrix} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix} \\ &= (qI - (I - KC)A)^{-1} \begin{bmatrix} (I - KC)B & Kq \end{bmatrix} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix} \end{aligned}$$

$$= (qI - (I-KC)A)^{-1} \begin{bmatrix} (I-KC)B & (I-KC)AK \end{bmatrix} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix} + Ky(t)$$

where in the last relation the property

$$\begin{aligned} (q - (I-KC)A)^{-1} &= q^{-1}I + (I-KC)Aq^{-2} + (I-KC)^2A^2q^{-3} + \cdots \\ &= q^{-1}\left(I + (q-(I-KC)A)^{-1}(I-KC)A\right) \end{aligned}$$

is used to eliminate the forward shift q with respect to the output $y(t)$. The linear time invariant controller that is equivalent to unconstrained model predictive control with the filter type observer is therefore given by

$$\begin{aligned} \hat{z}(t+1) &= (I-KC)(A-BF)\hat{z}(t) + (I-KC)AKy(t) \\ u(t) &= F\hat{z}(t) + FKy(t) \end{aligned}$$

The prediction of the state of the system is given by $\hat{x}(t) = \hat{z}(t) + Ky(t)$. Note the direct feedthrough from the measured output.
Assuming that the system dynamics obeys the state space relations, the prediction error $e(t) = x(t) - \hat{x}(t)$ for the two observer forms can be written as [7]

$$e(t+1) = (A-KC)e(t) \qquad\qquad e(t+1) = (A-KCA)e(t)$$

respectively. If the pair $\{A, C\}$ is observable, which implies observability of $\{A, CA\}$ if $A \neq 0$. Then the eigenvalues of these autonomous systems can be placed arbitrarily by the design of an appropriate feedback matrix K. This can be done with e.g. pole placement techniques or minimization of an LQ like function such that the well-known Kalman filter is obtained. A Luenberger observer is obtained if K is chosen such that $CK = I$. With this choice the output is reconstructed without error and the order of the observer can be reduced to $n - n_y$.
Both observer forms can be applied in model predictive control. However, if the computation time is significantly smaller than the sample time, usually the computed input sample is sent to the plant immediately as soon as the calculation is finished. In that case the filter type observer is especially appropriate because the measurement $y(t)$ is used to obtain the current input $u(t)$. This direct feedthrough of the output information can result in increased control performance.

Reference tracking

In the set up discussed above, model predictive is presented as a regulatory controller as no reference signal is present. Servo behaviour of the controller can be introduced as follows. Let the goal be to have the output $y(t)$ track a reference signal $r(t)$. For this

purpose the cost function can be altered to

$$J_{track} = \sum_{t=0}^{P-1} \{(r(t) - y(t))^T Q_1 (r(t) - y(t)) + u^T(t) Q_2 u(t)\} + x^T(P) Q_0 x(P).$$

In this cost function future values of the reference signal $r(t)$ can be used. However, in this setting still an offset between the desired output level $r(t)$ and and the plant output $y(t)$ can remain.

An important requirement for servo controllers is that they are able to track step like reference signals with no static error. In linear control design this can be introduced by adding an integrator to the controller. A similar approach can be followed for asymptotic tracking of model predictive control. Offset free tracking for asymptotically constant reference signals can be obtained in various ways. First the system can be augmented with an integrator at the input, such that $\Delta u(t) = u(t) - u(t-1)$ is the input and $y(t)$ the output. The integral action is then responsible for the offset free tracking. The integrator can also by placed at the output. A different approach is to add a constant disturbance observer to the controller. With this also integral action and asymptotic tracking is obtained.

2.4.5 Stability issues

In the early eighties it became clear that model predictive control did not provide a stable closed loop system for all values of the tuning variables. This is a property that for example LQ optimal control did possess. It is a favorable property because no tuning is necessary to obtain stability and the tuning can be adjusted on-line without the danger of running unstable. Especially the academic community tried to gain insight in the stability problem and developed numerous approaches to obtain a closed loop system with guaranteed stability.

In [122] it is shown how a *contraction constraint* can be used to force the state to decrease with time and stability follows independent of the various parameters in the objective function.

Another approach is to use the cost function J as a Lyapunov function for the closed loop. One wishes to show that

$$J(x(t)) - J(x(t+1)) > 0 \text{ for } x(\cdot) \neq 0 \tag{2.29}$$

because a Lyapunov function must be decreasing following theorem 2.2.15. There are several ways to assure that this is the case.

When a constraint for the final state $x(k+N) = 0$ is added, it can be shown [74] that the cost function is a Lyapunov function for the closed loop. A property of this approach is that the state constraint may not be feasible. This means that there is no input trajectory that satisfies all the constraints. Due to this infeasibility problem this approach may not

be appropriate for industrial systems.
In [126] it is used that the lefthand side of the requirement (2.29) can be written as

$$J_N(x(k)) - J_N(x(k+1)) = [x^T(k)Qx(k) + u_N(x(k))Ru_N(x(k))]$$

$$+[J_{N-1}(x(k+1)) - J_N(x(k+1)))]$$

where J_N is a quadratic cost function with prediction horizon N and $u_N(x(k))$ is the optimal input at time instant k. They go on to shown that the second term approaches zero as $N \to \infty$ and that there exists a finite N^* such that for $N > N^*$ the first term dominates over the second term resulting in a stable closed loop. Unfortunately the solution to a nonconvex minimax problem is necessary to determine the bound N^*. Therefore, for large scale industrial systems this approach does not seem feasible.
A third and very elegant approach is given in [128]. There it is shown that the requirement (2.29) is fulfilled if the output horizon is set to infinity $N = \infty$. They show how this infinite dimensional optimization problem is solvable with finite dimensional quadratic programming.

2.4.6 Performance issues

In MPC literature much attention is given to stability and robustness issues while relatively little attention has been given to the development of sensible performance measures and ways to increase the performance of constrained systems. One of the main difficulties in finding a suitable performance measure for constrained systems is its inherent nonlinearity. For linear systems a system norm can be used to compress performance information into one number while in the nonlinear case this is not feasible.
In model predictive control basically two performance measures can be detected. The first, denoted with *local or open-loop performance*, is a measure for the quality of the solution to the on-line optimization problem. The second, denoted with *global or closed-loop performance* is the performance of the interconnection of the system with the model predictive controller. In this thesis these types of performance are defined as follows.

Definition 2.4.2 Local performance. *The local performance of model predictive control algorithms is determined by the difference between the optimized control profile u^* and the optimal profile for an infinite horizon cost function.*

Clearly no mathematical performance measure is given here but an indicator of constrained performance.

Definition 2.4.3 Global performance. *With global performance of a model predictive controlled system, the tracking and servo properties of the nonlinear closed-loop system is denoted.*

Obviously, the objective of control design is to obtain satisfactory global performance. The local performance is therefore instrumental to obtain global performance.
Usually, the local behaviour will differ from the global one even if the model is equal to the system. This is because the sequence of inputs calculated on-line for the current time instant will generally be different from the sequence of first samples of the calculated input trajectories that is actually implemented on the plant. In some special cases the optimal input trajectory that is calculated at the current time instant is equal to the global behaviour of the control system in case no disturbances are present. These cases are the infinite horizon case and the terminal state constrained case. Because with these strategies the optimal policy that is calculated at this time instant is also optimal for the next and no constant switching between policies occurs. Therefore it seems more sensible from a performance point of view (as it was from a stability point of view) to use the infinite horizon strategy.
Still, measuring closed-loop performance for model predictive control is an open issue. For specific situations the closed loop behaviour of the model predictive controlled plant can be assessed by simulation. But it is laborious to simulate all relevant situations of with different reference signals, disturbance signals etc. It would be desirable to have a performance measure for the closed loop system as a whole, similar to the linear case where the H_∞ and H_2 norms are available to assess the performance of a control system without the need to simulate the behaviour for a large number of situation. Such an approach is lacking for model predictive control. One important reason for this is that at the current state of technology no sensible measure for the performance of nonlinear systems is available. Model predictive control is a nonlinear control technique. The closed-loop system will be nonlinear, even if the system is linear system. Certainly, it is more difficult to find a sensible measure of performance for the very rich class of all nonlinear systems, than for the highly structured case of model predictive controlled linear systems. But at this point in time, no measure is available.
In this thesis the local infinite horizon performance measure is used to assess the quality of control. With this performance measure, optimal performance is obtained if the applied control is a solution to the HJB equations.

2.4.7 Parametrization issues

An important determinant of the local as well as the global performance of model predictive control is the location and number of the free parameters in the input trajectory. This choice is denoted with the term *input parametrization*.
The highest local performance level can obviously be obtained if the input trajectory is parametrized freely over the prediction horizon. This leads to the largest possible freedom to decrease the cost function and hence the lowest possible value of this cost function.

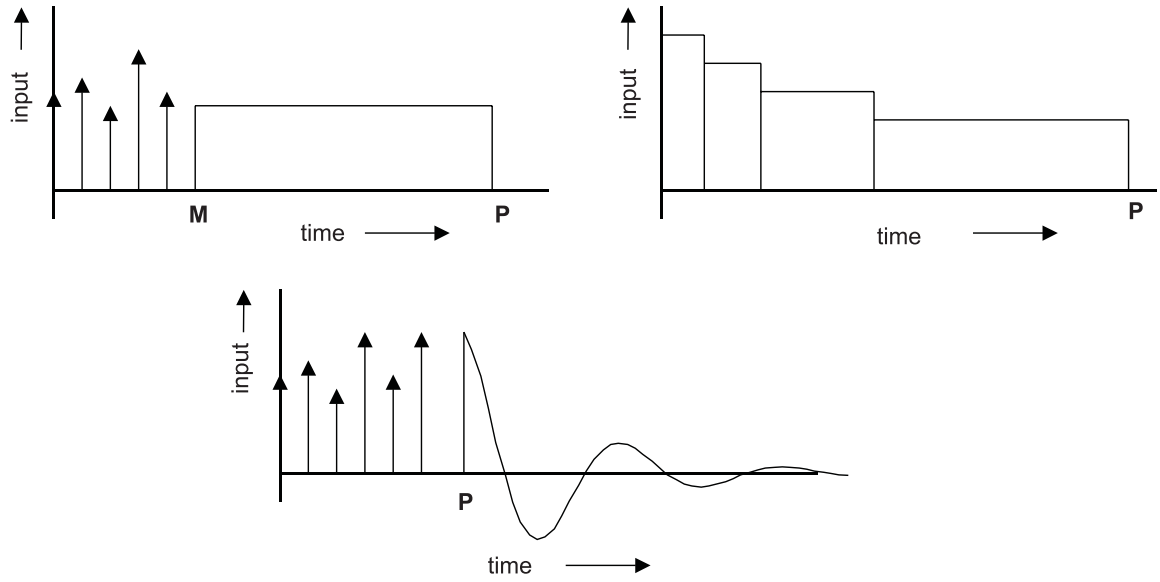

Fig. 2.3: *Common input parametrizations in model predictive control. Pulse parametrization with control horizon M (top left), blocking (top right) and parametrization for constrained infinite horizon LQ optimal control where from time instant P an LQ optimal control profile is assumed to be used (bottom).*

However, in many cases this will yield an on-line optimization problem that is intractable for fast sampled systems.

The input trajectory U can be parametrized in various ways to restrict the number of degrees of freedom. A general way to describe the parametrization of U is such that the class of obtainable input trajectories is a subspace of $\mathbb{R}^{Pn_u}$ can have a (considerably) lower dimension than the full space. This choice can be described by the linear parametrization

$$U(\theta) = \phi\theta \text{ with } \phi \in \mathbb{R}^{Pn_u \times n_\theta}, \theta \in \mathbb{R}^{n_\theta} \tag{2.30}$$

where θ is the free parameter and ϕ is a user-chosen matrix of which the columns form a basis for the space of all input trajectories that can be achieved. For each input sample over the horizon this parametrization is described by

$$u(t,\theta) = \varphi^T(t)\theta \text{ with } \varphi^T(t) \in \mathbb{R}^{n_u \times n_\theta}, \theta \in \mathbb{R}^{n_\theta}$$

where n_θ is the number of free variables in the parametrization. With the input space parameterized by the column span of ϕ, the optimization problem that has to be solved is, similar to conventional MPC: a quadratic programming problem. This is given by $\theta^* = arg\min_\theta J(\theta)$ with

$$J(\theta) = \theta^T\phi^T H\phi\theta + 2x_0^T g^T\phi\theta \tag{2.31}$$

subject to the constraints

$$K_U\phi\theta \leq k_U, K_X\phi\theta \leq k_X(x_0)$$

The matrix ϕ is clearly a design variable that is able to influence the controlled behaviour in a crucial way. The main question is, how this freedom can be used in a structured way to obtain a clear trade-off between closed-loop performance and optimization complexity. A first way to restrict the number of degrees of freedom is the introduction of a control horizon, i.e. parameterize the input freely over a certain horizon and keep the input fixed thereafter. In figure 2.3 this is graphically depicted. This parametrization is e.g. used in [24] in a finite horizon MPC and in [128] in an infinite horizon MPC. A second option is the use of blocking [134]. With blocking the input trajectory is built up by a number of blocks over which the input is held constant. A graphical representation is given in figure 2.3. The conventional input parameterization and parameterization using blocking can be obtained with respectively:

$$\phi_{pulse} = \begin{bmatrix} I & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & & I \\ & & \ddots & \vdots \\ 0 & \cdots & 0 & I \end{bmatrix} \in \mathbb{R}^{Pn_u \times Mn_u} \quad , \phi_{block} = \begin{bmatrix} I & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ I & 0 & \cdots & 0 \\ 0 & I & & \vdots \\ \vdots & \vdots & & 0 \\ & I & & 0 \\ & 0 & & I \\ & & & \vdots \\ 0 & 0 & 0 & I \end{bmatrix} \in \mathbb{R}^{Pn_u \times Mn_u}$$

where M is the control horizon. A drawback of this choice is that the performance can be bad for specific sets of active constraints. Another drawback is that, in case state constraints are present, the optimization problem can run into feasibility problems due to the restricted control freedom imposed by the input parametrization.

In predictive functional control (PFC, [131]) the input is parametrized in terms of polynomials and goniometric functions. In [137] a parametrization of the input is proposed in terms of one interpolation variable, such that in the unconstrained case LQ control is obtained and in the constrained case a convex combination of LQ and mean level control is calculated using linear programming such that all constraints are feasible. These approaches all have the property that the performance can be far from the optimal depending on the set of constraints that is active.

An approach that obtains optimal performance both in the unconstrained and constrained case in terms of the infinite horizon LQ criterion, is the algorithm presented in [141]. Here the input is parametrized freely for the first P samples and is fixed to unconstrained LQ optimal control thereafter. The contribution of this tail to infinity can be described by the terminal state weighting matrix Q_0 that satisfies the Lyapunov equation

$$(A - BF)^T Q_0 (A - BF) + (Q_1 + F^T Q_2 F) = Q_0$$

where F is the LQ optimal state feedback matrix given in (2.18). This provides constrained LQ optimal control if the prediction horizon P is sufficiently large, i.e. no constraints are active after this time instant. If a first choice of P is suboptimal it is increased and the control law is recalculated. For short horizons P this may be feasible strategy, however for longer horizons and/or shorter sampling intervals this may be computationally too demanding. This parametrization is depicted in figure 2.3.
With respect to parametrization in model predictive control the following questions are important. How is the trade-off between performance (loss) and number of degrees of freedom and hence optimization complexity and speed? How well is the designer able to tune the algorithm for this trade-off? Related to this question is the question whether the performance loss compared to the fully parametrized case can be quantified? If this quantification can be given, this provides a good tool to decide on a tuning setting. Is the parametrization such that stability and feasibility is guaranteed?

2.4.8 Computational complexity

The on-line computational load in linear model predictive control is almost exclusively determined by the quadratic programming problem that must be solved. Several optimization algorithms are available to solve a quadratic programming problem. An overview of recent developments in quadratic programming algorithms can be found in [79]. Two strategies are most widely used: active set methods and interior point methods.
Active set methods solve a quadratic programming problem roughly through iteration over the determination of an active set of constraints followed by a line search. After the line search a decision is made for the new active set of constraints which is again followed by a line search. This iteration is proceeded until convergence. The computational complexity is at best of the order $O(N^2)$ where N is the degrees of freedom in the quadratic programming problem [49]. The efficiency of an active set method can be improved greatly by the selection of a starting value for the iteration that has an active set that is not too far from the active set of constraints at the optimal solution.
The second class of algorithms is relatively new and, especially for large scale problems, also more efficient. This is the class of interior point methods [98][168] which covers a wide range of algorithms. The computational complexity of this type is $O(N)$ which is considerably faster than active set methods.
But even with a fast interior point algorithm high performance model predictive control of fast systems is not possible. The most important determinant in the computational load is the number of degrees of freedom in the quadratic programming problem. The computational load can be made smaller by lowering the degrees of freedom in the optimization. An important question is how the number of degrees of freedom can be decreased without degrading the solution too much.

2.5 System identification

2.5.1 Historic overview

System identification deals with the problem of building mathematical models of dynamical systems based on observed data from the system [81]. One of the oldest and well known estimation techniques is the maximum likelihood estimation. This has the property that it is asymptotically efficient and has the lowest possible parameter variance which is given by the Cramer-Rao bound. This favorable property is desirable but can for a large class of problems only be attained at a very high computational cost. Under mild conditions, however, the maximum likelihood estimation is equivalent to prediction error identification [81]. This approach to identification is followed in this thesis.
Apart from prediction error identification, which is a time-domain identification technique, a frequency domain identification approach can be followed [120][121]. In this approach the measurements, which are inevitably in the time domain, are transformed to a spectral estimate of the transfer using a nonparametric method like ETFE possibly with some degree of windowing. Next, a transfer function is fitted on this spectral estimate. One of the many discussion going on in the identification community was about the distinction between the two lines. Some consensus about this topic has been reached on this topic. There exists a unitary mapping between the time and frequency domain so any procedure in the time domain has a counterpart in the frequency domain and vice versa. The only point at which they differ is the algorithmic part. From an algorithmic point of view for example filtering in the frequency domain is just multiplication of the corresponding points of the frequency response which is computationally easier than processing a signal with a linear filter.
Another important question that is addressed in the identification community is which identification approach is best suitable for control design [48][153]. A result in this respect is the observation that for linear feedback control, closed loop experiments are optimal [60]. This is due to the fact that experiments in a closed-loop setting emphasizes that frequency range that is important to model accurately: the cross-over region. By now numerous approaches are available to estimate accurate models from closed loop data e.g. two stage method [152], coprime factor identification [154] and tailor-made parametrization [32].
A relatively recent alternative to prediction error identification is subspace identification e.g. N4SID [117] and MOESP [157]; see [158] for an overview on subspace identification. In subspace identification no minimization of a criterion is performed but a number of projections is executed to obtain a state space description from data. In this way statistical optimality is lost but it offers a number of computational advantages as projection is a relatively simple operation that can be executed efficiently up to very high dimensions. Subspace identification is an area of ongoing research where also the relations to predic-

tion error identification are becoming clearer [82].
In the sequel of this section the emphasis will be on open loop prediction error identification.

2.5.2 Elements of system identification

As a framework for identification the black-box identification setting according to Ljung [81] will be adopted. It is assumed that the system that has generated the data is the linear time-invariant, discrete time system

$$y(t) = G_0(q)u(t) + v(t) \tag{2.32}$$

with $G_0 \in H_2$, u a quasi-stationary signal, and v a stationary stochastic process with rational spectral density, that can be described by

$$v(t) = H_0(q)e(t) \tag{2.33}$$

with e a zero-mean white noise process with variance σ_e^2, and both $H_0, H_0^{-1} \in H_2$ and monic, i.e. $\lim_{|z|\to\infty} H_0(z) = I$. The system is determined by the pair

$$\mathcal{S} : \{G_0, H_0\}$$

which will be denoted by the data-generating system.
In system identification basically four steps have to be taken to come from a system to a model.

1. Test the system by an appropriate experiment.
2. Select a set of candidate models.
3. Use a rule by which candidate models can be assessed using the data.
4. Validate the accuracy and adequacy of the model.

These will be briefly explained in the sequel.

Experiment design
In the first step an experiment has to be designed to retrieve input/output data

$$\mathcal{D} : \{u(t), y(t)\}_{t=0,1,\ldots,N}$$

Important in this step is the input signal that is used to excite the plant. This must be such that the relevant dynamics of the plant is excited. Although experiment design is a very important topic, in this thesis no emphasis is placed on this. The reader is referred

to the many textbooks on the topic of experiment design [50].

Model set selection

In the second step a set of appropriate models has to be selected. This *model set* is formally described by

$$\mathcal{M} : \{(G(q,\theta), H(q,\theta)) \in H_2 \times H_2 \mid \theta \in \Theta \subset \mathbb{R}^n_\theta\}. \tag{2.34}$$

There are several black-box parametrizations available for the transfer of G and H. Most of them parametrize the two transfer functions as a quotient of two polynomial matrices. The coefficients in these polynomial matrices act as parameters. The most common ones are given in table 2.1. An important special case is the ARX model structure. For the SISO case this is determined by

$$\begin{aligned} A(q^{-1},\theta) &= 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_a} q^{n_a} \\ B(q^{-1},\theta) &= b_0 + b_1 q^{-1} + b_2 q^{-2} + \cdots + b_{n_b} q^{n_b} \end{aligned}$$

with

$$\theta = [a_1 \cdots a_{n_a}\ b_0 \cdots b_{n_b}]^T \in \Theta \subset \mathbb{R}^{n_b+n_a+1}$$

where Θ is the set for which the polynomial $A(q^{-1},\theta)$ is Hurwitz. An important property of this model structure is that the one step ahead predictor is *linear in the parameter*, i.e.

$$\hat{y}(t|t-1;\theta) = \varphi^T(t)\theta$$

where *regression vector* is $\varphi^T(t) = [y(t-1)\cdots y(t-n_a)\ u(t)\cdots u(t-n_b)]$. All these properties also hold for the multivariable case. This model structure is also denoted as a linear regression model structure because the optimization of the prediction error cost function defined below boils down to solving a linear regression problem. The other model structure that has this property is the finite impulse response model (FIR). This structure has disturbance model $H = I$. Model structures that have this property are also said to be of output error (OE) type.

Model structure	$G(q,\theta)$	$H(q,\theta)$
BJ	$F^{-1}(q^{-1},\theta)B(q^{-1},\theta)$	$D^{-1}(q^{-1},\theta)C(q^{-1},\theta)$
ARMAX	$A^{-1}(q^{-1},\theta)B(q^{-1},\theta)$	$A^{-1}(q^{-1},\theta)C(q^{-1},\theta)$
ARX	$A^{-1}(q^{-1},\theta)B(q^{-1},\theta)$	$A^{-1}(q^{-1},\theta)$
OE	$F^{-1}(q^{-1},\theta)B(q^{-1},\theta)$	I
FIR	$B(q^{-1},\theta)$	I
ORTFIR	$F^{-1}(q^{-1})B(q^{-1},\theta)$	I

Table 2.1: *Multivariable black-box model structures.*

The ORTFIR model structure, which plays an important role in this thesis, is also linear in the parameter and is of OE type. It is a model structure that is equivalent to the output error structure, but with a denominator polynomial that is not equipped with free parameters. The denominator is fixed a priori and for this reason this type of model structure is also denoted with fixed denominator model structure [107]. This model structure can also be viewed as generalization of the FIR model structure in the sense of a replacement of the shift operator for another (rational) transfer function, i.e.

$$G(q,\theta) = B_0 + B_1 f_1(q) + \cdots + B_n f_2(q)$$

where $f_i \in H_2^{n\times n_u}$ are dynamical basis functions that are chosen a priori. This model structure is more favorable than FIR because, especially for slow systems, the number of expansion coefficients that must be estimated for an accurate estimate, is large if $f_i(q) = q^{-i}$ while with an appropriate choice of the basis functions or, equivalently the fixed denominator $F(q^{-1})$, the number of free parameters can be reduced considerably. In the limit, the denominator is chosen according to the system denominator and the number of parameters needed to obtain an accurate estimate of the system transfer is only n_b. Some common choices for these basis functions are Laguerre [70], Kautz [66] and generalized system-based basis functions [153][104]. These will be discussed in detail in the next chapter.

Another property that the FIR and ORTFIR have in common is that the system transfer and noise transfer have no parameters in common, like in the Box Jenkins (BJ) and OE model structures. A structure with this property is denoted with *independent parametrization.*

Criterion selection

In the third step a criterion must be chosen by which one model out of the model class will be selected as the best one. The choice of the criterion depends on the intended use of the model and the assumption about the disturbance. If the disturbance is assumed to be a stationary stochastic process, a natural choice for the criterion is to minimize the variance of the prediction error over the observed samples. For this purpose first the prediction error is defined as

$$\varepsilon(t,\theta) = y(t) - \hat{y}(t|t-1;\theta)$$

where $\hat{y}(t|t-1;\theta)$ is the one-step ahead predictor

$$\hat{y}(t|t-1;\theta) = [1 - H(q,\theta)^{-1}]y(t) + H^{-1}(q,\theta)G(q,\theta)u(t).$$

Now, the least squares prediction error criterion function is given by

$$V(\theta) = \frac{1}{N}\sum_{t=1}^{N}\varepsilon(t,\theta)^T\varepsilon(t,\theta) \tag{2.35}$$

An additional degree of freedom can be obtained by minimizing the sum of squares of a prefiltered version of the prediction error $\varepsilon_f(t,\theta) = L(q)\varepsilon(t,\theta)$ where $L(q)$ is the prechosen prefilter. The construction of the prediction error signal is depicted in figure 2.4. By analyzing the properties of $\varepsilon(t,\theta)$ an indication is obtained of the model quality.
Of course, alternative cost functions and alternative norms can be used to specify the cost function. For example the infinity norm of the prediction error

$$V_1(\theta) = \|\varepsilon(t)\|_\infty$$

can be chosen as cost function. In the literature this is denoted as the *robust identification* problem [55]. Dependent on the available data also a linear combination of several criteria are possible to obtain a multi-objective identification criterion. Because all the requirements on the model are difficult to specify in a criterion function all these approaches can provide a suitable model. In this thesis the 2-norm criterion (2.35) or weighted versions thereof are used.

Model validation

One of the most difficult steps in experimental modeling is to assess whether the model is valid. In fact the only thing that can be done is to assess whether the obtained model

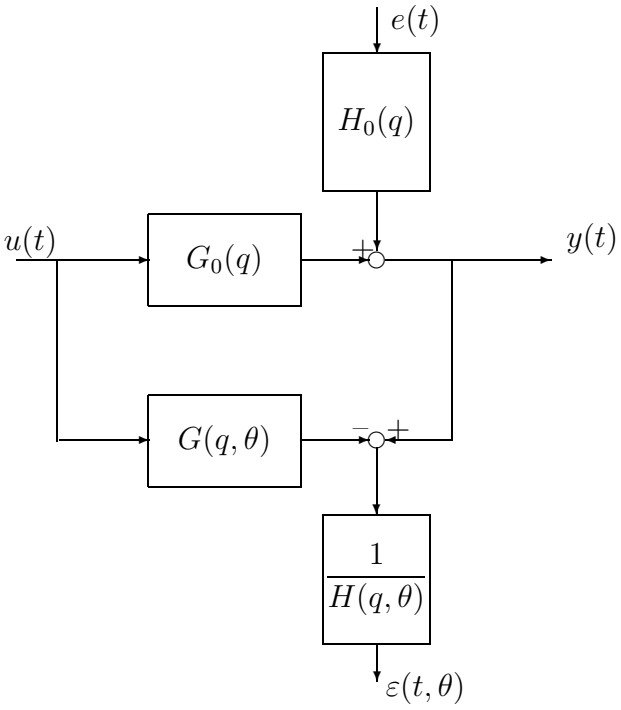

Fig. 2.4: *Data-generating system G_0, H_0 and predictor model (G, H).*

is consistent with all the available data and the assumptions made on the system.
Classical model validation tools are the comparison in the frequency domain of the Bode diagram of the identified parametric model with (nonparametric) spectral estimates. In the time domain the autocorrelation function of the prediction error is a pulse function, i.e. $R_\varepsilon(\tau) = \delta(\tau)$, if the system and model are equal, as the residual is the white noise signal e. Comparision of R_ε with a pulse provides information about both the system transfer and the disturbance transfer. In this situation it also holds that all information about the applied input is eliminated from the residue, implying that $R_{u\varepsilon}(\tau) = 0$. Hence, this cross-correlation can also provide valuable information about the model quality.
If all available information is consistent with the model one can be confident about the model but as soon as data becomes available that is inconsistent with the model and/or the assumptions we are forced to reject the model or change the assumptions on the system. Therefore, it is said that a model cannot be validated it can only be invalidated or, more positively, not invalidated [123]. This has strong relations to the work of the from origin Austrian philosopher Karl Popper [124].
From a less philosophical and a more practical point of view one can certainly validate a model. If the model is obtained, for example, for control design a natural way to validate the model is to apply it in the control design and see if it works. If the controller works, the model was apparently suitable for its purpose.
This type of validation can usually not be done in industry: simply trying out a controller to see whether it works is much too costly and sometimes too dangerous to be a feasible

strategy. Therefore, one wants to have some a priori guarantee that the model is suitable for its purpose. To obtain more insight in the quality of the model and to obtain more information about whether the controller will work with the model, information can be obtain from the data about the *model error*. In the literature several approaches are available to obtain these model error bounds [109][54][26]. Apart from the use of the model error bounds in (in)validation of the model, they can also be used in a robustness analysis of a controller or in the synthesis of a robust controller.

2.5.3 Convergence and consistency

An analysis can be made of prediction error methods where the number of data N tends to ∞. When the noise influence e on the measured data is indeed normally distributed and the model structure is chosen in accordance to the system, the prediction error identification is equivalent to the maximum likelihood estimate.
If the noise is not normally distributed still the following properties hold. The *convergence result* states the following. Let $\hat{\theta}$ be the parameter estimate $\hat{\theta} = arg\min_\theta V(\theta)$ with V the cost function (2.35). Then for $N \to \infty$ it holds that $\hat{\theta} \to \theta^*$ with probability 1, where

$$\theta^* = \arg\min_\theta \bar{E}\varepsilon_f^2(\theta).$$

As a result the asymptotic parameter estimate is independent of the particular noise realization.
The *consistency result* states the following. If the input signal u is persistently exciting of a sufficiently high order, then the asymptotic parameter estimate θ^* has the following properties:

a) If $\mathcal{S} \in \mathcal{M}$ then $G(q,\theta^*) = G_0(q)$ and $H(q,\theta^*) = H_0(q)$.

b) If $G_0 \in \mathcal{G}$ where $\mathcal{G} : \{G(q,\theta) \in H_2 \mid \theta \in \Theta \subset \mathbb{R}_\theta^n$, then $G(q,\theta^*) = G_0(q)$ provided that the model set is independently parametrized.

This implies that the system can be found exactly, at least in the asymptotic case, also if the noise model is not accurate enough, provided that the model set is independently parametrized.

2.5.4 Bias and variance

No estimation of a real world system is an exact representation. There is always model uncertainty. This is due to the fact that the information on the object is limited: the data is contaminated with disturbances and the number of data is finite. Also the assumptions about the system on which the estimation procedure is based does not capture the full complexity of the system, i.e. real world systems are hardly ever linear time invariant and

noise is usually not captured exactly in a stochastic framework. The latter inaccuracy is difficult to assess although progress is being made on this topic.
In this thesis the common and well-accepted approach is used to decompose the estimation error as

$$G_0(q) - G(q,\hat{\theta}) = \underbrace{G_0(q) - G(q,\theta^*)}_{\text{bias error}} + \underbrace{G(q,\theta^*) - G(q,\hat{\theta})}_{\text{variance error}}.$$

The first part is the structural or bias error usually induced by the fact that the model set is not rich enough to include the plant dynamics. The second part is the noise-induced or variance error which is due to the disturbances on the data.
With respect to the bias of a prediction error estimate the following relation holds [83] for the scalar case

$$\theta^* = \arg\min_{\theta} \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{|G_0(e^{i\omega}) - G(e^{i\omega},\theta)|^2\Phi_u(e^{i\omega}) + \Phi_v(e^{i\omega})}{|H(e^{i\omega},\theta)|^2}|L(e^{i\omega})|^2 d\omega. \tag{2.36}$$

where $L(e^{i\omega})$ is a stable prefilter. A similar result can be obtained for the multivariable case [176]. From this expression it becomes clear that when G and H are independently parameterized the bias error on the transfer G can be made zero irrespective of the quality of the noise model as long as $G_0 \in \mathcal{G}$ where

$$\mathcal{G} := \{G(q,\theta) \in H_2 \mid \theta \in \Theta \subset \mathbb{R}^{n_\theta}\}$$

If G and H have parameters in common, like in the ARX case, a trade off is made between fitting the system and noise transfer. In that case a bias results if $\mathcal{S} \notin \mathcal{M}$ even if $G_0 \in \mathcal{G}$. Therefore an independent parametrization of G and H is advisable.
With respect to the variance the following result [83] is appealing due to its simplicity and eloquence. Let n indicate the order of a model $(G(q,\theta), H(q,\theta))$ within $\mathcal{M}$.

- If $G_0 \in \mathcal{G}$ then

$$cov\ G(e^{i\omega},\hat{\theta}) \sim \frac{n}{N}\frac{\Phi_v(e^{i\omega})}{\Phi_u(e^{i\omega})} \quad \text{for} \quad n, N \to \infty, n << N$$

- If $\mathcal{S} \in \mathcal{M}$, then additionally

$$cov\ H(e^{i\omega},\hat{\theta}) \sim \frac{n}{N}|H_0(e^{i\omega})|^2 \quad \text{for} \quad n, N \to \infty, n << N$$

  and $G(e^{i\omega},\hat{\theta})$ and $H(e^{i\omega},\hat{\theta})$ are uncorrelated.

where $covG = E(G-E(G))(G-E(G))^T$. This result provides a clear relation between, on the one hand, the signal to noise ratio, the model order and the number of data samples and, on the other hand, the covariance of the model. A similar result is available for the multivariable case [176].

2.5.5 Parametrization issues

In prediction error identification the cost function

$$V(\theta) = \sum_{t=1}^{N} [H^{-1}(\theta)(y(t) - G(\theta)u(t))]^T [H^{-1}(\theta)(y(t) - G(\theta)u(t))]$$

where G, H is the unknown model that is to be determined. The objects G, H are equipped with free parameters θ over which the optimization is performed. The mapping V is a quadratic function in G and hence convex in G. Therefore, if G is parameterized linearly and $H = I$, the optimization remains convex which is computationally favorable. On the other hand, a free parametrization of both the system transfer G and the disturbance transfer H provides, if the disturbance e in (2.33) is Gaussian, the maximum likelihood (ML) estimate which is statistically optimal. To obtain this ML estimate the parametrization of G, H is nonlinear resulting in a (highly) nonconvex optimization problem of which the global optimum cannot be found with conventional optimization techniques. A challenge is to approach this ML estimate as well as possible while still using computational techniques that are feasible also for the large scale problems encountered in industry.
The maximum likelihood estimate is based on a stochastic assumption on the disturbance while usually the character of the disturbance on real process data cannot be explained totally with stochastic tools. However, from a deterministic point of view it still makes sense to minimize the least squares cost function, or a weighted version thereof with the inverse of the spectrum of the contribution of the disturbance, because it is the optimal fit in 2-norm sense of the output signal. This distinction between the stochastic and deterministic interpretation of maximum likelihood estimation is similar to the two viewpoints from which H_2 optimal control can be addressed. The stochastic motivation for this type of control comes from a desire to optimally suppress stochastic output and system noise while the deterministic motivation states that it is desirable to keep the closed loop transfer from disturbance to output small by minimizing the 2-norm, possibly with a dynamic weighting which is similar to the expected disturbance spectrum.

2.5.6 Computational complexity

The optimization problem that has to be solved to find a model with prediction error system identification is performed off-line. This implies that the computation time is not as critical as in on-line applications like model predictive control. Still, the computational load must remain within reasonable bounds. In identification problems, usually, the accuracy is more important than the computation time.
If a nonlinear parametrization of the model is utilized like an output error model structure, a nonconvex optimization problem must be solved. Various nonlinear optimization algorithms are developed. The most widely used are, so called Newton-like methods

[85][40][97]. These methods use a quadratic approximation of the cost function around the current estimate of the solution. This quadratic cost function is minimized to find the estimate of the solution for the next iteration step. If the exact second derivative of the cost function is applied the Newton method is obtained and if approximate information of the second derivative is used, the method is denoted with quasi-Newton method.
These methods all have the property that a local solution is obtained that can be near the initial point. This may not be an estimate with the desired degree of accuracy. In that case several initial values can be used to start the iteration. In large scale optimization problems for system identification it is difficult to find starting values for the parameters such that a model is obtained with the desired degree of accuracy.
It is therefore advisable to choose a model structure such that a convex optimization problem is obtained to guarantee a unique global minimum that can be found with standard optimization techniques. One way of obtaining a convex optimization problem in identification is the use of model structures that are linear in the parameters. An important question is how the linear parametrization can be chosen such that solution to the convex optimization is sufficiently close to the solution of the nonconvex optimization.

2.6 Model predictive control and system identification: a linear parametrization approach

2.6.1 Motivation

In the previous sections it is argued that the performance of industrial model predictive control of the class of systems mentioned above is limited by the large computational load of the control strategy. This impedes the use of sampling times smaller than several minutes. Even for systems with relatively fast dynamics, so called stiff systems, large sampling times are used and thereby neglecting the fast dynamics. Therefore, with the current generation of model predictive control technology it is not possible to increase the sampling frequency and thereby achieving higher performance levels.
System identification of these large scale and relatively fast sampled systems is also a difficult task. Because large (nonconvex) optimization problems have to be solved that can be badly conditioned. Due to the fast sampling time the emphasis is placed on the high frequent behaviour while the low frequent behaviour may be estimated inaccurately. Hence, new technology is needed to deal with high performance model predictive control and system identification of large-scale, fast sampled systems. The strategy we are following in this project is the following. It is investigated how the degrees of freedom in the optimization problems can be chosen carefully, such that a good trade-off between performance and complexity is obtained. Particular attention is paid to linear parametrizations. The model predictive control (or optimal control) problem and the prediction error iden-

tification (or maximum likelihood) problem have large similarities. This is by no means a new observation [132]. The similarities will be demonstrated in the sequel of this section and special attention is given to the role that a linear parametrization plays in both optimization problems.

2.6.2 Linear parametrization in identification

Given input-output data

$$\mathcal{D} : \{u(t), y(t) | t = 1, 2, \ldots, N\}$$

and the least squares prediction error identification problem

$$\mathcal{V} : \min_{\theta} V(\theta), \quad \text{with } V(\theta) = \sum_{t=1}^{N} (y(t) - G(q,\theta)u(t))^T (y(t) - G(q,\theta)u(t)) \tag{2.37}$$

possibly subject to constraints $K\theta < k$. The model can be parameterized with basis functions as

$$\mathcal{G} : G(z,\theta) = \sum_{k=1}^{n} \Theta_k^T g_k(z) \tag{2.38}$$

where $g_k(z)$ are user-chosen basis functions that contain prior knowledge of the system. The optimization problem (2.37) is usually a nonlinear, nonconvex problem while with a linear parametrization of the model it becomes convex.

With respect to the choice of basis functions, as in the choice of any model structure, there is a trade-off to be made between the bias and variance error. A low number of basis functions in the expansion provides a low variance of the estimate possibly at the cost of a higher bias. Therefore, the basis functions must be chosen such that both bias and variance are small.

2.6.3 Linear parametrization in model predictive control

In an analogous way, a linear parametrization can be applied in model predictive control. Let the prior information be given the system

$$\mathcal{S} : x(k+1) = Ax(k) + Bu(k), \; x(0) = x_0$$

The control problem is given by

$$\mathcal{J} : \min_{u(\cdot)} J(u(\cdot)), \text{ with } J = x^T(P)Q_0x(t) + \sum_{t=0}^{P-1} \{x^T(t)Q_1x(t) + u^T(t)Q_2u(t)\} \tag{2.39}$$

subject to:

$$K_u u(t) \leq k_u \text{ and } K_x x(t) \leq k_x, \forall t$$

The input trajectory can be parametrized with basis functions

$$\mathcal{U} : u(t, \rho) = \sum_{k=1}^{n} \rho_k^T \varphi_k(t) \tag{2.40}$$

where the basis functions $\varphi_k(t)$ can be chosen using prior knowledge of the system dynamics, constraints, control cost function or other available information. A considerable amount of prior information comes from the optimization result of the previous time instant, such as the optimal control profile U_{t-1}^* and Lagrange multiplier λ_{t-1}^*. The optimization problem (2.39) is, contrary to the identification problem, convex but the difficulty is caused by the constraints.

With respect to the choice of the basis functions there is a trade-off to be made with respect to complexity and accuracy. The functions should be chosen such that with a small number of basis functions a accurate approximation of the optimal input profile is obtained for all time instances and constraint situations.

2.6.4 Finding a suitable linear parametrization

From a more conceptual point of view the linear parametrization problem can be formulated as follows. Let an optimization problem be given by

$$\min_{x \in \mathcal{X}} J(x, \mathcal{I})$$

where an object x is searched for in some set $\mathcal{X}$ that is optimal in terms of a cost function J. Apart from the cost function additional exact information $\mathcal{I}$ is added to obtain the optimization problem. For the identification problem the object is the linear system description $x = G$ and the (assumed) exact information amounts to the data and the model structure

$$\mathcal{I}_{id} = \{\mathcal{D}, \mathcal{M}\}$$

For the nominal linear model predictive control problem the object is the control profile for the current time instant $x = u(t)$ and the (assumed) exact information is given by the system dynamics and the measured or estimated state[2]

$$\mathcal{I}_{mpc} = \{x(t), \mathcal{S}\}$$

There can be various reasons why it is not desirable or even tractable to solve the full order optimization problem.

First of all the problem can be infinite dimensional e.g. continuous optimal control problems or discrete infinite horizon optimal control problems. Infinite dimensional optimization problems are known to be difficult to solve and many are still intractable.

[2]Note that the optimal control problem does not assume measurements available of the system state but tries to find control profiles that are a function of this state.

Secondly, the problem can be highly nonconvex, e.g. the identification problem with a general input output model structure. With nonconvex optimization problems it is difficult to find a good local solution and it is virtually impossible in many cases to find the global optimum.
Thirdly, the full order solution can be calculated but has a large computational price which is not feasible for the application at hand. This is certainly the case for high performance model predictive control as the computation time is limited.
In both problems, model predictive control as well as identification, a convex cost function J is used which implies that if the object x is parametrized linearly, the optimization problem remains convex. By choosing a finite linear expansion the problem can also remain finite dimensional. The question is, however, what is a sensible linear parametrization?
The linear parametrization problem is to find a reduced order space $\mathcal{X}_r$ that is spanned by the finite number of basis functions f_k, i.e. the subspace

$$\mathcal{X}_r = sp\{\{f_k\}_{k=1,\dots,n}\}$$

or the linear variety

$$\mathcal{X}_r = b + sp\{\{f_k\}_{k=1,\dots,n}\}$$

To find a subspace that is optimal in some sense for specific applications the theory on n-widths can be used [119]. However, for the problems at hand these n-width problems are not tractable as will be shown in the sequel.

Therefore, to find sensible parametrizations we need to apply the information that is provided in the problem statement but also all additional information that is not provided in the problem statement. Additional information can for instance come from "soft" or approximate knowledge of the optimal solution of the optimization problem, i.e. the object x^*, that is searched for.
In the identification problem this type of additional information can come from prior knowledge about the dynamics or structure of G. This can e.g. be acquired from operators or prior identification results.
In model predictive control the additional information that can be used to find a parametrization could be about the character of the optimal input signal e.g. whether some smoothness properties are satisfied or the profile has a bang-bang character like in time optimal control problems. The additional information can also be about the active constraints. This type of information can e.g. be acquired by knowledge of the specific problem and its solution. Another source of additional information in model predictive control comes from the solution of the previous time instant that is available due to the receding horizon principle.

The search for efficient linear parametrizations is by no means new. Various choices for the basis function f_k are suggested in literature and are analyzed in different types of problem settings. Examples are:

- Classical orthogonal polynomials like Legendre, Chebychev and Hermite polynomials [147]. These polynomials are e.g. used in the solution of continuous time optimal control problems [62].

- Splines. This are piece-wise polynomial approximations that are available in both continuous time and discrete time [12]. The pieces lie between so-called knots and are glued together such that the first m derivatives are continuous. Splines are considered a powerful tool in a number of approximation tasks such as interpolation, data fitting and the solution of boundary value problems for differential equations. Especially the latter has made them popular in optimal control theory where they are used in e.g. collocation and Galerkin methods.

- Wavelets. Wavelets are a family of basis functions that are constructed through the translation and scaling of a single function that has usually the shape of a little wave, hence the name wavelet [25]. Wavelets are mainly used for the purpose of signal analysis. When a signal is convoluted with a set of wavelet functions this gives a picture of the distribution of the energy in the signal over various translations and scales.

These basis functions are useful in a wide variety of applications, but are not tailored to the use in optimization problems with dynamical systems. Linear dynamical systems are highly structured mappings. To obtain efficient parametrizations this structure must be exploited. Basis functions that exploit this are so called system-based basis functions such as Laguerre [70], Kautz [66] and generalized system-based basis functions [59][104]. System-based basis functions have already a wide application range in system identification and signal processing. One of the questions that is addressed in this thesis is to what extent this type of approach can lead to an efficiency improvement in model predictive control.

2.7 Detailed problem formulation and outline

In this thesis it is investigated to which extent a linear parametrization can contribute to the increased efficiency of model predictive control and identification techniques. Systematic ways have to be devised to choose linear parametrizations in the model predictive control and identification problems.

Investigate the possibilities to choose a linear parametrization in the optimization problems (2.38) *and* (2.40) *such that:*

- *high accuracy is obtained with a low computational complexity,*
- *a trade-off can be made between accuracy and complexity.*

With respect to identification a great number of results have already been obtained in this research direction on, so-called, system based basis functions e.g. Laguerre functions, Kautz functions and generalizations thereof. This line of research is extended in this thesis. The extensions are focused on the enhancement of the applicability of this approach to large-scale industrial type systems.
With respect to model predictive control less research results are available that is directed towards the systematic use of linear parametrizations in model predictive control although a number of linear parametrizations are proposed. Therefore, the research in the model predictive control part has a more exploratory nature than in the identification part.
The outline of the sequel of this thesis is as follows.

Chapter 3 System identification with system-based model parametrization

The existing theory on identification with system-based basis functions (SB-OBF's) is described with a particular focus on multivariable systems. A number of extensions are made to the basic theory to make it applicable to industrial identification.

Chapter 4 Iterative model improvement

Identification with SB-OBF's provides the possibility to improve the estimated model using an iteration. This iteration can be implemented efficiently and can therefore serve as a possible alternative for nonlinear optimization in identification problems for large-scale industrial processes. Several different viewpoints are explored.

Chapter 5 Model predictive control with system-based input parametrization

This chapter is the mirror image of chapter 3. An attempt is made to use similar ideas as in identification to use information of the dynamics of the system in the parametrization. Some possibilities and restrictions are explored.

Chapter 6 Constrained model predictive control with on-line input parametrization

This is in some ways the mirror image of chapter 4. Whereas in chapter 4 an alternative nonlinear optimization strategy is developed for identification, in this chapter an alterna-

tive quadratic programming strategy is provided. Both strategies use similar ideas and the similarities are highlighted.

Chapter 7 Identification and control of a fluidized catalytic cracking unit

The developed ideas are tested on an industrial type simulation example. Both identification and model predictive control algorithms are exposed to various case-studies to highlight the possibilities and limitations.

Chapter 8 Conclusions and recommendations

The main results are summarized in chapter 8 together with recommendations for future research.

Chapter 3

System identification with system-based basis functions

3.1 Introduction

In the identification of a model on the basis of industrial data a number of issues are important. First, the processes have many inputs and outputs. This large scale multivariable character implies that nonconvex optimization is intractable to find the optimal model parameters. Instead, convex or near convex problem formulations must be utilized as much as possible. Secondly, the experiment time is usually small, due to the large financial loss that is a consequence of the loss of product quality caused by the excitation signal. This also implies that the excitation signal is usually such that the data has a low signal to noise ratio. Clearly, the identification objective of an excitation that is as large as possible is conflicting with the economic objective to disrupt the process operation as little as possible. Last, the processes exhibit both fast and slow dynamics which must be modeled accurately. Time scales ranging from several hours to several seconds (or faster) must be captured in the model. Also the static behaviour of the system is important to estimate accurately although little information about the static gain of the system is present in the short data sets. In chapter 2 it is argued that a linear parametrization of the model can have important benefits with respect to the issues mentioned above. In this chapter this is further investigated and the basic theory of linearly parametrized models is extended to deal with specific challenges of identification of industrial processes.
A linear parametrization of the model for system identification has several advantages compared to nonlinear model structures. The largest advantage is that minimizing the prediction error criterion function for this model structure is a least squares problem that can be solved analytically. Due to convexity of the cost function the optimum is a global optimum, hence no better model can be found within the chosen model structure.

The general form of a (finite) linear parametrization of the system dynamics is

$$G(z,\theta) = \sum_{k=1}^{n} L_k(\theta) f_k(z) \tag{3.1}$$

where $L_k(\theta) \in \mathbb{R}^{n_y \times n_b}$ are the parametrized expansion coefficients with the free parameter θ that is to be estimated and $\{f_k(z)\}_{k=1,2,\ldots,n} \in H_2^{n_b \times n_u}$ is a set of dynamic functions which are chosen a priori. The finite number of functions span a subspace

$$\mathcal{G} = sp\{f_k\}_{k=1,\ldots,n} \subset H_2^{n_b \times n_u}$$

of the full space of stable discrete time systems. The question is how to choose this subset wisely for a specific situation.

Several requirements on the functions can be imposed irrespective of the specific application at hand. First, it is favorable that the functions f_k are linearly independent. This implies that each element in the space spanned by these functions is uniquely determined by its expansion coefficients. It is said that these functions then define a basis for the space they span. A second favorable property of this set of functions is that each stable transfer function can be described by a linear combination of these functions. This is denoted by completeness of this set of functions. As the set of stable transfer functions is an infinite dimensional, a complete set of basis functions for this space will necessarily also be an infinite set. Another property that can be advantageous in some cases is orthogonality. This refers to the property of the functions being perpendicular to each other. If the functions are moreover scaled to have length one, this is denoted with orthonormality. Hence, it is desirable for the basis functions $f_k(z)$ to form a (orthonormal) basis.

From a computational point of view it is desirable to solve a finite dimensional instead of an infinite dimensional problem. Therefore, in identification only a finite expansion is applied with a finite number of expansion coefficients to be calculated. Therefore it is important that a basis is chosen such that only a small number of expansion coefficients is needed to obtain a certain accuracy of the estimated model. This problem specific property is denoted with efficiency of the basis.

There is no limit to the number of bases that can be used to describe linear dynamical systems. Only some are more efficient than others in a particular situation, in the sense that only a small number of expansion coefficients is needed to obtain a certain accuracy of the model description.

The most widely used basis is the one which uses the shift operator

$$f_k(z) = z^{-k}.$$

This basis is also denoted with pulse basis. The model structure that results from this choice is the well known finite impulse response (FIR) model. For simple and fast systems this may be an efficient basis but for slow systems sampled at a rate that is fast compared

to the system dynamics, this can be very inefficient in the sense that a large number of coefficients is needed to obtain the desired accuracy. To mitigate this effect a Laguerre basis [160] can be used, which is given by

$$f_k(z) = \sqrt{1-a^2}\frac{(1-az)^k}{(z-a)^{k+1}} \tag{3.2}$$

where $|a| < 1$ is an a priori chosen real valued parameter. By choosing this parameter appropriately, e.g. using prior knowledge of the system dynamics, the number of functions that is needed to obtain a certain accuracy can be reduced with respect to the number that is needed with the FIR basis. For well damped systems this can be a very efficient basis but for moderately or badly damped systems still a large number of coefficients may be necessary for an accurate fit. For this class of systems an appropriate choice of basis is the two-parameter Kautz basis [65][161]

$$f_{2k}(z) = z\frac{\sqrt{1-c^2}(z-b)}{z^2+b(c-1)z-c}\left[\frac{-cz^2+b(c-1)z+1}{z^2+b(c-1)z-c}\right]^k \tag{3.3}$$

$$f_{2k+1}(z) = z\frac{\sqrt{(1-c^2)(1-b^2)}}{z^2+b(c-1)z-c}\left[\frac{-cz^2+b(c-1)z+1}{z^2+b(c-1)z-c}\right]^k \tag{3.4}$$

The Laguerre and Kautz bases are efficient for well damped and badly damped systems respectively. For systems with a combination of these dynamical phenomena these may not be efficient anymore, i.e. in both bases a large number of expansion coefficients are required. For this type of systems a dynamic basis is desirable that contains more general dynamics. This basis is available and is denoted with generalized orthonormal basis functions [59] [110][151]. This type of basis functions generalizes the bases given above.

Apart from the fact that the efficiency of the bases can be altered with respect to the specific problem at hand, the variance of the estimated model can be shaped by using a specific basis. These properties make the estimation of models using a parametrization in terms of system-based orthogonal basis functions an appealing approach to identify the type of industrial systems as indicated in the previous chapter.

In section 3.2 available results are discussed about the identification of scalar systems with system-based orthogonal basis functions. The focus in this thesis is on industrial systems which are usually multivariable. Therefore, in section 3.3 these results are extended to the multivariable case. Some results are known from literature and some results are not previously published. The short data sets with low signal to noise ratio encountered in industrial identification, result in model estimates with a large variance. To decrease the variance of a prediction error estimate, the identification of disturbance dynamics can be used. How the disturbance dynamics can be estimated using system-based basis functions and restricted increase in complexity is discussed in section 3.4. A number of extensions is necessary to improve the performance of system identification on the basis

of industrial data. In section 3.5 extensions to the basic theory are given like constrained estimation and the use of several data sets simultaneously, which are important to obtain an accurate estimate despite the short data sets. A simulation example is given in 3.6 to illustrate the properties of identification with system-based orthogonal basis functions and 3.7 concludes this chapter.

3.2 Scalar identification with orthogonal basis functions

3.2.1 Construction of an orthogonal system-based basis

There are several ways to construct an orthogonal basis for linear dynamical systems, in fact the number of bases that span this space is infinite. A well known procedure to obtain an orthogonal basis from a set of independent vectors is the Gram-Schmidt orthogonalisation. With this scheme a set of basis vectors say $\{g_k\}$ are chosen. With this set an orthonormal set $\{f_k\}$ can be constructed that spans the same space as $\{g_k\}$ using the recursion

$$\begin{aligned} w_n &= g_n - \sum_{k=0}^{n-1} \langle g_n, f_k \rangle f_k \\ f_k &= w_n / \|w_n\|_2. \end{aligned}$$

This holds for e.g. vectors g_k but also for dynamical functions. For $g_k = \frac{z^k}{z-a}$ the well known Laguerre functions are obtained. For $g_k = \frac{z^k}{z^2+b(c-1)z-c}$ the Kautz functions are obtained. For general dynamical $g_k = \frac{1}{z-a_k}$, where complex poles occur in conjugate pairs, the generalized orthonormal basis functions are obtained with

$$f_n(z) = \frac{\sqrt{1-|a_n|^2}}{z-a_n} \prod_{k=1}^{n-1} \frac{(1-\bar{a}_k z)}{(z-a_k)}. \tag{3.5}$$

The structure inherent in these functions is

$$f_n(z) = V_n(z) G_n(z)$$

where V_n is some transfer function and G_n is an **inner** transfer function. The relation (3.5) indicates that if orthogonality of the basis vectors is required, it is inevitable that each basis function inherits the dynamics of the previous ones. So due to the requirement of the orthogonality no decoupling of the dynamics is possible.
The development of orthogonal basis functions described above is based on transfer function descriptions of the elements g_k. There is also a state space approach available for the construction of orthogonal basis functions [57]. Let G_b, the basis generating systems, be an inner transfer function with a minimal balanced realization $\{A, B, C, D\}$ satisfying

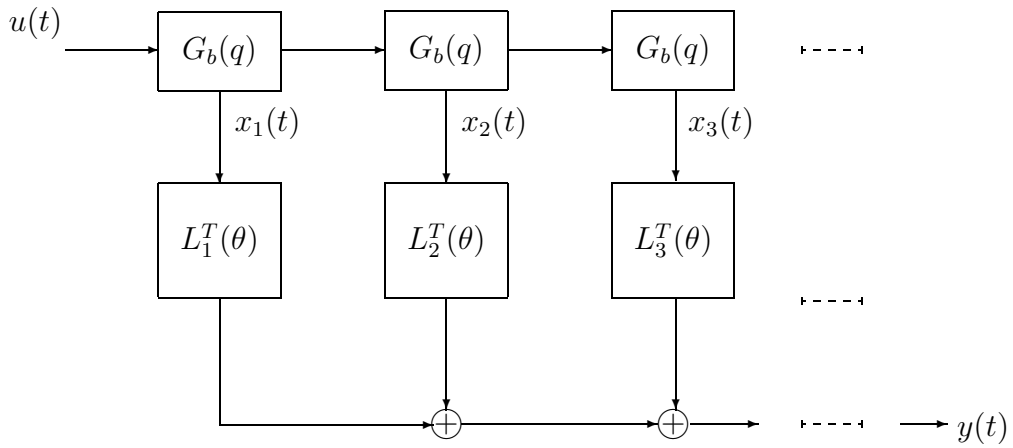

Fig. 3.1: *Series connection of inner transfer functions generating an orthogonal basis with batch wise recurring dynamics.*

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix}^T = I. \tag{3.6}$$

Then the input to state transfers of a series connection given in figure 3.1 of these balanced realizations of inner transfer functions constitute an orthogonal basis for $\mathbb{R}H_2$.
The basis functions above fit in this framework as follows.

- the pulse basis has inner function $G_b(z) = z^{-1}$ and minimal balanced realization $[A, B, C, D] = [0, 1, 1, 0]$.
- the Laguerre basis has inner function $G_b(z) = \frac{1-az}{z-a}$ with some real-valued $|a| < 1$ and balanced realization

$$A = (a, \sqrt{1-a^2}, \sqrt{1-a^2}, -a)$$

- the two-parameter Kautz basis has inner function $G_b(z) = \frac{-cz^2+b(c-1)z+1}{z^2+b(c-1)z-c}$ and balanced realization

$$\begin{aligned} A &= \begin{bmatrix} b & \sqrt{1-b^2} \\ c\sqrt{1-b^2} & -bc \end{bmatrix}, B = \begin{bmatrix} 0 \\ \sqrt{1-c^2} \end{bmatrix} \\ C &= \begin{bmatrix} -b\sqrt{1-c^2} & \sqrt{(1-c^2)(1-b^2)} \end{bmatrix}, D = -c \end{aligned}$$

  modulo some similarity transformation with a unitary transformation matrix.

With the transfer function approach discussed above the prior knowledge that is used to specify the basis functions is given the pole locations.

Another approach to construct the basis functions uses the state space matrices $\{A, B\}$ with $A \in \mathbb{R}^{n_b \times n_b}, B \in \mathbb{R}^{n_b \times n_u}$ as prior knowledge to construct the basis. With this prior information a $\{C, D\}$ pair where $C \in \mathbb{R}^{n_u \times n_b}, B \in \mathbb{R}^{n_u \times n_u}$ can be constructed such that the quadruple $\{A, B, C, D\}$ represents a balanced realization of an inner function for which the requirement (3.6) holds. This construction can be performed with the use of the singular value decomposition. Given an input balanced state space pair $\{A, B\}$ with its singular value decomposition given by

$$[A\ B] = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{3.7}$$

where the left and right singular vector matrices have partitions $U_1 \in \mathbb{R}^{n_b \times n_b}, U_2 \in \mathbb{R}^{n_b \times n_u}$ $V_1 \in \mathbb{R}^{(n_b+n_u) \times n_b}, V_2 \in \mathbb{R}^{(n_b+n_u) \times n_u}$ respectively. If the $\{C, D\}$ extension is taken to be

$$[C\ D] = V_2^T \tag{3.8}$$

the relation (3.6) holds. This can be easily deduced from the fact that $V = [V_1\ V_2]$ is a square unitary matrix and the input balanced pair $\{A, B\}$ satisfies the Lyapunov equation $AA^T + BB^T = I$.
The quadruple $\{A, B, C, D\}$ represents a balanced realization of an inner function G_b which generates vector valued basis functions

$$F_k(z) = (zI - A)^{-1} B G_b^{k-1}(z) \in H_2^{n_b} \tag{3.9}$$

using the series connection of inner functions of figure 3.1. This will be denoted as the $\{A, B\}$-variant. As a starting point also prior knowledge of the pair $\{A, C\}$ can be taken. Next, an appropriate pair $\{B, D\}$ is constructed using the left singular vectors of the singular value decomposition of $[A^T\ C^T]^T$. This results in the vector valued basis functions

$$F_k(z) = (zI - A^T)^{-1} C^T G_b^{k-1}(z) \in H_2^{n_b} \tag{3.10}$$

which will be denoted with the $\{A, C\}$-variant. In both variants the eigenvalues $\lambda(A)$ occur recurrently in the basis functions.
The number of basis functions needed to obtain a complete basis is infinite. Usually only prior knowledge of a finite set of poles is available. Basically two possible choices exist for the extension of this finite set of poles to an infinite set such that the basis is complete.

- Repetition of the finite set of poles. Examples are the repetition of one pole in the Laguerre case, of two poles in the two parameter Kautz basis or n poles in the generalized orthogonal basis discussed in the work by Heuberger and Van den Hof and coworkers. A consequence of the choice for repetition of the dynamics in

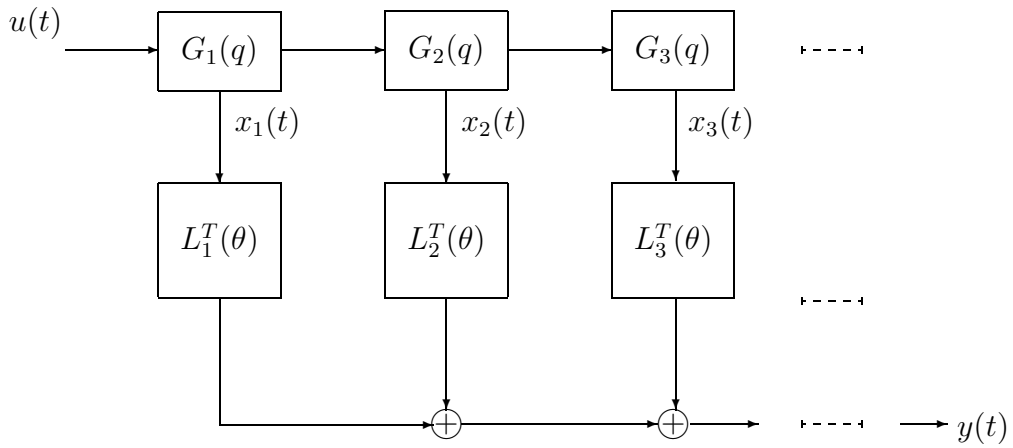

Fig. 3.2: *Series connection of inner transfer functions generating an orthogonal basis with general dynamics.*

the basis functions is that the estimated coefficients can be viewed as the Markov parameters of the system in a transformed domain, the so called Hambo domain [58]. A remarkable fact is that the system in the transformed domain is again a linear system. The generalized Markov parameters decrease with increasing index and an upper bound can be given for the convergence speed.

- No predefined location of the poles. To obtain a complete orthonormal basis it can be shown that the pole location must satisfy

$$\sum_{k=0}^{\infty}(1-|a_k|)=\infty$$

meaning that the pole location must not converge to the unit circle [110]. A consequence of this choice is that no upper bound for the convergence speed can be given, but it is more flexible with respect to the choice for dynamics in the basis functions. This is graphically depicted in figure 3.2 where the transfer functions G_i are inner functions and $x_i(t)$ are the balanced states. These functions can be constructed with either the transfer function or the state space approach.

The completeness proofs can be found in the respective papers.
Also a nonorthogonal basis is in use in the so called fixed denominator model structure [107]. In this model structure the basis functions

$$f_k(z)=\frac{z^{-k}}{D(z)}$$

are used, where $D(z)$ is a fixed stable polynomial that serves as the denominator of the estimated model. For the space that is spanned by these basis functions, also an

orthonormal basis can be constructed. Hence, an orthogonal and nonorthogonal basis can be used to describe the same model set. An orthogonal basis has some advantages over a nonorthogonal basis that will be made clear in the sequel [111].
For finite model orders any scalar model structure based on system-based orthogonal basis functions can be described by

$$G(z,\theta) = \theta^T (zI - A_e)^{-1} B_e \tag{3.11}$$

where the parameter vector $\theta \in \mathbb{R}^{nn_b}$ and $A_e \in \mathbb{R}^{nn_b \times nn_b}, B_e \in \mathbb{R}^{nn_b}$ where n is the number of repetitions in the expansion (3.1) and n_b is the order of the basis functions $f_b(z)$. The extended matrix pair $\{A_e, B_e\}$ has specific properties dependent on the choices made, either general or batch wise repetition of basis dynamics and orthogonal or nonorthogonal basis etc. The state space matrices for the repetition of the poles in the basis functions according to figure 3.1 amount to

$$A_e = \begin{bmatrix} A & 0 & \cdots & 0 \\ BC & A & 0 & \vdots \\ BDC & BC & A & \\ \vdots & \vdots & & \ddots \; \vdots \\ BD^{k-2}C & BD^{k-1}C & \cdots & BC \quad A \end{bmatrix}, \quad B_e = \begin{bmatrix} B \\ BD \\ BD^2 \\ \vdots \\ BD^{k-1} \end{bmatrix} \tag{3.12}$$

where $\{A, B, C, D\}$ is a balanced realization of the inner function G_b. The state space pair $\{A_e, B_e\}$ is again input balanced, i.e. $A_e A_e^T + B_e B_e^T = I$ because a series connection of balanced systems is again balanced. The general description (3.11) will be used in the sequel of this chapter.

3.2.2 Parameter estimation

The model structures discussed in the previous section are all linear in the parameters. The predicted output for this model structure, where the disturbance model is set to one, is given by

$$y(t,\theta) = \sum_{k=1}^{n} \theta_k^T f_k(q) u(t)$$

with $\theta_k \in \mathbb{R}^{n_b}$ and $f_k(z) \in H_2^{n_b}$. This yields a prediction error given in linear regression form by

$$\varepsilon(t,\theta) = y(t) - \varphi(t)^T \theta$$

with $\theta = [\theta_1^T \dots \theta_n^T]^T$ and $\varphi^T(t) = [f_1^T(q)u(t) \dots f_n^T(q)u(t)]$. For convenience of notation we define

$$Y = [y(1) \dots y(N)]^T, \Phi = [\varphi(1) \dots \varphi(N)]^T.$$

With this the prediction error cost function becomes

$$V(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \varphi^T\theta)^2 = \frac{1}{N}[Y - \Phi\theta]^T[Y - \Phi\theta]$$

The minimum of this cost function is given by

$$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y \tag{3.13}$$

which is also the least squares solution to the overdetermined set of equations

$$Y = \Phi\theta$$

This least square problem with orthogonal basis functions has special properties that do not apply if nonorthogonal basis functions are applied.

Property 3.2.1 *If the basis functions f_k are orthonormal and the input $u(t)$ is white noise with variance one, then for $N \to \infty$ the matrix $E\frac{1}{N}\Phi^T\Phi \to I$ and the parameter vector can asymptotically simply be found by the inner products $\hat{\theta} = \frac{1}{N}\Phi^T Y$.*

Property 3.2.2 *If the basis functions f_k are orthonormal and the input spectrum $\Phi_u(\omega)$ is sufficiently rich, then it holds that for $N \to \infty$* [106]

$$\min_{\omega\in[-\pi,\pi]} \Phi_u(\omega) \leq \lambda(\frac{1}{N}\Phi^T\Phi) \leq \max_{\omega\in[-\pi,\pi]} \Phi_u(\omega)$$

This property means that for a white input signal the least squares problem is, asymptotically in the number of data points, perfectly conditioned, i.e. $\kappa(\bar{E}\Phi^T\Phi) = 1$. If the input has some coloring the conditioning of the least squares problem remains good for a large class of input signals. It is important to note that this favorable property is not the case in the more realistic case of finite data. The consequences of the availability of a finite amount of data is analyzed in a later section.

3.2.3 Asymptotic bias

The asymptotic bias of a model estimated with orthogonal basis function structure are analyzed for a general choice of the pole location in [110], for the structure with a batch wise repetition of poles in [151] and for the fixed denominator model structure in [107]. The asymptotic bias for these model structures is given in the next theorem.

Proposition 3.2.3 *Let the system be given by the infinite series expansion*

$$G_0(z) = \sum_{k=1}^{\infty} L_k^{(0)} f_k(z)$$

and denote the first n expansion coefficients as $\theta_0 = [L_1^{(0)T} \dots L_n^{(0)T}]^T$. *Consider the identification setup of section 2.5 where* θ^* *can be described as* (2.36) *for any model structure. With the model structure* (3.1) *it moreover holds that*

$$\|\theta^* - \theta_0\|_2 \leq \frac{sup_\omega \Phi_u(\omega)}{inf_\omega \Phi_u(\omega)} \|\theta_e\|_2$$

where $\|\theta_e\|_2 = \sqrt{\sum_{k=n}^{\infty} L_k^{(0)} L_k^{(0)T}}$.

Proof: See [151]. □

This indicates that the bias is strongly related to the tail of the expansion coefficients. That observation is fruitfully used in many uncertainty identification methods to overbound the uncertainty due to undermodeling.
If the dynamics is repeated batch wise in the basis functions [151] the following eloquent result can be established. An upper bound for the convergence rate of the expansion coefficients is available as a function of the location of the system poles.

Proposition 3.2.4 *Let* $G_0(z)$ *have poles* $\mu_i, i = 1, \dots, n_s$, *and let* $G_b(z)$ *have poles* $\rho_j, j = 1, \dots, n_b$. *Denote*

$$\lambda := \max_i \prod_{j=1}^{n_b} \left| \frac{\mu_i - \rho_j}{1 - \mu_i \rho_j} \right| \tag{3.14}$$

Then there exists a nonnegative constant $c \in \mathbb{R}^+$ *such that for all* $1 > \eta > \lambda$

$$\|\theta_e\|_2 \leq c \frac{\eta^{n+1}}{\sqrt{1 - \eta^2}} \tag{3.15}$$

Proof: See [151]. □

Note that if the poles in the basis converge to the poles in the system, λ will tend to zero and the bias will obviously disappear. This relation also shows that the (upperbound on the) bias becomes larger if the set of basis poles is further away from the system poles. This indicates that from a bias point of view it is advisable to choose the poles of the basis function close to the system poles. Another way to decrease the bias is obviously by increasing the number of expansion coefficients n. In the limit any system can be described in terms of system based basis functions regardless whether the dynamics of these functions is close to the system dynamics or not. However, a small number of expansion coefficients is favorable for the variance of the estimate as will be demonstrated in the next section.

3.2.4 Asymptotic variance

The asymptotic variance is given in the next proposition.

Theorem 3.2.5 *Assume that the spectral density in the input $\Phi_u(\omega)$ is bounded away from zero and sufficiently smooth. Then for identification with model structure with scalar basis functions* (3.5) *and for $N, n \to \infty, n^2/N \to 0$, it holds that*

$$\left[\sum_{i=1}^{n} F_i^*(e^{i\omega_1})F_i(e^{i\omega_1})\right]^{-1} cov\{G(e^{i\omega_1}, \hat{\theta}_N), G(e^{i\omega_2}, \hat{\theta}_N)\} \to$$

$$\begin{cases} 0 & for\ G(e^{i\omega_1}) \neq G(e^{i\omega_2}) \\ \frac{1}{N}\frac{\Phi_v(\omega_1)}{\Phi_u(\omega_1)} & for\ \omega_1 = \omega_2 \end{cases} \tag{3.16}$$

Proof: See [151]. □

This theorem provides a closed form expression for the asymptotic covariance. It also implies that the variance of the transfer function estimate with a batch wise repetition of the poles, e.g. in (3.9) and (3.10), for a specific ω is given by

$$\frac{nn_b}{N} F_1^*(e^{i\omega})F_1(e^{i\omega})\frac{\Phi_v(\omega)}{\Phi_u(\omega)}$$

The variance is approximately equal to the signal to noise ratio weighted with an additional weighting function that is determined by the number of free parameters and the first basis function F_1 given in (3.9) or (3.10). Note that the number of expansion coefficients is linear in the number of basis functions n, hence a small number of expansion coefficients implies a relatively low variance level. This additional weighting, which is not present in the case of FIR or ARX estimation, has a low-pass character which provides a smaller variance for higher frequencies. The additional weighting can be used to influence the asymptotic variance by a specific choice of the basis functions.

Example 3.2.6 *The variance of an FIR estimate on the basis of data with a white input and disturbance signal is approximately equal to n_{FIR}/N for all frequencies. An orthogonal basis function estimate has a variance that is frequency dependent $(n_{ORT}/N)\, F_1^* F_1$. The dynamic weighting $F_1^* F_1$ will generally be smaller than one in some frequency ranges and larger than one in other regions. Hence whether the variance of the system based approach is smaller than the pulse basis for all frequencies depends on the gain in terms of the number free parameters in the basis function approach given by $n_{ORT} = nn_b$ compared to those in the pulse basis given by n_{FIR}. For e.g. slow systems which are sampled relatively fast n_{FIR} is generally high and hence the basis function model will generally have better variance properties for all frequencies.*

3.3 Multivariable model structures

3.3.1 Constructing of a complete orthogonal basis

There are several multivariable model structures that can be constructed with system based orthogonal basis functions (SB-OBF's) of the type

$$G(z,\theta) = \sum_{i=1}^{n} L_i(\theta) F_i(z)$$

where the estimated expansion coefficients are $L_i \in \mathbb{R}^{n_y \times n_x}$ and the basis functions $F_i(z) \in \mathbb{R}H_2^{n_x \times n_u}$ in which the number n_x is dependent on the type of basis functions that is chosen.

If the poles of a scalar system are available, the zeros and gain can be estimated from data with a simple convex optimization problem to form a minimal realization. Now, let the system $G(z) = C(zI - A)^{-1}B$ be multivariable with each matrix entry (i, j) given by the scalar transfer function $g_{ij}(z)$. Assume that the least common multiple of the poles of the transfer functions $f_{ij}(z)$, i.e. eigenvalues $\lambda(A)$, is given. Then the following parametrization is possible

$$G(z,\theta) = \frac{1}{a(z)} B(z,\theta) \tag{3.17}$$

where the roots of the monic scalar polynomial $a(z)$ are equal to $\lambda(A)$ and $B(z,\theta)$ is the parametrized polynomial matrix with the numerators of the transfer functions $g_{ij}(z)$. However, identification with this model structure will generally not yield a minimal representation due to pole zero cancellations that occur in the separate entries of the estimated transfer function matrix. Although the model structure (3.17) can be applied fruitfully in the identification of multivariable systems, it results in an over parameterized transfer function estimate. The use of an excessive number of parameters leads to a large variance error and can also cause numerical difficulties in solving the least squares optimization problem.

Additional information about the structure of the system, specifically the distribution of the poles over the consecutive entries of the transfer function matrix, can be used to reduce the number of parameters in the model structure. Several forms of structural information can be utilized in the construction of multivariable model structures, which makes it inherently different from the scalar model structures where this is not the case. In all the structures defined below different prior knowledge is assumed to be available. This is prior knowledge with regard to

- system dynamics, similar to the scalar case, and
- system structure, which is specific for the multivariable case.

Below a list is given of some possibilities together with the prior knowledge that is assumed to be available.

(A,B)-structure

A direct extension of the scalar $\{A, B\}$ structure to the multivariable case is obtained by starting with a state space $\{A, B\}$ pair of order n_b and constructing the basis functions

$$F_k(z) = (zI - A)^{-1} B G_b^{k-1}(z) \in H_2^{n_b \times n_u}$$

where $G_b \in H_2^{n_u \times n_u}$ is a square inner function which is constructed by extending the $\{A, B\}$ pair with a $\{C, D\}$ matrix as described in (3.7) and (3.8) such that the quadruple $\{A, B, C, D\}$ coincides with an inner system. This model structure can, similar to the scalar case, be described by

$$G(z, \theta) = \Theta^T (zI - A_e)^{-1} B_e = \frac{1}{det(zI - A_e)} \Theta^T adj(zI - A_e) B_e$$

with $\Theta^T \in \mathbb{R}^{n_y \times n n_b}$ and therefore the number of free parameters is $n_y n_b n$ and the pair $\{A_e, B_e\}$ is given in (3.12). The McMillan degree of G is generically equal to $n n_b$.

(A,C)-structure

Also the $\{A, C\}$ structure can be extended to the multivariable case, but in a less straightforward manner than in the scalar case. The expansion in basis functions is given by

$$G(z, \theta) = \sum_{i=1}^{n} F_i(z) L_i(\theta)$$

with

$$F_k(z) = G_b^{k-1}(z) C (zI - A)^{-1}. \tag{3.18}$$

The model structure is of the form

$$G(z, \theta) = C_e (zI - A_e)^{-1} \Theta = \frac{1}{det(zI - A_e)} C_e adj(zI - A_e) \Theta \tag{3.19}$$

with $\Theta \in \mathbb{R}^{n n_b \times n_u}$ and therefore the number of free parameters is $n_y n n_b$. The McMillan degree of the estimated model is generically equal to $n n_b$, the order of A_e.
The model structure can also be written in a similar form as the $\{A, B\}$ structure with the introduction of redundancy

$$G(z, \theta) = \sum_{i=1}^{n} \begin{bmatrix} \theta_1^T & & 0 \\ & \ddots & \\ 0 & & \theta_{n_u}^T \end{bmatrix} \begin{bmatrix} (zI - A_e^T)^{-1} & & 0 \\ & \ddots & \\ 0 & & (zI - A_e^T)^{-1} \end{bmatrix} \begin{bmatrix} c_1 & & 0 \\ & \ddots & \\ 0 & & c_{n_y} \end{bmatrix}$$

where $\Theta = [\theta_1 \cdots \theta_{n_u}]$ and $C_e^T = [c_1 \cdots c_{n_y}]$.

Scalar basis functions

Instead of using state space matrices to build a complete orthogonal basis, scalar basis functions can be used to build model structures that are tailored to the prior knowledge at hand. Several model structures are possible of which a few are given below.

- Scalar basis functions

$$G(z,\theta) = \sum_{i=1}^{n} \Theta_i f_i(z) = \sum_{i=1}^{n} \Theta_i \begin{bmatrix} f_i(z) & & 0 \\ & \ddots & \\ 0 & & f_i(z) \end{bmatrix} = \sum_{i=1}^{n} \begin{bmatrix} f_i(z) & & 0 \\ & \ddots & \\ 0 & & f_i(z) \end{bmatrix} \Theta_i \tag{3.20}$$

  with scalar basis functions $f_i(z) \in H_2$ and $\Theta_i \in \mathbb{R}^{n_y \times n_u}$ and therefore the number of free parameters is nn_un_y. The McMillan degree of G is generically equal to nn_u. The model class that is described here is equivalent to the one described by the fixed denominator model structure (3.17) only in the latter a nonorthogonal basis is utilized.

  If a state space pair $A \in \mathbb{R}^{n_b \times n_b}, B \in \mathbb{R}^{n_b \times n_u}$ is available, a model structure of the type (3.20) with poles $\lambda(A)$ can be obtained as follows. First, where b is an arbitrary vector such that $\{A, b\}$ is controllable, e.g. one of the columns of B. Next, perform a similarity transformation on this pair to obtain an input balanced pair $\{A_b, b_b\}$. Finally, obtain the model structure of the form (3.11) with

$$A_e = \begin{bmatrix} A_b & & 0 \\ & \ddots & \\ 0 & & A_b \end{bmatrix}, B_e = \begin{bmatrix} b_b & & 0 \\ & \ddots & \\ 0 & & b_b \end{bmatrix} \tag{3.21}$$

  which is equivalent to the model structure (3.20) with $n = n_b$. This is a model of order n_un_b. To increase the flexibility in the model structure, the model order can be increased either with arbitrary dynamics or with batch wise repetition of the basis poles following the $\{A, B\}$-variant (3.9). A similar approach can be followed for the $\{A, C\}$-variant yielding a model structure similar to the description on the far righthand side of (3.20).

- Scalar structure with different dynamics in each column

$$G(z,\theta) = \sum_{i=1}^{n} \Theta_i \begin{bmatrix} f_1^i(z) & & 0 \\ & \ddots & \\ 0 & & f_{n_u}^i(z) \end{bmatrix}$$

  with $\Theta_i \in \mathbb{R}^{n_y \times n_u}$ and therefore the number of free parameters is nn_un_y. The McMillan degree of G is generically equal to nn_y.

- Scalar structure with different dynamics in each row

$$G(z,\theta) = \sum_{i=1}^{n} \begin{bmatrix} f_1^i(z) & & 0 \\ & \ddots & \\ 0 & & f_{n_y}^i(z) \end{bmatrix} \Theta_i$$

with $\Theta_i \in \mathbb{R}^{n_y \times n_u}$ and therefore the number of free parameters is nn_un_y. The McMillan degree of G is generically equal to nn_y.

- Scalar structure with prescribed dynamics in each row and column of the basis function [108]

$$G(z,\theta) = \sum_{i=1}^{n} \begin{bmatrix} f_{l1}^i(z) & & 0 \\ & \ddots & \\ 0 & & f_{ln_y}^i(z) \end{bmatrix} \Theta_i \begin{bmatrix} f_{r1}^i(z) & & 0 \\ & \ddots & \\ 0 & & f_{rn_u}^i(z) \end{bmatrix}$$

with $\Theta_i \in \mathbb{R}^{n_y \times n_u}$ and therefore the number of free parameters is nn_un_y. The generic McMillan degree of G is nn_un_y.

- Scalar structure with different dynamics in each matrix entry

$$G(z,\theta) = \frac{1}{\sqrt{n_un_y}} \sum_{i=1}^{n} \Theta_i \begin{bmatrix} f_{11}^i(z) & \cdots & f_{1n_u}^i \\ \vdots & & \vdots \\ f_{n_y1}^i(z) & \cdots & f_{n_yn_u}^i(z) \end{bmatrix}$$

with $\Theta_i \in \mathbb{R}^{n_y \times n_y}$ and therefore the number of free parameters is nn_yn_y. The McMillan degree of G can be as high as nn_un_y if all poles of the basis functions are different. The scaling factor $\frac{1}{\sqrt{n_un_y}}$ is needed to make the multivariable basis functions orthonormal with respect to regular inner product $\langle F_i, F_j \rangle = \frac{1}{2\pi} \int_{-\pi}^{\pi} trace\{F_i(e^{j\omega}) F_i^*(e^{j\omega})\}$.

- Scalar structure with different dynamics in each matrix entry of the basis functions

$$G(z,\theta) = \frac{1}{\sqrt{n_un_y}} \sum_{i=1}^{n} \begin{bmatrix} f_{11}^i(z) & \cdots & f_{1n_u}^i \\ \vdots & & \vdots \\ f_{n_y1}^i(z) & \cdots & f_{n_yn_u}^i(z) \end{bmatrix} \Theta_i \tag{3.22}$$

with $\Theta_i \in \mathbb{R}^{n_u \times n_u}$ and therefore the number of free parameters is nn_un_u. The McMillan degree of G can be as high as nn_un_y if all poles of the basis functions are different.

- Scalar structure with dynamics that is prescribed for each entry of the transfer function matrix

$$g_{ij}(z) = \sum_{i=1}^{n^{(ij)}} \theta_i^{(ij)} f_i^{(ij)}(z) \tag{3.23}$$

with $\theta_i \in \mathbb{R}$ and therefore the number of free parameters is $\sum_{j=1}^{n_u} \sum_{i=1}^{n_y} n^{(ij)}$. If the number of basis functions is equal for each matrix entry, i.e. $n^{(ij)} = n$ for $i = 1, \ldots, n_y$ and $j = 1, \ldots, n_u$, the number of free parameters is nn_un_y.

Discussion

All these model structures are linear in the parameters and have the property that if $n \to \infty$ all systems in $\mathbb{R}H_2$ can be described. The output prediction of a model with these structures can, similar to the scalar case, simply be described by

$$y(t, \theta) = G(z, \Theta)u(t) = \Theta^T(zI - A_e)^{-1}B_eu(t) = \Theta^T\varphi(t) \tag{3.24}$$

where $\Theta \in \mathbb{R}^{n_y \times nn_x}$. The model structures differ in the properties of $\{A_e, B_e\}$ and prior knowledge that is assumed to be available about the dynamics and the structure. Structural information about the multivariable system can come in many forms, such as:

- knowledge of dynamics that is the same for each row or column.
- controlability indices. In this, maybe unlikely, case the prior knowledge of $\{A, B\}$ is usable.
- observability indices. In this, maybe unlikely, case the prior knowledge of $\{A, C\}$ is usable.

Clearly, the scalar basis functions can be used as building blocks to build multivariable model structures that fruitfully exploit all prior knowledge about the system dynamics and structure to decrease the number of parameters that must be estimated. It is important that only accurate knowledge about the structure of the system is applied in the construction of the model structure. The use of inaccurate structural information about the system in the construction of the model structure usually leads to a less efficient model structure. This implies that a larger number of parameters is needed to obtain a similar value of the cost function compared to a model structure in which no or less structural information about the system is applied. Put otherwise, given a certain number of free parameters, the model quality of the fist will be lower than that of the second. This is illustrated in the next example.

Example 3.3.1 *A multivariable identification is performed on data of the nonlinear simulation model of an industrial process that is also applied in chapter 7. The process has 5 inputs and 5 outputs. Input output data is available where the inputs are excited*

with pseudo random binary sequences. The data length is $N = 7800$ *samples. Similar to the procedure followed in chapter 7 the initial basis is obtained by identification with the realization algorithm presented in* [155]*, which provides us with a state space model* $\{A, B, C, D\}$ *of order* $n_b = 15$*. For this example the identification results for the following model structures are compared.*
The $\{A, B\}$ *variant for* $n = 5$ *and* $n = 10$ *yielding state space models of order* $nn_b = 75$ *and* $nn_b = 150$ *respectively with* $n_\theta = nn_b n_y = 375$ *and* $n_\theta = nn_b n_y = 750$ *free parameters respectively.*
The $\{A, C\}$ *variant for* $n = 5$ *and* $n = 10$ *yielding state space models of order* $nn_b = 75$ *and* $nn_b 150$ *respectively with* $n_\theta = nn_b n_u = 375$ *and* $n_\theta = nn_b n_u = 750$ *free parameters respectively.*
The scalar basis (3.20) *where* $eig\{A\}$ *is used as prior knowledge for all entries of the transfer function matrix following* (3.21)*. This structure is applied with batch wise repetition following* (3.9) *of the* n_b *poles for a number of repetitions equal to* $n = 1$ *and* $n = 2$*. This yields model structures with* $n_\theta = nn_b n_y = 375$ *and* $n_t heta = nn_b n_y = 750$ *free parameters respectively.*
The cost function values are compared for identification with these model structures where the number of parameters are the same and only the structure is different. The result is given in the table below.
Clearly, the model structure using scalar basis functions performs better than the other structures. A possible explanation for this is that the other structures contain information about the structure of the system. This prior information is apparently incorrect for this example.

	n_θ=375	n_θ=750
(A,B)	0.0672	0.064
(A,C)	0.073	0.0689
scalar	0.0629	0.054

Table 3.1: *Value cost function for model structures with* n_θ *free variables and structure according to the* $\{A, B\}$ *variant, the* $\{A, C\}$ *variant and based on scalar basis functions.*

From this example it becomes clear that a good guideline for model structure selection is the following. Use structural information if this information is known to be accurate but use a more flexible structure, e.g. one using scalar basis functions, if the structural information about the system is uncertain.

3.3.2 Parameter estimation

The multivariable model structure (3.24) is linear in the parameter matrix Θ. The prediction error cost function for this model structure is therefore quadratic in the free parameters and is given by

$$\hat{\Theta} = arg\min_{\Theta} \frac{1}{N} Trace \left\{ \sum_{k=1}^{N} \varepsilon(t,\Theta)\varepsilon^T(t,\Theta) \right\} \tag{3.25}$$

with $\varepsilon(t,\Theta) = y(t) - \Theta^T\varphi(t)$. In the multivariable case the linear regression problem that underlies this optimization can be formulated in terms of the matrix valued parameter Θ as

$$Y = \Phi\Theta \tag{3.26}$$

where

$$Y = \begin{bmatrix} y^T(1) \\ \vdots \\ y^T(N) \end{bmatrix} \in \mathbb{R}^{N\times n_y}, \Phi = \begin{bmatrix} \varphi^T(1) \\ \vdots \\ \varphi^T(N) \end{bmatrix} \in \mathbb{R}^{N\times n}$$

The least squares solution of the overdetermined set of equations (3.26) can be obtained in closed form as $\Theta = \Phi^\dagger Y$ with the pseudoinverse $\Phi^\dagger = (\Phi^T\Phi)^{-1}\Phi^T$. Note that the outputs are treated separately so that the result is similar to n_y MISO estimates. This can be seen by defining $\Theta = [\theta_1 \cdots \theta_{n_y}]$ and $Y = [Y_1 \cdots Y_{n_y}]$ where Y_i is the vector with measurements of output $y_i(t)$. The least squares solution of (3.26) in terms of the parameter matrix Θ is equivalent with the n_y least squares problems given by $Y_i = \Phi\theta_i$. It is computationally attractive to solve (3.26) instead of a least squares problem in the stacked parameter vector $\theta^T = [\theta_1^T \ldots \theta_{n_y}^T]$ as is also discussed in [81] for the case of an ARX model structure.

3.3.3 Asymptotic bias

The asymptotic bias for models estimated with a multivariable model structure with scalar basis functions for each entry of the transfer function matrix (3.23) is given in the next theorem.

Theorem 3.3.2 *Consider a scalar orthonormal basis $\{f_k\}_{k=1,\ldots,n}$ induced by the poles $\{\xi_1,\ldots,\xi_n\}$. Suppose matrix entry i,j of the system $G_{ij}^{(0)}(z)$ has partial fraction expansion*

$$G_{ij}^{(0)}(z) = \sum_{l=1}^{r} \frac{\alpha_l}{z - \gamma_{ij}^l}$$

where all the poles satisfy $|\gamma_{ij}^l| < 1$.
If the input u is white ($\Phi_u(e^{i\omega})$ is constant), then the limiting estimate θ^* satisfies

$$|G_{ij}^{(0)}(e^{i\omega}) - \hat{G}_{ij}(e^{i\omega},\theta^*)| \le \sum_{l=1}^{r} \left| \frac{\alpha_l}{e^{i\omega} - \gamma_{ij}^l} \right| \prod_{k=0}^{n} \left| \frac{\gamma_{ij}^l - \xi_k}{1 - \bar{\xi}_k\gamma_{ij}^l} \right|$$

where n is the number of expansion coefficients in (3.23).

Proof: The proof can be found in [103]. □

Note that this result also holds in the scalar case.

3.3.4 Asymptotic variance

In [81] the probability distribution of the parameter estimate is given for least squares estimates which also holds for model structures based on orthogonal basis functions. Let the least squares problem in terms of $\Theta = [\theta_1 \dots \theta_{n_y}]$ be given by (3.26). The probability distribution for each parameter estimate θ_i with $i = 1, \dots, n_y$ is given by the normal distribution

$$\hat{\theta}_i - \theta_i^* \in \mathcal{N}(0, P)$$

with the covariance matrix

$$cov\{\hat{\theta}_i\} = P = \Phi^\dagger E\{VV^T\}(\Phi^\dagger)^T \tag{3.27}$$

with $V = [H(q)e(1) \dots H(q)e(N)]$. For white noise disturbance this reduces to $P = \sigma^2(\Phi^T\Phi)^{-1}$. This result can be translated to an ellipsoidal confidence interval on the parameter as follows. The probability that the estimated parameter is outside the ellipsoid

$$(\hat{\theta}_i - \theta_i^*)^T P^{-1}(\hat{\theta}_i - \theta_i^*) < \alpha \tag{3.28}$$

is given by the α-level of a $\chi^2(n)$ distribution where n is the number of parameters in θ_i. The variance description (3.27) is valid for finite N and in an approximative setting, i.e. in the cases $\mathcal{S} \notin \mathcal{M}$ and $G_0 \notin \mathcal{G}$. However, it does not reveal how the choice of basis functions influences the variance. For the scalar model structure (3.20) the following result shows this dependence for scalar basis functions f_k asymptotically in both the model order and the data length.

Theorem 3.3.3 *Let $\Phi_u(\omega)$ and $\Phi_v(\omega)$ be the input and noise spectral densities, and let $\gamma_p(\omega)$ be defined as*

$$\gamma_p(\omega) = \sum_{k=1}^{p} \left| f_k(e^{i\omega}) \right|^2$$

Then for the model structure (3.20) *and $\Phi_u > 0$ and Φ_u having a finite dimensional spectral factorization, it holds that*

$$\lim_{p\to\infty} \lim_{N\to\infty} \frac{N}{\gamma_p(\omega)} cov\{G(e^{i\omega}, \hat{\theta})\} = \Phi_u(\omega)^{-1} \otimes \Phi_v(\omega)$$

Model structure	$G(q,\theta)$	$H(q,\theta)$	ORT model structure	$G(q,\theta)$	$H(q,\theta)$
FIR	$b(q,\theta)$	1	ORTFIR	$\frac{b(q,\theta)}{f(q)}$	1
ARX	$\frac{b(q,\theta)}{a(q,\theta)}$	$\frac{1}{a(q,\theta)}$	ORTARX	$\frac{b(q,\theta)g(q)}{a(q,\theta)f(q)}$	$\frac{g(q)}{a(q,\theta)}$
AR	0	$\frac{1}{a(q,\theta)}$	ORTAR	0	$\frac{g(q)}{a(q,\theta)}$

Table 3.2: *Linear black-box model structures and their orthogonal basis function analogue.*

Proof: The proof can be found in [102]. □

This shows that the variance can be influenced by a dynamic weighting with γ_p that acts on all elements of the transfer function matrix in the same way. This is a result that is similar to the work presented in [176] and [178] where the asymptotic variance of high order multivariable prediction error estimates is analyzed.

3.4 Identification of system and disturbance dynamics

3.4.1 Generalized ORTFIR structures

Until now the orthogonal basis function model structure is treated as an output error type model structure, i.e. $H = I$ in (2.34). Including an estimate of the disturbance dynamics has several advantages. From an estimation point of view, the statistically optimal estimate is a maximum likelihood estimate of which the solution is approached with prediction error identification if disturbance dynamics are accounted for. From a more deterministic point of view, the model error on the basis of the finite data set that is used during identification can be large at those frequency ranges where the (finite time) spectrum of the disturbance is large. To mitigate this effect a dynamic weighting can be applied with the inverse of the spectrum of the disturbance yielding a more accurate estimate in these frequency ranges.
Several black-box model structures can be constructed as is indicated in table 2.1 for which the prediction error is given by

$$\varepsilon(t) = H^{-1}(y(t) - Gu(t)).$$

Dependent on the choice of parametrization of G, H the optimization problem to find the optimal parameters can be simple or complex. Only a limited number of model structures

Model structure	$G(q,\theta)$	$H(q,\theta)$	ORT structure	$G(q,\theta)$	$H(q,\theta)$
ARARX	$\frac{b(q,\theta)}{a(q,\theta)}$	$\frac{1}{a(q,\theta)d(q,\rho)}$	ORTARARX	$\frac{b(q,\theta)g(q)}{a(q,\theta)f(q)}$	$\frac{h(q)g(q)}{a(q,\theta)d(q,\rho)}$
XAR	$b(q,\theta)$	$\frac{1}{d(q,\rho)}$	ORTXAR	$\frac{b(\theta)}{f(q)}$	$\frac{c(q)}{d(q,\rho)}$

Table 3.3: *Bilinear black-box model structures and their orthogonal basis function analogue.*

has the favorable property that the prediction error is linear in the parameter θ. This is only the case if H^{-1} and $H^{-1}G$ is parametrized linearly. The possibilities to obtain linearity in the parameters are given in table 3.2. These model structures can all be extended with a fixed filter to expand the flexibility. This is the fixed denominator or orthogonal basis function extension, which is also given in the table.
The fixed denominator polynomial $f(q)$ can be used to decrease the number of coefficients needed to estimate the system transfer with a certain accuracy, as explained previously. The fixed numerator polynomial $g(q)$ can be used to increase flexibility in the estimation of the disturbance transfer without the loss of linearity of the model structure. An appropriate choice for $g(q)$ can influence the trade-off between bias and variance positively. Apart from linear model structures, another class of parametrizations can have favorable properties with respect to optimization complexity. This is the class of bilinear model structures. The bilinear model structures and their orthogonal basis function analogue is given in table 3.3.

The ARARX model structure is analyzed in [143][81] together with an alternative optimization procedure that exploits the bilinearity of the model structure. The system and disturbance transfer is not parametrized independently due to the ARX part of the model structure. In the model structure we denote here by XAR, the system and disturbance transfer are parametrized independently. This is a favorable property because in this case an inaccurately estimated disturbance transfer does not introduce bias in the estimate of the system transfer. An attractive property of the ORTXAR extension is that both the system and disturbance transfer are fractions of polynomials which is the most flexible parametrization. By fixing the denominator of the system transfer and the numerator of the disturbance transfer, the model structure remains bilinear instead of fully nonlinear. This has an advantage with respect to the optimization procedure as will be made clear in the sequel.
In this section the ORTXAR model structure is analyzed. An optimization procedure is

described that is similar to the one proposed in [143] which exploits the bilinearity of the model structure. The scheme in [143] is denoted with generalized least squares (GLS) and the orthogonal basis function analogue is therefore denoted with ORT-GLS. It is a relaxation type optimization algorithm where in the consecutive steps either the linearly parametrized system transfer is fixed and the linearly parametrized disturbance transfer is optimized, or vice versa. The convergence properties of this scheme are analyzed. The treatment is based on a scalar setup for convenience of notations but can easily be extended to the multivariable case.

3.4.2 The basic principle of ORT-GLS

In the previous section the ORTXAR model structure is presented where the system transfer is parametrized with a fixed denominator and the disturbance transfer with a fixed numerator. This model set can also be described by an orthogonal basis function expansion where the favorable properties of orthogonality of the functions can be exploited. Let the system and disturbance transfer be parametrized as follows.

$$G(q,\rho) = \sum_{k=1}^{n_g} g_k f_k(q) \tag{3.29}$$

$$H(q,\eta)^{-1} = 1 + \sum_{k=1}^{n_h} h_k w_k(q) \tag{3.30}$$

with $\rho^T = [g_1 \cdots g_{n_g}]$, $\eta^T = [h_1 \cdots h_{n_h}]$ and f_k and w_k are basis functions of the form (3.5); each of the sequence of functions f and w are constructed by a particular given sequence of poles $\{a_k\}$.
The prediction error is given by

$$\varepsilon(t;\rho,\eta) = H^{-1}(q,\eta)y(t) - H^{-1}(q,\eta)G(q,\rho)u(t) \tag{3.31}$$

which is bilinear in the parameters ρ and η, and as a result the optimization problem

$$\begin{pmatrix} \hat{\rho} \\ \hat{\eta} \end{pmatrix} = \arg\min_{\rho,\eta} V(\rho,\eta) \tag{3.32}$$

$$\text{with } V(\rho,\eta) = \frac{1}{N}\sum_{t=1}^{N} \varepsilon(t;\rho,\eta)^2 \tag{3.33}$$

is a non-convex optimization problem.
A general nonlinear optimization procedure as discussed in section 2.5.6 can be utilized. For fast convergence to a local minimum usually the gradient and for some algorithms also the Hessian information must be calculated. This can be done for this cost function, however an optimization approach is available that does not need information about the

derivatives.
The alternative optimization procedure contains mainly the following steps.

$$\hat{\rho}_k = \arg\min_{\rho} V(\rho, \hat{\eta}_{k-1}) \tag{3.34}$$

$$\hat{\eta}_k = \arg\min_{\eta} V(\hat{\rho}_k, \eta). \tag{3.35}$$

The procedure is similar to generalized least squares [143] but the ARX/AR estimation is substituted by orthogonal basis function estimates which can be used to improve convergence properties of the consecutive steps. In the next section the convergence properties of this iteration are investigated.

3.4.3 Convergence properties of ORT-GLS

The iteration indicated with ORT-GLS over the steps (3.34) and (3.35) can be proceeded for a number of times but the question is whether the iteration converges. And if it converges to which parameter vector. These questions are addressed in this section.
Let the complete ORT-GLS iteration be given by the following steps.

Definition 3.4.1 Relaxation algorithm

0. *Set* $\rho_k = \rho_0$. *Set* $k = 1$.

1. $\rho_k = arg\min_{\rho} V(\rho, \eta_{k-1})$

2. $\eta_k = arg\min_{\eta} V(\rho_k, \eta)$

3. *Set* $k = k + 1$ *and go to* **1.**

With respect to this, relaxation type, algorithm the following result can be obtained.

Proposition 3.4.2 *Under persistency of excitation conditions on the input signal, the iterative algorithm 3.4.1 converges to a stationary point of the cost function* (3.33).

Proof: The proof of the proposition is along lines as indicated in [81]. It will be shown that the separate steps in the relaxation algorithm allow a recursive formulation:

$$\begin{bmatrix} \hat{\rho}_k \\ \hat{\eta}_k \end{bmatrix} = \begin{bmatrix} \hat{\rho}_{k-1} \\ \hat{\eta}_{k-1} \end{bmatrix} - \begin{bmatrix} F_1 & 0 \\ F_{21} & F_2 \end{bmatrix} \begin{bmatrix} \frac{\partial V(\rho,\eta)}{\partial \rho} \\ \frac{\partial V(\rho,\eta)}{\partial \eta} \end{bmatrix}_{\hat{\rho}_{k-1}, \hat{\eta}_{k-1}} \tag{3.36}$$

where the matrices F_1, F_2 on the righthand side are positive definite. Then it is guaranteed that in a stationary point of the iteration the first derivative of the loss function is zero. The Hessian can be shown to be positive definite under the persistency of excitation condition, therefore the point where the gradient vanishes is a minimizer. A detailed proof is given in appendix A. □

It can be proven that the iteration converges to a stationary point, i.e. a point where the gradient vanishes. Dependent on the second derivative in this point, this can be either a minimizer or a saddle point. If the system dynamics can be described exactly by the model, i.e. $G_0 \in \mathcal{G}$ it can be shown that asymptotically in the number of data points the second derivative is always positive definite. This is proven in the next proposition.

Proposition 3.4.3 *Under persistency of excitation conditions on the input signal and $G_0 \in \mathcal{G}$, the iterative algorithm 3.4.1 converges to a minimum point of the cost function 3.33 for $N \to \infty$.*

Proof: The proof is in appendix A. □

Note that the conditions are relatively strict. Dependent on the data, also for less demanding conditions the convergence point can be a minimizer. This can always be checked by inspection of the second derivative of the cost function in the specific point.
The relaxation algorithm has linear convergence properties and can therefore be slow compared to e.g. (quasi-) Newton algorithms. The advantage of the relaxation algorithm is that no calculation of the gradient information is needed to run the algorithm. It provides an algorithm that is easy to implement and provides a good approximation of the nearest stationary point in a small number of iterations. These can be attractive properties for especially large scale industrial identification problems.

3.4.4 Practical implementation

The iteration over the two optimization problems (3.34) and (3.35) is equivalent with the following steps of least squares estimation and filtering.

0. Initialize the disturbance model estimate, e.g. with $\hat{\eta}_{k-1} = 0$.

1. Estimate a model $G(q, \hat{\rho}_k)$ with structure (3.29) on the basis of a fixed noise model $\hat{H}(q, \hat{\eta}_{k-1})$ with structure (3.30). Define the residual as $\varepsilon_1(t) = y(t) - G(q, \hat{\rho}_k)u(t)$

2. Estimate a model $H(q, \hat{\eta}_k)$ with structure (3.30) [105] on the residual $\varepsilon_1(t)$ of the estimation result of step 1.

3. If the model quality is sufficient for its intended use terminate the iteration, else go to step 1 with the new noise model $H(q, \hat{\eta}_k)$.

In step 1 an ORTFIR estimate is performed which is a least squares problem. The filter that is utilized is the inverse of the estimated disturbance dynamics $H^{-1}(q, \hat{\eta})$. In the scalar case the filtered prediction error identification is equivalent with identification on the basis of filtered input and output signals. In the multivariable case the prediction

error must be filtered. The proposed initialization $\hat{\eta}_{k-1} = 0$ amounts to $H(\hat{\eta}_{k-1}) = I$.
In step 2 of the ORT-GLS algorithm an estimate is made of the disturbance transfer on the basis of the residual $\hat{\varepsilon}_1(t) = y(t) - G(q, \hat{\rho})u(t)$ of the transfer function estimate of step 1. A common strategy to obtain a parametric dynamical model of a signal, e.g. the disturbance signal $v(t)$, is by autoregressive (AR) modeling. In this section it is shown how AR modeling can be improved by the use of orthogonal basis functions.
Let the spectrum of the disturbance be given by

$$\Phi_v(e^{i\omega}) = \sigma^2 |H(e^{i\omega})|^2$$

In [81] it is described how the disturbance transfer can be estimated from a realization of the signal $v(t)$ by prediction error methods. The one-step ahead prediction $\hat{v}(t)$ can be derived as follows

$$v(t) = H(q)e(t) \Rightarrow \hat{v}(t) = [H(q) - 1]e(t) = [1 - H^{-1}(q)]v(t) = \sum_{k=1}^{\infty} \tilde{h}(k)v(t-k)$$

where $\tilde{h}(k)$ are the impulse response coefficients of

$$\tilde{H}(z) = 1 - H^{-1}(z). \tag{3.37}$$

Now a finite impulse response model can be estimated for the transfer $\tilde{H}(z)$ by using the AR model structure given by

$$\mathcal{M}_{AR} : \hat{v}(t, \theta) = -\theta_1 v(t-1) - \theta_2 v(t-1) - \cdots - \theta_{n_H} v(t - n_H)$$

This AR estimation has the same inefficiency as the FIR estimate of system transfers: for slow dynamics in the disturbance filter a large number of coefficients has to be estimated to obtain an accurate estimate. To remedy this, system-based orthogonal basis functions are applied to specify a more efficient AR model structure as described in [105]. This model structure, denoted as ORTAR, is given by

$$\mathcal{M}_{ORTAR} : v(t, \eta) = \sum_{k=0}^{n_h} h_k w_k(q) v(t) \tag{3.38}$$

with the scalar basis functions $w_k(z)$ of the form (3.5). Note that $\tilde{H}$ is estimated with the orthogonal basis function structure rather than H, but the disturbance dynamics can be calculate from the estimation result with (3.37). Therefore in (3.30) the parametrization is chosen as such.
Asymptotically in N and n_h this estimate is consistent and the asymptotic variance is given in the next theorem.

Theorem 3.4.4 *Assume that the spectral density $\Phi_v(\omega)$ has a finite dimensional spectral factorization and is bounded away from zero and is sufficiently smooth. Then for $N, n_h \to \infty, n_h^2/N \to 0$, it holds that*

$$\frac{N}{2} cov\{H^{-1}(e^{i\omega_1}, \hat{\rho}_N), H^{-1}(e^{i\omega_2}, \hat{\rho}_N)\} \to$$

$$\begin{cases} 0 & for\ H_b(e^{i\omega_1}) \neq H_b(e^{i\omega_2}) \\ \sum_{k=1}^{n_h} w_k^*(e^{i\omega_1}) w_k(e^{i\omega_1}) {\Phi_v}^2(\omega_1) & for\ \omega_1 = \omega_2 \end{cases}$$

Proof: Follows from [105]. □

Also the basis functions with batch wise repetition can be utilized here. These basis functions are given by

$$W_k(z) = (zI - A_H)^{-1} B_H H_b^{k-1}(z) \tag{3.39}$$

with the inner function $H_b(z) = C_H(zI - A_H)^{-1}B_H + D_H$ of order n_b.
For this choice of basis function the covariance of the estimate reduces to

$$\frac{2n_b n_h}{N} W_1^*(e^{i\omega_1}) W_1(e^{i\omega_1}) {\Phi_v}^2(\omega)$$

where only the first basis function W_1 influences the covariance.
Prior knowledge of the dynamics of $\tilde{H}(z)$ can be incorporated in these basis functions to obtain a fast convergence of the expansion coefficients. This prior knowledge can be obtained e.g. by an ARMA estimate on the signal $\varepsilon(t)$, or an AR estimate followed by a realization algorithm to obtain a state space realization of the transfer. The parametrization in terms of the basis functions can reduce the number of free parameters that are needed to obtain an accurate estimate if an AR estimation is performed.

3.5 Design issues

3.5.1 Motivation

To utilize multivariable model structures to identify reliable models on the basis of industrial data, a number of issues must be resolved. In this section several extensions to the basic estimation procedure with orthogonal basis functions are presented, that improve the estimation accuracy and are in some cases indispensable. Due to the convexity of the optimization problem these features can usually be added without a large increase in computational complexity.
First, the consequences of a finite length of the data are analyzed. The basis functions are orthogonal on a semi-infinite axis but in practice a finite number of data points is available. The orthogonality property disappears and the perfect conditioning may be lost. In this section a number of considerations with respect to finite data and conditioning are given.
Secondly, constraints can be added to the least squares optimization problem to add certain prior knowledge of the system to the problem. Useful in the identification of industrial processes, is a constraint that fixes the steady state of the model to a certain prespecified value. The estimation of the static gain and low frequency behaviour of the

model can be inaccurate due to the fact that the data sets that are available are usually too short to capture the static behaviour of the system. If prior knowledge of the system gain is available due to e.g. available step test results or operator knowledge, the static behaviour of the model can be improved considerably by adding this knowledge to the identification procedure. In this section it is shown how this can be done in the estimation with orthogonal basis functions. Both hard and soft constraints are demonstrated.
Next, if experiments are conducted with a plant that is not in steady state the estimated models usually exhibit a bias due to the initial response of the system if this is not accounted for in the estimation. Hence, estimation of initial conditions is important if limited data is available. In this section it is shown how this works out in ORTFIR estimation.
Lastly, it is possible that only a number of short data sets are available because e.g. only short experiments were allowed. In that case it is important to be able to obtain an optimal model for all these data sets. This multiple data set estimation is analyzed in this section.

3.5.2 Finite data aspects

In the case of finite data the perfect conditioning property 3.2.1 for white input signals is lost. The regression matrix ϕ in (3.26) consists of input data filtered with the basis functions. Therefore, there are two causes for the loss of the perfect conditioning:

- finite time aspects of the input signal,
- finite time aspects of the basis functions.

In this section the finite time properties of the least squares problem (3.26) are analyzed. Conditioning and variance of the estimate have a strong relation. A least squares estimate is said to be badly conditioned if the condition number $\kappa(\Phi^T\Phi) = \frac{\sigma_{max}}{\sigma_{min}} = \frac{\lambda_{max}}{\lambda_{min}}$ is large. The last relation follows from the fact that for symmetric matrices the singular values are equal to the eigenvalues. The variance of a least squares estimate can be geometrically represented by the ellipsoidal region given by (3.28). For white noise disturbances with variance σ the ellipsoid is determined by $P = \sigma^2(\Phi^T\Phi)^{-1}$. The long axis of the ellipsoid, i.e. the direction with large variance, is determined by the smallest eigenvalue of $\Phi^T\Phi$ and its corresponding eigenvector. The short axis is determined by $\lambda_{max}(\Phi^T\Phi)$ and its corresponding eigenvector. Hence, bad conditioning reflects the size of the variance ellipsoid. Perfect conditioning coincides with a circular variance region, bad conditioning with a long slim ellipsoid. Note that conditioning is a relative measure and does not reflect the size, only the shape of the variance. The size of the variance can be best read from $\lambda_{min}(\Phi^T\Phi)$ which indicates the large axis of the variance ellipsoid.

A small value for $\lambda_{min}(\Phi^T\Phi)$ indicates that the cost function relatively insensitive for variations in certain parameter directions. Therefore, bad conditioning of the least squares problem can express itself in senseless (e.g. very large) values for the calculated parameters. Note that the values of the parameter vector can be large, although the input output description, e.g. in terms of the Markov parameters, can have sensible values.
There are several reasons why it is desirable to obtain a parameter estimate without unnecessary large coefficients although the Markov parameters are estimated well. First, it appears that no guarantee can be given about the properties of the Markov parameter after the time window N. An example of this can be found in [5]. Secondly, the state space description (3.11) $\{A_e, B_e, C_e(\hat{\theta})\}$ of the estimated model can be far from a balanced realization which make operations on this system description like balancing, needed for e.g. balanced reduction, inversion and control design techniques badly conditioned.
The problem of finding a sensible solution to ill conditioned least squares problems has a long history and is called regularisation. In [99] the history of this topic of research and several regularisation techniques are discussed to mitigate the negative aspects of ill-conditioning. In this section the source of the ill-conditioning in the specific case of orthogonal basis function estimation is analyzed. Several remedies are given that are specific for the orthogonal basis function estimation and cannot be found in regularisation literature due to their specific nature.
How the fact that only finite length measurement is available leads to ill conditioning can be seen from the following rewritten version of the LS problem. The overdetermined set of equations can be written as

$$\underbrace{\begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} u(1) & & \\ \vdots & \ddots & \\ u(N) & \cdots & u(1) \end{bmatrix}}_{\mathcal{T}_u} \underbrace{\begin{bmatrix} B^T \\ \vdots \\ B^T A^{T\,N-1} \end{bmatrix}}_{g} \theta \tag{3.40}$$

$$= \underbrace{\begin{bmatrix} B^T & & \\ \vdots & \ddots & \\ B^T A^{T\,N-1} & \cdots & B^T \end{bmatrix}}_{\mathcal{T}_g} \underbrace{\begin{bmatrix} Iu(1) \\ \vdots \\ Iu(N) \end{bmatrix}}_{g_u} \theta \tag{3.41}$$

where for simplicity of notation the case of identification of a scalar system is given. The least squares optimal solution is given by

$$\hat{\theta} = (g^T\mathcal{T}_u^T\mathcal{T}_u g)^{-1} g^T\mathcal{T}_u^T Y \tag{3.42}$$

$$= (g_u^T\mathcal{T}_g^T\mathcal{T}_g g_u)^{-1} g_u^T\mathcal{T}_g^T Y \tag{3.43}$$

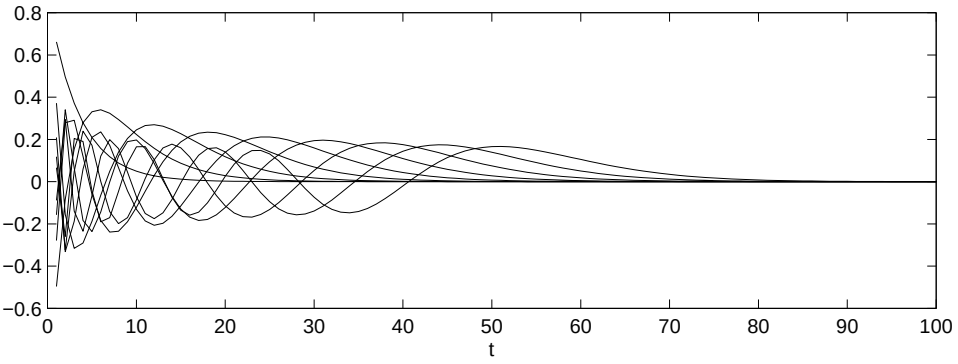

Fig. 3.3: *First 10 Laguerre functions for $a = 0.75$ plotted in one figure to illustrate the effect of the build up of support over the time axis for increasing function numbers.*

The matrix $(g\mathcal{T}_u\mathcal{T}_u^T g^T)$ has to be inverted. For infinite time $\frac{1}{N}T_u^T T_u = I$ for white input signals and $g^T g = I$ because of the input balancedness of the pair $\{A, B\}$. In the finite time case there are two causes for the loss of this favorable property:

- finite times aspects of the input signal causing T_u to be nonunitary, and
- finite time aspects of the basis functions causing g to be nonunitary.

Several aspects influence the finite time conditioning of the least squares problem

- the dynamics of the basis functions. If the dynamics of the basis functions are damped out within the time window of length N the conditioning of $\mathcal{T}_g$ is better than for basis functions that have energy outside this window. Note that it is not sufficient to choose the data window length N such that the basis functions are damped out sufficiently. This will yield $g^T g = I$ but still bad conditioning can occur because $T_u^T T_u \neq I$ according to equation (3.42). Only if the input signal is a pulse the requirement on the basis functions is sufficient for optimal conditioning.
- the number (of repetitions) of basis functions. The energy in the basis function shifts to later time instances as the number n increases. Intuitively this is necessary to be able to span the whole space of stable transfer functions. The negative effect of this is that by increasing n, the basis function shifts out of the data window and conditioning is lost. This effect is illustrated in figure 3.3 for a Laguerre basis. This effect is stronger for slow dynamics.

This points to the design rule that the dynamics in the basis functions and the order of the expansion should be chosen according to the length of experimentation time. This can improve the conditioning of the least squares problem. In some cases, however, it is not possible. An example of such a situation is the iterative scheme presented in the next

chapter where slow dynamics may be present in the system and hence in the basis while only a short data set is available.

Several regularisation techniques are available to improve the conditioning of a general least squares problems. In [51] the near rank deficient least squares problem is treated and several alternatives are provided to find a good approximation of the solution, e.g. using the pseudo inverse with a certain cut-off value for the small singular values and appropriate diagonal scaling. The use of singular values is the most sensible approach but can be time consuming. A large class of regularisation techniques append the least squares problem to

$$\hat{\theta} = (\Phi^T\Phi + X)^{-1}\Phi^T Y$$

where X is some prechosen positive definite matrix. In Tikhonov regularisation or ridge regression $X = \varepsilon^2 I$ [149], in generalized ridge regression more flexible choices are made [36]. In [99] a more extensive and detailed overview is given of this topic of research.

Although these regularisation methods mitigate the effect of bad conditioning, in some cases the choices that have to be made are not clear and must be done by trial and error. This is undesirable and therefore a regularisation technique is developed that is tailored to the problem at hand and that fruitfully applies the orthogonality property of the basis functions.

In [5] a regularization is proposed that is specific for identification with orthogonal basis functions and makes use of the insight in the origin of the bad conditioning. Basically, they propose to extend the blockcolumns of the matrix $\mathcal{T}_g$ in (3.41) together with an extension of the output vector

$$\left[\begin{array}{c} y(1) \\ \vdots \\ y(N) \\ \hline Y_{ext} \end{array}\right] = \underbrace{\left[\begin{array}{ccc} B_e^T & & \\ \vdots & \ddots & \\ B_e^T A_e^{T\,N-1} & \cdots & B_e^T \\ \hline B_e^T A_e^{T\,N} & \cdots & B_e^T A_e^T \\ \vdots & & \vdots \\ B_e^T A_e^{T\,N+m} & \cdots & B_e^T A_e^{T\,m} \end{array}\right]}_{\mathcal{T}_{ext}} \left[\begin{array}{c} Iu(1) \\ \vdots \\ Iu(N) \end{array}\right] \theta \tag{3.44}$$

In this way the transients of basis functions generated are extended outside the data window, improving the numerical conditioning of the regression matrix. If the length of the extension m is long enough it will hold that $\mathcal{T}_{ext}^T\mathcal{T}_{ext} \approx I$ due to orthogonality of the basis functions. The extension of the output Y_{ext} is not known a priori due to the influence of the unknown parameter θ. Therefore, an improvement of the conditioning is obtained at the cost of a bias due to the error in the output extension.

This approach indeed provides an increase in the numerical conditioning but at the cost of

a (possibly) large increase in size of the least squares problem, dependent on the number of additional expansion coefficients needed to obtain an acceptable numerical conditioning. Also, the number m needs to be chosen by trial and error to obtain a reasonable trade-off between computational complexity and numerical conditioning. Therefore, an alternative procedure is suggested in this thesis.
Let a model be available identified using an ORTFIR model structure (3.24), given by

$$G(z, \hat{\Theta}) = \hat{\Theta}^T (zI - A_e)^{-1} B_e$$

where $\hat{\Theta}$ is obtained by the solution of an ill-conditioned least squares problem. The estimated parameter $\hat{\Theta}$ contains large values due to the bad numerical conditioning. Assume that at least the first N Markov parameters of $\hat{G}$ have values that have a realistic size. This is usually the case due to the properties of A_e, B_e which mitigate the effect of the large numbers in $\hat{\Theta}$.
Now, regard the problem of fitting the first N Markov parameters of the estimated model given by

$$\begin{bmatrix} \hat{g}(1) \\ \vdots \\ \hat{g}(N) \end{bmatrix} = \begin{bmatrix} B_e^T \\ \vdots \\ B_e^T A_e^{T\ N-1} \end{bmatrix} \hat{\Theta}. \tag{3.45}$$

Let the overdetermined set of equations be extended as follows

$$\begin{bmatrix} \hat{g}(1) \\ \vdots \\ \hat{g}(N) \\ \hline \hat{g}_{ext} \end{bmatrix} = \underbrace{\begin{bmatrix} B_e^T \\ \vdots \\ B_e^T A_e^{T\ N-1} \\ \hline X_{ext} \end{bmatrix}}_{X} \Theta_{reg}$$

where Θ_{reg} is the regularized parameter that must be calculated and $\hat{g}_{ext}, X_{ext}$ are extensions that must be chosen a priori. If $\hat{g}_{ext} = 0, X_{ext} = 0$ no regularization is obtained, i.e. $\Theta_{reg} = \hat{\Theta}$. For other choices a trade-off is made between fitting the first N parameter of the estimated model $\hat{G}$ and fitting $X_{ext}\Theta_{reg}$ to $\hat{g}_{ext}$. This extension can be utilized to restrict the size of the estimated parameters. The question is how to choose the extensions $\hat{g}_{ext}, X_{ext}$ wisely.
A choice for X_{ext} that makes fruitful use of the orthogonality of the basis is given by the infinite dimensional matrix

$$X_{ext} = [A_e^N B_e \ A_e^{N+1} B_e \ A_e^{N+2} B_e \cdots]^T \tag{3.46}$$

With this extension it holds that $X^T X = I$, due to orthogonality of the basis functions which implies that the pair $\{A_e, B_e\}$ is input balanced. With respect to $\hat{g}_{ext}$ several

extensions can be made which lie between $g_{ext} = 0$ and g_{ext} is equal to the tail of the impulse response of the estimated model $\hat{G}$. In the later case the regularized parameter is equal to the original parameter estimated and no regularization is obtained. In the first case a regularization of the parameter estimated is achieved, but at the cost of a bias introduced by ignoring the tail of the pulse response. However, in this specific least squares problem that results for an identification with orthogonal basis functions, a particularly nice computational procedure can be used to obtain the regularized parameter. For $\hat{g}_{ext} = 0$ and X_{ext} given in (3.46) it holds that

$$\Theta_{reg} = \sum_{t=0}^{N-1} A_e^t B_e B_e^T A_e^{T\,t} \hat{\Theta} = (I - A_e^N A_e^{T\,N}) \hat{\Theta}. \tag{3.47}$$

In the last equality it is used that the controlability Gramian of the input balanced pair $\{A_e, B_e\}$ can be written as

$$\begin{aligned} I &= \sum_{t=0}^{\infty} A_e^t B_e B_e^T A_e^{T\,t} = \sum_{t=0}^{N-1} A_e^t B_e B_e^T A_e^{T\,t} + \sum_{t=N}^{\infty} A_e^t B_e B_e^T A_e^{T\,t} \\ &= \sum_{t=0}^{N-1} A_e^t B_e B_e^T A_e^{T\,t} + A_e^N \left(\sum_{t=0}^{\infty} A_e^t B_e B_e^T A_e^{T\,t} \right) A_e^{T\,N} = \sum_{t=0}^{N-1} A_e^t B_e B_e^T A_e^{T\,t} + A_e^{T\,N} A_e^N. \end{aligned}$$

Therefore, the regularized parameter can simply be calculated from the estimated parameter by $\hat{\Theta}_{reg} = T\hat{\Theta}$ with $T = I - A^N A^{T\,N}$, which can be calculated fast, also for large dimensions.

The consequences of bad conditioning are illustrated in the next example as well as the result of the proposed regularisation.

Example 3.5.1 Consequences of finite data length. *The system is a scalar subtransfer of the nonlinear model of the fluidized catalytic cracking unit, discussed in chapter 7, exhibiting both fast and slow dynamics. Let* $N = 25$ *pulse response coefficients* $g(k)$ *be available of this system. A basis generating pair* $\{A_b, B_b\}$ *is available of order* 9 *with poles located at* $\lambda = \{-0.0605 \pm 0.8066i, 0.3573, 0.7510 \pm 0.2531i, 0.8605, 0.9957, 0.9955 \pm 0.0103i\}$. *The ill-conditioned least squares problem is observed that arises when an ORT-FIR model is estimated of order 18 with a basis construction method in which the dynamics is repeated. The source of the ill-conditioning is the short data set in combination with the slow dynamics in the basis functions. In the figure the pulse response data is given together with the pulse response of the model estimated by calculating the least squares solution to* (3.45) *and the pulse response of the regularized estimate. Also the estimated parameter vectors are given.*

The parameter estimate has unrealistically large values of up to 10^{12} *due to ill-conditioning. The regularization approach discussed above provides more realistic parameter values without a visible deterioration of the estimated Markov parameters.*

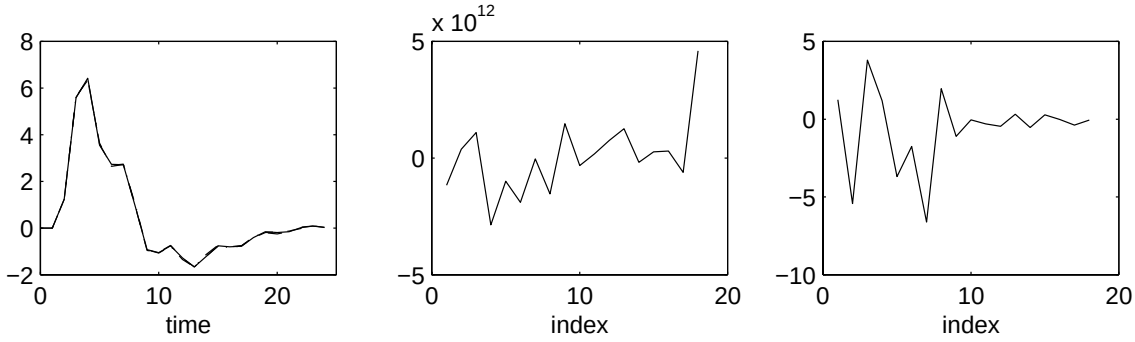

Fig. 3.4: *Left: Measured pulse response (solid) and estimation without regularization (dashed) and with regularization following equation* (3.47) *(dash dotted) which all practically coincide. Middle: estimated parameters* $\hat{\Theta}$ *without regularisation. Right: estimated parameters* $\hat{\Theta}_{reg}$ *with regularisaton.*

3.5.3 Selection of poles

An important question that is addressed next is how to choose the dynamics in the basis functions, i.e. the location of the poles a_k in the basis functions (3.5), the matrices $\{A, B\}$ in the basis functions (3.9) and the matrices $\{A, C\}$ in the basis functions (3.10).
Several theoretical results have been obtained with respect to the optimal pole choice for Laguerre functions [170][112], two parameter Kautz functions [113] [27][162] and batch wise repeating general dynamics [115][114]. Basically two scenarios are analyzed in these papers:

- The situation is observed that an estimated expansion (3.1) of a specific order is available. The question that is addressed is when is the pole location of the basis functions optimal in terms of the mean square error (MSE). An eloquent result with respect to Laguerre functions (3.2) is that the pole location a is optimal in an expansion (3.1) of order n if $L_n = 0$ and/or $L_{n+1} = 0$. Optimality is measured here as function of the pole location. A similar result can be found for an expansion of $2n$ Kautz functions (3.3) and (3.4) which states that the pole location is optimal if $[L_{2n-1}\ L_{2n}] = 0$ and/or $[L_{2n+1}\ L_{2n+2}] = 0$.

- The second situation that is observed in literature is the situation where prior knowledge is available that the poles of the system belong to some subset of the interior of the unit circle. The question that is addressed is which is the best set of n basis functions such that the approximation of the form (3.1) has the smallest possible squared error for the worst possible nominal model that satisfies the prior knowledge about the pole location. A revealing result is the following. If the only prior knowledge that is available about the system, is that it has its poles in the region $|z| < \lambda_{max} < 1$, then the best basis is the FIR basis $f_k = z^{-k}$. If more information

about the system is available, which generally is the case, other bases are preferable.

From a more practical point of view it is relevant to know how numerical values of the pole locations can be found. An obvious way to start is to estimate a model with another model structure, e.g. ARX which can be advantageous due to linearity in the parameters, or another technique, e.g. subspace identification which can be advantageous because no (nonconvex) optimization is required to find a state space model for the given data. This provides a model $\{\hat{A}, \hat{B}, \hat{C}, \hat{D}\}$ of which the poles $\lambda(A)$, the state space matrices $\{\hat{A}, \hat{B}\}$ or $\{\hat{A}, \hat{C}\}$ or structural information about the distribution of the poles can be used as prior knowledge to construct the basis functions.
If the prior information about the system dynamics and structure coming from $\{\hat{A}, \hat{B}, \hat{C}, \hat{D}\}$ is not accurate, this will be reflected in the large number of expansion coefficients that needs to be estimated to obtain a certain accuracy. This results in an estimate $\hat{G}(z) = \hat{\Theta}^T(zI - A_e)^{-1}B_e$ that can have a a high order. In the next chapter it is shown how this high order model can be used to find a set of improved basis functions.

3.5.4 Estimation of initial conditions

In the measured data of a process sometimes a transient is present due to an initial condition that is not a stationary working point. If ample experimentation freedom is available one will usually wait until the process is in rest before the excitation signal is applied. If the experimentation freedom is limited or the system has very slow modes it may be necessary to start experimenting with the system while it is not at rest. The result of this is that the transient behaviour of the system is present in the data which cannot be explained as a response to the applied input. If no precautions are taken this results in a biased estimate.
To account for this, the initial condition can be estimated simultaneously with the system dynamics. This can be done without losing the linear regression structure if $\{A, C\}$ is used as prior information. For this purpose the model structure ($\{A, C\}$ variant) is extended to

$$\hat{y}(t, \theta) = C_e(qI - A_e)^{-1}B_e(\theta)u(t) + C_e A_e^{t-1}x_0 \tag{3.48}$$

where $\{A_e, C_e\}$ are a priori chosen state-space matrices and $B_e(\theta)$ and $x(0)$ are the parameters that are to be estimated from the data. It is important to note that with the $\{A, B\}$ variant an estimation problem is obtained that is nonlinear in the parameters. This can be seen by writing

$$\hat{y}(t, \theta) = C_e(\theta)(qI - A_e)^{-1}B_e u(t) + C_e(\theta)A_e^{t-1}x_0$$

where products occur of the free parameters in $C(\theta)$ and x_0.

Calculation of the parameters with model structure (3.48) boils down to solving the least-squares optimal parameter vector for the overdetermined set of equations

$$Y = [\phi \ \phi_{x0}] \begin{bmatrix} \theta \\ x_0 \end{bmatrix} := \phi_{ext}\theta_{ext}$$

where the rows of ϕ_{x0} are given by CA^{t-1}. Estimation of initial conditions can be used to reduce the bias due to unknown initial conditions at the expense of an increased variance. That the variance increases can be seen from the finite time covariance of the parameter estimate. For white output disturbances with covariance σ^2 this is similar to (3.27) given by

$$cov(\hat{\theta}_{ext}) = \sigma^2(\phi_{ext}^T\phi_{ext})^{-1} = \sigma^2 \begin{bmatrix} \phi^T\phi & \phi^T\phi_{x_0} \\ \phi_{x_0}^T\phi & \phi_{x_0}^T\phi_{x_0} \end{bmatrix}$$

By the matrix inversion lemma [175] for partitioned matrices it follows that

$$\frac{1}{\sigma^2}cov\hat{\theta} = (\phi^T\phi)^{-1} + (\phi^T\phi)^{-1}\phi^T\phi_{x_0}\Delta^{-1}\phi_{x_0}^T\phi(\phi^T\phi)^{-1}$$

where the Schur complement is given by $\Delta = \phi_{x_0}^T\phi_{x_0} - \phi_{x_0}^T\phi\ (\phi^T\phi)^{-1}\phi^T\phi_{x_0}$. The covariance for the parameter estimate with estimation of initial conditions is therefore equal to the covariance of the estimate without estimation of initial conditions $\sigma^2(\phi^T\phi)^T$ added with a positive definite matrix. Therefore the variance increases due to estimation of initial conditions. The gain is, however, in a decrease in the bias.

An important prerequisite to estimate the initial conditions is that the matrix $[\phi \ \phi_{x_0}]$ has full column rank. If this is not the case, the contribution of the initial condition cannot be distinguished from the influence coming from the excitation signal. In that case either a change in basis functions or experimental conditions like data length, input spectrum etc. is required.

3.5.5 A bilinear model structure

The orthogonal basis function model structures denoted with the $\{A, B\}$ and $\{A, C\}$ variant can be very efficient if the structural information present in the state space matrices resembles the structural properties of the system. If the structural information is incorrect, a larger number of parameters is required to identify the system with the same accuracy. Instead of increasing the number of expansion coefficients, and with that also the model order, to improve the model accuracy, another extension of the number of parameters can be utilized. This extension does not increase state space order of the model but is no longer linear but bilinear in the parameters. Such an extension of the (A, B) and (A, C) structures is given by the following (bilinear) model structure

$$G(z, \Theta_L, \Theta_R) = \Theta_L(zI - A)^{-1}\Theta_R$$

where both the B and the C matrix are parametrized freely. Linearity in the parameters and guaranteed convexity of the optimization problem are lost. What is gained is an increased flexibility in the parametrization that results in an improved value of the cost function for the estimated model. Due to the fact that the model structure is bilinear, again, a relaxation type of algorithm can be used as an alternative for more conventional nonlinear optimization solvers. The following scheme approaches this bilinear estimation problem in consecutive linear steps.

1. Estimate a model $\{A_1, B_1, C_1\}$ using the $\{A, C\}$ variant, i.e. estimate the matrix B_1, if desired including initial conditions.
2. Subtract the estimated initial condition effect from the data.
3. Input balance the pair $\{A_1, B_1\}$ to $\{A_2, B_2\}$.
4. Estimate a model $\{A_2, B_2, C_2\}$ using the $\{A, B\}$ variant, i.e. estimate the matrix C_2, without initial conditions, from the revised data.
5. Output balance the pair $\{A_2, C_2\}$ to $\{A_3, C_3\}$.
6. Estimate a model $\{A_3, B_3, C_3\}$ using the $\{A, C\}$ variant, i.e. estimate the matrix B_3, if desired including initial conditions, using the original data.
7. Subtract the estimated initial condition effect from the original data.
8. Terminate if the model quality is sufficient, else go to 1.

For the case without initial conditions this scheme will always converge, since the criterion function can only decrease. Experimental experience indicates that in general more than 95 % of the improvement (in terms of decrease of the criterion function) is reached within one or two iteration steps.
An important feature of this iterative scheme is that it creates flexibility to correct errors in the imposed structure. Also the possibility to detrend the data with an estimate of the initial condition influence can be advantageous, especially in cases where the system is not in rest during the excitation and the data set is short compared to the settling time of the system.

3.5.6 Constrained estimation

In some cases prior knowledge of the system is available that can be expressed in terms of a constraint on the model. An example is the steady state gain of the system or the frequency response at other frequencies. The low frequent and static behaviour of the system is barely present in experimental data if the data length is relatively short

compared to the slowest time constant, which is often the case in industrial process identification. Therefore the static gain of the estimated models can be inaccurate. To remedy this, the static gain can be enforced on the model by means of a constraint, that is linear in the parameter vector. The static gain of the model with model structure 3.24 is given by

$$K_{ss}(\theta) = C(I - A)^{-1}\Theta := Q\theta \tag{3.49}$$

The steady state gain can be enforced on the estimated model by solving the constrained optimization problem

$$\min_{\theta} V(\theta) \text{ subject to } Q\theta = K_{ss}$$

where the cost function $V(\theta)$ is the standard prediction error cost functions and K_{ss} is the prior knowledge about the steady state gain. This is an equality constrained optimization problem that is convex. This problem can be solved, similar to the unconstrained case, by linear regression techniques using the Lagrangian (2.5). This can be seen by observing the Lagrange necessary conditions (2.6) and (2.7) of the constrained optimization problem, which boil down to

$$\begin{bmatrix} \phi^T\phi & Q^T \\ Q & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \lambda \end{bmatrix} = \begin{bmatrix} \phi^T Y \\ K_{ss} \end{bmatrix}$$

where λ is the Lagrange multiplier. This set of equations is uniquely solvable because the matrix on the left hand side is square and invertible.
The constraint is enforced on the model such that the steady-state gain is equal to the specified one. However, the steady-state gain taken from the step response data may not be accurate; therefore possibly unnatural behaviour is enforced on the model. To alleviate this, *soft* constraints are used, which are constraints that can be violated. With a soft constraint a term is added to the cost function to penalize the deviation of the steady state of the model and the prior knowledge of the steady state

$$V_{ext}(\theta) = V(\theta) + (K_{ss} - Q\theta)^T P(K_{ss} - Q\theta) \tag{3.50}$$

where $P \in \mathbb{R}^{n_y \times n_y}$ is a weighting matrix that can be chosen a priori. If (an estimate of) the covariance matrix of the prior knowledge on the steady state $cov(K_{ss})$ is available and is invertible, the choice $P = cov(K_{ss})^{-1}$ provides a larger weighting on the accurate prior information and a lower on the inaccurate parts. It can be shown that this is optimal in maximum likelihood sense. Note that if P goes to infinity, the soft constraint is enforced on the model and becomes a hard constraint.

3.5.7 Combination of data sets in system identification

In many cases the experimentation time is limited but there is a possibility to conduct several short experiments with the system. Also measurements are available of the system without excitation or with insufficient excitation that still contain information about the system and disturbance dynamics. In that case a procedure is required to combine several experiments in one identification.
In this section a systematic procedure is described and analyzed to identify a model with a linear regression structures with a deterministic regressor (FIR, ORTFIR) on the basis of a number of separate data sets. The basic principle holds true for all model structures but the analysis is restricted to FIR and ORTFIR model structures. The input signal of these data sets can have a different character, i.e. random inputs, step inputs and not sufficiently exciting inputs can be combined. In [81] a similar approach is mentioned but not further analyzed.
Let n_d input/output data sets of length N be given by:

$$\mathcal{D} : \{U_1, Y_1\}, \ldots, \{U_{n_d}, Y_{n_d}\}$$

where

$$U_i^T = [u^T(1) \ldots u^T(N)] \in \mathbb{R}^{Nn_u}, Y_i^T = [y^T(1) \ldots y^T(N)] \in \mathbb{R}^{Nn_y} \text{ for } i = 1, \ldots, n_d$$

are the measured inputs and outputs stacked in a vector. For convenience of notation the data sets have the same length but this is not required. The input can be either a known realization of a stochastic process such as filtered white noise, step or impulse signals, a multisine or any other excitation signal. At this point sufficiency of excitation for each data set is not required therefore also free run experiments can be applied.
The problem is to identify a model and uncertainty bounds on the basis of these data sets. An optimal parameter estimate can be obtained by solving the following optimization problem

$$\mathcal{I} : \hat{\theta} = arg \min_{\theta} \{V_1(\theta) + \ldots + V_{n_d}(\theta)\} \tag{3.51}$$

where $\phi_i^T = [\varphi_i(1) \cdots \varphi_i(N)]$ is the regression matrix of the ith data set and

$$V_i(\theta) = \|Y_i - \phi_i\theta\|_2 = \theta^T\phi_i^T\phi_i\theta + 2Y_i^T\phi_i\theta + Y_i^TY_i \text{ for } i = 1, \ldots, n_d \tag{3.52}$$

is the least squares cost functions for the ith data set. Hence, the overall cost function (3.51) is the sum of the least squares cost functions of the separate data sets.
For convenience of notation but without loss of generality the number of data sets is taken $n_d = 2$ and the Hessian and gradient of the quadratic cost function (3.52) are respectively denoted as

$$H_i := \phi_i^T \phi_i \text{ and } g_i := 2\phi_i^T Y_i \tag{3.53}$$

The solution to the optimization problem (3.51) is equivalent to the least squares solution to the overdetermined set of equations given by

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} \theta$$

The optimal solution can be calculated analytically by

$$\hat{\theta} = \left(\sum_i^{n_d} H_i \right)^{-1} \left(\sum_i^{n_d} g_i \right) \tag{3.54}$$

which is a rather obvious extension to the least squares solution of one data set. This points however to a different definition of sufficiency of excitation.

Definition 3.5.2 *A set of inputs $\{U_i\}_{i=1,\ldots,n_d}$ is denoted as sufficiently exciting of order n if*

$$rank \left\{ \sum_{i=1}^{n_d} H_i \right\} \geq n$$

where H_i is given by (3.53)*.*

Under this definition experiments can be used where the input is not sufficiently exciting of order n for one data set, i.e. $rank(H_i) < n$ as long as the combination of experiments satisfies the sufficiency of excitation condition. The following lemma shows that if one of the experiments is sufficiently exciting, the combination is also sufficiently exciting.

Lemma 3.5.3 *Let a set of inputs be given by $\{U_i\}_{i=1,\ldots,n_d}$. A sufficient condition for persistency of excitation of these inputs according to definition 3.5.2 is that at least one input is persistently exciting of order n.*

Proof: Let input U_j be sufficiently exciting of order n, i.e. $rank\{H_j\} \geq n$ and the other input are sufficiently exciting of order lower than n. It holds that $H_j = \phi_j^T \phi_j > 0$ and $H_i \geq 0$. Hence, $\sum_{i=1}^{n_d} H_i > 0$. □

If all experiments are sufficiently exciting of the prespecified order, the separate single data parameters can be calculated. There exists a relation between the multiple data estimate and single data parameter estimates. This is given in the next proposition.

Proposition 3.5.4 *Let a set of inputs be given by $\{U_i\}_{i=1,\ldots,n_d}$ where each of the elements is sufficiently exciting of order n. Let the single data estimate be given by*

$$\hat{\theta}_i = H_i^{-1} g_i \tag{3.55}$$

Then the multiple data estimate $\hat{\theta}$ *of* (3.51) *is the weighted average*

$$\hat{\theta} = \left(\sum_{i=1}^{n_d} H_i\right)^{-1} \left(\sum_{i=1}^{n_d} H_i \hat{\theta}_i\right) \tag{3.56}$$

Proof: Filling in (3.55) in (3.56) yields (3.54) □

This indicates that the multiple data estimate could have been derived equally well by taking the weighted average of the optimal parameter estimate based on the separate input output data. Note that this result is only valid of all the inputs are separately sufficiently exciting, while the multiple data identification is valid under the sufficiency of excitation definition 3.5.2 which is much weaker.
The bias and variance results for this model structure are fairly similar to the bias and variance results of a single linear regression estimate [81]. Under similar conditions the estimation based on multiple data sets is asymptotically unbiased.

Definition 3.5.5 Standard conditions. *The following conditions are denoted as* standard conditions*:*

1. *the regressor* $\varphi(t)$ *is deterministic,*
2. *the set of inputs* $\mathcal{U}$ *is sufficiently exciting in the sense of definition 3.5.2.*

Under these condition nice expressions for the asymptotic bias and variance of the multiple-data estimation can be obtained, which are similar to the ones obtained in the regular least squares identification setting. First the result is given with respect to the bias.

Proposition 3.5.6 *Given the solution* $\hat{\theta}$ *of a multiple-data identification* (3.51)*. Under the standard conditions of definition 3.5.5 the multiple data estimate is asymptotically unbiased.*

Proof: The bias is given by $\theta_0 - E\{\hat{\theta}\}$. For convenience of notation the number of data sets is taken $n_d = 2$. Under condition 1 it holds that

$$E\{\hat{\theta}\} = E\{(\phi_1^T\phi_1+\phi_2^T\phi_2)^{-1}(\phi_1^TY_1+\phi_2^TY_2)\} = E\{(\phi_1^T\phi_1+\phi_2^T\phi_2)^{-1}(\phi_1^T(\phi_1\theta_0+W_1)+\phi_2^T(\phi_2\theta_0+W_2))\}$$

where $W_i^T = [v_i^T(1)\cdots v_i^T(N)]$ is the stacked noise vector. Condition 2 implies that $E\{\phi_i^TW_i\} = 0$. Hence, $E\{\hat{\theta}\} = \theta_0$ which proves the result. □

The variance of the multiple-data estimate is also similar to the results for regular least squares estimation.

Proposition 3.5.7 *Given the solution $\hat{\theta}$ to* (3.51)*. Let the standard conditions of 3.5.5 hold Then the multiple data estimate has an asymptotic variance given by*

$$cov\theta = \left(\sum_{i=1}^{n_d} H_i\right)^{-1} \left(\sum_{i=1}^{n_d} \phi_i^T E\{W_i W_i^T\}\phi_i\right) \left(\sum_{i=1}^{n_d} H_i\right)^{-1}$$

where $W_i^T = [v_i^T(1) \cdots v_i^T(N)]$ is the stacked noise vector for output i.

Proof: Again for convenience of notation the number of data sets is $n_d = 2$. The covariance is defined as $cov\hat{\theta} = E\{(\hat{\theta} - E\hat{\theta})(\theta - E\hat{\theta})^T\}$. It holds that

$$\hat{\theta} - E\theta = (\phi_1^T\phi_1 + \phi_2^T\phi_2)^{-1}(\phi_1^T W_1 + \phi_2^T W_2)$$

This yields a covariance

$$cov\theta = (\phi_1^T\phi_1 + \phi_2^T\phi_2)^{-1}(\phi_1^T E\{W_1 W_1^T\}\phi_1 + \phi_2^T E\{W_2 W_2^T\}\phi_2)(\phi_1^T\phi_1 + \phi_2^T\phi_2)^{-1}$$

□

It is noteworthy that this result holds for finite data length and also for approximate system identification, i.e. $G_0 \notin \mathcal{G}$. This is a consequence of the deterministic regressor. Therefore, this result does not hold for an ARX model structure but exclusively for FIR and ORTFIR.

If the output disturbances $v_i(t)$ with $i = 1, \ldots, n_y$ are white noise sequences with variance σ_i^2 which are mutually uncorrelated, then the multiple data estimate has an asymptotic variance that reduces to

$$cov\theta = \left(\sum_{i=1}^{n_d} H_i\right)^{-1} \left(\sum_{i=1}^{n_d} \sigma_i H_i\right) \left(\sum_{i=1}^{n_d} H_i\right)^{-1}$$

This comes from the fact that due to the whiteness condition on the noise it holds that $E\{W_i W_i^T\} = \sigma_i^2 I$. If, moreover, the variance over all data sets is equal for all data sets, i.e. $\{\sigma_i\}_{1,\ldots,n_d} = \sigma$, the variance expression can be simplified to

$$cov\hat{\theta} = \sigma^2 \left(\sum_{i=1}^{n_d} H_i\right)^{-1} \tag{3.57}$$

In multiple data identification it is possible to combine data sets that are not sufficiently exciting by itself. Still the variance of the estimate is decreased if these data sets are used. This is stated in the following theorem.

Proposition 3.5.8 *Let θ_1 be a parameter estimate* (3.54) *based on n_1 data sets and let θ_2 be an estimate based on $n_2 > n_1$ data sets with the first n_1 equal to the ones of the previous estimate. Let the standard conditions hold in both cases and the variance on each measured output $\{\sigma_i\}_{i=1,\ldots,n_2} = \sigma$. Then it holds that*

$$cov\theta_2 \leq cov\theta_1$$

Proof: Let the sum of Hessians for θ_1 be given by $\bar{H}_1$ and that of θ_2 with $\bar{H}_2$. It holds that

$$\bar{H}_2 - \bar{H}_1 = \sum_{i=n_1+1}^{n_2} H_i \geq 0$$

as the Hessian of each experiment is positive semi-definite. Hence $\bar{H}_2 \geq \bar{H}_1$ so $\bar{H}_2 \geq \bar{H}_1$. With relation (3.57) it follows $cov\theta_2 \leq cov\theta_1$. Equality is only obtained if the added Hessians are all zero, so no additional experiments are performed. □

This proposition indicates that also data that is not sufficiently exciting by itself improves the variance of the estimate. This is intuitively clear as the added data certainly increases our knowledge of the system although it is not sufficient by itself to obtain a unique parameter estimate.
Next, to illustrate the consequences of the developed results, an example is given of how the parameter variance looks like for separate identification results and combined results.

Example 3.5.9 *In* [81] *the probability distribution of the parameter estimate is given for least squares estimates which can be readily extended to the multiple data case. This is given by the normal distribution*

$$\hat{\theta} - \theta^* \in \mathcal{N}(0, P)$$

with the covariance matrix $P = \sigma^2 \left(\sum_{i=1}^{n_d} H_i\right)^{-1}$. *This result can be translated to an ellipsoidal confidence interval on the parameter as follows. The probability that the estimated parameter is outside the ellipsoid*

$$(\hat{\theta} - \theta^*)^T P^{-1} (\hat{\theta} - \theta^*) < \alpha$$

is given by the α*-level of a* $\chi^2(n)$ *distribution where* n *is the number of parameters. The* largest *variance is in the direction of the eigenvector belonging to the* smallest *eigenvalue* $\underline{\lambda}(P^{-1})$ *and the* smallest *variance is in the direction of the largest eigenvalue* $\bar{\lambda}(P^{-1})$.
The ellipsoidal confidence interval is illustrated in figure 3.5 for the estimation of two parameters θ_1 *and* θ_2 *based on data with noise disturbance with variance* $\sigma^2 = 1$ *and Hessians* H_1 *and* H_2. *The confidence intervals are given for the case where either one of the Hessians is used and where both Hessians are used. The center of the estimates is translated to the origin.*
From the picture 3.5 it becomes clear that the variance ellipsoid for the multiple data estimate is contained in the variance ellipsoids for the separate data sets as could be expected from proposition 3.5.8. But what is striking is that the multiple data ellipsoid is considerably smaller than the intersection of the two other ellipsoids. Intuitively this can explained by noticing that every experiment, no matter how bad, contains information about the system that is not in other experiments. Therefore the information about the system increases with every added experiment and hence our uncertainty about the system is ever decreasing.

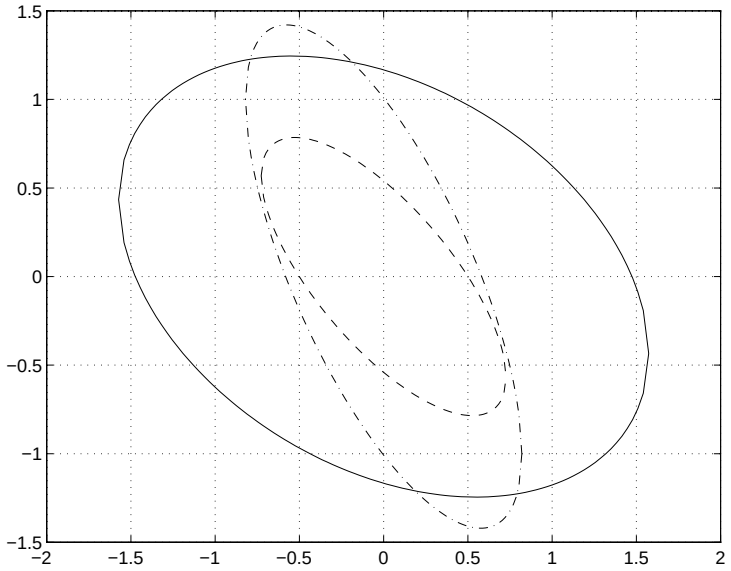

Fig. 3.5: *Uncertainty ellipse for the parameter estimate based on data set one (solid), data set two (dash dotted) and the combination of data set one and two (dashed).*

This example demonstrates that even "bad" data gives us information about the system that can be used to decrease our uncertainty about the system.
Next, it is demonstrated that combination of several data sets in the way discussed above coincides with the maximum likelihood estimate for the identification setup used.
First we discuss the maximum likelihood (ML) estimation. The likelihood function $L(\theta; y_1, y_2, \ldots)$ is a function of the observations $\{y(1), y(2), \ldots\}$ and the unknown parameter θ. The value of $L(\cdot)$ is the probability density function of the observations evaluated at the given observations Y and the parameter θ. Let the observations be independent. This is the case if the output noise $v(t)$ is white. Then the likelihood function is given by

$$L(\theta; y_1, y_2, \ldots) = p(y_1, \theta)p(y_2, \theta)\ldots$$

where $p(\cdot)$ is the probability density function of the random variable y. If the observations are normally distributed, this function becomes

$$p(y, \theta) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y-m}{2\sigma^2}}$$

where m is the mean value and σ is the standard deviation. The logarithm of the likelihood function can in this case be written as

$$-logL = \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - m)^2 + Nlog\sigma + \frac{N}{2}log2\pi$$

where m and σ are (possibly) functions of θ. If the standard deviation σ is a fixed number, i.e. no function of other variables, minimization of L is obviously equivalent to

minimization of the sum of squares. If the standard deviation is not fixed this still holds. This theory is applied to the estimation of a dynamical system which satisfies $G_0 \in \mathcal{G}$. Let the system be given by

$$\mathcal{S}_{lin} : y(t) = \varphi^T(t)\theta_0 + v(t)$$

with the output noise $v(t) \in \mathcal{N}(0, R)$. Then the loglikelihood function is given by

$$-logL(\theta) = \frac{1}{2}\sum_{t=1}^{N}(y(t) - \varphi^T(t)\theta)^T R^{-1}(y(t) - \varphi^T(t)\theta) + NlogdetR + \frac{Nn_y}{2}log2\pi.$$

Minimization of this function is equivalent to minimization of the (weighted) least squares (LS) problem

$$V(\theta) = \sum_{t=1}^{N}(y(t) - \varphi^T(t)\theta)^T R^{-1}(y(t) - \varphi^T(t)\theta).$$

Hence, under the given assumptions the ML estimate is equivalent to the LS problem with weighting $\sqrt{R}^{-1}$.
The ML estimation described above can also be applied to the case where n_d data sets are available. The estimate for this situation is equivalent to the solution of the LS problem with the following cost function

$$V(\theta) = \sum_{i=1}^{n_d}\sum_{t=1}^{N}(y_i(t) - \varphi_i^T(t)\theta)^T R^{-1}(y_i(t) - \varphi_i^T(t)\theta)$$

Here it is assumed that the noise has the same characteristic for each data set. The cost function above is equivalent to

$$V(\theta) = \sum_{i=1}^{N_d} V_i(\theta)$$

with $V_i(\theta)$ are the weighted LS cost functions of the separate data sets.

3.6 Simulation example

In this section a simulation example is given. The system that is considered is a nonminimum-phase system with two resonance modes given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 + 3.8630z^2 - 2.6426z + 0.8084}$$

The open-loop step response is given in figure 3.6.

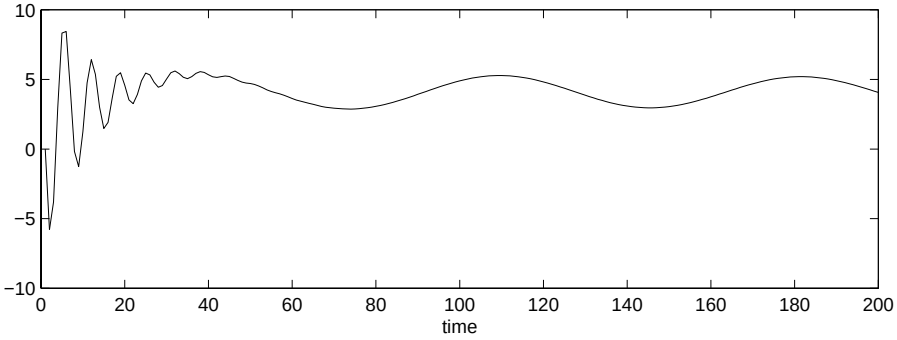

Fig. 3.6: *Step response of the plant.*

The input is constrained by $-0.275 \leq u(t) \leq 0.275$ which restricts the excitation signal design. The number of data is $N = 1000$. The output disturbance v is white and has variance $\sigma_v^2 = 0.26$. The input signal is a PRBS.
Low order prediction error models and models identified with a subspace technique (implemented in the *System Identification Toolbox* for Matlab) all show the fast resonance mode, as can be concluded from figure 3.7. However, the slow mode is less present in the data due to the finite length of the data set and the disturbance on the data. Also by increasing the model order it is difficult to capture the slow mode of the system.

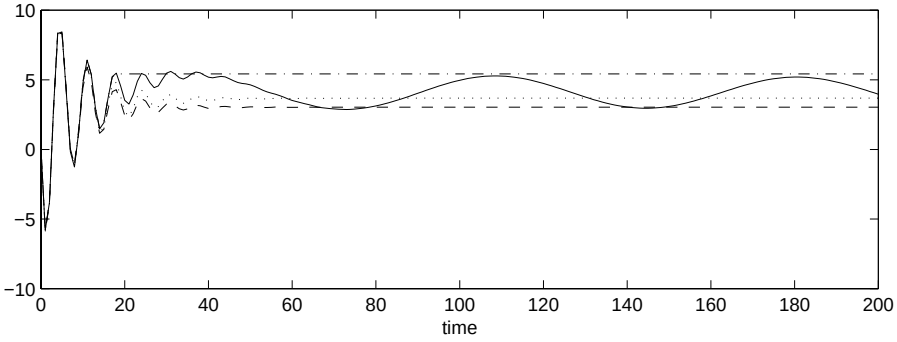

Fig. 3.7: *Step response system (solid), an ARX model with order $n_a = 4, n_b = 4, n_k = 1$ (dashed), an FIR model with $n_b = 18, n_k = 1$ (dash-dotted) and a subspace model with state space order 4 (dotted).*

Using system-based orthogonal basis functions the slow mode can be captured with a relatively small number of parameters. The pole location for the generation of the basis are taken to be

$$\lambda_1 = 0.5, \lambda_2 = 0.7, \lambda_3 = 0.8.$$

The construction mechanism is used where the poles are repeated. Already with a number of repetitions $n = 6$ and a state space order of $n_s = 18$ the slow mode is becoming visible in the model.

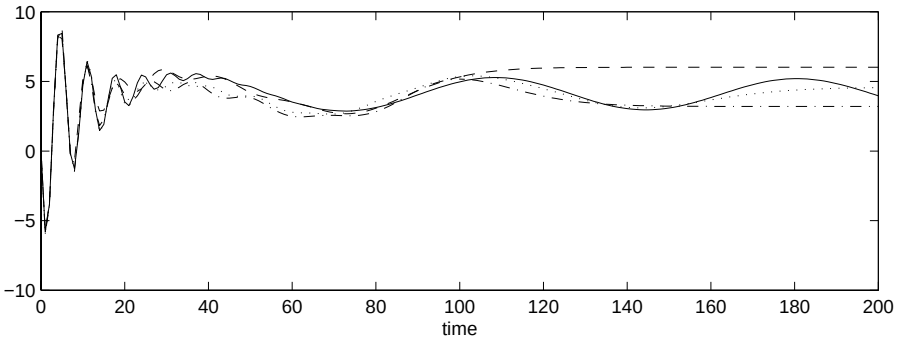

Fig. 3.8: *Step response system (solid) and ORTFIR models with state space order 18 (dashed), 21 (dash dotted) and 30 (dotted).*

Clearly, identification with the ORTFIR model structure provides a relatively close approximation of the system over a wider frequency range than with other model structures using relatively low orders. Therefore, especially in the identification of systems with a large dynamic range it is an efficient model parametrization. However, the model order is still relatively high compared to the system order. In the next chapter possible approaches are described to decrease this overfit in a systematic way.

3.7 Discussion

In this chapter some basic results are discussed of the theory of prediction error identification with model structures based on orthogonal basis functions, denoted with ORTFIR identification. To apply this model structure to industrial identification problems, a number of extensions are necessary to deal with the large scale and large dynamic range of the system as well as the short data sets with a low signal to noise ratio. In this chapter new results are given such that ORTFIR models can be readily applied to industrial identification problems.

Many results are available for ORTFIR identification of scalar systems, but only a small number for multivariable systems which is the most relevant for industrial identification. An important issue is the construction of multivariable model structures with orthogonal basis functions. Unlike in the scalar case, in the multivariable case it is not sufficient to specify the poles of the system to construct a model structure which contains a minimal realization of the system. In multivariable model structures additional information about the system structure, e.g. information the distribution of the poles over the entries of the transfer function matrix or Kronecker indices, is required to construct a minimal realization. If no prior knowledge about the system structure is available the non-minimal parametrization can be utillized but at the cost of a large number of parameters. By using structural prior knowledge about the system the number of parameters needed to fit the system can be reduced considerably resulting in an decrease of the variance of the

estimate.
A number of multivariable model structures is discussed and design rules are given with respect to the choice for a suitable structure given a certain type of prior knowledge about the system. Applying prior knowledge about dynamics and structure, such as used in the $\{A, B\}$ and $\{A, C\}$ variant, in the model structure can lower the number of free parameters in the model structure that is needed to obtain a certain accuracy. The use of inaccurate prior knowledge about the system structure can result in an increase of the number of free variables needed to obtain a certain accuracy. Therefore, the advised design rule is to use prior knowledge of the system structure only if this knowledge is known to be accurate and otherwise use a more flexible structure like e.g. one which uses scalar basis functions.
In industrial identification it is economically advantageous to minimize the experimentation time which results in short data sets. The signal to noise ratio can be low due to the presence of large disturbances. This implies that estimates can have a large variance. It is well known that estimation of a noise filter decreases the variance of the estimate. ORTFIR models have been described in literature mainly as output error model structures. An overview is given of the possibilities to estimate a disturbance model with orthogonal basis function model structures. Both linear model structures and bilinear model structures are presented. A bilinear model structure with orthogonal basis functions is further analyzed. The model structure, denoted with ORTXAR, describes the system dynamics and the inverse of the disturbance dynamics in terms of orthogonal basis functions. The estimated models for the system and disturbance dynamics are rational transfer functions which are parametrized independently. This is an advantage over e.g. the ARX model structure in which the disturbance and system dynamics are parametrized dependently.
The model structure is bilinear in the parameters and has a strong relation with the model structure used in generalized least squares (GLS) [143]. An alternative optimization strategy similar to the one proposed in [143] is analyzed. This optimization strategy is denoted with ORT-GLS. The ORT-GLS approach consists of consecutive steps of least squares identification and filtering, which can has the advantage over conventional nonlinear optimization algorithms that no information about the derivatives of the cost function are required. It is shown that the alternative optimization algorithm converges, under mild conditions, to the nearest local minimum.
ORTFIR identification can be extended with hard and soft constraints, estimation of initial conditions and can be applied to several data sets simultaneously without loosing convexity of the optimization. Constrained estimation is important in the identification of industrial processes as information about the static behaviour is frequently barely present in the short data sets. In industry a large amount of data is available in the form of data bases of normal operation data which is not sufficiently exciting enough, but by applying several data sets simultaneously the insufficiently excited data can still be utilized to de-

crease the variance on the estimate.
Hence, in this chapter we have extended the basic theory of ORTFIR identification such that it can be used to identify multivariable industrial systems on the basis of data sets that are short compared to the slowest time constant. The basis function description of the model enables the use of prior knowledge about system dynamics and structural information about the system, allowing an accurate model to be obtained with relatively low complexity provided sufficient prior knowledge is available.

Chapter 4

Model reduction and basis improvement

4.1 Introduction

Identification of a model with orthogonal basis functions usually results in an estimated model of relatively high order because the basis functions, which are chosen on the basis of a priori knowledge of the system dynamics, may not resemble the system dynamics. In many situations it is desirable not to work with a high order model but reduce the order of the model.
One reason to reduce the model order is to improve the quality of the overall model. A model with high order can have a large variance which is expressed in e.g. a lower quality of the output prediction for a data set other than the one used for identification. By sensible model reduction this variance influence can be decreased. Another reason for model reduction is that for some applications a low model order is desirable or required. This is the case if the model is used in a model based control design technique in which the model order is reflected in the controller order, e.g. robust and optimal control techniques [175]. For these techniques a lower order model generally yields a lower order controller which is easier to implement.
Several model reduction techniques are available in literature. The question is which reduction technique performs well in the specific case of the reduction of a high order model identified with a model structure using orthonormal basis functions. This question is addressed in section 4.2.
An important benefit of identification with a model structure based on orthogonal basis functions is that prior knowledge about the system dynamics can be utilized. A model is estimated with basis functions that applies the currently available prior knowledge. The identified model contains new information about the system. This new information can be used to update the prior knowledge about the system. This updated prior knowledge can be utilized to construct a new and improved set of basis functions. The improvement of the basis is reflected in the fact that fewer expansion coefficients need to be estimated to obtain

a model with the same bias error. The smaller number of estimated expansion coefficients results in a smaller variance error and hence a better model is obtained with the improved basis functions. These steps can be proceeded until no further model improvement is obtained. This results in an iterative process in which in consecutive steps the level of prior knowledge is increased while the optimization complexity is decreased resulting in an estimated model that is more and more accurate.
In [56] this basic idea is implemented in the following fashion where model reduction plays an important role to extract the updated prior knowledge about the system dynamics from the high order estimate. Let a high order model be identified in a basis that is given by a set of poles. If the high order model is reduced, dependent on the applied model reduction technique, the reduced order model has different dynamics from the high order model. This new information about the pole location can be used to construct a new basis which is used in the following identification step. This iterative scheme is further discussed in section 4.3.

4.2 Model reduction for high order ORTFIR estimates

4.2.1 Model reduction techniques

Many model reduction techniques are available from literature [167]. An important class of model reduction methods utilizes a balanced realization as defined in definition 2.2.1. Let a partitioned balanced state space description of the high order discrete time model G be given by

$$A = \begin{bmatrix} A_1 & A_{12} \\ A_{21} & A_2 \end{bmatrix}, B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, C = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \tag{4.1}$$

with Gramians $P = Q = \Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_n)$ where the Hankel singular values are ordered following $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$. Various reduction approaches can be defined that have an explicit solution. The discrete truncated balanced reduced order model of order k is given by

$$G_r(z) = C_1(zI - A_1)^{-1}B_1.$$

It can be shown that the model error is bounded by $\|G_{red}(z) - G(z)\|_\infty \leq 2(\sigma_{k+1} + \ldots + \sigma_n)$. Another method is obtained if the discrete time model is first transformed to the continuous time domain. Next continuous time model reduction is performed on this model and finally the reduced order model is transformed back to the discrete time. A closed form expression for the reduced order model $G_r(z) = D_r + C_r(zI - A_r)^{-1}B_r$ can be found in [57] and is given by

$$\begin{aligned} A_r &= A_1 - A_{12}(I + A_2)^{-1}A_{21}, B_r = B_1 - A_{12}(I + A_2)^{-1}B_2 \\ C_r &= C_1 - C_2(I + A_2)^{-1}A_{21}, D_r = -C_2(I + A_2)^{-1}B_2 \end{aligned}$$

Both methods provide an approximation that is especially accurate in the high frequency range.

Instead of just partitioning the high order transfer function, the singular perturbation principle can be used. In this approach the states corresponding to the lower part of the partitioned description (4.1) is assumed to be infinitely fast which results in a set of k difference equations and $n-k$ algebraic equations. This results in the singular perturbational balanced reduction method with a closed form reduced order model given by

$$\begin{aligned} A_r &= A_1 + A_{12}(I+A_2)^{-1}A_{21}, B_r = B_1 + A_{12}(I+A_2)^{-1}B_2 \\ C_r &= C_1 + C_2(I+A_2)^{-1}A_{21}, D_r = C_2(I+A_2)^{-1}B_2 \end{aligned}$$

This method has the property that especially the low frequency dynamics are approximated better.

In [39] an extension of balanced reduction is proposed that enables specific frequency ranges to be emphasized by means of dynamic filters. This extension is denoted with frequency weighted balanced reduction. Let in addition to the high order model $G(z)$, the stable filters $W_i(z) = C_i(zI - A_i)^{-1}B_i$ and $W_o(z) = C_o(zI - A_o)^{-1}B_o$ be given. Now, the objective is to find a lower order model G_r such that

$$\|W_o(G - G_r)W_i\|_\infty$$

is made as small as possible. This can be done by applying balanced reduction to that part of the series connection W_oGW_i that corresponds to the dynamics of the high order model G. Let the series connection be given in state space form $W_oGW_i = \bar{C}(zI - \bar{A})\bar{B}$ with

$$\begin{aligned} \bar{A} &= \begin{bmatrix} A & 0 & BC_i \\ B_oC & A_o & 0 \\ 0 & 0 & A_i \end{bmatrix}, \ \bar{B} = \begin{bmatrix} BD_i \\ 0 \\ B_i \end{bmatrix} \\ \bar{C} &= \begin{bmatrix} D_oC & C_o & 0 \end{bmatrix} \end{aligned}$$

and Gramians $\bar{P}$ and $\bar{Q}$. The part of the Gramians that belongs to the dynamics of the high order model is given by $P = [I_n\ 0]\bar{P}[I_n\ 0]^T, Q = [I_n\ 0]\bar{Q}[I_n\ 0]^T$. Now, balancing and reduction can be performed on the model using these Gramians instead. This results in a reduced order model with an improved fit in those frequency ranges where the weighting filters W_i, W_o are large. An important benefit is that the reduced order model can be calculated easily.

4.2.2 Model reduction of estimated high order models

In the previous section some standard model reduction techniques are described. However, we are searching for a reduction technique that is specifically suitable for the reduction

of models that are a result of an identification procedure. Identified high order model can have a large error due the large variance influence of the estimated parameter vector. Usually information about the way the variance of the model is distributed over the frequencies is available or can be calculated using the data. It seems sensible to use a reduction method in which those frequency ranges are deemed more important in which the high order model is accurate. Such a model reduction approach is proposed in [159].

Let a high order model $\hat{G}(z)$ be given that is estimated with a prediction error identification method. If the model structure is sufficiently flexible such that the system is in the model set, the identified model is consistent regardless of the applied model structure. The asymptotic properties of a transfer function estimate is derived in [81] and is given by the normal distribution.

$$\hat{G}(e^{i\omega}) \sim As\mathcal{N}(0, \frac{n}{N}\frac{\Phi_v(e^{i\omega})}{\Phi_u(e^{i\omega})}). \tag{4.2}$$

where n is the number of estimated parameters and N is the number of data points. This relation provides information about the frequency ranges in which the model is accurate. In [159] it is proposed to find a low order model $G(z,\theta)$ by minimizing the cost function

$$L(\theta) = \int_{-\pi}^{\pi} |\hat{G}(e^{i\omega}) - G(e^{i\omega},\theta)|^2 Q(e^{i\omega}) d\omega \tag{4.3}$$

with the weighting function

$$Q(e^{i\omega}) = \frac{\Phi_u(e^{i\omega})}{\Phi_v(e^{i\omega})}.$$

From an intuitive point of view this seems like a sensible choice because the weighting is proportional to the inverse of the covariance of the high order estimate, which provides a high weighting in frequency ranges where the high order model is accurate and a low weighting in frequency ranges where the high order model is inaccurate. From a mathematical point of view the weighting can be motivated using the fact that the minimum of (4.3) is equivalent to the maximum likelihood (and prediction error) estimate of the model based on observations of the stochastic process (4.2). A proof of this can be found in [159].

A problem with this approach is that in weighting function $Q(e^{i\omega})$ the noise spectrum $\phi_v(e^{i\omega})$ is unknown. However, the exact spectrum can replaced by an estimate

$$\hat{\phi}_v(e^{i\omega}) = \hat{\sigma}^2 \hat{H}(e^{i\omega})$$

where $\hat{H}$ is the estimated disturbance spectrum which can be obtained e.g. with the technique described in section 3.4.4 and

$$\hat{\sigma}^2 = \frac{N}{N - n_\theta} V(\hat{\theta}_i)$$

is a consistent estimate of the noise variance where N is the number of data and n_θ is the number of parameters used in the estimation of the model.
A common choice of input signal is filtered white noise $u(t) = F(q)e(t)$ where $e(t)$ is a white noise noise signal with variance σ_e^2 and $F(q)$ is some stable filter. Then we can take

$$\hat{\phi}_u(e^{i\omega}) = \phi_u(e^{i\omega}) = \sigma_e^2 |F(e^{i\omega})|^2.$$

If the input spectrum is unknown it can be estimated. In this way an approximation of the weighting filter $Q(e^{i\omega})$ can be found that is given by $\hat{Q} = \hat{\phi}_u \hat{\phi}_v^{-1}$ which can be calculated from the data.
The reduction problem stated in (4.3) amounts to an L_2-norm model reduction with a weighting function that is the inverse of the variance of the high order estimate. This reduction problem is a nonlinear optimization problem that may have several local minima in which a nonlinear optimization procedure might get stuck. To alleviate this problem it is proposed in [159] to use frequency weighted balanced reduction, as discussed in section 4.2.1, with the weighting filter $\hat{Q}$. This weighting filter needs to be available in terms of a stable finite dimensional transfer function which can be obtained as discussed above.

4.2.3 An asymptotic approach to ORTFIR model reduction

In this section the approach to model reduction of [159] is analyzed in case the high order model is obtained using an orthogonal basis function model structure. First, the scalar case is discussed and next the extension to the multivariable case is given.
To implement the model reduction strategy of [159], it is necessary to have the inverse covariance of the high order transfer function estimate available in terms of a finite dimensional transfer function that can be used as weighting function $Q(z)$. The covariance of an ORTFIR estimate is given in (3.2.5). This implies that weighting function $Q(z)$ must satisfy

$$|Q(e^{i\omega})|^2 = (F_1^*(e^{i\omega})F_1(e^{i\omega}))^{-1} \frac{\Phi_u(\omega)}{\Phi_v(\omega)}.$$

where it is assumed for convenience of notation that the basis dynamics is repeated batch-wise. To calculate the transfer function of the weighting filter $Q(z)$, finite dimensional spectral factors need to be calculated of $\Phi_u(\omega)$, $\Phi_v^{-1}(\omega)$ and $(F_1^*(e^{i\omega})F_1(e^{i\omega}))^{-1}$.
The input spectrum $\Phi_u(\omega)$ is constant if a white noise signal is applied. For simplicity of notation this is assumed in the sequel. However, if a colored noise signal is applied as input, the required transfer function is equal to the transfer function of the coloring filter.
A consistent estimate of a spectral factor of the noise spectrum $\Phi_v^{-1}(\omega)$ can be obtained by the technique described in section 3.4.4. The estimate of this term is given by [81]

$$\hat{\Phi}_v^{-1}(\omega) = \frac{2\pi}{\hat{\sigma}^2} |\hat{H}(e^{i\omega})|^{-2}$$

where $\hat{H}(e^{i\omega})$ is the estimate of the disturbance model and

$$\hat{\sigma}^2 = \frac{N}{N - n_\theta} V(\hat{\theta}_i)$$

is a consistent estimate of the noise variance where N is the number of data and n_θ is the number of parameters used in the estimation of the model. With these estimates a spectral factor for the term $\Phi_v^{-1}(\omega)$ is obtained.
Obtaining a finite dimensional spectral factor of the term $(F_1^*(e^{i\omega})F_1(e^{i\omega}))^{-1}$ is slightly more involved. Searched for is a finite dimensional transfer function $R(e^{i\omega})$ that satisfies

$$|R(e^{i\omega})|^2 = (F_1^*(e^{i\omega})F_1(e^{i\omega}))^{-1} \tag{4.4}$$

Define the numerator and denominator of the terms as follows

$$F_1^*(z^{-1})F_1(z) = \frac{N(z)}{D(z)} \tag{4.5}$$

The denominator of this terms is easily obtained as $D(z) = a(z^{-1})a(z)$ where $a(z) = det(zI - A)$ is the characteristic polynomial of the state space matrix A used for the basis generation. This can be easily seen by rewriting

$$\begin{aligned} F_1^T(z^{-1})F_1(z) &= B^T(z^{-1}I - A^T)^{-1}(zI - A)^{-1}B \\ &= \frac{1}{det(z^{-1}I - A^T)} B^T adj(z^{-1}I - A^T) adj(zI - A) B \frac{1}{det(zI - A)} \end{aligned}$$

Let the numerator polynomial $N(z)$ in (4.5) be given by

$$N(z) = \sum_{|k| \leq n} d_k z^k$$

Let the zeros of this polynomial be given by $\{z_i\}_{i=1,\ldots,2n}$. This is a set of n stable and n anitstable zeros. Let the monic stable part be given by the polynomial $b(z)$ which has the stable zeros of $N(z)$ as its zeros. Then the spectral factor we are searching for can be written as

$$R(z) = \frac{a(z)}{Kb(z)}$$

with a real-valued constant $K = \frac{(-1)^n d_n}{\prod_{i=1}^n b_i}$. This procedure to obtain a spectral factorization can also be found in [144].
Now we are ready to specify the weighting filter that can be used in frequency weighted balanced reduction. This weighting filter is given by

$$W(z) = \hat{\sigma}^{-1} R(z)(\hat{H}^{-1}(z))$$

This is the weighting filter for the scalar case. For the multivariable case the weighting can be determined as follows. Assume that the disturbances on the n_y outputs are mutually

independent. Define the inverse of the estimated disturbance filter for each output as $\hat{H}_i^{-1}$ for $i = 1, \ldots, n_y$. Let the state space matrix of the basis generating function be partitioned as $B = [B_1 \cdots B_{n_u}]$. Then the input and output weighting transfer function matrices that must be utilized in frequency weighted balanced reduction are given by

$$W_i = \begin{bmatrix} F_1 & & \\ & \ddots & \\ & & F_{n_u} \end{bmatrix}, W_o = \begin{bmatrix} \hat{\sigma}_1^{-1}\hat{H}_1^{-1} & & \\ & \ddots & \\ & & \hat{\sigma}_{n_y}^{-1}\hat{H}_{n_y}^{-1} \end{bmatrix}$$

where $\{F_i\}_{i=1,\ldots,n_u}$ are spectral factors of

$$B_i^T(z^{-1}I - A^T)^{-1}(zI - A)^{-1}B_i\Phi_{u_i}$$

that can be obtained along the same lines as discussed above. If the filters W_i, W_o are applied in frequency weighted balanced reduction, they provide a weighting that is large for frequency ranges with small variance, i.e. certain parts, and small for frequency ranges with high variance, i.e. uncertain parts. In this way the accurate part of the estimated model is preserved in the reduced model.
The calculation of the weighting filter described in this section is based on the asymptotic variance expressions. There exists also a nonasymptotic variance expression for ORTFIR models that can be used to calculate the weighting filter. This is described in the next section.

4.2.4 A nonasymptotic approach

In the previous section the asymptotic variance expression is utilized in the construction of an appropriate weighting filter for frequency weighted balanced reduction. For linear regression model structures with a deterministic regressor, i.e. FIR and ORTFIR, a finite time expression for the parameter variance is available. This expression is given in (3.27) and holds for finite data length N and finite model order n. The asymptotic variance expression can differ considerably from the nonasymptotic one. This is especially true for badly conditioned least squares problems. In this case it is more sensible use the nonasymptotic variance expression in frequency weighted model reduction. For the multivariable case such a procedure is proposed below.
Let the applied model structure be given by

$$G(e^{i\omega}, \theta) = \Theta^T(e^{i\omega}I - A_e)^{-1}B_e$$

with $\Theta = [\theta_1 \cdots \theta_{n_y}] \in \mathbb{R}^{nn_b \times n_y}$ and $B = [B_{e\,1} \cdots B_{e\,n_u}] \in \mathbb{R}^{nn_b \times n_u}$. Each transfer function element is therefore given by

$$\hat{G}_{ij}(e^{i\omega}) = \theta_i^T(e^{i\omega}I - A_e)^{-1}B_{e\,j} \tag{4.6}$$

which is a linear mapping in θ. The finite data parameter variance (3.27) is given by

$$cov\{\hat{\theta}_i\} = P = \Phi^{\dagger} E\{VV^T\}(\Phi^{\dagger})^T$$

with $V = [H(q)e(1) \dots H(q)e(N)]$. For white noise disturbance this reduces to $P = \sigma^2(\Phi^T\Phi)^{-1}$. If an estimate of the disturbance dynamics is available the covariance matrix can be calculated. The covariance matrix determines an ellipsoidal confidence region for the parameter that can be mapped into a confidence interval for the frequency response using the linear mapping (4.6). This yields the covariance matrix of the transfer function estimate

$$cov\hat{G}_{ij}(e^{i\omega}) = B_{e\,j}^T(e^{-i\omega}I - A_e^T)^{-1} cov\hat{\theta}_i (e^{i\omega}I - A_e)^{-1} B_{e\,j}$$

This is a transfer function of which the numerator and denominator can be calculated and factorized in stable polynomials to obtain the desired weighting filter. However, the numerator denominator descriptions are known to be numerically badly conditioned and yield inaccurate transfer functions. Therefore an approach is followed that utilizes only state space descriptions of the system. For this purpose the covariance matrix is rewritten to

$$cov\hat{G}_{ij}(e^{i\omega}) = -e^{i\omega} B_{e\,j}^T(e^{i\omega}I - A_e^{-T})^{-1} A_e^{-T} cov\hat{\theta}_i (e^{i\omega}I - A_e)^{-1} B_{e\,j}$$

where it is assumed that A_e is invertible. Now, the description of the covariance of the transfer function is dependent only on z and not also on $1/z$. A stable spectral factor $F_{ij}(e^{i\omega})$ can be determined by selection of the stable poles and zeros purely on the basis of state-space descriptions. Especially in badly conditioned identification problems manipulation with (balanced) state space descriptions can improve the accuracy of the solution.

Due to the structure of the model, the covariance of each column of the estimated transfer function matrix is the same. This implies that the weighting can be applied for each input channel separately. Let it again be assumed that the disturbances on the consecutive outputs are mutually independent. Then the weighting in the model reduction can be divided in an input weighting W_i and an output weighting W_o that are given by

$$W_i = \begin{bmatrix} F_1 & & \\ & \ddots & \\ & & F_{n_u} \end{bmatrix}, W_o = \begin{bmatrix} \hat{\sigma}_1^{-1}\hat{H}_1^{-1} & & \\ & \ddots & \\ & & \hat{\sigma}_{n_y}^{-1}\hat{H}_{n_y}^{-1} \end{bmatrix}$$

where $\hat{H}_i^{-1}$ for $i = 1, \dots, n_y$ is the inverse of the estimated disturbance filter for each output i and

$$\hat{\sigma}_i^2 = \frac{N}{N - nn_b} V(\hat{\theta}_i).$$

is an estimate of the noise variance in output channel i.

The nonasymptotic approach is computationally more demanding than the asymptotic

approach because the weighting filter is of order nn_b as opposed to n_b, for batch wise repeated basis dynamics. However, especially for badly conditioned least squares estimates the nonasymptotic approach performs better because it utilizes information about the conditioning of the regression matrix in terms of $cov\hat{\theta}$.

4.3 Iterative basis improvement

4.3.1 The basic principle

A difficult part in identification with orthogonal basis functions is the choice of the dynamics that is utilized in the construction of the basis. In section 3.5.3 it is mentioned that a possible way to find sensible dynamics for the basis functions is by identification of a model $\hat{G}_{init}$ of order n with a different model structure, e.g. ARX to benefit from the linearity in the parameters. The dynamics of this model possibly supplemented with structural information can be utilized to build a basis with the approach described in section (3.2.1) for scalar basis functions and (3.3.1) for multivariable basis functions. If the n system poles are correctly estimated, only n parameters need to be calculated in the scalar ORTFIR model structure to find an estimate of the system zeros. But generally the pole location is incorrect. In that case the number of expansion coefficients can be extended such that the model accuracy is improved. In extending the number of expansion coefficients that are estimated, the bias of the estimate is decreased while the variance is increased. A model order, i.e. number of expansion coefficients, can be chosen such that the bias variance trade-off is such that a gain in model accuracy is obtained.
Now, we have obtained a model $\hat{G}_{high}$ of possibly high order. This model is more accurate than the initial estimate and therefore contains more information about the system that has generated the data. An even more accurate model $\hat{G}_{low}$ can be obtained by reducing the variance influence in the model by model reduction, e.g. with reduction approach discussed in the previous section. Let the reduction order be given by n_r. Dependent on the reduction approach, the reduced order model has different dynamics from the initially estimated model $\hat{G}_{init}$.
If the model is not sufficiently accurate for its intended use, the low order model $\hat{G}_{low}$ can be used, as the initial estimate $\hat{G}_{init}$ was used previously, to construct a new set of basis functions with the updated information about the system dynamics (and system structure). This points to the iterative process of basis improvement discussed in the next section.

4.3.2 An alternative nonlinear optimization strategy

The objective that is pursued here is to find a (multivariable) model $\hat{G}$ of the system dynamics G_0 and no disturbance model is required i.e. the identification of an output error

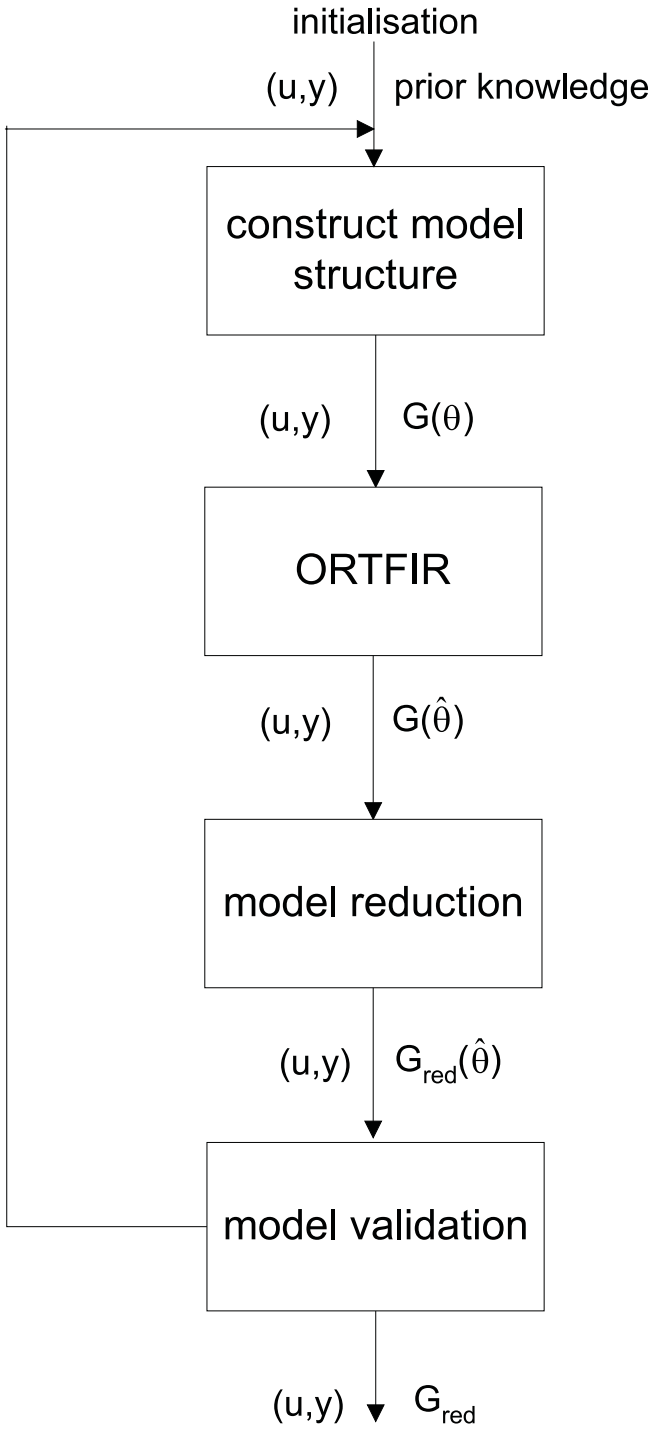

Fig. 4.1: *Iteration of high order system identification and model reduction.*

model is regarded. This boils down to finding a solution to the nonconvex optimization problem with cost function (2.35) given by

$$V(\theta) = \frac{1}{N}\sum_{t=1}^{N} \varepsilon(t,\theta)^T \varepsilon(t,\theta) \text{ with } \varepsilon(t,\theta) = y(t) - G(q,\theta)u(t)$$

where $G(z,\theta) = C(\theta)(zI - A(\theta))^{-1}B(\theta)$ is a parametrized transfer function of order n.
A nonlinear optimization algorithm as discussed in 2.5.6 can be used to find a solution to this optimization problem. Usually a local optimum is found and it is difficult to find starting point for the iterative search such that an improved solution is found. Instead, the iteration sketched in the previous section can be utilized which contains the following steps.
First a model is identified with a model structure based on orthogonal basis functions. This is followed by model reduction that changes the dynamics of the model. The updated information about the dynamics can be used to obtain a new set of basis functions that are applied to construct a new model structure, which is used in a new identification step. This iteration, graphically depicted in figure 4.1, can be continued until convergence (if the iteration converges) or until no further improvement is obtained. In this section this strategy, denoted with ORTIT , is investigated and further developed.

Algorithm 4.3.1 ORTIT algorithm. *To find a solution to the nonlinear optimization problem that originates in a prediction error identification with an output error structure, regard the following iteration:*

Step 0. *Acquire prior knowledge about the system dynamics in terms of a set of poles* $\{\lambda_i\}$*, state space matrices* $\{A, B\}$ *or* $\{A, C\}$ *and information about the system structure if this is available.*

Step 1. *Construct a system-based basis* $\{f_i\}_{i=1,2,...,}$ *using the available information about the system dynamics and system structure, using the approach discussed in section 3.3.1.*

Step 2. *Identify a high order model using the model structure which is a finite expansion in the chosen basis.*

Step 3. *Reduce the order of the identified model to obtain a low order model* G_{red} *with poles that are different from the poles of the high order model.*

Step 4. *Validate the model* G_{red}*. If the accuracy is sufficient, terminate the iteration else go to Step 1 with the updated information about the system dynamics and structure.*

In this iteration the computational load is concentrated in the step 2, identification, and step 3, model reduction. The identification step is a linear regression problem that can be

solved efficiently for large scale problems. The computational load of the model reduction step can be limited if a model reduction technique is utilized that requires no (nonconvex) optimization. Such a technique is e.g. balanced reduction or frequency weighted balanced reduction. Hence, the iteration ORTIT can be performed with limited computational load.
The properties of the iteration can be influenced by a specific choice for the degrees of freedom in the algorithm. These degrees of freedom or tuning parameters are

- high model order
- low model order
- model reduction tool

An appropriate choice for the high model order is obtained through the inspection of the expansion coefficients. The model order can be increased until the expansion coefficients become small. In that case the decrease of the bias of the estimate due the contribution of an additional coefficient becomes negligible. A more sophisticated approach is the use of uncertainty identification techniques [109][54] to find the model order such that the bias and variance trade-off is optimal.
The low model order can be chosen on the basis of indicators that are specific for the applied reduction technique. If (frequency weighted) balanced reduction is utilized a sensible reduction order can be obtain by inspection of the Hankel singular values. The tail of smaller singular values can be discarded.
The type of model reduction utilized in the iteration influences the final estimation result considerably. Model reduction plays the role in the iteration of compressor of information stored in the high order model. It is crucial that the dynamics of the reduced order model is different from the dynamics of the high order model. This rules out e.g. modal reduction or minimal realization, because these techniques yield pole locations of the reduced order model that are a subset of the pole locations of the high order model. A possible reduction strategy is frequency weighted balanced reduction with the weighting chosen as proposed in the previous section.

4.3.3 An alternative nonlinear optimization strategy: Box-Jenkins case

The iterative approach discussed in the previous section is presented in the output error case, i.e. a fixed noise model is utilized. The objective that is pursued in this section is to find a (multivariable) model $\hat{G}$ of the system dynamics G_0 and a model $\hat{H}$ of the disturbance dynamics H_0, i.e. identification of a Box-Jenkins model is regarded. This boils down to finding a solution to the nonconvex optimization problem with cost function

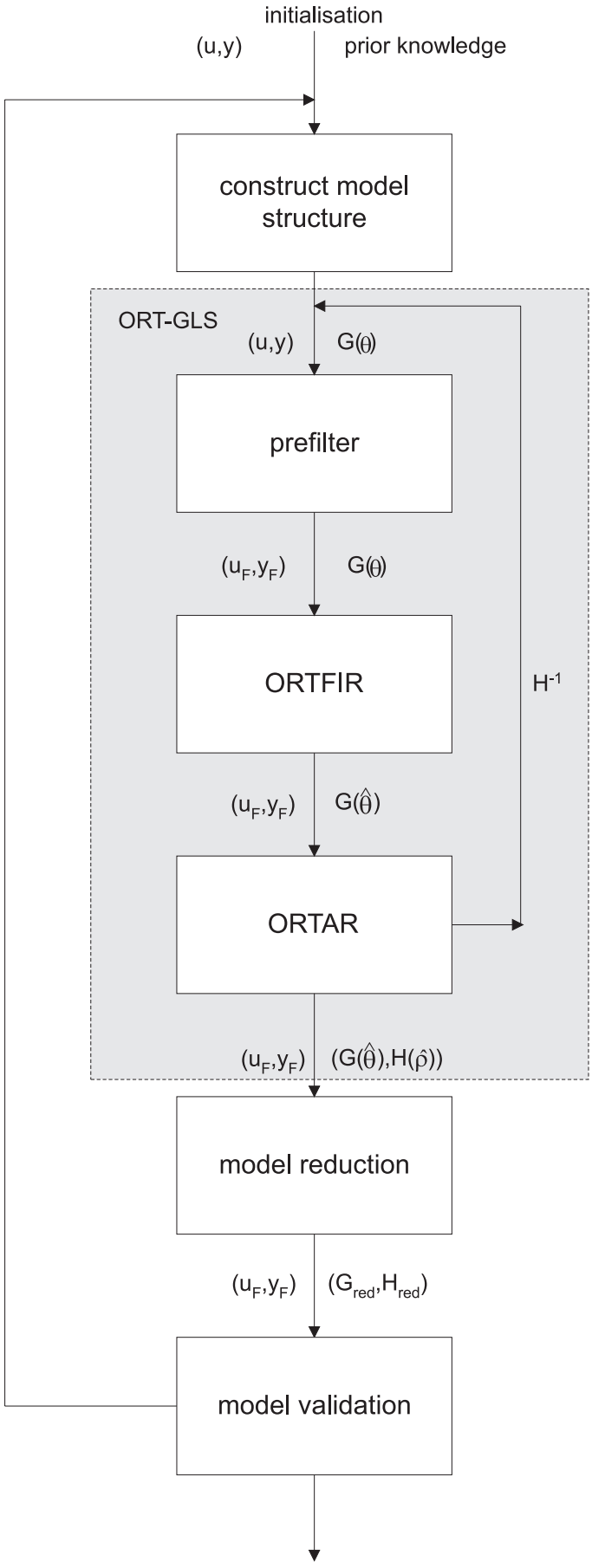

Fig. 4.2: *Iteration of high order system identification and model reduction.*

(2.35) given by

$$V(\theta,\rho)=\frac{1}{N}\sum_{t=1}^{N}\varepsilon(t,\theta,\rho)^T\varepsilon(t,\theta,\rho) \text{ with } \varepsilon(t,\theta,\rho)=H^{-1}(q,\rho)(y(t)-G(q,\theta)u(t))$$

where $G(z,\theta)=C(\theta)(zI-A(\theta))^{-1}B(\theta)$ is a parametrized transfer function of order n and $H(z,\rho)=C_H(\rho)(zI-A_H(\rho))^{-1}B_H(\rho)$ is a parametrized transfer function of order n_H. It is assumed that G,H are parametrized independently.
The ORT-GLS procedure discussed in section 3.4 can be utilized to provide an estimate of the disturbance dynamics. If ORT-GLS is utilized this provides an additional iteration in an inner loop as depicted in figure 4.2. In the iteration the following steps are taken. First, a high order model $G(\hat{\theta})$ is estimated for the system dynamics with an ORTFIR model structure. Next, a high order model $H(\hat{\rho})$ is estimated for the disturbance dynamics with an ORTAR model structure based on the residual $\varepsilon(t,\hat{\theta})=y(t)-G(\hat{\theta})$. From this point, two possible strategies can be followed. In the first strategy the high order estimates $\hat{G},\hat{H}$ are reduced. For this purpose frequency weighted balanced reduction can be applied with the weighting function proposed in section 4.2. A similar strategy can be used for the reduction of the disturbance model where the variance information that is required for the computation of the weighting is given in theorem 3.4.4. Instead of one high order estimation of the system dynamics followed by one high order estimation of the disturbance dynamics, an iteration can be performed over these steps using the ORT-GLS algorithm. With this strategy the variance influence of the estimated model can be decreased before the models are reduced. It increases the complexity of the total algorithm but can contribute to an increased accuracy of the overall estimation result. In figure 4.2 this iteration is indicated with the inner loop. Note that figure 4.2 holds for the scalar case because the prefiltering of the prediction error is replaced by prefiltering of the input and output signals. For the multivariable case the prefiltering can only be applied to the prediction error signal.

4.3.4 A motivating example

Consider the problem of finding the pole of a first order system

$$G_0(z)=\frac{1}{z-a_0}$$

from data $\{u(t),y(t)\}_{t=1,\ldots,N}$ using a prediction error criterion. The output $y(t)$ is disturbed with white noise with variance σ_e^2. The identification of this system with an output error model structure amounts to the scalar optimization problem $\min_a V(a)$ with cost function

$$V(a)=\sum_{t=1}^{N}(y(t)-G(q,a)u(t))^2$$

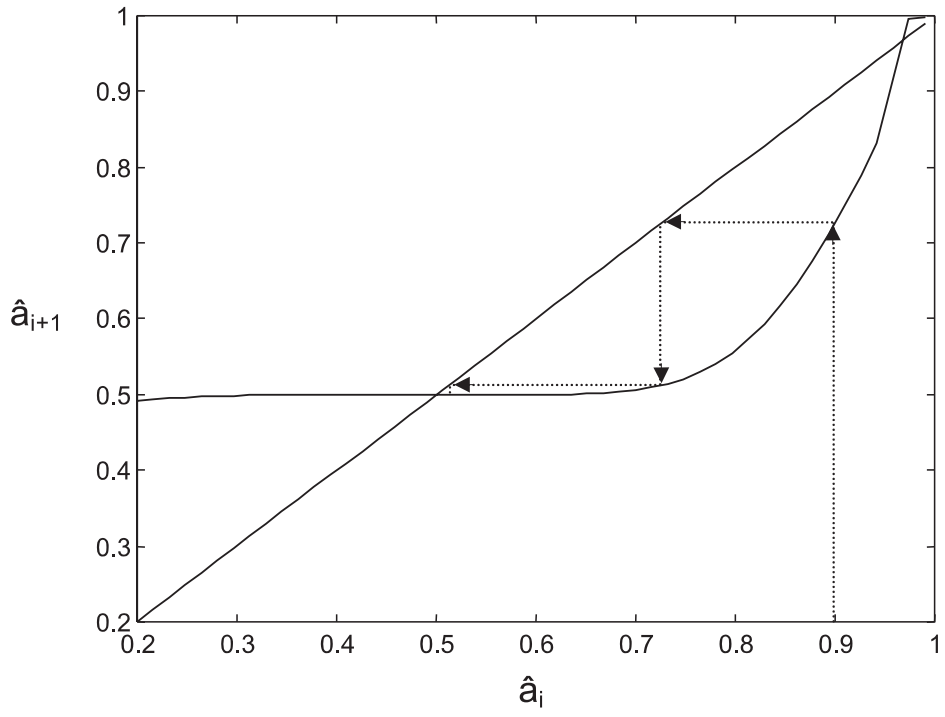

Fig. 4.3: *Mapping* $\hat{a}_{i+1} = T(\hat{a}_i)$ *together with* $\hat{a}_{i+1} = \hat{a}_i$ *and an example iteration starting at* $\hat{a}_0 = 0.9$.

where $G(z, a) = (z - a)^{-1}$ which is clearly nonlinear optimization problem due to the nonlinear mapping $G(z, a)$ and may therefore be nonconvex.
In the basis construction the chosen pole is repeated, i.e. the approach presented in [151], and further the initial pole location $a_0 = 0.5$, the number of data points $N = 100$, the number of expansion coefficients $n = 3$, the disturbance variance $\sigma_e^2 = 0.25$ and the reduction tool is balanced reduction [92].
The iteration discussed previously can be analyzed for this simple situation using the mapping from one estimate of the pole to the other, i.e.

$$\hat{a}_{i+1} = T(\hat{a}_i)$$

where the mapping T contains the following steps. First, a Laguerre model with n basis functions (3.2) with pole location $\hat{a}_i$ is identified. Next, this model is reduced to order 1 with balanced reduction. Finally, the pole of the first order system that is the result of the reduction is denoted with $\hat{a}_{i+1}$. The effect of these steps is denoted with the mapping T.
The mapping T is calculated for the example at hand and is depicted in the figure 4.3.
If an iteration is started at e.g. $\hat{a}_0 = 0.9$ the iteration converges to the true value of the pole, i.e. 0.5 which is the point of intersection of the mapping T and the line $\hat{a}_{i+1} = \hat{a}_i$. This iteration is graphically depicted in the figure. Hence, in 2 iteration steps the correct

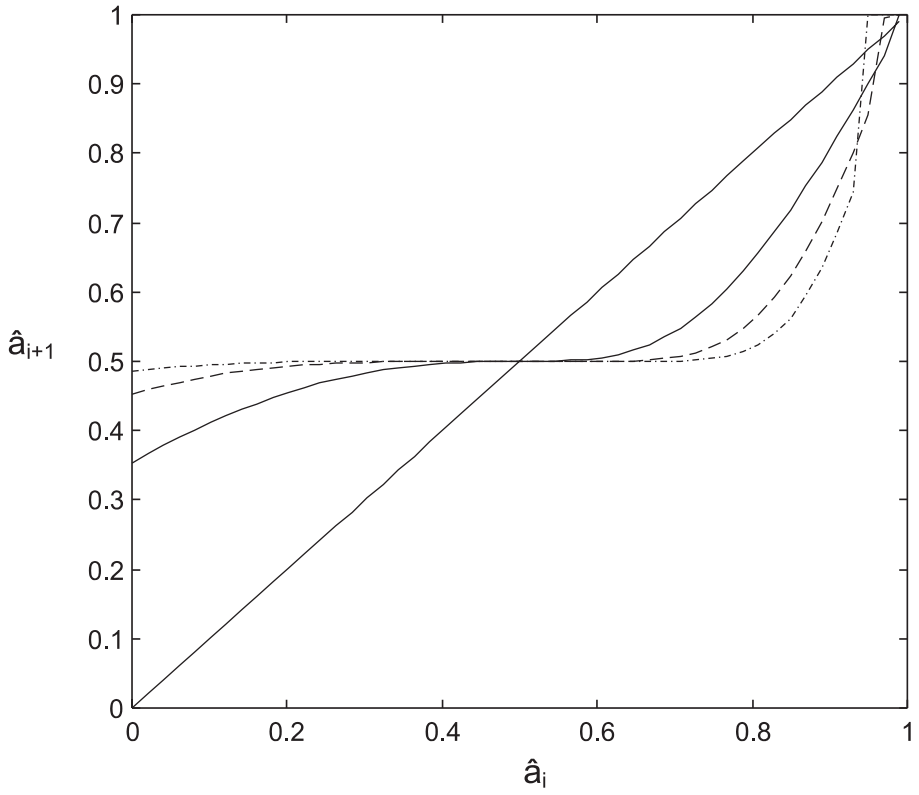

Fig. 4.4: *Mapping $\hat{a}_{i+1} = T(\hat{a}_i)$ together with $\hat{a}_{i+1} = \hat{a}_i$ for three values of the high model order: $n = 2$ (solid), $n = 3$ (dashed) and $n = 4$ (dash dotted).*

value for the pole location is found.

The points of intersection between the graph of the mapping and the line $y = x$ indicate the stationary points of the iteration. One of the stationary points is the true parameter value 0.5, which is a stable stationary point. The other is 0.97 and is unstable. This unstable point is undesirable and is caused by ill conditioning of the least squares problem. By inspection of the mapping T the influence of several degrees of freedom in this optimization algorithm can be assessed. These degrees of freedom or tuning parameters are the high model order, low model order and model reduction tool.

The influence of high model order is illustrated for the example in figure 4.4. The mapping is flattened out near the stable convergence point which results in a faster convergence of the iteration to the right parameter. Hence, an increased high model order improves the convergence but is less beneficial for the conditioning of the least squares problem.

Using the mapping T also a comparison can be made between ORTIT and other optimization algorithms in terms of e.g. speed of convergence. In the figure the mapping T is plotted for a gradient search algorithm and ORTIT. Clearly the steepest descent method has a very slow convergence compared to the orthogonal basis function approach. Although it must be said that steepest descent is known to converge slowly, it still indicates that the orthogonal basis function method is potentially an improvement over current

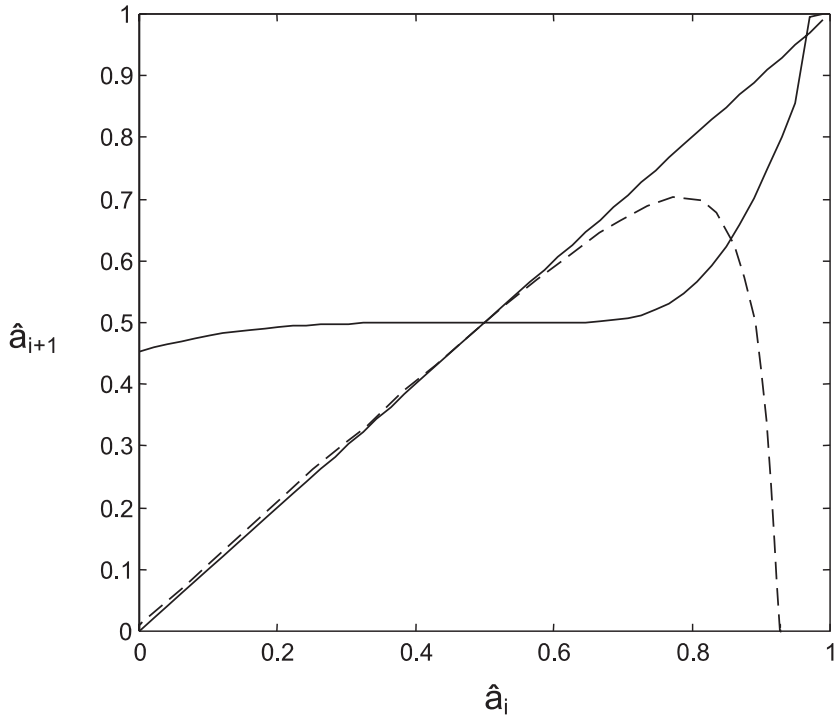

Fig. 4.5: *Mapping $\hat{a}_{i+1} = T(\hat{a}_i)$ together with $\hat{a}_{i+1} = \hat{a}_i$ for the orthogonal basis function approach (solid) and the method of steepest descent (dashed).*

practice.

4.3.5 Discussion

In this section an alternative optimization strategy is presented to approximate the solution to the output error and Box Jenkins identification problems for both scalar and multivariable systems. Instead of utilizing a standard nonlinear optimization algorithm an iteration is proposed of high order identification with orthogonal basis functions and model reduction. Both the identification and the model reduction can be performed without the use of nonlinear optimization, which makes the approach efficient also for large scale problems.

The optimization approach discussed here can be viewed as an alternative approximation concept, which seems to have a wider range of application than finding a solution to the optimization problem for identification purposes. This can be seen as follows. In a large class of optimization algorithms the following updating rule is applied $\hat{\theta}_{i+1} = \hat{\theta}_i + P \left.\frac{\partial V}{\partial \theta}\right|_{\hat{\theta}_i}$. Some special cases are

- $P = \alpha \frac{\partial^2 V}{\partial \theta^2}$, which amounts to a pure Newton step if $\alpha = 1$ and a, so called, modified Newton step if $\alpha \neq 1$ typically obtained from a line search yielding the smallest value of the cost function along the search direction.

- $P = \alpha I$ yields the method of steepest descent or gradient method. Again, α is typically obtained by a line search.

In these approaches the *cost function* $V(\theta)$ is approximated *locally* by a first or second order function to obtain a search direction.
Typically a Taylor expansion is used to the find a local optimimum around a current estimated solution θ_k, i.e.

$$V(\theta) = V(\theta_k) + \left.\frac{\partial V}{\partial \theta}\right|_{\theta_k} (\theta - \theta_k) + \frac{1}{2} \left.\frac{\partial^2 V}{\partial \theta^2}\right|_{\theta_k} (\theta - \theta_k)^2 + O((\theta - \theta_k)^3)$$

This type of approximation has a long history in the context of optimization algorithms. Many convergence results for optimization algorithms are based on a linear or quadratic approximation of the cost function.
One interpretation of ORTIT is the following. The approximation used in ORTIT is not an approximation of the cost function around a certain parameter value. In ORTIT a distinction is made between the model G that is searched for and the identification function J, i.e. $V(\theta) = J(G(\theta))$. In ORTIT an approximation is made of the model, or more generally the object, that is searched for. This approximation is a *global approximation of the object* rather than a local approximation of the cost function as was the case for the class of algorithms mentioned above. This concept seems a powerful approach to find a global optimum for specific classes of nonlinear optimization problems.
The concept of globally approximating an object with a linear parametrization can be used fruitfully in any optimization problem of which the cost function can be written as $V(\theta) = J(G(\theta))$ where J is convex and therefore relatively simple, and G may be non convex. With this line of reasoning the information about convexity of part of the cost function can be used fruitfully.
An open problem for ORTIT is still a mathematical analysis of the convergence properties of the iterative scheme. No guarantees can be given whether the iteration converges, which are the stationary points etc. The main complication in the analysis is the model reduction step. This reduction introduces a nonlinearity in the mapping T that is difficult to analyze.
A possible way is to investigate the convergence of the iteration is by inspection the properties of the mapping T. A possible way is to view the iterative process $\hat{\theta}_{k+1} = T(\hat{\theta}_k)$ as an autonomous nonlinear system of which the passivity or stability properties need to be assessed. In this way no monotonicity of the value of the cost function is necessary as is the case for most of the currently available algorithms. This implies that the algorithm can wander over hills in the cost function if it knows there is a low valley behind it.

4.4 Simulation example

In this section the simulation example of chapter 3 is revisited. The objective in this section is to obtain a low order model for the system which contains both the fast and the slow mode from a short data set. In the section 3.6 it is shown that the slow mode is particularly difficult to retrieve from the data because of the short data set and the bad signal to noise ratio. The applied subspace identification technique and output error prediction error identification with conventional nonlinear optimization to obtain an output error model fail to spot the slow mode. In the latter approach it is difficult to find an initialization of the nonlinear programming problem such that a model is obtained that contains this slow mode.
The iteration of high order ORTFIR identification and model reduction discussed in section 4.3.2 is applied to this identification problem. It is assumed that no prior knowledge of the system dynamics and model order is available and the initial poles are located at

$$\lambda_1 = 0.5,\ \lambda_2 = 0.7,\ \lambda_3 = 0.8$$

The basis generation was done by repeating the poles of the basis generating system following (3.9). The applied model reduction tool is balanced reduction as discussed in section 4.2.1. The least squares problems for the high order ORTFIR estimation are badly conditioned which made regularisation necessary. The applied regularisation procedure is the one discussed in section 3.5.2. In the table below an example is given of an iteration using ORTIT.

Iteration	High order n_h	V	V_{reg}	Low order n_l	V_{red}
1.	15 × 3=45	1.0175	1.0175	4	1.0547
2.	5 × 4=20	1.007	1.0157	4	1.0266
3.	5 × 4=20	1.0095	1.0168	4	1.0277
4.	5 × 4=20	1.01	1.0168	4	1.0277

Table 4.1: *Iteration with high order $n_h = n \times n_b$ model yielding cost function value V, reduced model using balanced reduction with order n_l yielding cost function value V_{red} and the value of the cost function for the regularized estimate V_{reg}.*

The best iterate, i.e. the reduced model of step 2, is used as initial model for the nonlinear programming problem to find the nearest optimal output error estimate. This optimal model has cost value $V_{opt} = 1.0265$. Clearly, the best result of ORTIT is near a local optimum of the prediction error cost function and served as an excellent starting value for the nonlinear optimization.
Important in this iteration is that for a given basis as much dynamics is captured as

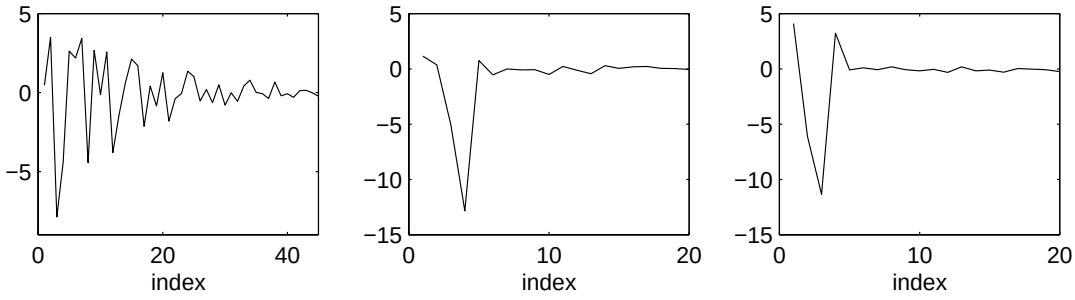

Fig. 4.6: *Estimated parameter or, equivalently, generalized impulse response for iteration 1 (left), 2 (middle) and 3 (right).*

possible in the high order estimate. This can be checked by inspection of the generalized expansion coefficients, which is in this case equivalent to the estimated parameter $\hat{\theta}$. When the parameters have damped out, i.e. are approximately zero, no significant improvement can be obtained by adding additional parameters. This provides a clear way to choose the model order during the iteration. The evolution of the expansion coefficients over three iterations is depicted in figure 4.6.

Clearly, as the dynamics of the reduced order model resemble the system dynamics more and more, the expansion coefficients in the tail become smaller and smaller. If, in the limit, the poles of the reduced order model coincide with those of a (local) minimum, only the first n_b coefficients will be nonzero.

The comparison in terms of the amplitude of the Bode plot is depicted figure 4.7.

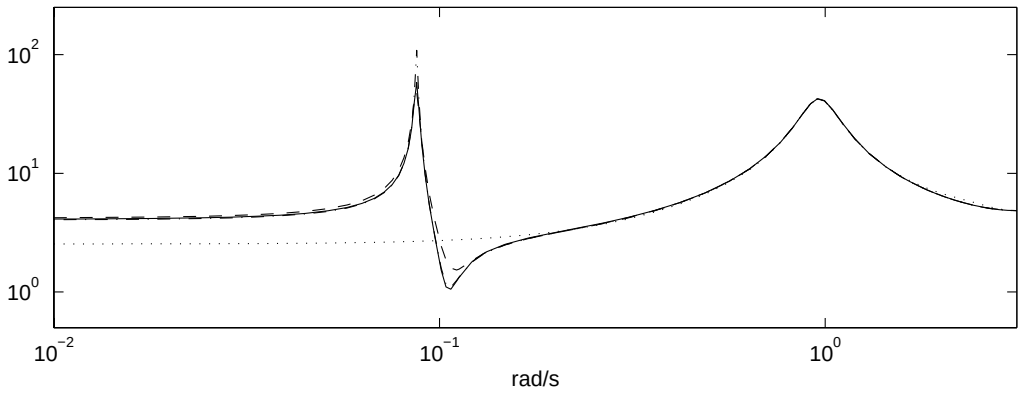

Fig. 4.7: *Amplitude plot of the system (solid), estimated model of order* 4 *after the second iteration (dashed) and optimal model of order* 4 *(dashed-dot) which practically coincides with the system and an output error model of order* 4 *with a conventional optimization approach (dotted).*

The model estimated in the second iteration provides a good approximation of both the slow and fast dynamics of the system. It also is a suitable initial model for a nonlinear optimization procedure.

Frequency weighted balanced reduction can have better results than balanced reduction, especially in the case of badly conditioned identification problems. To illustrate this an iteration is performed without the regularisation step. This results in badly conditioned estimates reflected in the large values of the estimated parameters, i.e. large values in the state space matrix C. The large values in C compared to the moderate values in the input balanced $\{A, B\}$ makes input output balancing an ill conditioned problem. This is illustrated in table 4.4 where the iteration results are given for the case where no regularisation step is applied.

Iteration	High order n_h	V	Low order n_l	V_{red}	V_{red}^{FW}
1.	15×3=45	1.0175	4	1.0546	1.0656
2.	5×4=20	1.0117	4	15.5679	1.0352
3.	5×4=20	1.0095	4	1.808×10^{17}	94.38

Table 4.2: *Iteration without regularisation. The high order model with order n_h yields a prediction error cost function value V. The low order model with reduced order n_l yielding cost function value V_{red} if balanced reduction is utilized and V_{red}^{FW} if weighted balanced reduction is utilized.*

Clearly, with balanced reduction the iteration degenerates after only one iteration step, due to the bad conditioning of the problem. This effect can be mitigated by the use of frequency weighted balanced reduction with the weighting proposed in this chapter. For this case, in the second iteration a fairly accurate model is obtained, which is not as accurate as the best model in the iteration with regularisation. In the third iteration the reduced order model is inaccurate. This is due to the bad conditioning of the estimate, which can be deduced from the unrealistically large coefficients in the estimated parameter vector. Clearly, frequency weighting in the reduction step can help to improve the estimate but no guarantees can be given about the value of the cost function compared to other reduction techniques.

In the given example the iteration converges, but not to the lowest value that is reached during the iteration. At the end of the second iteration the lowest value is reached. This is due to the fact that in the current set-up of the iteration no guarantee can be given that the iteration finds a lower value each iteration step. Also convergence cannot be guaranteed. The main difficulty here is the reduction step.

Due to the lack of convergence guarantees the iteration described in this example is not (yet) a mature alternative for nonlinear optimization. However, it can be used to find good initial estimates for the nonlinear optimization. Especially in highly nonconvex optimization problems the experience is that good local (sometimes global) optima are found with this procedure. If no optimality is required from the model it is usually

sufficient to perform a few (typically two or three) iteration steps to estimate a models of high quality.

4.5 Discussion

In industrial identification it is desirable to have an identification approach that provides an estimate with as little bias and variance as possible. This is important to deal with the short data sets with small signal to noise ratios. Moreover, the applied procedure must be applicable to multivariable systems. Due to the large scale of many industrial identification problems it is important that the approach is computationally efficient.
In this chapter the iterative scheme of high order ORTFIR modeling and model reduction introduced in [57] is further developed. The original iteration was intended for identification in an output error model structure. It is shown how the iteration can be extended to also identify a disturbance model such that the procedure can be used to identify a model in a Box-Jenkins model structure. By estimating a disturbance model the variance of the estimation of the system transfer can be reduced. In this way the maximum likelihood estimate, which is statistically the most efficient and has the smallest variance given by the Cramer-Rao bound, is approached more closely. Therefore, to improve the variance of the estimate it is advisable to estimate a disturbance model. This is especially true for short data sets with high signal to noise ratios.
It is shown that the underlying principle is applicable to a wider class of optimization problems than the optimization problem originating in linear system identification. However, no convergence guarantees can be given for the iteration. The concept appears to be powerful and more research in the convergence properties seems worth the effort.
An important question is which model reduction tool is appropriate for application in this iterative scheme. In this chapter a model reduction procedure is proposed, analogous to the procedure described in [159] for the scalar case and in [178] for the multivariable case. With this approach a low order model is obtained that has, under mild assumptions, similar properties as a maximum likelihood estimate. The procedure has the advantage that no nonlinear optimization has to be performed, only two projections which can be calculated efficiently also for large order models.
The procedure starts with a high order estimate of the system transfer with a model structure based on orthogonal functions. On the residual of this model a high order disturbance model is estimated with a moving average structure also based on orthogonal functions. With these estimates and their asymptotic properties a weighting filter is derived. This filter is such that, asymptotically in the number of data points and the model order, a weighted reduction in the L_2 norm yields a low order model with the same properties as a maximum likelihood estimate. This reduction method needs a nonlinear optimization procedure which is likely to get stuck in a local minimum and is computationally

expensive. Therefore a frequency weighted balanced reduction is applied with the proper weighting filter that can be calculated on the basis of the data.
The iterative scheme of high order identification and model reduction is an appropriate method for the identification of industrial processes. The estimation procedure consists of steps of least squares optimization and (frequency weighted) balanced reduction. Both steps can be solved with non-iterative procedures and can be performed with a limited computational load. This implies that the method is applicable to large scale industrial identification problems. The identification of a disturbance model can be used to improve variance properties of the model which is especially attractive if only short data sets are available with a low signal to noise ratio. In the simulation example of chapter 7 the iterative approach is shown to give good results on an industrial type identification problem.

Chapter 5

Model predictive control with system-based input parametrization

5.1 Introduction

In model predictive control, similar to system identification, an optimization problem has to be solved. In system identification an optimal model is to be found, while in model predictive control an optimal input signal is searched for. Dependent on the length of the prediction horizon, this can be either a signal of finite or infinite dimension. To obtain a tractable optimization problem which can be solved on-line, a finite number of free parameters is selected in the input signal. The free parameters can be located at various positions. It is desirable, as explained in section 2.6, to use a linear parametrization because for linear parametrization the optimization problem remains a quadratic programming problem that is convex which provides a unique global optimum.
The more freedom is allowed in the parametrization of the input, the higher the performance of the model predictive controller can be. However, this is at the cost of a high computational complexity. Because no closed-form solution for quadratic programming is available, the problem is solved with an iterative search algorithm as explained in section 2.4.8. This implies that the computational complexity is relatively high which limits the use of model predictive control to relatively slow systems with relatively low dynamic performance specifications. One way to make high performance model predictive control possible for faster systems is the selection of an efficient input parametrization.
The input parametrization problem in model predictive control can be viewed as a problem of finding a suitable subspace in the full input space. The reduced order space must be such that the performance of the controller is decreased as little as possible with respect to a free parametrization. Another important requirement of the subspace is that a trade-off can be made between performance and complexity. This line of reasoning is further developed in this chapter.

In section 5.2 a low dimensional subspace of the full input is selected systematically. For infinite horizon model predictive control this subspace is generated by a linear dynamical system, hence the name of the chapter. This parametrization has a strong relation with the system-based parametrization, discussed in chapter 3, utilized in system identification. The system-based input parametrization is utilized in section 5.3 in a model predictive control algorithm. In this section some properties of this approach are analyzed. The system based approach provides a way to use standard model reduction techniques to reduce the number of degrees of freedom. This is explained in section 5.4. The ideas presented in these sections are partly previously published in [29][30]. In section 5.5 it is shown how the constrained performance can be increased. The potential of the approach is illustrated in section 5.6 with two simulation examples and section 5.7 concludes the chapter.

5.2 Finding a suitable input parametrization

From a conceptual point of the view, the input parametrization problem, as stated in the previous section, is the following. Let a model be given of the system that is to be controlled. The input of the system is given by $u(t) \in \mathbb{R}^{n_u}$ and the state by $x(t) \in \mathbb{R}^n$. The inputs and states are constrained by $K_u u(t) < k_u$ and $K_x x(t) < k_x$ respectively. At this point the system can be either linear or nonlinear but it is assumed to be discrete time. Let the on-line optimization problem for receding horizon control be given by

$$\inf_{U \in \mathcal{U}} J(U, x_0) \tag{5.1}$$

where U is the input trajectory which is an element of some set $\mathcal{U}$ which is totally specified by the input and state constraints imposed on the system, $J(\cdot)$ is a cost function that reflects the desired performance and x_0 is some variable that is a function of the measurement with which the optimization problem is initialized at each time instant t. The variable x_0 can be either a measured state (full information state feedback) or a predicted state (partial information state feedback) or the measured output (general dynamic output feedback).

The input parametrization problem is to find a suitable subspace $\mathcal{U}_r$ with a lower dimension than the full space $\mathcal{U}$ such that the solution to the reduced order optimization problem

$$\inf_{U \in \mathcal{U} \cap \mathcal{U}_r} J(U, x_0) \tag{5.2}$$

provides *good control* for all possible measurements x_0.

A possible way to quantify *good control* is by making the reduced problem deviate as little as possible from the original problem. A possible way to choose a space $\mathcal{U}_r$ of dimension p is then by solving

$$\min_{\mathcal{U}_p, dim(\mathcal{U}_p) \leq p} V(\mathcal{U}_p) \tag{5.3}$$

where $V(\cdot)$ reflects the deviation of the reduced order problem to the full order problem. Possible cost functions are

- minimize the worst-case deviation in value of the cost function

$$V_1(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \left(\min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0) - \min_{U \in \mathcal{U}} J(U, x_0) \right) \tag{5.4}$$

  where the measurement x_0 varies over some set $\mathcal{X}$.

- minimize the worst-case deviation between the optimal input trajectories

$$V_2(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \left\| arg \min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0) - arg \min_{U \in \mathcal{U}} J(U, x_0) \right\|_\gamma \tag{5.5}$$

  where this deviation is measured in some norm $\| \cdot \|_\gamma$.

- minimize the worst-case difference between the first samples of the optimal input trajectories

$$V_3(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \| [I_{n_u} \, 0 \cdots 0] (U_r^*(\mathcal{U}_p, x_0) - U^*(x_0)) \|_\gamma \tag{5.6}$$

  with $U^*(\mathcal{U}_p, x_0) = arg \min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0)$ and $U^*(x_0) = arg \min_{U \in \mathcal{U}} J(U, x_0)$. This function seems suitable because only the first sample of the input trajectory is actually applied in a receding horizon strategy.

The optimization problems discussed above are related to n-width problems [119] where an optimal n-dimensional subspace is calculated by an optimization of a cost function with a free variable that is a set of a certain dimension. Unfortunately, the problems stated above are as such intractable. This is mainly because the cost functions must be evaluated for all possible sets of active constraints which is combinatorial in size. Therefore, only in specific cases a solution can be found.
Let the applied cost function be given by the quadratic function

$$J(u(\cdot)) = \sum_{t=0}^{P-1} \{ x^T(t) Q_1 x(t) + u^T(t) Q_2 u(t) \} + x^T(P) Q_0 x(P). \tag{5.7}$$

which is equivalent to the one used in the LQR problem (2.4.1) and the MPC problem (2.19). In terms of the stacked input vector $U^T = [u^T(0) \cdots u^T(P-1)]$ this cost function is given by

$$J(U) = U^T H U + 2 x_0^T g^T U \tag{5.8}$$

as explained in section 2.4.4.
The first case in which a solution to the input parametrization problem (5.3) can be found, is the *unconstrained case*. In this case the optimal control profiles are generated by a time-varying state feedback which can be calculated a priori. This controller is given in (2.16) where use is made of the Riccati difference equation (2.17). Note that the

control problem that is solved is an open loop control problem, while the solution is a closed loop controller. The only variable that cannot be computed a priori is the state x_0 with which the optimization problem is initialized at each time instant t. This simple observation leads to the following lemma for the subspace which contains all solutions to the discrete-time LQR problem.

Lemma 5.2.1 *Consider the quadratic cost function given by* (5.7) *with finite horizon* P *and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{H^{-1}g\}$$

with $H \in \mathbb{R}^{Pn_u \times Pn_u}, g \in \mathbb{R}^{Pn_u \times n}$ *given in* (5.8) *is a solution of* (5.3) *with any of the cost functions* (5.4), (5.5) *or* (5.6) *and* $\mathcal{X} = \mathbb{R}^n$.

Proof: The lemma follows from the fact that the unconstrained solution to (5.8) is given by $U(x_0) = -H^{-1}gx_0$. □

Apparently if one wants to find the unconstrained optimum for all possible initializations x_0, a search over an n-dimensional subspace is sufficient instead of a Pn_u-dimensional one.
Lemma 5.2.1 holds for a finite horizon criterion but a similar simple result also holds for the infinite horizon case.

Lemma 5.2.2 *Consider the quadratic cost function given by* (2.19) *with infinite horizon* $P = \infty$ *and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{\begin{bmatrix} F \\ F(A-BF) \\ F(A-BF)^2 \\ \vdots \end{bmatrix}\} \subset l_2^n[0,\infty), \tag{5.9}$$

with $\{A, B\}$ *state space matrices given in* (2.20) *and* F *the LQ-optimal state feedback given by* $F = (B^TXB + Q_2)^{-1}B^TXA$ *with* X *the unique nonnegative definite solution of the Algebraic Riccati Equation*

$$X = A^T[X - XB(B^TXB + Q_2)^{-1}B^TX]A + Q_1,$$

is a solution of (5.3) *with any of the cost functions* (5.4), (5.5) *or* (5.6) *and* $\mathcal{X} = \mathbb{R}^n$.

Proof: The unconstrained solution to the problem (2.24) with infinite horizons is the LQ optimal control profile given by

$$u(t, x_0) = -F(A-BF)^t x_0, \quad t = 0, 1, 2, \ldots$$

which directly shows the result. □

The lemma above indicates that an efficient input parametrization for infinite horizon model predictive control is generated by a dynamical system $\{F, A - BF\}$. With this parametrization the infinite dimensional optimization problem is reduced to a finite dimensional optimization problem with a number of free variables that is equal to the model order.

5.3 Model predictive control with input parametrization

In this section it is described how the input parametrization for infinite receding horizon control discussed in the previous section can be applied in an efficient model predictive control algorithm. The properties of this algorithm are investigated where special attention is paid to nominal performance and constrained closed-loop stability.

5.3.1 Infinite horizon model predictive control

If input and state constraints are incorporated in infinite receding horizon control, a finite parametrization of the input space is needed to obtain a finite dimensional constrained optimization problem. In section 2.4.7 several approaches to this problem are discussed that can be found in the literature. An important approach is the one described in [140] where the input is parametrized freely over the first P samples and is fixed to the LQ optimal state feedback for the tail to infinite time.
In this section the parametrization of lemma 5.2.1 is applied to obtain an alternative parametrization for model predictive control. This controller and some of its properties is given in the following proposition.

Proposition 5.3.1 *Let the linear discrete-time system and the model both be given by* (2.20) *subject to input and state constraints* (2.22). *Let the receding horizon controller cost function be given by* (2.23) *with* $P = \infty$ *and let* x_0 *be either the measured state vector (full information case) or a prediction thereof (partial information case). Let the input over the infinite horizon be parametrized as* $u(t, \theta) = F(A - BF)^t\theta$. *Then*

1. *the optimal control input is given by* $u(0, \theta^*) = F\theta^*$ *with* θ^* *the solution to the finite dimensional quadratic programming problem*

$$\min_{\theta \in \mathbb{R}^n} \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} Y \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \tag{5.10}$$

$$\text{subject to: } K_u F(A - BF)^t\theta < k_u \quad \text{and}$$

$$\begin{bmatrix} 0 & K_x \end{bmatrix} \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} < k_x, t = 0, 1, \ldots, N_c$$

where $\tilde{A} = \begin{bmatrix} A - BF & 0 \\ BF & A \end{bmatrix}$, $\tilde{C} = \begin{bmatrix} \sqrt{Q_1}F & 0 \\ 0 & \sqrt{Q_2} \end{bmatrix}$ *and* N_c *is the constraint horizon that is chosen such that after this time instant no constraints are active. Finally,* Y *is the solution of the Lyapunov equation*

$$\tilde{A}^T Y \tilde{A} + \tilde{C}^T \tilde{C} = Y$$

2. *if no constraints are active this controller is equivalent to LQ control with state feedback* F*.*

Proof: The input and state trajectories over the infinite horizon are given by

$$\begin{bmatrix} u(t,\theta) \\ x(t,\theta,x_0) \end{bmatrix} = \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \tag{5.11}$$

Substituting this in the cost function yields

$$J(\theta, x_0) = \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} \left(\sum_{t=0}^{\infty} \tilde{A}^{Tt} \tilde{C}^T \tilde{C} \tilde{A}^t \right) \begin{bmatrix} \theta \\ x_0 \end{bmatrix}$$

The matrix in this expression can be calculated with the Lyapunov equation in statement 1. Statement 2 can be proven by the fact that the unconstrained optimal solution is given by $\theta^* = -x_0$ which is the state measurement or prediction. From the description (5.11) it follows that this yields a state feedback control with state feedback F which is LQ-optimal. □

As described in the proposition the quadratic programming problem is either initialized with a state measurement or a state estimate. The latter is more likely because usually measurement of all the state variables is too costly or even impossible. It is well known in literature that an LQ optimal state feedback combined with a Kalman filter gives an LQG controller [3][72]. This dynamic output feedback controller is an optimal controller with respect to white noise disturbances on the outputs and states with known covariance matrices.

With the approach discussed in this section the optimization problem can be built up by solving one Riccati equation for the solution of the LQ control problem and a Lyapunov equation to specify the cost function. This can be done quickly because good software tools are available for solving Riccati and Lyapunov equations, also for large scale problems. Therefore, the procedure is flexible for on-line changes in the internal model, the parametrization and the controller cost function. This flexibility of the proposed method can be a successful property for constrained control of nonlinear systems

with switching linear predictive controllers such as nonlinear quadratic dynamic matrix control (NLQDMC, [46]).
Another property is that the tuning of the proposed algorithm is simple. A standard LQG control design is needed. The only additional choice that has to be made is the number of samples in the future over which the constraints are evaluated. This is given by the constraint horizon N_c which must be chosen such that possible constraint activation can be detected sufficiently long in advance. The parameter N_c is no tuning variable for nominal unconstrained performance as it has no influence on the closed-loop performance. The constraints horizon can also be chosen automatically as in [128].
The parametrization is based on the observation that in the unconstrained case the optimization problem with cost function (5.8) has a closed-form solution if x_0 is known. The proposed parametrization contains the optimal solutions for all x_0. This amounts to a subspace $\mathcal{U}_r$ which is also used to parametrize input profiles for the constraint case. Therefore the input profiles that are obtained in the constrained case are suboptimal. Because the parametrization is based on unconstrained observations the constrained performance may in some situations not be good. This issue will be addressed in section 5.5. But for many situations this parametrization provides a controller that attains optimal performance in the unconstrained case and good performance in the constrained case.

5.3.2 Nominal stability under constraints

In this section constrained stability properties are analyzed of the controller described in the previous section. In the next proposition it is proven that under mild conditions the controller provides a stable closed loop, also if constraints are active.

Proposition 5.3.2 *The predictive control strategy given in proposition 5.3.1 is globally asymptotically stable if and only if the optimization problem* (5.10) *is feasible*

Proof: The global time index is denoted with t and the local time index within the optimization is denoted with k. Let the input trajectory $u_t^*(k) = -F(A-BF)^k\theta_t^*$ be a feasible but possibly not optimal solution at time t. Let the corresponding cost be given by $J(t)$. The first sample of this trajectory is applied as current input $u(t) = F\theta^*$. This yields a state $x(t)$ which is equal to the predicted state if no disturbances are present and the model and plant are equal. Then a feasible trajectory for $t+1$ is given by $u_{t+1}^*(k) = -F(A-BF)^k\theta_{t+1}^*$ with $\theta_{t+1}^* = (A-BF)\theta_t^*$ as this is equivalent with the previous trajectory without the first sample. Denote the corresponding cost function with $J(t+1)$. This performance cost level need not be optimal therefore it holds that

$$J(t+1) \leq J(t) - x^T(t)Q_1x(t) - u^T(t)Q_2u(t)$$

Because $Q_1, Q_2 > 0$ the sequence $J(t)$ is decreasing. It is bounded from below by zero and therefore $J(t)$ converges to zero, hence $x(t), u(t)$ also converge to zero. Therefore the nonlinear state feedback is stabilizing. □

A similar result holds for the partial information case. Due to the separation principle this stabilizing state feedback combined with a stable observer yields a stabilizing dynamic output feedback [174]. The proposition implies that also in the presence of constraints the closed loop system remains stable if and only if the optimization problem is feasible. Feasibility can only be lost if hard state (output) constraints are used: only then is it possible that there is no input trajectory in the set of feasible input trajectories that renders the state (output) inside the feasible set of states (outputs). In [139] it is described how the problem of feasibility can be avoided. Often applied methods are constraint softening [174] or discarding constraints that are not crucial until the problem becomes feasible [41]. In both cases the proposed algorithm is also stabilizing in the presence of constraints.

5.4 Systematic reduction of the complexity

5.4.1 The basic principle

In this section it is described how the complexity of the on-line optimization of the model predictive control algorithm described in the previous section can be reduced while keeping track of the performance loss. This is done by choosing the parametrization of input in terms of a linear combination of profiles that have the largest contribution to the cost function. If the complexity must be reduced, it is possible to base the system-based input parametrization on a reduced order model i.e.

$$u(t, \theta) = F_r(A_r - B_r F_r)^t \theta \tag{5.12}$$

where $\{A_r, B_r\}$ are state-space matrices for the reduced order system and F_r is the LQ-optimal state feedback for this reduced order model.
This approach inherits the constrained stability properties of the full order case under the condition that the basis generating system is stable. This is proven in the next corollary.

Corollary 5.4.1 *The predictive control strategy given in proposition 5.3.1 with input parametrization generated by a stable autonomous dynamic system with state space matrices* $\{A_p, C_p\}$ *following*

$$u(t, \theta) = C_p A_p^t \theta$$

is globally asymptotically stable if and only if the optimization problem (5.10) *is feasible.*

Proof: Along identical lines as the proof of proposition (5.3.1) only with $\{A_p, C_p\}$ instead of $\{A - BF, F\}$. □

Note that this theorem implies that constrained closed-loop nominal stability with the proposed controller is preserved if the input is parametrized with any stable system.
The model reduction algorithm that is applied should be such that the reduced order basis functions have the largest contribution to the cost function and the functions that are discarded have the smallest. In the next sections two possible approaches are discussed.

5.4.2 LQG balanced reduction

For LQG control a tailor made model reduction can be performed with LQG-balanced reduction [63]. With this reduction technique first a similarity transformation is applied on the state space system that provides a coordinate system in which each state is equally well controllable with an LQ-controller as it can estimated with a Kalman filter. This similarity transformation is obtained by forcing the solution of the control discrete algebraic Riccati equation (CDARE) and the filter discrete algebraic Riccati equation (FDARE) to be equal and diagonal. These equations are given respectively by

$$X = A^T X A - A^T X B (B^T X B + I)^{-1} B^T X A + C^T C \tag{5.13}$$

$$Y = AYA^T - AYC^T(CXC^T + I)^{-1}CYA^T + B^T B. \tag{5.14}$$

These Riccati equations hold for the case $Q_1 = C^T C, Q_2 = I$, but LQG balanced reduction can be utilized for general $Q_1, Q_2 > 0$. In [63] it is proven that there exists such a similarity transformation in the continuous-time case. The same transformation holds for the discrete time case as is discussed in the next proposition.

Proposition 5.4.2 [63] *Let a linear time-invariant discrete-time system be given by* (2.20) *and let the positive definite solutions to the Riccati equations* (5.13),(5.14) *be* $X, Y > 0$. *Let* T *be given by*

$$T^{-1} = R^T U \Sigma^{-\frac{1}{2}}$$

with a Cholesky decomposition $R^*R = Y$ *and the eigenvalue decomposition* $RXR^* = U\Sigma^2 U^*$ *where* $U^*U = I$ *and* $\Sigma = diag\{\sigma_1, \ldots, \sigma_n\}$.
Then the transformed system $\{\tilde{A}, \tilde{B}, \tilde{C}\} = \{TAT^{-1}, TB, CT^{-1}\}$ *satisfies the Riccati equations* (5.13) *and* (5.14) *with*

$$\tilde{X} = \tilde{Y} = \Sigma$$

Proof: The proof follows from the fact that the solutions to the Riccati equations for the transformed system are given by

$$\begin{aligned}\tilde{X} &= TXT^* = \Sigma^{-\frac{1}{2}}U^*RPR^*U\Sigma^{-\frac{1}{2}} = \Sigma \\ \tilde{Y} &= (T^*)^{-1}YT^{-1} = \Sigma^{\frac{1}{2}}U^*(R^*)^{-1}(R^*R)R^{-1}U\Sigma^{\frac{1}{2}} = \Sigma\end{aligned}$$

For a full proof see [63]. □

The states that have a "small" corresponding value on the diagonal of $\Sigma = diag(\sigma_1, \ldots, \sigma_n)$, are both "easy" to filter and have a low contribution to the controller cost function. These can be discarded if the order of the controller is to be reduced. The states that have a "large" corresponding value on the diagonal of Σ, are both "difficult" to filter and are essential states to control and must certainly be accounted for in a reduced order controller. The diagonal elements are invariants for linear systems and can be used to decide on the reduction order in a similar manner as Hankel singular values are used in balanced reduction. The cost function for the LQG balanced system is given by

$$J(x(0)) = x(0)^T X x(0) = \sum_{i=1}^{n} \sigma_i x_i^2(0)$$

with the initial state $x^T(0) = [x_1(0) \ldots x_n(0)] \in \mathbb{R}^n$. Therefore the states corresponding to a large value of the LQG invariant σ_i has a large contribution to the value of the cost function. For good performance, i.e. a low value of the cost function, it is important that this state is actively controlled so that is remains small. Therefore it makes sense to discard the states corresponding to a low value of σ as they do not contribute much to the performance. In this way optimization complexity can be traded-off against nominal performance in a more quantitative way.
Let the model in LQG-balanced form be given by $\{A, B, C, D\}$ and the reduction order is given by n_r. Then the resulting reduced order model can simply be obtained by

$$A_r = [I_{n_r}\ 0]A[I_{n_r}\ 0]^T, B_r = [I_{n_r}\ 0]B, C_r = C[I_{n_r}\ 0]^T, D.$$

Then the reduced order input parametrization can be constructed with (5.12). The input parametrization based on the reduced order model is again generated by a stable dynamic system because the LQ-optimal state feedback is guaranteed to be stabilizing. Due to this fact and stability of the system it holds that for any reduction order the receding horizon controller is stable also in the constrained case.

5.4.3 Frequency weighted balanced reduction

Another reduction approach utilizes frequency weighted balanced reduction. Consider the series connection of the basis generating system and the plant given by

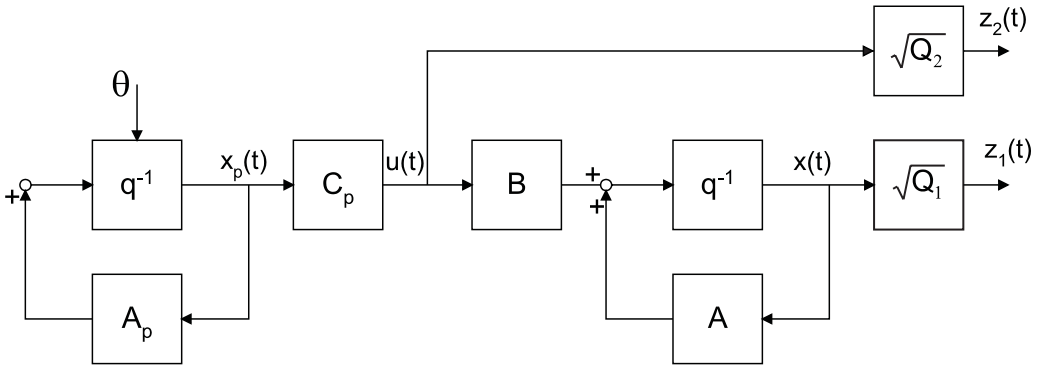

Fig. 5.1: *System used for model predictive controller reduction using frequency weighted balanced reduction.*

$$\begin{bmatrix} x_p(t+1) \\ x(t+1) \end{bmatrix} = \begin{bmatrix} A_p & BC_p \\ 0 & A \end{bmatrix} \begin{bmatrix} x_p(t) \\ x(t) \end{bmatrix}, \begin{bmatrix} x_p(0) \\ x(0) \end{bmatrix} = \begin{bmatrix} \theta \\ x_0 \end{bmatrix}$$
$$\begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix} = \begin{bmatrix} X_{Q_2}C_p & Q_1 \end{bmatrix} \begin{bmatrix} x_p(t) \\ x(t) \end{bmatrix}$$

where $Q_1 = X_{Q_1}^T X_{Q_1}$ and $Q_2 = X_{Q_2}^T X_{Q_2}$. This system is graphically depicted in the figure and was previously denoted with the pair $\{\tilde{C}, \tilde{A}\}$.
The observability matrix of the pair $\{\tilde{C}, \tilde{A}\}$

$$Q = \sum_{t=0}^{\infty} \tilde{A}^{Tt} \tilde{C}^T \tilde{C} \tilde{A}^t = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix}$$

has strong relation to the cost function (5.10) if a system based parametrization is utilized. If the system-based basis functions are generated with an autonomous system given by $\{A_p, C_p\}$, then it can easily be shown that the model predictive control cost function is given by

$$J(\theta, x_0) = \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} \theta \\ x_0 \end{bmatrix}$$

The part Q_{11} is related to the state $x_p(t)$ of the parametrization generating system. This state dimension is equal to the number of free parameters which is to be reduced. The singular values, retrieved from the singular value decomposition $Q_{11} = U\Sigma U^T$, indicate the directions that have a large contribution to the cost function and those that have a low contribution. Let an equivalence transformation be given by

$$\bar{A} = \begin{bmatrix} U^T & 0 \\ 0 & I \end{bmatrix} \tilde{A} \begin{bmatrix} U & 0 \\ 0 & I \end{bmatrix}, \; \bar{C} = \tilde{C} \begin{bmatrix} U & 0 \\ 0 & I \end{bmatrix}.$$

It follows easily that the Q_{11} block of the observability Gramian for this pair is Σ. With the following model reduction for the parametrization generating system, the largest n_r singular values are selected

$$A_r = [I_{n_r}\ 0]U^T A_p U[I_{n_r}\ 0]^T, C_r = C_p U[I_{n_r}\ 0]^T$$

The singular values can be used to assess the unconstrained performance degradation. This approach is equivalent to frequency weighted (output) balanced reduction of the system $\{C_p, A_p\}$ using as a dynamics weighting the system dynamics with the proper outputs z_1, z_2 as indicated in figure 5.1.

5.5 Improving the constrained performance

5.5.1 Motivation

In the previous sections an approach is described to choose the degree of freedom in model predictive control that is entirely based on unconstrained considerations. If constraints become active the performance of the proposed model predictive control algorithm can deteriorate. This can be seen as follows for the continuous time case but the line of reasoning holds equally well for the discrete time case. The system based basis functions are differentiable, while the constrained profiles are not as they usually possess edges. It is known to be inefficient to fit nondifferentiable functions with a linear combination of differentiable ones. An illustration is the Gibbs phenomenon for the Fourier series representation of a square wave [116] where a ripple occurs near the edges of the squares. For the specific case of model predictive control this type of inefficiency can be handled in various ways. The following two approaches are further discussed here.

- Embedding of the subspace $\mathcal{U}_r$ generated by the system based basis functions, by a larger space $\mathcal{U}_r^{ext}$ to improve the solution for active constraints.
- Making the parametrization time-varying $\mathcal{U}_r(t)$ so that the parametrization can be tailored to the current situation at hand. This approach will be denoted with parametrization scheduling.

These two approaches are discussed in the sequel.

5.5.2 Parametrization embedding

The flexibility of the parametrization can be improved by extension of the subspace in which an optimal solution is searched for. The infinite horizon parametrization discussed in this chapter given by

$$u(t, \theta) = F(A - BF)^t \theta \tag{5.15}$$

spans a subspace that is explicitly tailored to the unconstrained case. Mild constraint activation can be dealt with, however in the multivariable case constrained inputs cause a large performance degradation. An important reason for this performance degradation is the fact that, in general, the parametrization (5.15) does not provide an independent parametrization for each input $u_i(t)$. There exists coupling of inputs due to the structure in the pair A_p, C_p that generates the input space. This provides a very efficient parametrization for unconstrained and mildly constraint situations but is less robustness for other constraint situations like e.g. fixed inputs.

It is shown that the subspace $\mathcal{U}_r$ can be extended to $\mathcal{U}_{ext}$ such that all unconstrained optimal input profiles are contained in the set, but each input is independently parametrized. Let the following parametrization of the separate inputs be given

$$u_i(t, \theta_i) = C_i A_i^t \theta_i$$

where $\{C_i, A_i\}$ is a minimal realization of $\{F_i, (A - BF)\}$ with F_i the i-th row of F. This provides a parametrization

$$u(t, \theta) = \begin{bmatrix} C_1 & & \\ & \ddots & \\ & & C_{n_u} \end{bmatrix} \begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_{n_u} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_{n_u} \end{bmatrix}$$

With this parametrization the dimension is increased from n to at most $n_u n$ but at the benefit of an independent parametrization of each input. This approach has, obviously, a strong analogy to the identification case where $\mathcal{U}_r$ corresponds to a parametrization of a multivariable model with $\{A, C\}$-structure and $\mathcal{U}_{ext}$ to the use of scalar basis functions for each output separately.

Another embedding of the parametrization is the following. In chapter 3 several ways are described to obtain an orthogonal basis for the class of l_2 signals starting from dynamic information using Laguerre, Kautz and general basis functions. This is also a possible embedding approach that is followed in the work by Dumont and coworkers [37][38] where a Laguerre basis is used to parametrize the input trajectory. This has, however, a large drawback in the context of constrained model predictive control. This is due to the fact that approximating a non-differentiable signal with differentiable ones can be highly inefficient.

Hence, the use of information about system dynamics is efficient in a limited amount of situations. Specifically in cases where the constraints play a moderate role, a system-based parametrization can improve the efficiency considerably. It is inefficient where the constraints play a dominant role.

By parametrization embedding the constraint performance can be increased. This is at the cost of an increase in the number of free variables in the optimization. Let n be the dimension of the A-matrix in (5.15) and n_u be the number of inputs. If the input

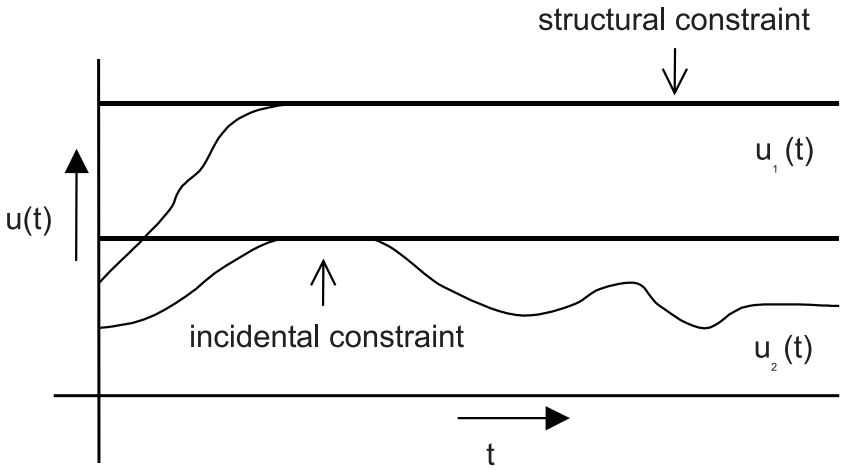

Fig. 5.2: *Graphical representation of structural and incidental constraints.*

parametrization is made independent along the lines discussed above, the number of free variables can be as high as nn_u. Dependent on the problem at hand, this will provide improved performance over other parametrizations, e.g. pulse or blocking, with the same number of degrees of freedom.

5.5.3 Parametrization scheduling

In this section another approach is discussed to improve the constrained performance of model predictive control with system-based input parametrization. In the approach at each time instant the degrees of freedom are chosen according to the specific situation, specifically the constraints, at hand. This strategy is denoted with parametrization scheduling and boils down to choosing a time-varying input parametrization. For this purpose two classes of active constraints are distinguished

- structural constraints. This refers to the situation where the working point is such that one or several inputs are saturated and cannot be used for control purposes. This situation can occur if an output reference is specified that is such that it cannot be obtained within the given input constraint set. Also a persistent disturbance can drive the system to this situation.

- incidental constraint. This refers to constraints that are active for only a short time. Incidental constraints can be caused by a large disturbance or a badly damped dynamical phenomenon in the system that has driven the inputs or states temporarily to their constraints.

The difference between these classes is graphically depicted in figure 5.2.
The system-based input parametrization provides good performance for incidental constraints, but structural constraints can deteriorate the performance. Therefore a proce-

dure is proposed that leaves the parametrization unaltered for incidental constraints but changes the parametrization if structural constraints are detected. For this purpose a detection mechanism for structural constraints is required.

There is only one reason for structural constraints if no disturbances are present. This is when a reference value r is chosen that is infeasible for the given input and state constraints and the static behaviour of the system. If disturbances are present, structural contraints can occur for constant disturbances d. A constant input or output disturbance can be estimated with an observer as discussed in section 2.4. This situation can be easily accounted for in the procedure that is given in the sequel. For convenience of notation it is assumed that no disturbance is present.

If a reference value is such that constraints are permanently active on one or several inputs, the structure of the system has changed from a full $n_u \times n_y$ dynamical system to a $n_{uc} \times n_y$ system with $n_{uc} < n_u$. If the model predictive control problem is solved for the system of full size, several inputs must be fixed. This is done by consuming a part of the freedom that is present in the parametrization of the input trajectory. This is an inefficient parametrization in the sense that this freedom could have been better spent on the free inputs. Because the input-dimension of the system has changed, the parametrization should be tailored to the new situation. This can be done with the following parametrization scheduling algorithm. In the description an input parametetrization $\mathcal{P}$ is used to denote a parametrization of the form $u(t,\theta) = C_p A_p^t \theta$ with some state space matrices A_p, C_p.

Algorithm 5.5.1 *Parametrization scheduling.*

Step 1. *Calculate the optimal input trajectory in the current parametrization $\mathcal{P}_0$. If the first sample of one or more inputs is constrained go to Step 2 else terminate.*

Step 2. *Assess whether one or more of the active constraints on the first sample are structural. If so proceed with Step 3 else terminate.*

Step 3. *Change the input/output dimension of the system such that the fixed inputs are taken to be fixed such that no optimization freedom is consumed in these inputs. Because the number of degrees of freedom is reduced the reference value must be adapted such that a feasible target is specified.*

Step 4. *Obtain a new parametrization $\mathcal{P}$ for the system with the new structure along the lines described in the previous section.*

Step 5. *Perform the optimization as specified in 5.3.1 with the parametrization obtained in Step 4 to obtain the optimal input trajectory .*

In the sequel these steps are further discussed.

Step 1. Calculate the optimal input with the current parametrization $\mathcal{P}_0$
The parametrization applied in the previous time instant corresponds to the structure of the system at that time instant. The control profile is calculated for this structure and on the basis of this solution it is assessed whether a structural change has to be made.

Step 2. Assessment of structural constraints
Assessing whether the constraints are incidental or structural is a difficult task in the most realistic case of an inaccurate model and the presence of disturbances. If it is assumed that the model is equivalent to the system and there are no disturbances the only cause of structural constraints is a reference value that cannot be obtained within the given constraint set. For this ideal situation a possible procedure is the following.
An input constraint is structural if the reference r cannot be attained in steady state, such that the input lies in the feasible region determined by the input constraints. This is the case if for a given desired reference value $r \in \mathbb{R}^{n_y}$ there does not exist a static input vector $u_{ss} \in \mathbb{R}^{n_u}$ which satisfies the static input output relation

$$r = K_{ss}u_{ss} \text{ such that } K_u u_{ss} \leq k_u \tag{5.16}$$

where $K_{ss} = C(I - A)^{-1}B \in \mathbb{R}^{n_y \times n_u}$ is the steady-state gain of the system. If the reference r is found infeasible a decision must be made about a new reference vector r^* that is feasible. In the next step several strategies are discussed to obtain this.

Step 3. Reconfiguration of the control problem

Many strategies are available to choose a feasible reference vector r^* on the basis of an infeasible reference vector r. A strategy is to define the new working point such that the difference between the desired reference and the obtainable outputs is minimized in some norm, i.e. the static optimization problem

$$u_{ss} = arg\min_{u_{ss}} \|r - K_{ss}u_{ss}\|_\gamma \text{ such that } K_u u_{ss} \leq k_u.$$

If $\gamma = 2$ the distance in minimized in least squares sense [96], $\gamma = \infty$ the maximum distance is minimized and $\gamma = 1$ the sum of absolute values is minimized. The formula is given if only input constraints are present but it can easily be extended to the general case of input and state constraints.
Another simple strategy that can be utilized to decide on a new reference r^* if several reference values r are not crucial and can be discarded without serious consequences. In that case the references can be ordered from essential to unimportant. Now, reference signals with a low priority can be discarded until a reference vector r^* is obtained that can indeed be attained while satisfying the input constraints. Also a combination of the approaches discussed above is possible. In that case some reference values can be

discarded, some reference values should be obtained in least squares sense and others in infinity norm sense.
Now the steady state value of the input is u_{ss} and of the reference is $r^* = K_{ss}u_{ss}$. Some of the inputs, denoted with u_{fixed} will be equal to a constraint. These inputs are denoted as structural constrained inputs. The other inputs are denoted with u_{free}. The new steady state relation is given by

$$r^* = \begin{bmatrix} K_{ss}^{fixed} & K_{ss}^{free} \end{bmatrix} \begin{bmatrix} u_{fixed} \\ u_{free} \end{bmatrix} \text{ such that } K_u u_{ss} \leq k_u \tag{5.17}$$

where r^* are those reference values that remained, and

$$\begin{bmatrix} K_{ss}^{fixed} & K_{ss}^{free} \end{bmatrix} = \begin{bmatrix} C^*(I-A)^{-1}B_{fixed} & C^*(I-A)^{-1}B_{free} \end{bmatrix} \tag{5.18}$$

where C^* corresponds with the outputs that are given a setpoint, B_{fixed}, B_{free} correspond to the fixed and free inputs, u_{fixed} is the input fixed to the active structural constraint, u_{free} is the input that is still available for control purposes.
The control problem for the system

$$y^*(t) = C^*(qI-A)^{-1}B_{free}u(t) + C^*(I-A)^{-1}B_{fixed}u_{fixed} \tag{5.19}$$

where the influence of the structurally constrained input is equivalent to a fixed output disturbance.
The approach followed here has similarities with the concept of reference governing [8] where the reference is adapted to satisfy constraints. Only here the reference values that are incorporated are unchanged while in reference governing the reference values are adapted constantly to satisfy constraints on input and state variables. Also a strong relation is present with reconfigurable control as presented in [86]

Step 4. Calculate a new parametrization $\mathcal{P}$
For the reconfigured system (5.19) with state space description $\{A, B_{free}, C^*, 0\}$ a new optimal control problem must be specified. For this control problem another parametrization is efficient. This parametrization is typically one that does not consume degrees of freedom to fix one or several inputs but applies all the freedom to the true degrees of freedom. In this step a redistribution of degrees of freedom takes place.
The following steps are proposed

- if necessary do (LQG or frequency weighted balanced) reduction with the new input/output dimension of the system
- calculate the new LQ-optimal state feedback for the full or reduced order system
- build the constrained infinite horizon optimization problem given in proposition 5.3.1.

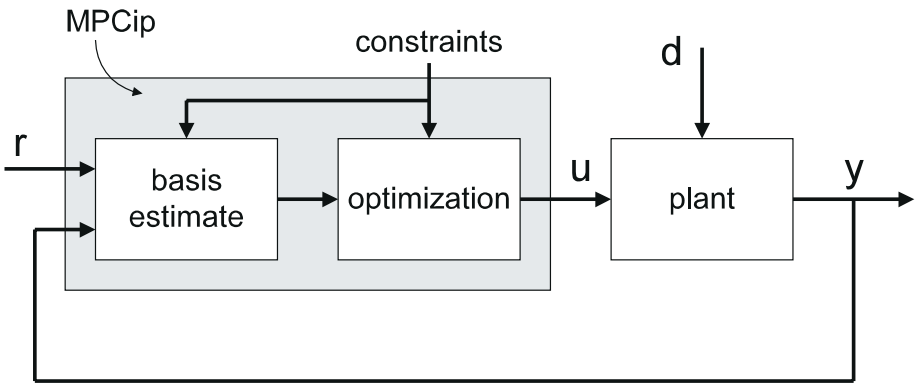

Fig. 5.3: *Graphical representation of the architecture of model predictive control with input parametrization.*

After this step the new constrained optimization problem is complete.

Step 5. Solve reduced optimization problem

Perform the optimization as specified in proposition 5.3.1 with the parametrization obtained in Step 4 to obtain the optimal input trajectory.

This procedure provides a way to parametrize the optimization problem efficiently, also if the structure of the control problem has changed due to structural constraints. Therefore no free parameters are consumed to fix a certain input to a constraint value but if an input is structurally constrained a redistribution of optimization freedom takes place. With this approach the constrained performance can be improved with only a relatively small computational effort: i.e. a Riccati equation has to be solved to obtain the state feedback vector for the new structure and a Lyapunov equation has to be solved to calculate the quadratic programming problem. Good and efficient solvers are available to find the solution to Riccati and Lyapunov equations that are also applicable to large-scale systems. However, the different control problems with the different input output dimensions can also be calculated off-line. In this case only an on-line switch has to be made between the different control problems which increases the computational speed at the cost of an increased off-line computation and on-line computer storage.

The architecture of the control system is given in figure 5.3. The block labeled with *basis estimation* is the element in which the determination of the structure takes place as it is described in Step 1 until Step 4. The actual optimization takes place in the block labeled with *optimization* as described in Step 5.

5.6 Simulation example

In this section two simulation examples are given to demonstrate the properties of the proposed approach, denoted with MPC_{ip}, compared to model predictive control with a

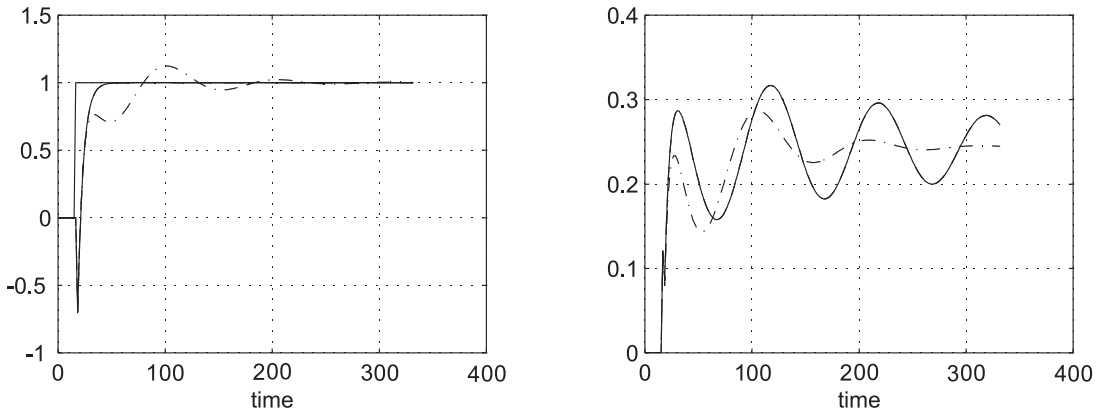

Fig. 5.4: *Input (right) and output (left) of the closed-loop step response conventional MPC with* $M = P = 100$ *(solid) and* $P = 100, M = 4$ *(dash dotted) and* MPC_{ip} *(dashed) with* 4 *degrees of freedom which coincides with the solid line.*

finite prediction and control horizon, denoted with MPC. The first simulation example concerns a scalar system where a large number of free variables is needed to obtain satisfactory performance with model predictive control with a conventional pulse parametrization while with the proposed model predictive control algorithm high performance is obtained with only a few free variables. The second example is multivariable and is used to show the potentials of the reduction method of section 5.

Simulation example 1

The first system that is considered is a highly oscillatory nonminimum-phase system given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 + 3.8630z^2 - 2.6426z + 0.8084}$$

The input is constrained by $-0.275 \leq u(t) \leq 0.275$. The open-loop step response is given in figure 3.6. If conventional MPC is applied to this system it is necessary to take a prediction horizon that is long enough to incorporate at least one full period, i.e. P=100. If a pulse parametrization is utilized, the choice of the control horizon is critical for this system. A control horizon which is equal to the prediction horizon gives good performance. However, decreasing the degrees of freedom easily gives bad performance as can be seen from figure 5.4. Reduction of the control horizon until $M = 75$ is possible without considerable loss of performance, further reduction gives bad performance due to the slow oscillation. Suitable tuning parameters are given by $Q_1 = 1, Q_2 = 1$.
Due to the long prediction and control horizon the computational burden is large for MPC. With the approach presented in this chapter, here denoted with MPC_{ip}, the number of degrees of freedom is equal to the model order, i.e. n=4. Suitable tuning parameters are

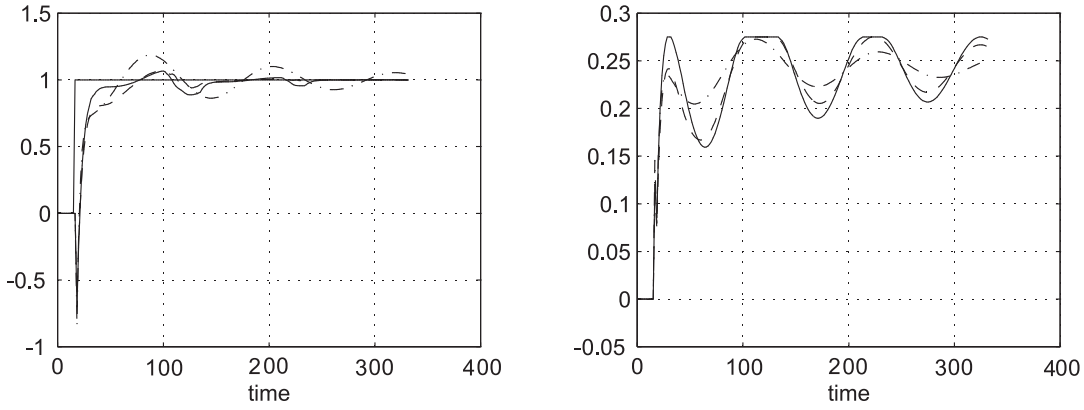

Fig. 5.5: *Input (right) and output (left) of the constrained closed-loop step response conventional MPC with $M = P = 100$ (solid) and P=100,M=4 (dash dotted) and MPC_{ip} (dashed) with 4 degrees of freedom.*

$Q_1 = 1, Q_2 = 1$. This yields a controlled performance, depicted in figure 5.4, that is practically identical to the fully parametrized controller only at a much lower computational cost.

Also if constraints are active MPC_{ip} performs better than conventional MPC with the same number of degrees of freedom as can be seen from figure 5.5. It performs slightly less than the fully parametrized MPC but with a much lower computational burden.

To give an indication of the computational load, on a Pentium 233 MHz computer the simulation of 200 time samples cost 511.83 seconds for fully parametrized conventional MPC and only 10.885 seconds for MPC_{ip}. To illustrate the idea of using basis functions to build up the space of allowable input trajectories, the four basis functions are plotted in figure 5.6.

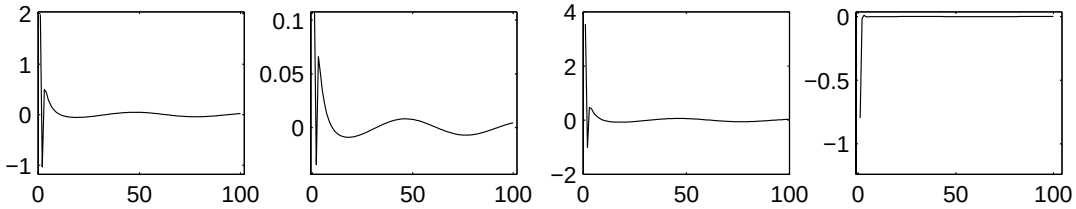

Fig. 5.6: *Four basis functions used in MPC_{ip}.*

Simulation example 2

The second system that is considered is a 4 input 4 output subsystem of the nonlinear simulation model of a fluidized bed catalytic cracking unit (FCCU) given in [89]. The four inputs are the fresh feed F_3, the slurry recycle F_4, reactor/regenerator differential pressure ΔP and lift air blower setpoint V_{lift}. The four outputs are the regenerator temperature

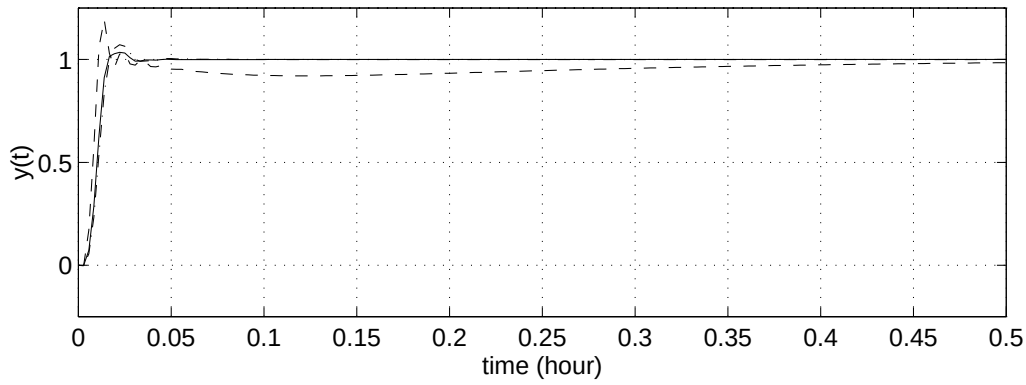

Fig. 5.7: *Controlled output T_{reg} on a step on the reference signal of $2^\circ C$. The applied controllers, both with the same number 8 degrees of freedom, are MPC (dashed), MPC_{ip} with infinite prediction horizon (solid) and with finite horizon (dashdotted) which practically coincides with the solid line.*

T_{reg}, reactor temperature T_r, the oxygen concentration in the stack gas outlet of the regenerator C_{O_2} and the reactor stand-pipe level l_{sp}. A detailed flowsheet of the process and an explanation of the input and output variables can be found in chapter 7. The system has large interaction, a combination of fast and slow dynamical phenomena and it is nonlinear. A linear model of this system is obtained that has order 8 that is accurate around the working condition.

Model predictive control with the proposed input parametrizations from lemma 5.2.1 and 5.2.2 are applied. In both cases the number of degrees of freedom is 8. To detect constraint activation well in advance the constraint horizon is chosen $N_c = 100$, which is large because of the slow dynamics in the system. For comparison a model predictive controller is designed with a pulse parametrization with the same number of degrees of freedom, i.e. $M = 2$. The prediction horizon is taken to be $P = 100$. Both controllers are tuned with tuning parameters $Q = I, R = I$. In this comparison the complexity of the online optimization for MPC_{ip} and MPC is identical. The unconstrained closed-loop behaviour is tested by a step signal on the reference of the first output. The simulation result is depicted in figure 5.7.

For this example the MPC_{ip}-controller is better able to deal with the combination of fast and slow dynamics than MPC with a pulse parametrization with the same number of degrees of freedom. This is because the latter has a higher overshoot as well as undershoot. The reduction procedure of section 5.4.2 is applied to assess whether the complexity can be reduced without significant loss of performance. For this purpose the LQG invariants of the model are determined. These are plotted in figure 5.8.

From this figure it can be seen that in the LQG invariants a jump occurs at order 5. The first 5 LQG invariants are larger or equal to one and the rest is significantly smaller. This jump in the value of LQG invariants indicates that the number of free variables in the

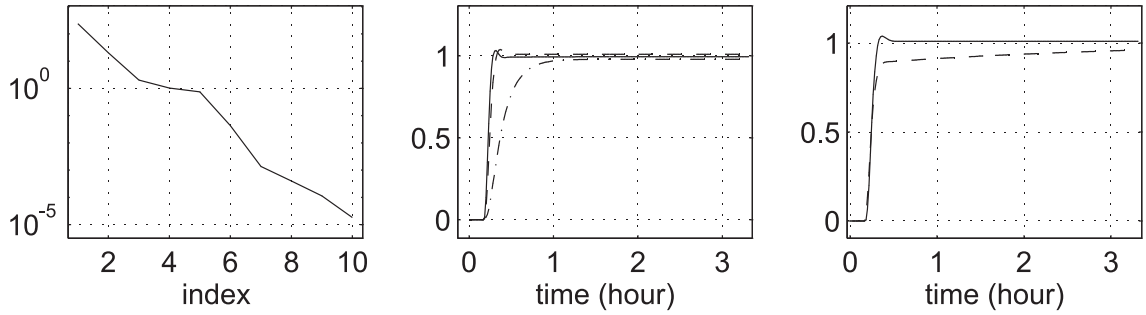

Fig. 5.8: *Left: LQG invariants of the model. Middle: the controlled output T_{reg} on a step on the reference signal of 2° C; unreduced MPC_{ip} of order 10 (solid), reduced to order 5 (dashed) and reduced to order 4 (dash dotted). Right: MPC with the input parametrization proposed in this chapter, reduced to 5 degrees of freedom (solid) and MPC with conventional parametrization with 5 degrees of freedom (dashed).*

input parametrization can be reduced to order 5 without considerable loss in closed-loop performance.
The closed-loop step response for the full order and reduced order situations is also given in figure 5.8. This figure indicates that the loss in performance is indeed small. Further reduction to order 4 shows a large loss in performance which is in accordance with the LQG invariants. Hence, these invariants give a good indication of the smallest number of free variables that is needed to obtain good unconstrained performance. In figure 5.8 also a comparison is made with MPC_{ip} and MPC with the same number of degrees of freedom. From this it becomes clear that the loss of unconstrained performance of the former is much smaller than for the latter parametrization.

5.7 Discussion

In this chapter it is investigated how a linear parametrization can be utilized to improve the efficiency of model predictive control algorithms. An efficient parametrization is obtained using the observation that the class of all solutions to a finite and infinite horizon LQ control problem can be parametrized with a number of free parameters that is equal to the model order, without loss of unconstrained performance.
For an infinite horizon cost function the finite dimensional parametrization is generated by an autonomous linear dymamical system. This parametrization has a strong similarity with the system-based basis functions utilized in system identification as discussed in chapter 3. It is shown that model predictive control with the system-based parametriza-

tion yields a nominally stable closed loop system if the constrained optimization problem is feasible.

An important requirement of a parametrization is that a trade-off can be made between performance and complexity as argued in section 2.7. For the proposed system-based input parametrization it is shown that this trade-off can be made using standard model reduction tools. This reduction procedure utilizes either an LQG-balanced reduction or frequency weighted balanced reduction of the basis generating model to select those basis functions that have the largest influence on the value of the cost function. The smallest number of basis functions that is needed to obtain approximately the unreduced performance can conveniently be read from the LQG invariants or the Hankel singular values of the basis generating model. The closed-loop system with the infinite horizon controller with the reduced order parametrization is also shown to be stable in both the constrained and unconstrained case.

In the present algorithm the choice of the basis is independent of the constraints. Hence, the more the constraints play a role in the control problem, the more performance can be lost compared to a free parametrization. To deal with this problem basically two strategies are discussed to improve the constrained performance: the embedding approach and the scheduling approach.

In the embedding approach the dimension of the space spanned by the basis functions is increased to improve the flexibility of the parametrization, at the cost of a higher computational load. If the complexity is increased systematically using the basis generation as discussed in chapter 3, i.e. utilizing Laguerre, two parameter Kautz or generalized basis functions, it is found that no efficient parametrization is obtained. The main reason for this is that these basis functions are differentiable functions while the constrained input trajectories are non-differentiable functions. Approximating a non-differentiable function with a set of differentiable ones is known to be inefficient. An example of this is the Gibbs phenomenon that occurs when a square wave is approximated by sinusoidal functions.

The scheduling approach can be used to improve the constrained performance for multivariable systems constraints of one or several inputs are permanently active. In this case a model predictive control problem can be formulated without the fixed input signals. By reformulation of the control problem no free parameters are consumed to keep the input at a fixed value but all free parameters are utilized in the input signal that are not fixed. The reformulation can be done either on-line which increases the computational load considerably, or off-line which increases the need for more storage capacity in the controller.

Clearly, the system based approach has its limitations in the constrained case. For mildly constrained situations and specific constrained situations the system based approach provides an efficient parametrization. However, the more constraints play a role in the control problem, the less efficient the system based approach becomes and the more likely

it becomes that other parametrizations, like e.g. pulse or blocking, are more efficient. Therefore, in the next chapter a parametrization is discussed that is specifically tailored to the constraint situation at hand rather than utilizing the system dynamics to construct a set of basis functions.

Chapter 6

Constrained model predictive control with on-line input parametrization

6.1 Introduction

In the previous chapter a model predictive control algorithm is developed with an input parametrization that utilizes a low dimensional subset of the full input space that is generated by an autonomous linear dynamical system. This provides an efficient parametrization for mildly constrained systems, but for situation where the constraints play a dominant role a large loss in performance can result. The constrained performance can be improved by a scheduling of the parametrization according to the current active constraints at hand. Still, the system based basis functions fail to be efficient for a situation where constraints play a major role.

As opposed to the proposed input parametrization strategy in the previous chapter, in this chapter an approach is developed that aims to utilize prior knowledge about the active constraints rather than knowledge about the system dynamics to construct an input parametrization. An algorithm is developed that selects on-line a low dimensional subset of the full input space at each time instant. It is shown how the efficiency of model predictive control can be increased by the on-line choice of a low number of degrees of freedom in the parametrization of the input trajectory that is specifically tailored to the constraint situation at hand.

In section 6.2 it is shown how a subset of the full input space can be selected such that it is tailored to the specific situation at hand. Crucial in this procedure is the use of prior information about the active constraints that can be retrieved from the solution of the quadratic programming problem in the previous time instant. In section 6.3 algorithmic properties are addressed. In this section an emphasis is placed on the computational complexity compared to other optimization techniques like the active set method and interior point algorithms. This subset selection can be utilized in model predictive control,

which is analyzed in 6.4. A simulation example is given in section 6.5 to illustrate the potential of the approach and section 6.6 concludes this chapter. The results presented in this chapter are partly previously published in [31].

6.2 Selection of degrees of freedom in quadratic programming

In model predictive control a possibly large scale quadratic programming problem has to be solved on-line. Characteristic for quadratic programming in model predictive control applications is the possibly large scale, the large amount of prior information due to the receding horizon principle, the high requirement on efficiency and the relatively low requirement on accuracy of the solution because only the first sample of the optimal trajectory is applied.
Several optimization algorithms are available to solve a QP such as active set methods and interior point methods. The first is most widely used but the second class of algorithms is much more efficient [168]. An overview of recent developments in quadratic programming algorithms can be found in [79]. But no matter how efficient the applied numerical routines are, the large number of decision variables remains a source for an increase in computation time. Therefore a strategy is searched for that reduces the number of free variables in the problem.
The high computational load of quadratic programming solvers for large scale problems is mainly due to the fact that the active set of constraints has to be searched for. If there is a large number of decision variables, the number of possible active sets is very large. However once the active set is known, the solution can be computed fast from the Lagrange necessary conditions. From the prior knowledge that is available at each time instant it should be possible to make a selection of constraints that are certainly not active, certainly active and a set that is either active or inactive. By searching only over the uncertain constraints a decrease in computational load should be obtainable. This strategy basically involves following steps.

Main procedure

Step 1. *Divide the set of constraints in a set of inactive, active and possibly active constraints.*

Step 2. *Determine a low dimensional subset of the full input space* $\mathbb{R}^{Nn_u}$ *that contains all input trajectories that comply with the three sets of constraints.*

Step 3. *Solve a reduced order quadratic programming problem by searching a solution in the selected subset.*

This strategy is further developed in this section.

Step 1: Dividing the set of constraints

The success of the approach discussed in this section depends on a good estimation of the set of active constraints. Also information is needed about the constraints that might be active. The active set of constraints can be determined in several ways (see e.g. [85][169]). However, with these approaches it is difficult to extract information about the possibly active constraints. Hence, an approach is needed where this information is obtained. A procedure that we will pursue is based on Hildreth's quadratic programming approach [84].

Let the applied linear model predictive control cost function again be given by

$$J(u(\cdot)) = \sum_{t=0}^{N-1} \{x^T(t)Q_1x(t) + u^T(t)Q_2u(t)\} + x^T(N)Q_0x(N). \tag{6.1}$$

subject to the process constraints. In terms of the stacked input vector

$$U^T = [u^T(0) \cdots u^T(N-1)] \in \mathbb{R}^{Nn_u}$$

the optimization problem that is to be solved on-line, as explained in section 2.4.4, is given by the quadratic program

$$\min_U \{U^T HU + 2x_0^T G_x^T U\} \tag{6.2}$$

$$\text{subject to } KU \leq k \tag{6.3}$$

where both the input and state constraints are described by one set of linear inequalities defined by $K \in \mathbb{R}^{n_c \times Nn_u}$ where n_c is the number of constraints.

In a constrained optimization problem the Lagrange multiplier λ indicates whether a specific constraint is active or not. By definition all elements of the Lagrange multiplier in a constrained optimization problem are nonnegative. If the element $\lambda_i = 0$ then the i^{th} constraint is inactive and if $\lambda_i > 0$ the corresponding constraint is active.

A possible way to estimate the Lagrange multiplier is to find an approximate solution of the dual quadratic program. The quadratic program (2.24) has a dual (2.14) which is an optimization over the Lagrange multiplier λ given by

$$\min_\lambda \{\lambda^T P\lambda + d^T\lambda\} \text{ subject to } \lambda \geq 0 \tag{6.4}$$

with $P = KH^{-1}K^T$ and $d = KH^{-1}G_x x_0 + k$.

This optimization problem has n_c free variables which in generally much more than the number of free parameters. However, the constraints are merely nonnegativity constraints for which dedicated algorithms exist that are faster than algorithms for general linear

constraints [40][49].
To obtain an approximate solution of this optimization problem, Hildreths quadratic programming procedure can be used [84].

Definition 6.2.1 Hildreth's quadratic programming procedure. *Let the dual quadratic program be given by* (6.4) *and let the optimal solution of this quadratic program be given by* λ^* *with the corresponding solution to the primal problem* $U^* = -H^{-1}(G_x x_0 + K^T \lambda^*)$. λ^* *can be obtained with successive optimization of each element of* λ *separately. This can be expressed explicitly as*

$$\lambda_i^{k+1} = max(0, w_i^{k+1})$$

where $w_i^{k+1} = -\frac{1}{p_{ii}} \left(d_i + \sum_{j=1}^{i-1} p_{ij} \lambda_j^{k+1} + \sum_{j=i+1}^{n} p_{ij} \lambda_j^k \right)$. *Here index* i *denotes the element of the vector and* $k+1$ *denotes the updated elements of the Lagrange multiplier while* k *denotes the values that are not updated. Finally,* p_{ij} *is the element* (i, j) *of matrix* P *in* (6.4). *A full cycle of these calculations that updates all elements of the Lagrange multiplier* λ^k *to the updated values* λ^{k+1} *is denoted as* Hildreth iteration.

An iteration of Hildreth iterations as defined in definition 6.2.1 converges monotonically to the optimal Lagrange multiplier in a finite number of iterations as explained in [84]. In each iteration the cost function of the dual is decreased due to the optimization problems used to find w_i [84].
A distinction can be made between certainly active, certainly inactive and uncertain constraints by inspection of the elements of w rather than λ using the following procedure. The three sets of constraints are denoted with $\mathcal{A}_c, \mathcal{A}_i, \mathcal{A}_u$ respectively.

Algorithm 6.2.2 Estimation of active constraints.

0. *Check whether the unconstrained solution is feasible. If so terminate else go to step 1.*

1. *Start with a feasible Lagrange multiplier. Take the Lagrange multiplier from the previous time instant* λ_{t-1}^* *and shift the corresponding input and output constraints one sample back in time.*

2. *Apply* n_H *Hildreth iterations to obtain an approximation of the Lagrange multiplier.*

3. *Set the set of certainly active constraints* $\mathcal{A}_c$ *to those constraints that coincide with a value* $w_i > \delta$. *Set the possibly active constraints* $\mathcal{A}_u$ *to those constraints that coincide with a value* $|w_i| \leq \delta$ *and the set* $\mathcal{A}_i$ *to those that coincide with* $w_i < -\delta$ *where* δ *is a positive real value.*

This algorithm can be used to estimate the set of active constraints which is used to define the low dimensional subspace as discussed in the next part of this section.

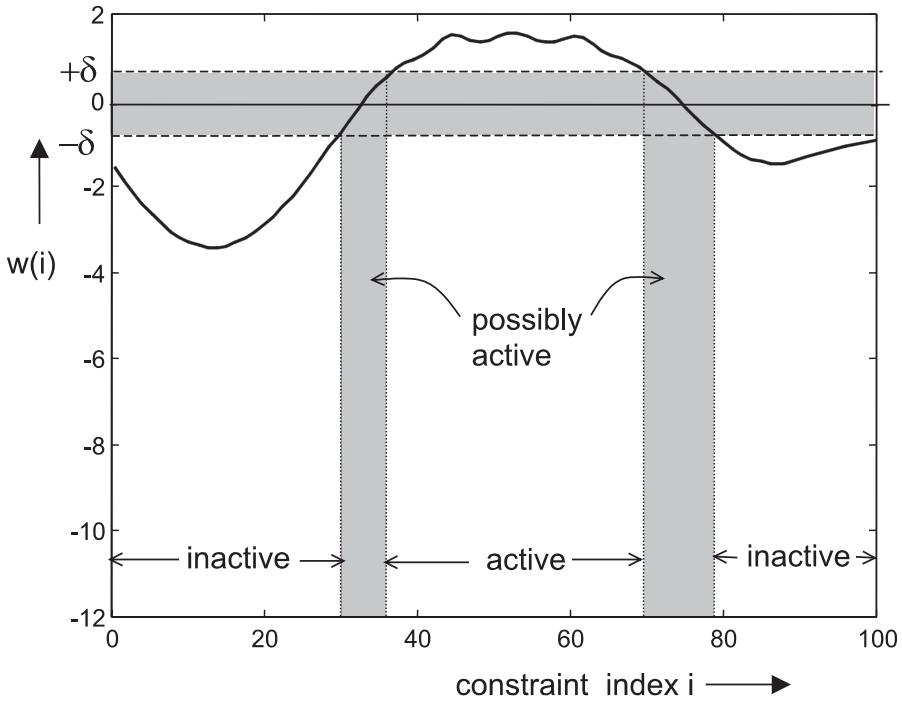

Fig. 6.1: *Illustration of the δ-region that is used to make the division in the set of constraints. The solid line is the extended Lagrange multiplier w of which the positive elements are equal to the Lagrange multiplier.*

Step 2: Construction of a parametrization

Let the set of active constraints at the optimal solution of the quadratic program be given by $LU = l$. Then the optimal input trajectory can simply be obtained by minimization of the Lagrangian which is equivalent to solving the set of equations

$$\begin{bmatrix} H & L^T \\ L & 0 \end{bmatrix} \begin{bmatrix} U \\ \lambda \end{bmatrix} = \begin{bmatrix} -G_x x_0 \\ l \end{bmatrix} \tag{6.5}$$

So, once the active set of constraints is known, the optimization problem is simple. In fact the optimum can be calculated analytically. Generally the active set is not known but let us assume that we are certain that the set of constraints $L_1 U = l_1$ denoted with $\mathcal{A}_c$ is active at the optimal solution and that we are uncertain whether the constraints $L_2 U = l_2$ denoted with $\mathcal{A}_u$ are active or not. Further assume that the other constraints are inactive.

Now, we can define an optimization problem that only tries to find out whether the uncertain constraints are active or not. This can be done by choosing as free parameter θ, the bound l_2. This results in a set of equations given by

$$\begin{bmatrix} H & L_1^T & L_2^T \\ L_1 & 0 & 0 \\ L_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -G_x x_0 \\ l_1 \\ \theta \end{bmatrix} \tag{6.6}$$

All optimal input trajectories that satisfy the assumption about the constraints can be described by

$$U(\theta) = [\phi_x \ \phi_l \ \phi] \begin{bmatrix} x_0 \\ l_1 \\ \theta \end{bmatrix} = \varphi + \phi\theta \tag{6.7}$$

where ϕ_k is calculated from the inverse of the partitioned matrix on the left hand side of (6.5) with

$$[\phi_x \ \phi_l \ \phi] = [I \ 0 \ 0] \begin{bmatrix} H & L_1^T & L_2^T \\ L_1 & 0 & 0 \\ L_2 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} G_x & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

Note that the matrix inversion is only possible provided $[L_1^T \ L_2^T]^T$ has full column rank which is not guaranteed here. A selection of the rows of this matrix must be made to comply with this condition which can be done efficiently with QR factorization [51].
The subset (6.7) is a linear variety. Therefore, the origin is not necessarily contained in this set if $\varphi \neq 0$. For some situations the origin is a guaranteed feasible point of the input constraints. If the parametrization (6.7) is used, no guarantee can be given about feasibility for these situations. In this situation feasibility can be guaranteed by extension of the space to a subspace generated by the columns of

$$\phi_{ext} = [\phi \ \varphi]$$

provided φ is independent of the columns of ϕ. This increases the dimension of the space with only one, but improves the flexibility and feasibility.
Now we can specify the following optimization problem

$$\min_{\theta} \{U^T(\theta) H U(\theta) + 2x_0^T G_x^T U(\theta)\} \tag{6.8}$$

subject to the constraints and input parametrization (6.7). This is still a quadratic programming problem. An important property of this parametrization is the following.

Lemma 6.2.3 *Let the division of the constraints be such that that $\mathcal{A}_c$ contains only constraints that are active at the optimal solution and the remaining active constraints at the optimal solution are contained in $\mathcal{A}_u$. Then the optimal solution to the full optimization problem* (2.24) *and the reduced optimization problem* (6.8) *are identical.*

Proof: The reduced order optimization problem is equal to the full order optimization problem with a restriction on the space over which is searched. The unrestricted optimal solution is contained in this space and will therefore also be the solution to the reduced order problem. □

In the next section some properties of the full algorithm are discussed.

6.3 Algorithmic properties

The algorithm has two important tuning variables, δ and n_H, that can be used to trade-off computational speed and accuracy. A larger value for δ gives a larger set of uncertain constraints and as a consequence a larger computational load for the reduced QP. On the other hand, the chance becomes larger that the input parametrization contains the optimal input trajectory or a very good approximate thereof which gives an improved accuracy. A larger value for n_H increases the computational load but also increases the chance that the correct division of the constraint set. If n_H is such that the Hildreth iteration has converged or the sign of Lagrange multipliers is correct, no optimization is needed and the optimal input trajectory can simply be calculated with (6.5). Both tuning parameters provide a way to trade off computational complexity against accuracy: n_H in Step 1 and δ in Step 2 of the Main Procedure.
The computational price of the strategy proposed in this chapter consists mainly of two parts: computation of the division of the set of constraints in Step 1 and solution of the low order QP in Step 2. First, the proposed algorithm for the estimation of the active constraints has complexity $O(N)$. This is because a finite number of computations is necessary for the correction of each element of the Lagrange multiplier while the number of constraints is an integer times the number of free variables, i.e. input upper and lower bound, input rate constraints and (possibly) output upper and lower bound. Secondly, the low order QP has a computational price of $O(n_\theta)$ if an interior point algorithm is used. As a result the overall complexity of the algorithm is of order $O(N)$, which is comparable to interior point algorithms.
The computer storage that is needed for the described approach is larger than active set QP solvers because the dual problem has to be stored. The storage requirements will be similar to primal-dual interior point methods where also the dual problem is stored.

6.4 Model predictive control with on-line input parametrization

In the previous sections it is explained how the computational efficiency of quadratic programming can be increased by selection of a low dimensional subset of the full parameter

space followed by an optimization over this low dimensional subset. In this section a model predictive control algorithm is presented that applies this quadratic programming approach.

Algorithm 6.4.1 *Optimize at each time instant the quasi infinite receding horizon controller cost function given by*

$$J(u(\cdot)) = \sum_{t=0}^{N-1} \{x^T(t)Q_1x(t) + u^T(t)Q_2u(t)\} + x^T(N)Q_0x(N).$$

with prediction horizon N such that after this time instant no constraints are active and $Q_0 = Y$ the solution of the Lyapunov equation

$$(A - BF)^T Y(A - BF) + F^T Q_2 F + Q_1 = Y.$$

where F is the LQ optimal state feedback (2.18)*. The initial state x_0 is either the measured state vector (full information case) or a prediction thereof (partial information case). Parametrize at each time instant the stacked input $U^T = [u^T(0) \cdots u^T(N-1)]$ over the finite horizon N linearly by*

$$U(\theta) = \phi\theta + \varphi$$

with ϕ, φ given in (6.7) *where the division of the constraint set in $\mathcal{A}_u, \mathcal{A}_c$ is determined with Procedure 1.*

This model predictive control algorithm can be viewed as a standard quasi infinite horizon model predictive controller with an alternative quadratic programming approach, but it is also a model predictive controller in which at each time instant the input parametrization is updated. By selection of the tuning variables δ, n_H a trade-off can be made between controller performance and computational efficiency.
Some properties of this controller are given in the following corollary.

Corollary 6.4.2 *The controller of algorithm 6.4.1 is:*
1. equivalent to LQ control if no constraints are active,
2. equivalent to constrained LQ control provided that the division of the constraint set is correct.

Proof: Property 1 follows directly from Procedure 1 and the fact that a quasi infinite horizon cost function is used. Property 2 follows directly from lemma 6.2.3. □

In the following proposition it is proven that the control algorithm is nominally stabilizing in the presence of constraints.

Proposition 6.4.3 *The predictive control strategy given in algorithm 6.4.1 is globally asymptotically stable if and only if the optimization problem* (6.8) *is feasible*

Proof: The proof follows from a Lyapunov argument. The value of the cost function at t is always lower than or equal to the one at $t-1$ because the Lagrange multiplier at $t-1$ is used as starting point for the computation at time t. □

The constrained infinite horizon model predictive control approach described here is similar to the one presented in [141] but it differs in the applied parametrization of the input. In their approach constrained LQ optimality is obtained at each time instant possibly at the cost of a large computational load. Here optimality at each time instant is sacrificed, i.e. optimality is obtained only under the condition that the division of the constraint set is correct, in return for a gain in computational efficiency. The separately property of constrained stability is preserved with the presented approach.

6.5 Simulation example

In this section two simulation examples are given. In the first example the properties of model predictive control with on-line input parametrization are illustrated. In the second example a specific quadratic programming problem is solved with the active set method, an interior point algorithm and with the approach that is presented in this chapter which consists of a subset selection followed by a low dimensional quadratic program. Especially the properties of the Hildreth iteration used to estimate the active constraints are investigated.

Example 1

The system that is considered is a highly oscillatory nonminimum-phase system given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 + 3.8630z^2 - 2.6426z + 0.8084}$$

The open-loop step response is given in figure 3.6. The input is constrained by $-0.275 \leq u(t) \leq 0.275$. To obtain integral action in the controller the system is augmented with an integrator at the input. Simulations are performed on a PC with a 233 MHz Pentium processor.

If MPC is applied to this system it is necessary for satisfactory performance to take a prediction horizon that is long enough to incorporate at least one full period, i.e. $P = 100$. If a pulse parametrization is applied, the choice of the control horizon M is critical for this system. Reduction of the control horizon down to $M = 75$ is possible without considerable loss of performance, further reduction gives bad performance due to the slow oscillation. Therefore the computational burden is relatively large for high performance MPC of this system.

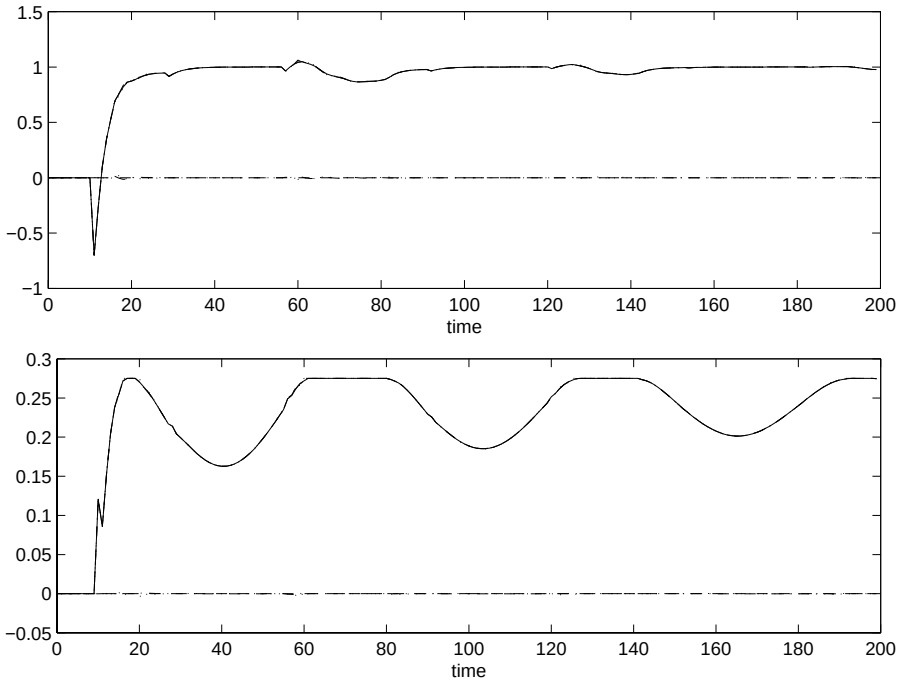

Fig. 6.2: *Output (top) and input (bottom) of the closed-loop step response with constrained LQ control with P=100 (solid) and Algorithm 1 $\delta = 0$ and $n_H = 2$ (dashed).*

In figure 6.2 the closed loop response is given for the proposed Algorithm 6.4.1 together with constrained LQ optimal control. The two simulations practically coincide. A more precise indication of the trade-off between accuracy and complexity is given in table 6.1.

n_H	computation time 200 simulation steps (CPU s)	error $\|y - y_r\|^2$
1	22.55	3.1×10^{-3}
2	25.63	1.2×10^{-3}
5	38.49	8.428×10^{-9}

Table 6.1: *Computation time of 200 simulation steps to illustrate the trade-off between performance and complexity for $\delta = 0$ and several values for n_H.*

If the active set quadratic programming routine is used, the simulation of 200 steps of the model predictive controlled plant takes 367.348 CPU sec. Hence, for this specific example the computation time can be reduced with more than a factor 10 without loss in control performance.

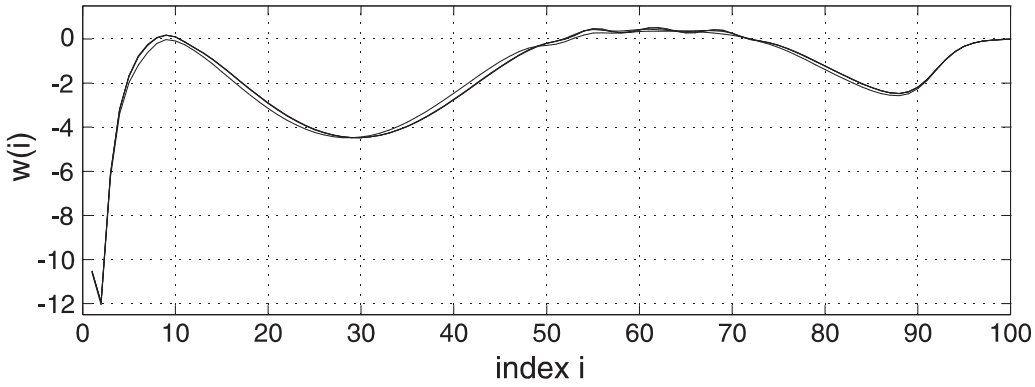

Fig. 6.3: *Evolution of the estimate of w for 2, 5, 10 and 50 iterations.*

Due to the choice for $\delta = 0$ in the proposed algorithm no reduced QP has to be solved because the set of uncertain constraints is empty. Clearly, this has a favorable influence on the computational load. This tuning is possible without considerable closed loop performance loss because of the high quality of the estimates of the constraint set. Note that this is obtained with only a few Hildreth iterations. The relatively fast convergence of the Hildreth iterations to an approximate solution of the Lagrange multiplier can also be seen in figure 6.3 where the evolution of w is depicted for several values for n_H.
The active set method is known to be less efficient than e.g. interior point methods for moderate and large size problems. Therefore in the next example an interior point method is used for comparison.

Example 2

Next, a typical optimization problem that must be solved on-line in the model predictive control application of Example 1 is selected to illustrate the convergence properties of Hildreth iterations and the effect of δ on the computational load and the accuracy of the solution. The horizon is taken $P = 100$ which implies a number of constraints equal to $n_c = 200$, upper and lower bound of the input.
In table 6.2 the computation time of several quadratic programming algorithms is given. If the active set quadratic programming routine is used that is implemented in the Optimization Toolbox of Matlab, the simulation a typical optimization time is 4.456 seconds. This computation time could not be decreased using a warm start with the clipped unconstrained input trajectory.
In Table 6.2 also the performance of an interior point algorithm for quadratic programming is given. It is the QP solver implemented in the software package MOSEK [2] which provides a solution to the optimization problem in 0.66 seconds which is considerably faster than the active set method.
With the approach presented in this chapter the computational load can be decreased even further to 0.56 seconds to find the exact solution. This is an improvement of approx-

Method	tuning	computation time QP (CPU sec)	error $\|u - u_{opt}\|^2$
QP act.set	$P = 100$ $n_c = 100$	4.46	0
QP int.pt.	$P = 100$ $n_c = 100$	0.66	0
Alg. 1	$n_H = 5$ $\delta = 0.05$	0.56	0
Alg. 1	$n_H = 2$ $\delta = 0$	0.47	1.78×10^{-7}

Table 6.2: *Comparison of the computational load for a typical quadratic programming problem with* $P = 100$ *degrees of freedom in the input trajectory* u *and* $n_c = 100$ *constraints (upper bound on the input). The applied methods are the active set method, interior point method and the described algorithm 1 for several values of* δ *and* n_H*.*

imately 15 % with respect to the interior point algorithm. For tuning $\delta = 0.05, n_H = 5$ the optimal solution is obtained but if a small error of order 10^{-7} is permitted a computation time of 0.47 seconds can be obtained with tuning $\delta = 0, n_H = 5$. Clearly, by selection of a small number of degrees of freedom prior to optimization, the overall optimization complexity can be reduced.

In table 6.3 some computational statistics are given for $\delta = 0$ where n_H is the number of Hildreth iterations, n_F is the number of incorrect constraints, J is the value of the cost function, Total is the total CPU time in seconds which is the sum of the CPU time of Step 1, Step 2 and Step 3, finally n_θ is the dimension of the subset constructed in Step 2. Similar computational statistics are given in table 6.4 for $\delta = 0.05$.

From the tables it becomes clear that if an exact solution is required, $n_H = 5, \delta = 0.05$ are appropriate values. This provides the solution in 0.4010 seconds. From the simulation result in figure 6.2 it is clear that a rough solution is sufficient in model predictive control. Therefore $n_H = 2, \delta = 0$ is an appropriate tuning for near optimal constrained performance. With this tuning the algorithm provides an approximate solution in 0.24 s. An increase in efficiency of more than a factor 20.

The gain in efficiency compared to active set methods is larger if more constraints are active at the optimal solution. An illustration of this is the calculation of the optimal trajectory for tighter constraints, i.e. $-0.17 \leq u(t) \leq 0.17$. Now 71 of the 100 degrees

n_H	n_F	J	Total CPU s	Step 1 CPU s	Step 2 CPU s
2	3	94.1255	0.2400	0.0900	0.1500
5	1	93.9941	0.4010	0.2310	0.1700
10	1	93.9941	0.6210	0.4600	0.1610
15	0	93.9897	0.8420	0.6810	0.1600
50	0	93.9897	2.3930	2.2530	0.1400

Table 6.3: Computational statistics for $\delta = 0$.

of freedom are constrained at the solution. The active set method requires $T = 10.16$ seconds to obtain the exact solution. The proposed algorithm with $n_H = 10, \delta = 0$ also provides the optimal solution only in 0.431 s. An increase in efficiency with a factor more than 20 compared to the active set method.
Note however, for the software implementation of the interior point method an optimization has taken place with respect to numerical properties. On the other hand, for the simulations above the approach presented in this chapter is implemented in a straightforward way, therefore another algorithmic implementation can improve the computational efficiency. Therefore, further development of the algorithm appears useful.

n_H	n_F	J	Total CPU s	Step 1: CPU s	Step 2: CPU s	Step 3: CPU s	n_θ
2	2	93.9941	0.3500	0.0900	0.1500	0.06	4
5	0	93.9897	0.5610	0.3310	0.1700	0.06	4
10	0	93.9897	0.6810	0.4600	0.1610	0.06	4
15	0	93.9897	0.9020	0.6810	0.1600	0.06	4
50	0	93.9897	2.5030	2.2530	0.1400	0.06	4

Table 6.4: Computational statistics for $\delta = 0.05$.

6.6 Discussion

In this chapter an approach is presented to improve the efficiency in model predictive control by choosing on-line a parametrization of the input trajectory with a small number

of degrees of freedom. The degrees of freedom are selected such that the optimal solution differs as little as possible from the solution of the problem with a free parametrization of the input. The approach consists basically of two steps. In the first step, the set of constraints is divided in three sets: a set with constraints that are certainly inactive, certainly active and possibly active. This is done on the basis of an estimation of the active constraints using Hildreth's QP procedure. Next, a subset of the full input space is specified that contains all optimal input trajectories that comply with the division in the three sets of constraints. In the second step a low-order optimization problem is solved where the input can vary over this subset, leading to a large decrease of the computation time.

This procedure can be viewed as an overhead over an optimization algorithm, e.g. an interior point algorithm, to speed up the computation for large scale problems by a quick selection of a limited number of degrees of freedom. If the solution to the problem is not required to be exact, like in the case of model predictive control, the procedure can be sufficient. If optimality is required the procedure can be repeated until optimality is obtained or the solution can be used as starting value for a further optimization.

The basic idea is used in a constrained infinite horizon model predictive controller that attains LQ optimal control if no constraints are active, constrained LQ optimal control if the division of the constraint set is correct and is closed loop stable. The tuning parameters in the algorithm provide a way to trade-off computational complexity and performance.

On the basis of a simulation example it is shown that the proposed approach can provide a large gain in efficiency compared to an active set strategy to solve the optimization.

Chapter 7

Identification, optimization and control of a cracking unit

7.1 Introduction

Industrial systems, e.g. (petro-)chemical processes and systems encountered in the iron and steel industry, often have a high complexity both in terms of the number of inputs and outputs and the system dynamics. The systems possess dynamics in the (very) low frequency range as well as the (very) high frequency range. The experiment time is preferably as short as possible to avoid unnecessary product loss although the signal to noise ratio can be poor. Experiment design is moreover limited by process constraints. Identification of such systems on the basis of limited experimental data is a difficult task as explained in chapter 1 and 2. In chapter 3 and 4 new theory is developed to identify multivariable systems which is particularly suitable for this type of systems.
Also model predictive control of these processes is a difficult task, especially if the performance requirements are high. In current practice the principle of division of time scales is used to justify the fact that the second to second behaviour of the system is controlled with a basic control layer and the minute to minute dynamics with model predictive control. In this work it is shown that it is technologically possible to perform constrained multivariable control on a fast sampling rate and that it has advantages to do so. However, utilizing less basic controllers has a price. A wider dynamic range must be covered by the model predictive controller. To provide high performance to the system despite the large dynamic range, usually a large optimization complexity is necessary to meet the specifications. For this efficient model predictive control algorithms are required. In chapter 5 and 6 new approaches are developed to improve the efficiency such that it can be utilized in the above mentioned situation.
In this chapter multivariable system identification and model predictive control of an industrial type system is discussed to indicate how the developed approaches can contribute

to an improvement in identification and model predictive control. Also static optimization is performed to find an improved setpoint and to test the regulatory behaviour of the control system. For this purpose a nonlinear simulation model of a Model IV fluidized catalytic cracking unit is used. This model is presented in [89] and is intended as a benchmark problem for identification, control and optimization.
The fluidized catalytic cracking (FCC) process is used to crack crude oil into lighter and more valuable components. The overall economic performance of a refinery largely depends on the economic operation of the FCC unit [148]. Therefore accurate modeling, control and optimization of this process is of large importance.
Several challenging phenomena are present in the system. The system is multivariable with large interaction between the several input/output-channels. Characteristic for this system is the combination of fast and slow physical phenomena. Both frequency ranges need to be estimated accurately for high performance control design. This means, however, that long data sequences at a high sampling rate need to be used to capture both slow and fast phenomena in the data.
In section 7.2 the overall strategy of identification, optimization and control is described. In section 7.3 the system is described in more detail. Next, in 7.4 it is described how identification with orthogonal basis functions as described in chapter 3 and 4, can be utilized to obtain an accurate model for this system. The identification results are partly previously published in [33][34]. In 7.5 static optimization is utilized to determine a new setpoint for the system that is on the one hand an improvement with respect to the economics of the plant and on the other hand provides sufficient robustness for plant model mismatch and disturbances. In 7.6 model predictive control with system based input parametrization as described in chapter 5 is applied to bring the system from the initial setpoint to the improved setpoint. Finally, 7.7 concludes the chapter.

7.2 Overall strategy

The applied strategy for process improvement contains the following steps.

- First, the system is identified around its present working condition $\{u_o, y_o\}$ which provides a linear model of the system dynamics $\hat{G}$, a linear model of the disturbance dynamics $\hat{G}$ and a characterization of an upperbound on the modeling error ΔG.

- Secondly, the information provided by the identification is used to find an improved working condition $\{u_1, y_1\}$ by a static optimization. A cost function is optimized such that a trade-off is made between optimizing the plant economics and providing sufficient robustness for plant model mismatch and disturbances.

- Lastly, a model predictive control algorithm with several parametrization approaches

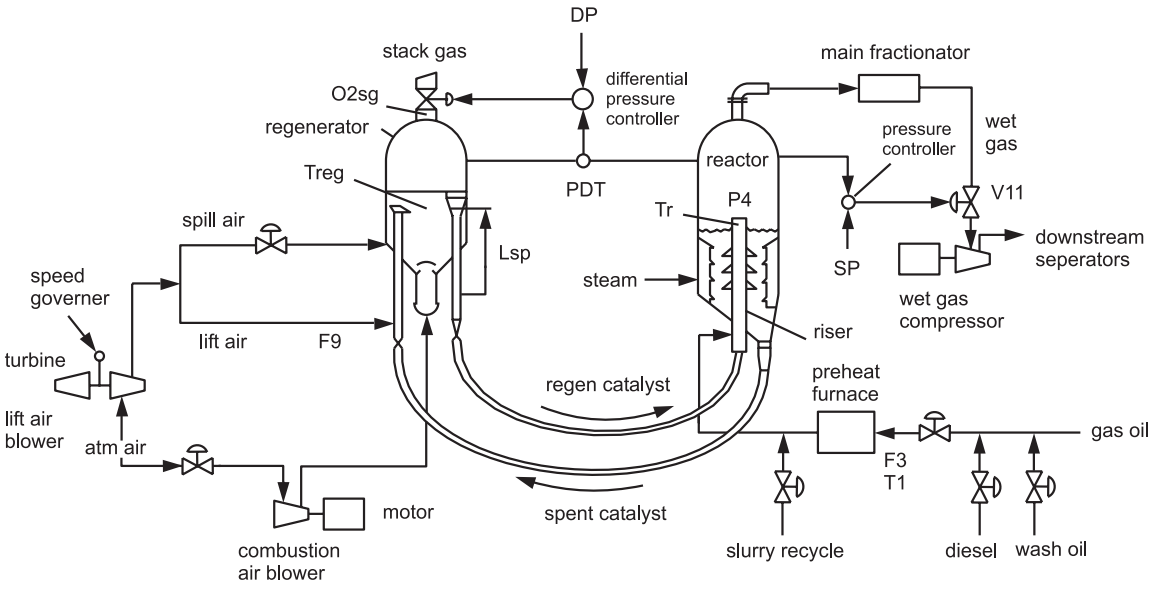

Fig. 7.1: *Schematic of a Model IV fluid catalytic cracking unit.*

is used to lead the system from the initial working condition $\{u_o, y_o\}$ to the new working point $\{u_1, y_1\}$, while satisfying all process constraints.

In the next sections the consecutive steps are further discussed and results are presented.

7.3 The process

The system to be identified is the nonlinear FCCU model described in [89]. In figure 7.1 a flow sheet is given of a typical Model IV fluidized catalytic cracking unit is shown. The system consists basically of two subsystems: the riser or reactor and the regenerator.
In the reactor fresh feed of a blend of oil products with a high boiling point and hot catalyst coming from the regenerator is mixed which induces the cracking reaction which makes the crude oil to fall down into lighter and more valuable components. These components leave the reactor at the top as gas and are separated in the downstream separators. In the reaction the catalyst is contaminated with carbonaceous material (coke). The spent catalyst is transported to the regenerator to be regenerated.
In the regenerator spent catalyst is regenerated by means of air injection provided by the air blowers (figure 7.1). The air injection fluidizes the catalyst and removes the coke by a exothermal reaction. The heat induced by this reaction is used to supply the heat for the endothermic reaction in the reactor.
Hence, no additional heat is supplied to the reactor. Because of this, the reactor and the regenerator are highly coupled. The multivariable system shows large interaction between the several input-output channels.
The system also shows both fast and slow dynamic behaviour. The fast behaviour comes

from flow and pressure phenomena while the slow behaviour stems from the fact that it takes a long time to reach a thermal equilibrium.
A control objective that can be formulated for this system is

1. maximize the total feedrate while maintaining riser temperature as close to its metallurgical constraint as possible to provide a cracking that is "deep" enough such that a large amount of gasoline is obtained.

2. Reject modeling disturbances. The primary disturbances are changes in the feed composition, which is modeled by the coking factor Ψ_f, and ambient temperature which have a cyclical character due to the changes from day to night and vice versa.

The disturbances acting on the system are the following. A measurable disturbance is the ambient temperature $T_{atm}(t)$ and a disturbance that is not measurable is the changing coking factor $\psi_F(t)$ of the incoming fresh feed. The number of inputs and outputs are denoted as n_u and n_y respectively.
The sampling time is chosen according to the fastest time constant and is given by $\Delta T = 10\ s$. The fastest dynamics are due to pressure effects in combination with the basic feedback controller that controls the pressure difference between the reactor and riser.
The simulation example in this chapter is mainly to demonstrate the properties of some of the algorithms presented in the previous chapters rather than to obtain the optimal operating strategy for the process. For identification and model predictive control separate configurations are used in order to improve the emphasis on certain specific challenges in both areas. The following configurations are utilized.

Configuration for identification

The inputs that can be manipulated for identification purposes are given by

$$u(t) = [F_3(t)\ T_2(t)\ F_9(t)\ p_4(t)\ \Delta p(t)]^T$$

where the $F_3(t)$ is the fresh feed flow, $T_2(t)$ is the temperature of the fresh feed flow, $F_9(t)$ is the lift air flow, $p_4(t)$ is the pressure in the riser and $\Delta p(t)$ is the pressure difference between regenerator and riser. The measured output vector is given by

$$y(t) = [l_{sp}(t)\ T_{reg}(t)\ T_r\ O_{2sg}(t)\ V_{11}(t)]^T$$

where $l_{sp}(t)$ is level in the stand pipe of the riser, $T_{reg}(t)$ is the temperature in the regenerator, $T_r(t)$ is the temperature in the reactor, $O_{2sg}(t)$ is the concentration oxygen in the stack gas coming out of the regenerator and $V_{11}(t)$ is the valve position at the point where the wet gas is sent to the main separators. In this setup the local controllers discussed in [89] are assumed to be active. Due to the presence and settings of the local controllers small high frequent dynamics are present which are relatively difficult to identify.

Configuration for optimization and control

In contrast to the configuration applied for identification, in the configuration for optimization and control a large part of the local controllers are not used to avoid adverse effects of these local controllers on the process dynamics. Instead, the model predictive controller controls these loops at a high sampling rate while regarding process constraints. One variable that is controlled locally is the differential pressure ΔP, which is controlled by a PID controller that manipulates the valve V_{14}. This valve is overdimensioned therefore no constraint activation will occur. This local control is used to speed up the initial configuration and decreases in this way the dynamic range. It is common control practice to also control locally the reactor pressure P_4 with the valve V_{11} and the liftair flow rate F_9 with valve V_{lift}. This is not done here because by increasing the economic gain these valves will saturate. If they are controlled locally an additional nonlinearity is introduced in the system that can be circumvented by using model predictive control to control the valves V_{11} and V_{lift} while accounting for the constraints.
The inputs that can be manipulated for control purposes are given by

$$u(t) = [F_3(t)\ F_4(t)\ \Delta p(t)\ V_{lift}(t)\ V_{11}(t)]^T$$

where the F_3 is the fresh feed flow, F_4 is the slurry recycle flowrate, $\Delta p(t)$ is the pressure difference between regenerator and riser, V_{lift} is the lift air valve and V_{11} is the wet gas compressor suction valve. The vector with controlled variables is given by

$$y(t) = [T_{reg}(t)\ T_r(t)\ O_{2sg}(t)\ l_{sp}(t)]^T$$

where $T_{reg}(t)$ is the temperature in the regenerator, $T_r(t)$ is the temperature in the reactor, $O_{2sg}(t)$ is the concentration oxygen in the stack gas coming out of the regenerator and $l_{sp}(t)$ is level in the stand pipe of the riser.
The input constraints are given by

$$\begin{aligned} 0 &\leq u_1 \leq 144 \\ 0 &\leq u_2 \leq 10 \\ -5 &\leq u_3 \leq 2 \\ 0 &\leq u_4 \leq 1 \\ 0 &\leq u_5 \leq 1 \end{aligned}$$

The output constraints are given by

$$\begin{aligned} 1265 &\leq y_1 \\ & \quad y_2 \leq 995 \\ 1.5 &\leq y_3 \\ 0 &\leq y_4 \leq 20 \end{aligned}$$

The control objective must be obtained within the limitations of the system. The sampling in this example is considerably faster than what is common in the process industry nowadays, where usually a sampling time of one or several minutes is applied for model predictive control. This results in the need to identify a model and control a system with a large dynamic range. It is shown that the identification and model predictive control tools developed in this project can be applied to attain an accurate model and high performance control of this type of system.

7.4 System identification

7.4.1 Experiments

The experiment design follows closely the approach advocated in [178]. First preliminary experiments are performed to assess disturbance dynamics, assessment of linearity of the system and to obtain rough system knowledge. The preliminary experiments that are performed are free run and step response experiments.

Free run experiments

First several free run experiments are conducted to assess disturbance dynamics. From these experiments the following observations are made:

- an initial condition disturbance is present, i.e the initial condition of the system is not an equilibrium of the system. This induces transients in the data that are, for some output variables considerable. It is important for the identification approach to account for the transients in the data. Especially the transient in the temperature in the regenerator and the reactor are severe.

- substantial changes of the coking factor occur once every 5-7 hours which has a large influence on the measured output. The coking factor is a disturbance which is not measurable. If parametric identification is performed on data which is disturbed by a changing coking factor, a considerable bias can be expected. For this reason only those parts of the data are used in parametric identification for which the coking factor can assumed to be approximately constant.

It is important in the identification approach to account for these disturbances.

Step response experiments

Step response experiments are performed to assess nonlinearity and obtain a first indication of system dynamics. This knowledge is needed to choose an appropriate sampling

time and experiment length.
The inputs are successively excited with a step function and the five outputs are measured. The experiments are performed with the amplitudes: u_{amp} = [2.4 12.5 0.90 0.11 0.10], $2u_{amp}$, $-u_{amp}$ and $-2u_{amp}$.
The measured step responses are detrended for the initial condition disturbance with the mean of several free runs. By comparison of the results with different step sizes, it can be concluded that the system behaves fairly linearly apart from possible activation of valve constraints. It becomes clear that the system has very fast phenomena, therefore decimating is not possible.

Experiments for parametric identification

The following experiments for parametric identification are performed.

- Pseudo random binary sequence (PRBS) experiments [81]. With this input signal the high frequency behaviour of the system is dominantly present in the data because the data length is approximately 5 hours, while the slowest settling time is approximately 2 hours (in the transfer to the temperature of the regenerator and the reactor).
- Random binary sequence (RBS) experiments. To emphasize the low frequent behaviour more, RBS experiments are used with a low switching probability [150].

An important aspect of multivariable experiment design is that the inputs are as much uncorrelated as possible to keep the identification problem well conditioned. If different realizations of the signals mentioned above are used for the different input channels, this is approximately satisfied.
The length of the experiment is limited first of all due to the change in coking factor which occurs every 5 hours. Also the experimentation time is intentionally chosen short compared to the slowest time constant to decrease the experimental cost. The experimental time of 5 hours is relatively short, because the slowest settling time is approximately 2 hours (in the transfer to the temperature of the regenerator and the reactor). A rule of thumb is that for an accurate identification of the low frequent behaviour an experimentation time of at least 10 times the settling time is advisable. Hence, the short length of the data sets induces a challenge for the identification algorithm.

7.4.2 Parametric identification and validation

The aim of the identification approach is to identify a model which accurately describes all the data that is present: the response to the PRBS signal which contains the high frequent behaviour more than the low frequent, the response to the RBS signal which emphasizes the low frequent behaviour more and the step response data with a major

emphasis on low frequent dynamics.
The approach followed here involves basically three steps:

1. a realization algorithm based on step response data is used to obtain a rough parametric model of the system.

2. an orthonormal basis function model is identified using the parametric model obtained in the first step to generate an initial basis. The model is iteratively improved.

3. The previous steps are performed for five multi-input/single-output (MISO) problems. In the last step a full multivariable model is estimated with a basis generated by the identification results of the previous step.

These steps are describe in the sequel of this section.

Step response realization algorithm

First the realization algorithm described in [155] is used to obtain a state-space description directly from the step response coefficients. The algorithm is similar to the well known Ho-Kalman algorithm [61][71] but does not act on the Hankel matrix with pulse response coefficients but with the Hankel matrix with step response coefficients. This has the advantage that no discrete differencing has to be applied to the step response data to obtain impulse response coefficients, thus avoiding a large increase in the influence of the disturbances. An important property of this approach is that the emphasis of the obtained models is more on the low frequent behaviour than with the algorithm of Ho-Kalman.
The identification of the MIMO model is split into 5 separate MISO identification problems. The reason for this is that the identification problem becomes computationally more tractable. Also the input and output weighting and compensation for time delays can be performed on each transfer function separately. This flexibility is necessary to obtain accurate models.
The order of the estimated models are: from u to $y_1(t)$ 10th order, 6th to y_2, 9th to y_3, 10th to y_4 and 6th order to y_5. This makes a 41st order MIMO model.
The MISO realization models describe the step response data accurately. However, the models are not capable of predicting the output of the PRBS and RBS data well.

ORTFIR identification

The system is identified with prediction error identification with an ORTFIR model structure as described in chapter 3. For each MISO transfer the corresponding part, $\{A, c_i\}$, is taken of the $\{A, C\}$ matrix of the initial model, obtained with the realization algorithm. The $\{A, C\}$ rather than the $\{A, B\}$ variant is chosen because the number of unknown

parameters that is estimated in the former is larger which increases the model flexibility. Also, the $\{A, B\}$ matrix of a MISO model contains information about the Kronecker indices that may be different from that of the system, which can affect the estimate negatively as explained in section 3.3. Finally, in the $\{A, C\}$ variant the initial conditions can be estimated simultaneously with the system dynamics while the optimization problem remains a least squares problem, as explained in section 3.5.4. The prior information of the $\{A, C\}$ pair can be used to construct a set of basis functions (3.18) yielding the model structure given by

$$G(z, \theta) = C_e(zI - A_e)^{-1}\Theta = \frac{1}{det(zI - A_e)} C_e adj(zI - A_e)\Theta \tag{7.1}$$

with A_e, C_e as defined in (3.19). In the expansion a repetition of the dynamics is used as described in section 3.3.
Several extensions to the basic procedure are applied that are discussed next.

Estimation of initial conditions

In the measured data of the FCCU a transient is present due to an initial condition that is not a stationary working point. To account for this, the initial condition is estimated simultaneously with the system dynamics. This can be done without loosing the linear regression structure as follows. The model structure is extended to

$$\hat{y}(t, \theta) = (D(\theta) + C_e(qI - A_e)^{-1}B(\theta))u(t) + C_e A_e^{t-1} x_0 \tag{7.2}$$

where $\{A_e, C_e\}$ are selected a priori and $D(\theta), B(\theta)$ and $x(0)$ are the parameters that are to be estimated from the data. This boils down to solving the least-squares optimal parameter vector for the overdetermined set of equations

$$Y = [\phi \ \phi_{x0}] \begin{bmatrix} \theta \\ x_0 \end{bmatrix} := \phi_{ext}\theta_{ext}$$

where the t^{th} block-row of ϕ_{x0} is given by CA^{t-1}. Estimation of initial conditions can be used to reduce the bias due to unknown initial conditions at the expense of an increased variance.
The estimated transient of the initial condition and the measured output for the reactor temperature are given in figure 7.2. Clearly, the transient due to the nonstationary initial condition is fitted accurately.

Enforcement of the static gain

The low frequent and static behaviour of the system is barely present in the RBS data due to the relatively short data length compared to the slowest time constant. Therefore

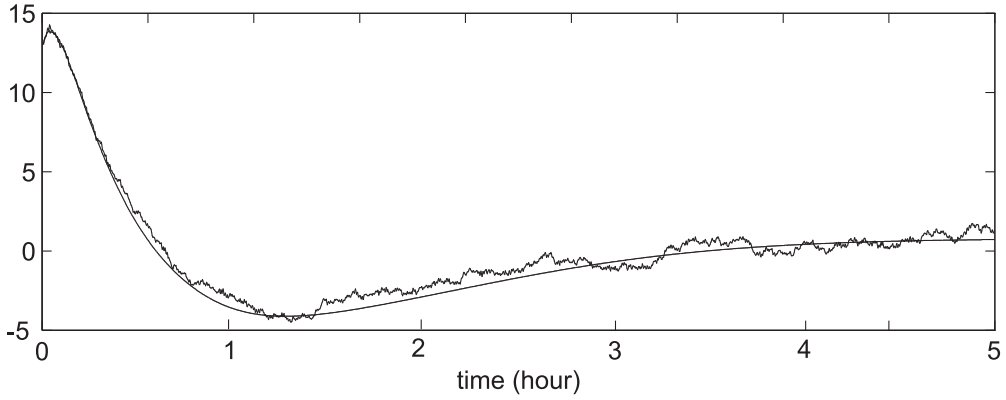

Fig. 7.2: *Measured data $T_r(t)$ with RBS experiment (dotted) and the estimated initial condition contribution (solid).*

the static gain of the estimated models can be inaccurate. This effect can be seen clearly in figure 7.3. To remedy this, the static gain is enforced on the model by means of a constraint, that is linear in the parameter vector. Therefore the linear regression structure is preserved. The static gain of the model (3.11) is given by

$$K_{ss}(\theta) = D(\theta) + C(I - A)^{-1}B(\theta) =: Q\theta \tag{7.3}$$

Any static gain K_{ss} can be enforced on the estimated model by using the Lagrangian of the constrained optimization problem, discussed in section 3.5.6. This boils down to solving

$$\begin{bmatrix} \phi_{ext}^T\phi_{ext} & (Q\ 0)^T \\ (Q\ 0) & 0 \end{bmatrix} \begin{bmatrix} \theta_{ext} \\ \lambda \end{bmatrix} = \begin{bmatrix} \theta_{ext}^T Y \\ K_{ss}^T \end{bmatrix}$$

where λ is the Lagrange multiplier. This is uniquely solvable because the matrix on the left hand side is square and invertible.

The prior knowledge required for the specification of the constraint is obtained from the step response data. The constraint is enforced on the model such that the steady-state gain is equal to the specified one. However, the steady-state gain taken from the step response data is not accurate; therefore possibly unnatural behaviour is enforced on the model. To alleviate this, *soft* constraints are used, which are constraints that can be violated. A soft constraint can be implemented as in (3.50) by adding an additional term to the prediction error cost function identification which penalizes a violation of the steady state equation 7.3.

Figure 7.3 shows the measured step responses, together with the step response of the model resulting from applying no static state constraint as well as from using a soft constraint. The model with the soft constraint fits the measured step response well, while the model with no constraint has a considerable deviation in the steady state gain.

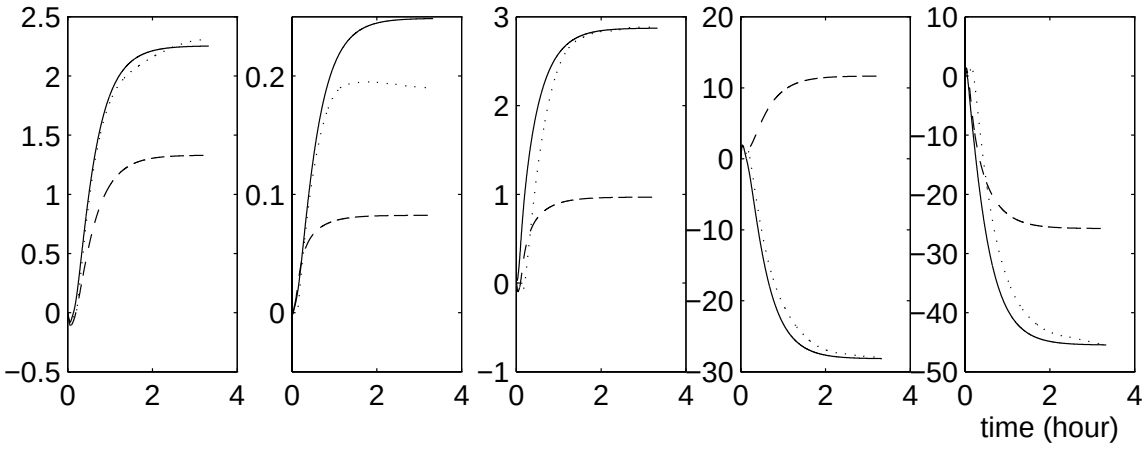

Fig. 7.3: *Measured step response for the input $u(t)$ and the output $T_r(t)$ (dotted). Step responses of the estimated models with no static gain constraint (dashed), and a with a soft constraint (solid).*

Iterative model enhancement

For further improvement of the model, the iterative scheme of ORTFIR identification and model reduction is applied, which is presented in section 4.3.2. In this iteration the following steps are applied:

Step 1. generate basis functions,

Step 2. estimate a high order model with the ORTFIR model structure (3.11),

Step 3. reduce the high order model with a linear model reduction technique, and use the reduced order model to generate a basis in the first step.

This iteration is applied to the data of the cracking unit with balanced reduction, described in section 4.2.1, as reduction method. With this approach the criterion value can be improved considerably. In figure 7.4 the minimal criterion value is given for a number of iterations for the estimation of the transfer between the input and the temperature of the reactor $T_r(t)$. From the figure 7.4 it can be concluded that the criterion value is decreased during the iterations.
In each iteration the low and high model orders are $n_b = 5$ and $nn_b = 20$ respectively. The high order is chosen such that all dynamical phenomena are incorporated in the model. This can be assessed by inspection of the estimated expansion coefficients $L_i(\theta)$. In each iteration a repetition of $n = 4$ times the basis poles is sufficient for an accurate estimation of this transfer. Equivalent to the impulse response coefficients, these coefficients go to zero for stable systems for high enough values of the number of repetitions n. An example is given in figure 7.5.

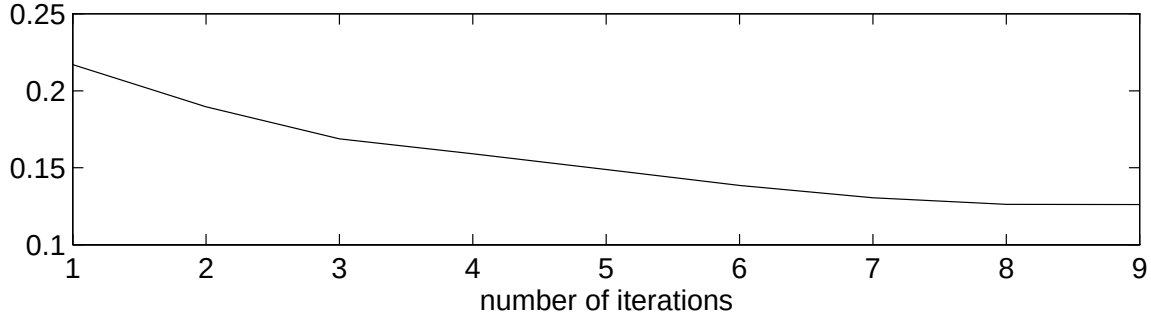

Fig. 7.4: *Value of the criterion function for 9 iterations (low model order: 5, high model order: 20) for the estimation of the transfer between $u(t)$ and $T_r(t)$.*

The aim of the iteration is to concentrate the energy of the estimated model in the first few expansion coefficients such that a low order model can be obtained. This trend is indeed observed during the iterations but can in general not be guaranteed.

Conditioning of the optimization problem

To calculate the optimal parameter vector the Toeplitz matrix $\phi^T\phi$, given in (3.13), needs to be invertible. This implies that ϕ must have full column rank to obtain an estimate without numerical problems. There are several reasons why this may not be the case as also explained in section 3.5.2.
First of all, this can occur if the dynamics present in the basis functions is slow compared to the data length. In that case the inputs filtered with the basis functions forming the columns of ϕ may not be independent. This can be detected by inspection of the impulse responses of the basis functions. If the impulse response has considerable energy outside the time interval given by the data length, the orthogonal basis functions are not orthogonal on the finite time interval and numerical problems are likely to occur. This effect will be denoted the shifting of the basis functions outside the data window.
Secondly, if a high number of repetitions of the basis dynamics n is used, the energy in

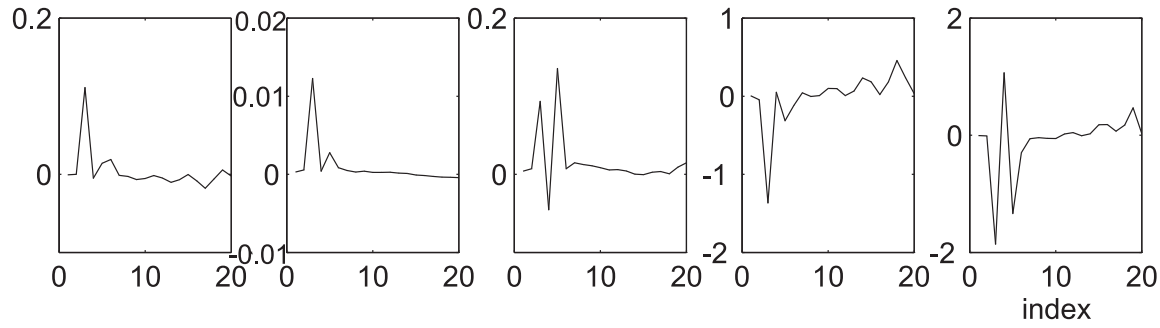

Fig. 7.5: *Generalized impulse responses of the estimated model with soft constraints for the transfer from $u(t)$ to $T_r(t)$.*

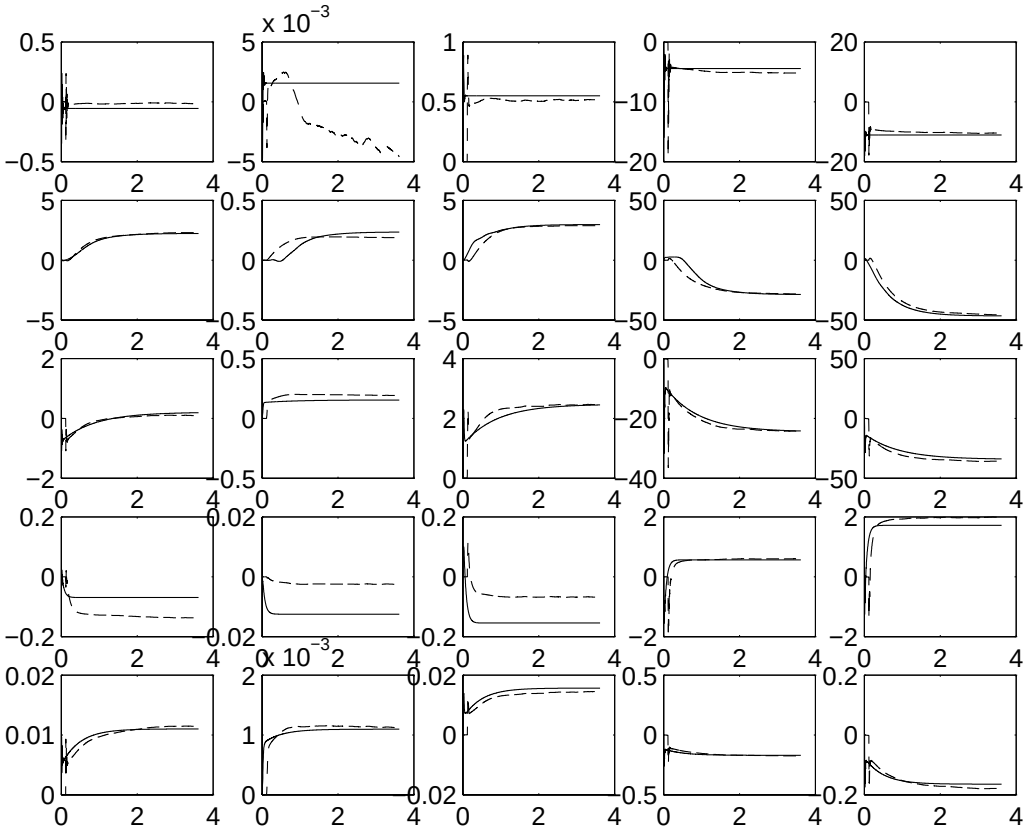

Fig. 7.6: *Step responses of the 33th order multivariable model (solid), and the undisturbed step response (dashed).*

the impulse response of the basis functions shifts to later time instances as is illustrated in figure 3.3. This makes the basis functions shift out of the data window, resulting in the loss of orthonormality and possibly a badly conditioned optimization problem.
To avoid numerical problems, the following strategy is followed. Because slow dynamics is present in the basis functions, the number of repetitions is restricted to $n = 1$. Fast dynamics is added to the dynamics of the basis functions to add extra flexibility in the model. The added dynamics can be any set of stable poles. In the identification of the FCCU, poles are added in the origin comprising a fixed denominator model (3.17).
The part of the regression matrix regarding the initial conditions ϕ_{x0} has another character than the regression matrix for the dynamic part ϕ: the columns of the first consists of transient responses and the columns of the second of responses to signal with mean value zero. Due to this difference the number of repetitions n that can be used in ϕ_{x0} is larger than that can be used in ϕ before bad conditioning occurs. Therefore, the number of repetitions of the basis functions in ϕ_{x0} is taken to be $n = 3$ to give the estimation of the transient additional flexibility.

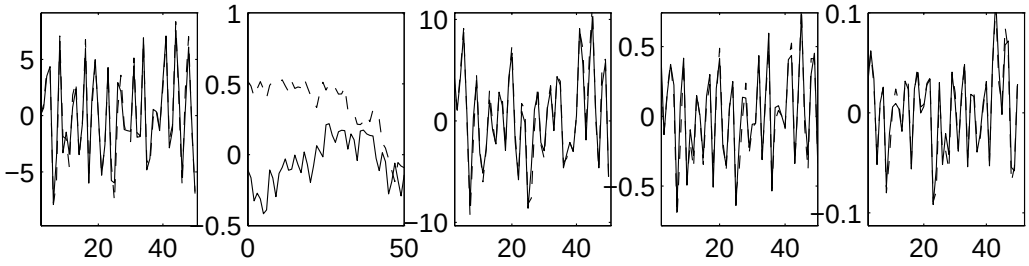

Fig. 7.7: *Measured output (solid) and output prediction (dashed) for the 33rd order MIMO model.*

The full MIMO model

The five identified MISO models, of order 5, 7, 7, 6 and 8 respectively, are combined into one MIMO model. The dynamics of this model is used to generate a basis of order 33 for the full MIMO system with an $\{A, B\}$-structure. Only one group is used to parametrize the model, i.e. $n = 1$. This yields a high order model of $nn_b = 33$ with $n_y nn_b = 165$ free parameters.

The optimal value of the identification cost function could be improved from $V_{opt} = 0.11$ for the combination of the five MISO models to $V_{opt} = 0.099$ for the MIMO model. For this an RBS data set is used as identification data and a PRBS data set is used for validation. Similar results are observed if an RBS data set, other than the identification data, is used as validation set.

The step responses of this model are given in figure 7.6 together with the measured step responses. The output prediction of the MIMO model is given in figure 7.7. This is based on a PRBS validation set. The output prediction of the second output seems inaccurate, however this is mainly due to the initial condition disturbance in the validation set. This is only accounted for by the mean of five free runs which is rather inaccurate. The other outputs are predicted accurately. As conclusion, the identified model with the described approach is consistent with the step response data, the RBS data set and the PRBS data set. However, the order is relatively high and there seems to be room for improvement of the estimation accuracy. In the sequel it is shown how frequency weighted balanced reduction can provide such an improvement where especially the variance error is decreased.

Multiple data identification

In the identification of the model described previously, the (P)RBS data set is used for parametric identification while the step response experiments are used for the determination of an initial model and for validation purposes. In this part, the data of the PRBS and step experiments are combined in a multiple data identification as described in section 3.5.7. The step response of the identified high order model is given in figure 7.8. The

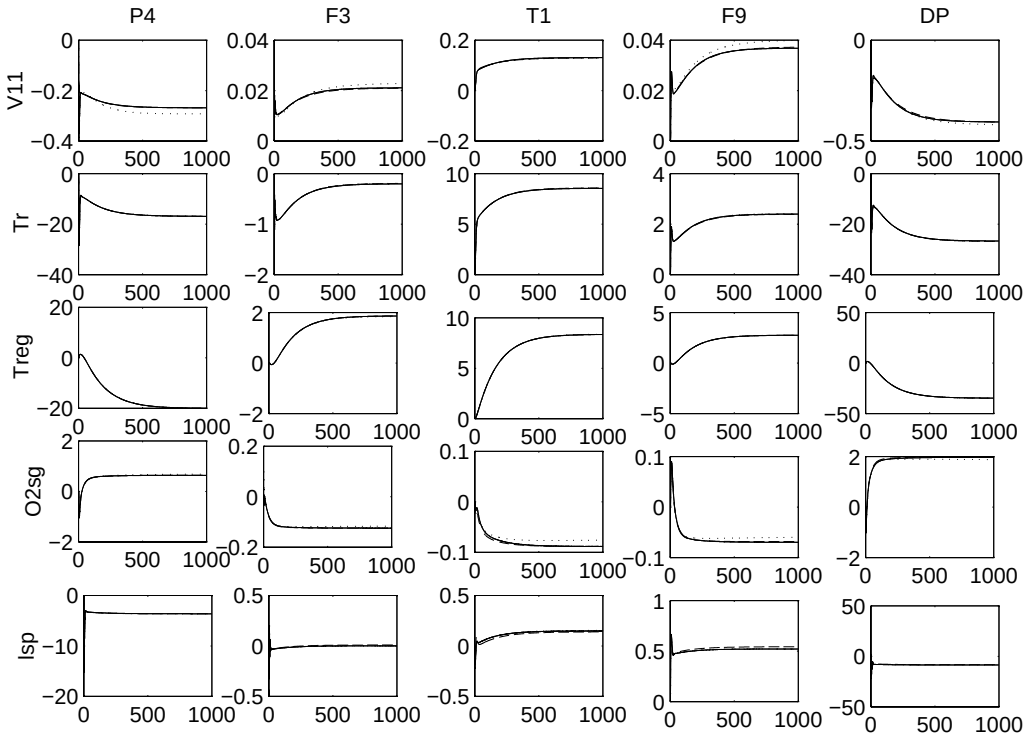

Fig. 7.8: *Step responses of the elements of the transfer function matrix of the 18th order multivariable model (dashed), the initial model obtained with realization algorithm (dotted) and the undisturbed output (solid).*

output prediction depicted in 7.10 is clearly improved.
The figures show that the model accuracy in the low frequent part is improved. This shows that the flexibility of identification on the basis of multiple data sets can be used to improve the model accuracy in certain frequency regions by performing dedicated experiments and combining the data as proposed in this thesis. This is especially useful if only a number of short plant tests can be performed, like in an industrial environment.

Frequency weighted balanced reduction

Next, the high order model is reduced with several standard reduction techniques and frequency weighted balanced reduction with the weighting filter proposed in section 4.2.4 that is calculated from the nonasymptotic variance description. The quality of the reduced models in terms of the quadratic cost function based on validation data is given in figure 7.9. From this figure it can be seen that with frequency weighted balanced reduction with the weighting filter described in section 4.2.4, much lower order models can be obtained without significant increase of the cost function. This is a consequence of the fact that in the reduction method the variance of the estimated high order model is taken into account.

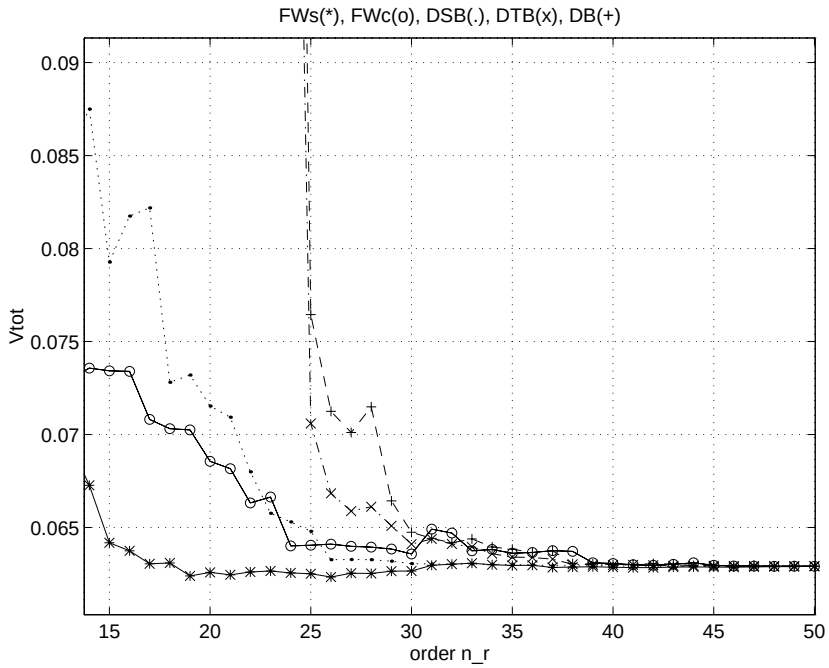

Fig. 7.9: *Optimal value of the criterion function V_{tot} of the reduced order models plotted against the reduced order n_r. The high order model is of 75^{th} order and reduction methods are frequency weighted balanced reduction (FWs), discrete truncated balanced reduction (DTB), discrete balanced reduction (DB) and discrete singular perturbational balanced reduction (DSB).*

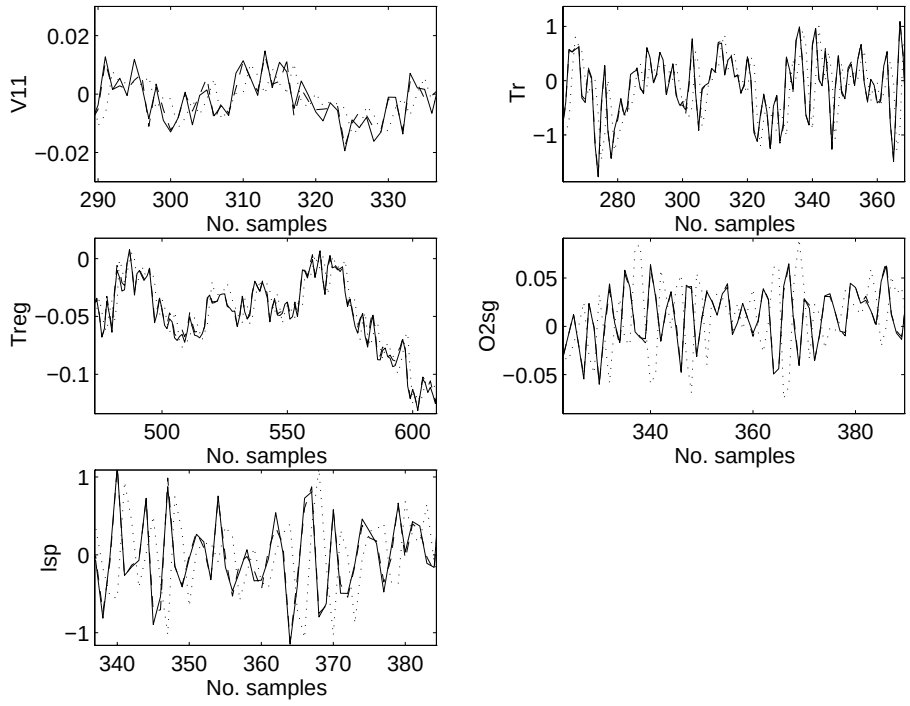

Fig. 7.10: *Undisturbed output (solid) and the output prediction for the initial model (dotted) and the MIMO model of order 18 after 3 iterations (dashed).*

Iterative model improvement

The iterative scheme discussed in section 4.3.2 is performed over high order ORTFIR identification on the basis of multiple data sets and the frequency weighted model reduction approach presented in section 4.2.4. After three iterations progress became slow and the iteration was terminated. During the iteration steps the optimal value of the cost function for the low order model is monotonically decreased from 0.048 for the initial model till 0.013 in the third iteration step. The largest gain is made in the first two steps, thereafter the improvement is marginal.
The output prediction of the model of order 18 after 3 iterations is given in figure 7.10. Clearly, a considerable improvement is obtained with respect to the initial estimate. Hence, with the approach followed an accurate model of limited order is obtained on the basis of a limited amount of data. Both the fast and slow dynamics are estimated accurately.
Although no convergence guarantees of the iteration can be given, it is a good alternative for nonlinear optimization. For this example nonlinear optimization of the cost function with an output error model structure is time-consuming and gets stuck in bad local optima and is therefore not practically applicable for this large scale application. Certainly, the model identified with the iterative scheme can be used as an initial model for a nonlinear optimization to find the nearest model that is optimal in terms of the quadratic cost function. This decreased the value of the cost function only marginally and no visible improvement of the model is observed. This indicates that the iterative scheme provided for this identification problem a near optimal model.

7.5 Static optimization

To find an improved working point for an industrial process static optimization can be utilized. For a suitable working condition it is important that, on the one hand the economics of the plant are optimized while on the other hand sufficient robustness is introduced to account for disturbances and model uncertainty. Economic optimization tends to drive a plant to one or several constraints. This may induce a situation where not enough control freedom is present. Hence, economic performance and robustness are conflicting goals which must be traded off against each other. In this section an approach is described to make this trade-off by specification of a suitable optimization problem.
A steady state input output pair is given by $\{u_{ss}, y_{ss}\}$ and satisfies the steady state relation

$$y_{ss} = K_{ss} u_{ss}$$

where $K_{ss} \in \mathbb{R}^{n_y \times n_u}$ is the steady state system transfer. To obtain an optimal working point the following linear cost function is minimized

$$f(u) = f_u^T u + f_y^T y = f_u^T u + f_y^T K_{ss} u$$

subject to input and output constraints

$$\begin{aligned} \underline{u} &\leq u \leq \bar{u} \\ \underline{y} &\leq y = K_{ss} u \leq \bar{y} \end{aligned}$$

This provides the working condition $\{u^*, y^*\}$ which is steady state optimal in terms of the linear cost function $f(u)$. An optimum can be found with a linear programming algorithm.
If output constraints are active it is advisable to choose a working condition that is somewhat away from the constraints to be able to deal with

- disturbances and
- model uncertainty

that may cause the system to violate the constraints. The introduction of such a safety margin is denoted with *back-off*. Usually, for economic reasons it is favorable to choose the working point as close to the constraints as possible. In the sequel an approach is presented that uses information about the output disturbance and uncertainty to determine a suitable back-off. For clarity of exposition the two approaches are treated separately but can be applied simultaneously.

Robustness for output disturbances

To obtain an appropriate back-off to account for output disturbances the following situation is regarded. Let the steady state model be extended to

$$y = K_{ss} u + d$$

where $\underline{d} \leq d \leq \bar{d}$ is the steady state value influence of disturbances on the output. The information about the disturbance can e.g. be obtained by identification techniques. Now we can find a suitable working point that is optimal in the face of a worst case disturbance by the minimization of the following cost function.

$$\begin{aligned} f(u) &= f_u^T u + \sup_d \{f_y^T y(u,d)\} = f_u^T u + f_y^T K_{ss} u + \sup_d \{f_y^T d\} \\ \underline{u} &\leq u \leq \bar{u} \\ \underline{y} &\leq y = K_{ss} u \leq \bar{y} \\ \underline{d} &\leq d \leq \bar{d} \end{aligned}$$

This is a convex minimax problem of which the supremum can be found by convex programming techniques. It can also be found by the solution to the following simple linear programming problem where the variable d has been eliminated.

$$\begin{aligned} f(u) &= f_u^T u + f_y^T K_{ss} u \\ \underline{u} &\leq u \leq \bar{u} \\ \underline{y} + \underline{d} &\leq K_{ss} u \leq \bar{y} - \bar{d} \end{aligned}$$

In this case the worst case disturbance is the disturbance that decreases the feasibility region the most.
If more information is available about the disturbance the disturbance model can be extended to e.g. $d = H_{ss} e$ where H_{ss} is the steady state transfer from a set of disturbances e, which is e.g. bounded by $\underline{e} \leq e \leq \bar{e}$, to the output y.

Robustness for model uncertainty

To obtain an appropriate back-off to account for model uncertainty the following situation is regarded. Let the steady state gain be linear in some parameter matrix $\Theta \in \mathbb{R}^{n_x \times n_u}$

$$K_{ss} = Q\Theta$$

where Θ is given to lie in the element wise uncertainty region given by

$$\underline{\Theta} \leq \Theta \leq \bar{\Theta}$$

where $\underline{\Theta}, \bar{\Theta}$ contain the lower and upper bound of the elements of the parameter matrix. Such an uncertainty region can e.g. be obtained with the identification techniques discussed in this thesis. The following optimization problem can be used to obtain a working point that is suitable for the worst-case steady state matrix.

$$f_u^T u + \sup_{\Theta} \{ f_y^T Q \Theta u \}$$

subject to

$$\begin{aligned} \underline{u} &\leq u \leq \bar{u} \\ \underline{y} &\leq Q\Theta u \leq \bar{y} \qquad (7.4) \\ \underline{\Theta} &\leq \Theta \leq \bar{\Theta} \qquad (7.5) \end{aligned}$$

The cost function is convex because the maximum over convex functions is also convex and linear functions are convex [13]. For this reason convex optimization algorithms can

be used to find a maximizer but the cost function can be nondifferentiable. Therefore convex programming algorithms must be used that do not use explicit expression of the gradient and Hessian.

The main difficulty, however, with this optimization problem is the nonlinear nonconvex constraint (7.4). In general this constraint makes it difficult to find a global maximum for the cost function. However, in this case prior knowledge about the solution can be utilized to find the global optimum. One way is to check all vertices of the uncertainty polytope which is, already for relatively small scale problems, computationally intractable. Instead we will pursue the use of realistic prior knowledge to find a global solution to the nonconvex problem.

Let the following additional information be given. At the optimal solution the output constraints with index $i = \{a_1, \ldots, a_y\}$ are active. This information can be found in specific cases. As a good starting point the active output constraints of the (nonrobust) static optimization can be used. With this information the nonconvex constraint can be eliminated and replaced by a linear equality constraint. How this is done is explained next.

Let $K_{ss}^T = [\theta_1 \cdots \theta_{n_y}]$ with $\theta_i \in \mathbb{R}^{n_u}$ where $i = 1, \ldots, n_y$. Assume that the constraints

$$\begin{aligned} \theta_2^T u &\leq \bar{y}_2 \\ \underline{y}_3 \leq \theta_3^T u & \end{aligned}$$

are active at the optimal solution. Therefore the other nonconvex constraints can be discarded. The two inequality constraints result in the following values for the parameter vector

$$\begin{cases} u_i \geq 0, \theta_{2i} = \bar{\theta}_{2i} \\ u_i < 0, \theta_{2i} = \underline{\theta}_{2i} \end{cases} \text{ and } \begin{cases} u_i \geq 0, \theta_{3i} = \underline{\theta}_{2i} \\ u_i < 0, \theta_{3i} = \bar{\theta}_{2i} \end{cases}$$

This provides the vertices of the polytope that coincide with active constraints. Hence, the parameter vectors θ_2, θ_3 have known values. This yields the following optimization problem.

$$f_u^T u + \sup_{\theta_1, \theta_4, \ldots} \{f_y^T Q \Theta u\}$$

subject to inequality constraints

$$\begin{aligned} \underline{u} &\leq u \leq \bar{u} \\ \underline{\theta_i} &\leq \theta_i \leq \bar{\theta}_i \text{ for } i = 1, 4, 5, \ldots \end{aligned}$$

and equality constraints

$$\begin{aligned} \theta_2 u &= \bar{y}_2 \\ \theta_3 u &= \underline{y}_3 \end{aligned}$$

where θ_2, θ_3 are known. This is a convex optimization problem which can be solved with conventional convex programming techniques.
This way to treat uncertainty in static optimization is conservative because it is based on open-loop considerations. Usually a feedback controller is active that is able to maintain setpoints closer to the output constraints

Static optimization of an FCCU

The optimization strategy described above is applied to the FCCU model. The objective is to maximize the total feedthrough while maintaining the reactor temperature at the highest possible level. This objective is translated to the cost function given by

$$f(u) = -\left(F_3 + F_4 + 100T_r\right)$$

The sum of fresh feed F_3 and slurry recycle F_4 in this cost function provides for a maximization of the feedthrough. The reactor temperature T_r weighted with a factor 100 makes sure that this variable is maximized with priority over the other terms.
The initial working condition was

$$u_0 = [126\ 2\ -3.18211\ 0.4015\ 0.95]^T \quad y_0 = [1261\ 994.95\ 2.2641\ 11.064]^T$$

which is the initial condition provided in [89]. The steady state transfer is retrieved from the estimated model using

$$K_{ss} = \hat{\Theta}(I - A_e)^{-1}B_e.$$

The disturbance variance σ_d is identified and the 3σ-boundary is used to specify the back-off. Additionally, the covariance matrix of the parameter estimate is used to describe the uncertainty polytope to provide sufficient robustness in the back-off. Note that the error due to undermodelling is discarded which can in this case be justified because the generalized expansion coefficients of the tail are very small compared to the part belonging to the model (see figure 7.5). With this information the optimization problem described previously is solved. This yields the new working point

$$u_0 = [133.28\ 0\ -3.355\ 1\ 0.95]^T \quad y_0 = [1263.1\ 994\ 2\ 12.62]^T$$

So the amount of fresh feed is increased with 7.

7.6 Model predictive control

7.6.1 Introduction

In this section model predictive control is applied to control the system from the initial working point to the improved working condition provided by the static optimization

while satisfying all process constraints. A comparison is made between model predictive control with a conventional pulse parametrization versus a system-based parametrization as proposed in chapter 5. It is also investigated whether scheduling of the system based input parametrization as discussed in section 5.5.3 contributes to an improved performance.

To compare only the influence of the parametrization, the controllers that are used differ solely in the parametrization. The controllers that are compared have a filter type state observer given in (2.28). The controller cost function that is optimized is the infinite horizon LQ cost function (2.23) with P infinite, subject to the process constraints given in section 7.3. This infinite dimensional constrained optimization problem is approximated with a finite dimensional problem by specification of a finite dimensional parametrization of the input trajectory. The following parametrizations are utilized.

- The quasi infinite horizon approach [128] is used. For this parametrization a finite prediction horizon P is selected that is such that after this time instant no constraints are active. For the samples $t = P, P+1, \ldots$ it is assumed that an LQ optimal state feedback controller F as given in (2.18) is active. The contribution of this tail to infinity can be described by the terminal state weighting matrix Q_0 that satisfies the Lyapunov equation

  $$(A - BF)^T Q_0 (A - BF) + (Q_1 + F^T Q_2 F) = Q_0$$

  With this approach constrained LQ optimal control is obtained if the input is parametrized freely over the prediction horizon. The number of free parameters in the optimization for this approach is Pn_u.

- A standard way to lower the complexity of constrained LQ optimal control is to use a pulse basis. For this purpose a control horizon M is selected. The input is parametrized freely for the first M samples and is fixed to the value $u(M-1)$ for the samples $t = M, \ldots, P-1$. The number of free parameters in this approach is given by Mn_u.

- The system based input parametrization for infinite horizon model predictive control is used as discussed in proposition 5.3.1. The number of free parameters in this approach is given by the order n of the linear model on which the controller is based.

In the sequel of this section a comparison is made between model predictive controllers with these parametrizations. Especially the efficiency, i.e. the relationship between performance and computational complexity, of the various approaches is compared. This is done in two ways. The first way is to compare the performance of two model predictive controllers that have the same complexity. The second way is to assess the computational complexity for two controllers with (approximately) the same performance.

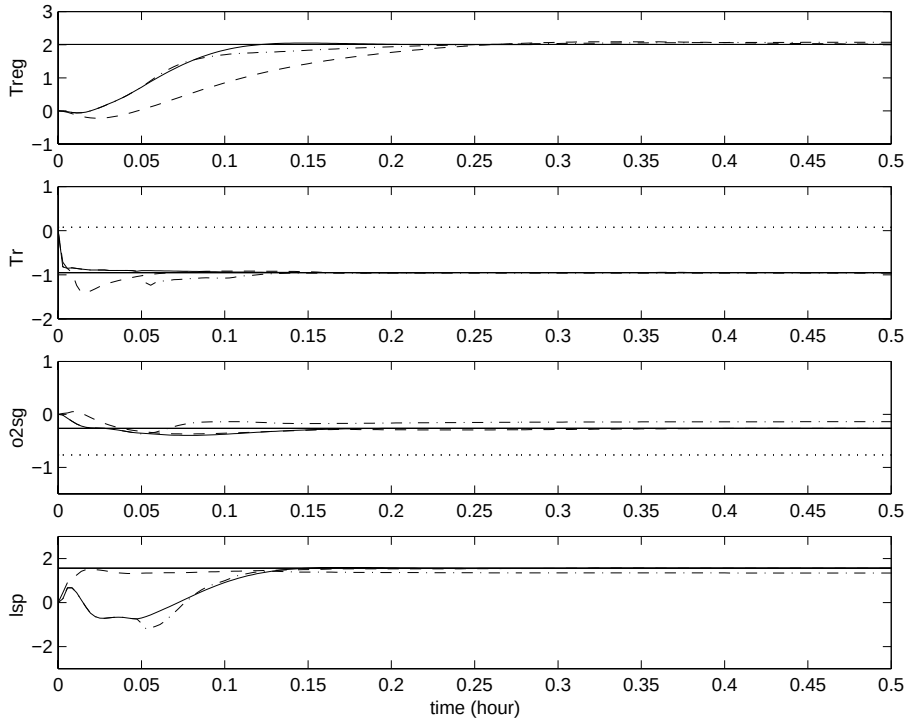

Fig. 7.11: *Simulated output for model predictive control with conventional (dashed), system based (solid) and scheduled system based (dash dotted) input parametrization.*

7.6.2 MPC with system-based input parametrization

First, a linear state space model $\{A, B, C, D\}$ is obtained which is a linearisation of the system around the initial working condition given in section 7.3. The order of this discrete time model is $n = 10$. This model is used both in the observer and the specification of the optimization problem and corresponds with a sampling time of 10 s.
To obtain enough preview of the slow system dynamics the prediction horizon is set to $P = 50$. With a sampling time of 10 s this implies that 500 s in the future is predicted. Tuning parameters that provided good performance appeared to be $Q_1 = C^T diag(1, 1, 2, 0.1)C, Q_2 = diag(1, 1, 1, 1, 1)$. Note that the output rather than the states are weighted.
First a comparison is made between the pulse parametrization and the system based approach given a fixed complexity. The system based parametrization yields an optimization problem with $n + n_u = 15$ degrees of freedom. The term n_u appears because the model is augmented with an integrator at the input to obtain integral action in the controller as explained in section 2.4.4. The pulse parametrization with (approximately) the same degrees of freedom has a control horizon $M = 3$. This induces an on-line optimization problem with $Mn_u = 15$ degrees of freedom. This results in control strategies with the same on-line computational load.

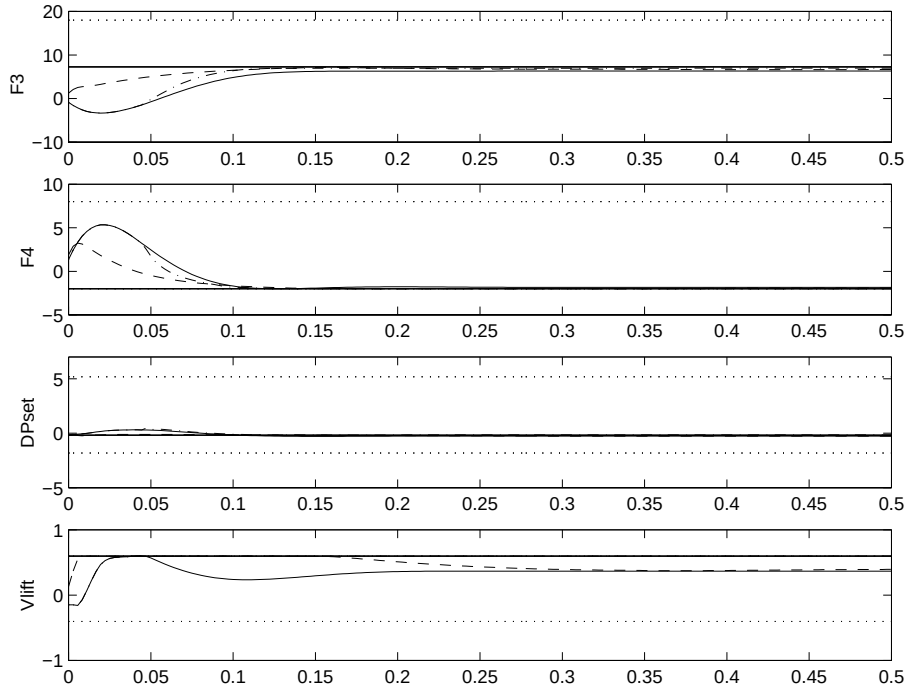

Fig. 7.12: *Simulated input for model predictive control with conventional (dashed), system based (solid) and scheduled system based (dash dotted) input parametrization.*

The simulated outputs are given in figure 7.11. The model predictive controller with system based parametrization attains a faster response than the one with conventional parametrization which is particularly visible in the regenerator temperature. The simulated inputs are given in figure 7.12. The conventional parametrization attains slightly faster the final level of the fresh feed F_3.

Next, instead of a comparison of the control performance given a certain computational load, a comparison is made between the computational load given a certain level of performance. For this purpose the control horizon M is increased until the performance of model predictive control with a pulse parametrization is equivalent to the performance of model predictive control with a system-based parametrization. In figure 7.13 the response of the temperature of the regenerator T_r for various controllers. In this figure the performance for a control horizon $M = 3$ is again depicted. If the control horizon is increased to $M = 10$, which implies a quadratic programming problem with 50 degrees of freedom, the performance is increased but the response is slower than with a system-based input parametrization. Only if the input is parametrized freely over the full prediction horizon $P = M = 50$ (250 degrees of freedom in the quadratic program) the responses practically coincide.

Although for this specific industrial type problem efficient parametrization is not indispensible, the sampling time of 10 s provides enough calculation time to perform a moderate size quadratic program, it still shows that improved performance can be obtained at a

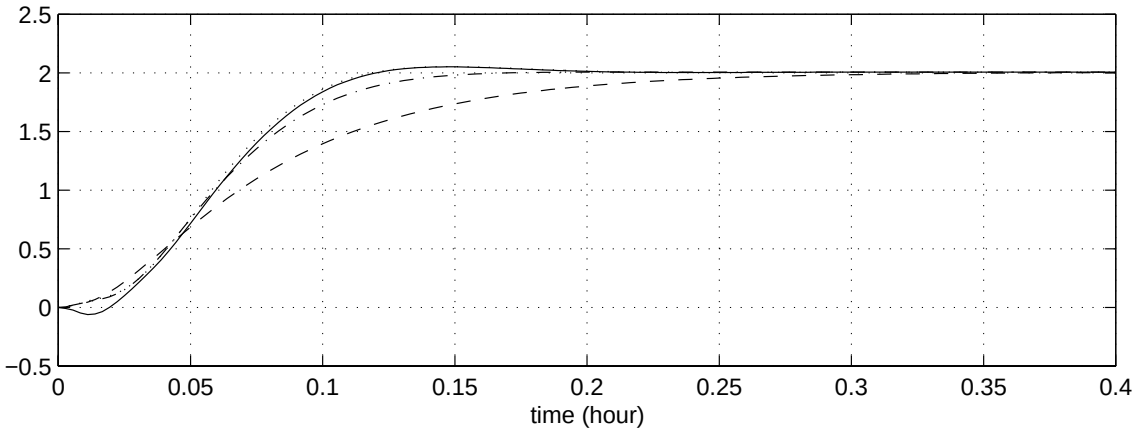

Fig. 7.13: *Response of the regenerator temperature with model predictive control with a system-based parametrization (solid) and with a pulse parametrization for control horizons $M = 3$ (dashed), $M = 10$ (dash dotted) and $M = P = 50$ (dotted) which almost coincides with the solid line.*

lower computational price by applying a parametrization that is tailored to the problem at hand.

7.6.3 MPC with input parametrization scheduling

In section 5.5.3 it is described how the performance of MPC_{ip} can be improved by scheduling of the parametrization in case a structural constraint is encountered. From the static optimization described in section 7.5 it is predicted that the V_{lift} is constrained if the new steady state value is reached. If the scheduling approach proposed in section 7.5 is used, this implies that the input V_{lift} is fixed once it reaches the constraint value.
If the scheduling approach is utilized such that the input V_{lift} is fixed after 0.04 hours, the following effects are observed in figure 7.12 and 7.11. The speed of the fresh feed is increased for the scheduling approach. This is, however, at the cost of the speed of response in the output, which expresses itself in a slightly slower response of the regenerator temperature.
The scheduling approach discussed in section 7.5 is not robust for the mismatch between the plant and the model, which becomes clear in this example. The static optimization, based on the linear model, predicts that the input V_{lift} is constrained in the new working point. In fact, this is not the case because the nonlinear model is different from the linear model. Therefore, the scheduling approach, which is based on the linear model, is not able to provide an improved performance. Hence, in order to improve the performance with scheduling, it is important to utilize a scheduling strategy that accounts for plant model mismatch.
It can be concluded that a system based input parametrization can be used to increase the

performance of model predictive control while keeping the computational cost the same. Or, vice versa, the computational cost can be decreased with only a small or no decrease in performance. Scheduling the parametrization can help to improve certain performance aspects but it is important to choose a suitable scheduling strategy that accounts for the mismatch between plant and model.

7.7 Discussion

In this chapter the identification, static optimization and model predictive control of a nonlinear simulation model for the Model IV catalytic cracking unit is described. The system is multivariable and contains both fast and slow dynamical phenomena. For economic reasons it is favorable to identify an accurate model on the basis of short experiments. Therefore, as additional challenge the length of the data set is restricted to five hours which is approximately two times the settling time. It is common industrial practice to use several local controllers without regarding process constraints at a high sampling rate, e.g. in the order of a second. On top of this so called basic control layer a model predictive controller is active that regards the process constraints but runs at a larger sampling rate, e.g. in the order of minutes. In this case study only one local controller is used and model predictive control is utilized at a higher sampling rate. The sampling time is chosen according to the fastest dynamical phenomenon and is taken to be 10 seconds.
Prediction error identification with multivariable system-based orthogonal basis functions, discussed in section 3.3 is utilized successfully. Both fast and slow dynamics are captured accurately. Due to the short length of the data set, the slow behaviour of the system is barely present in the data. Therefore, in the identification the slow dynamics have less emphasis although it is important for control design to have an accurate model for this frequency range. By selection of an initial basis with accurate low frequent dynamics, retrieved from step response experiments, the ORTFIR identification procedure captured both dynamic ranges accurately. Also the use of a soft steady state constraint proposed in section (3.5.6) and the combination of data sets discussed in section 3.5.7 has a favorable effect on the low frequent fit of the model.
With the iterative scheme of high order identification and model reduction discussed in section 4.3.2 a considerable decrease of the cost function can be obtained. This decrease is large for the first two steps but thereafter no considerable improvement can be made. The model reduction technique used in the iterative scheme has a large influence on the final model. The frequency weighted balanced reduction approach discussed in section 4.2.4 performs particularly well. With this approach an accurate model is already obtained for relatively low model orders.
Model predictive control with system based input parametrization (MPC_{ip}), described in chapter 5, is applied to this system. MPC_{ip} is compared to model predictive control

with a conventional (pulse) parametrization in two ways. First, it is shown that MPC with the proposed input parametrization provides higher control performance than with a conventional parametrization with the same number of degrees of freedom. Secondly, a considerably higher computational load is necessary to obtain the same level of performance with a conventional parametrization. Hence, the proposed parametrization can provide an improved efficiency although in this example efficiency is not crucial because the sampling time of 10 s provides enough calculation time to obtain high performance. An investigation of the properties of the parametrization scheduling approach discussed in section 5.5.3 showed that no considerable improvement can be made with this strategy for this example. The main reason is that there is a mismatch between the nonlinear system and the linear model used for model predictive control, while the scheduling approach is based on nominal considerations. This indicates that improvement can be obtained with scheduling approaches that are more robust against a plant model mismatch.

Chapter 8

Conclusions and recommendations

8.1 Review

The ever increasing competition between industrial companies is an incentive to push the limits of the performance of industrial plants in which large sums of money are invested. If there is enough demand for the product that is produced, operating the plant at or near constraints is optimal. Automatic control and system identification of these, often large scale, plants remains challenging.
Systematic incorporation of constraints in the control design is currently only possible for relatively slow systems with low dynamic performance specification, e.g. petrochemical plants. This is due to the high computational load of the available technology: model predictive control. In other industries, such as iron and steel, food processing, industries in which robots are used in manufacturing etc., faster dynamics need to be controlled within the constraints of the system. In this research a possibility is investigated to improve the efficiency of model predictive control technology.
System identification of industrial processes mentioned above is still a challenging enterprise. The experimental freedom is limited due to the constraints and the desire to keep the economic loss during experimentation as small as possible. This results in short data sets with moderate excitation of the plant dynamics, i.e. a bad signal to noise ratio. Due to the large scale of processes nonconvex optimization techniques are likely to get stuck in local optima or fail to converge at all. Therefore, it is still a difficult and specialistic task to obtain a model with system identification techniques in this situation. In this research it is investigated how large scale industrial systems can be accurately identified without the need for nonconvex optimization. Aimed for are efficient algorithms that attain high accuracy with relatively low computational complexity.
More specifically, in this thesis it is investigated to what extent a flexible (linear) parametrization in model predictive control and system identification can be used to improve the efficiency of these techniques applied to large scale industrial systems exhibiting both fast

and slow dynamics.

8.2 Contributions of this research

With respect to system identification techniques the following results were obtained.
Model structures based on system-based orthogonal basis functions, denoted in short with ORTFIR structures, can be used to identify both scalar and multivariable systems. For scalar ORTFIR the basis functions are determined by a prechosen set of poles, i.e. dynamics. Multivariable ORTFIR structures, enables the use of prior knowledge with respect to both *dynamics* and *structure* of the multivariable system. Structural information that can be used is e.g. knowledge about the distribution of the poles over the entries of the transfer function matrix or controllability indices. The use of the prior knowledge can decrease the complexity of the model structure yielding a lower bias and variance of the model than e.g. FIR and ARX structures.
It is well known from prediction error identification that the estimation of a noise model can have a favorable effect on the variance of the estimate. In literature ORTFIR identification is usually presented without a noise model. In this thesis, various possibilities are investigated to incorporate the estimation of a noise model in ORTFIR identification. A new model structure is proposed that utilizes system-based orthogonal basis functions in the specification of both the system transfer and the disturbance transfer. The structure is bilinear in the free parameters making a nonlinear optimization routine necessary for the solution of the free parameters. A relaxation type algorithm is proposed, which consists of consecutive steps of least squares optimization and is proven to converge to a stationary point under mild conditions. The approach is similar to generalized least squares (GLS), hence the name ORT-GLS.
Dependent on the accuracy of the prior knowledge used in the construction of the basis functions, the identified ORTFIR model can still have a relatively high order. The model order can be reduced systematically with consecutive steps of high order ORTFIR identification and model reduction. The reduced order model is used for the generation of improved basis functions. This increases the prior knowledge about the system in each iteration step. In this thesis the iterative scheme of identification and model reduction is improved such that it can be used as alternative for nonlinear optimization for large scale identification problems. The main improvement is obtained by using frequency weighted balanced reduction with the weighting filter that includes information about the variance of the high order estimate.
With respect to model predictive control the following issues have been elaborated.
The question is addressed to what extent it is possible to improve the efficiency of MPC with the use of system information in the parametrization of the input trajectory, similar to the use in system identification. It is shown that for situation where constraints play a

moderate role, a system-based parametrization can be used fruitfully. An efficient infinite horizon MPC algorithm is proposed and analyzed that utilizes this information. It is shown that the complexity can be reduced systematically with conventional linear model reduction techniques because the parametrization is generated by a linear dynamical system. The indicators that are used in these techniques to decide on an appropriate model order can be used in this context to keep track of the unconstrained performance loss.
Although useful in large class of situations, the system based approach certainly has limitations. The use of system-based parametrization is limited because the performance loss with respect to free parametrization grows larger, the more constraints play a role. This is due to the fact that optimal input profiles of constrained systems are inherently non-differentiable, i.e. they exhibit corners etc. In the continuous time case, trajectories generated by systems are differentiable. Approximating a non-differentiable signal with differentiable ones is known to be highly inefficient. Hence, the use of information about system dynamics in input parametrization for model predictive control is confined to a limited class of applications.
Instead of utilizing information about the system dynamics to construct a linear parametrization, other information can be used. In this thesis, the question is addressed to what extent information about the active constraints can be used to improve the efficiency of model predictive control. It is shown that for general constrained situations this information can indeed be used fruitfully. An approach is proposed that, starting with a high dimensional optimization problem, selects a low dimensional subset of the full space over which the optimization takes place. The computational load can be decreased by optimization over the low instead of the high dimensional space. This strategy is worked out in detail for the quadratic programming problem, which provides a novel way of solving a high dimensional quadratic programming problem. The use of this optimization strategy yields an efficient model predictive control routine. This can be interpreted as a controller in which at each time instant a low dimensional linear parametrization of the input is selected on-line according to the specific situation at hand.

8.3 Recommendations for future research

The following points deserve attention in research projects in the (near) future. A thorough analysis of the convergence properties is desirable for the iterative scheme of high order identification with basis functions and model reductions. There are strong indications that such a nonlinear programming approach is more generally applicable to optimization problems in which a transfer function is searched for. The underlying principle consists basically of two steps: first, solve a convex optimization problem to find a global approximation of the object that is searched in terms of a basis that is tailored to the object; second, transform the object in the chosen basis back to the format that is desired.

The underlying principle is applicable to an even wider range of optimization problems. Also an analysis of the convergence properties is desirable for the alternative quadratic programming approach that is proposed in this thesis. The prototype software developed in this research is, with respect to computational speed, comparable with interior point methods. Further development of efficient implementations seem worthwhile. The principle of estimating the constraints together with an uncertainty region and use this information to construct a low dimensional subset to lower the optimization complexity, seems to have a wider range of application in constrained optimization than quadratic programming. An investigation of the potential of this approach to general constrained optimization may be fruitful.

In this thesis the use of flexible linear parametrizations is investigated for linear system identification and model predictive control. A similar approach is conceivable for nonlinear identification and model predictive control. Therefore, an important question that can be addressed is to what extent linear parametrizations can contribute to the improvement of efficiency in nonlinear identification and model predictive control.

Appendix A
Proofs for chapter 3

Proof of proposition 3.4.2

The specific (deterministic) optimization problem that has to be solved is the following

$$\min_{\rho,\eta} V(\rho,\eta) = \frac{1}{N}\sum_{t=1}^{N} \varepsilon^2(t,\rho,\eta) \tag{8.1}$$

with

$$\begin{aligned} \varepsilon(t,\rho,\eta) &= H^{-1}(\eta)(y(t) - G(\rho)u(t)) \\ &= H^{-1}(\eta)y(t) + \varphi_F^T(t,\eta)\rho \\ &= H^{-1}(\eta)v(t,\rho) \end{aligned}$$

where

$$\begin{aligned} \varphi_F(t,\eta) &:= -H^{-1}(\eta)[f_1 u(t) \cdots f_{n_g} u(t)] = H^{-1}(\eta)\varphi(t) \\ \varphi(t) &:= -[f_1 u(t) \cdots f_{n_g} u(t)]^T \\ v(t,\rho) &:= y(t) - G(\rho)u(t) = y(t) + \varphi^T(t)\rho. \end{aligned}$$

Moreover let

$$\begin{aligned} \psi(t,\rho) &= [w_1 v(t,\rho) \cdots w_{n_h} v(t,\rho)]^T \\ &= \begin{bmatrix} w_1\varphi^T(t) \\ \vdots \\ w_{n_h}\varphi^T(t) \end{bmatrix}\rho + \begin{bmatrix} w_1 y(t) \\ \vdots \\ w_{n_h} y(t-n_c) \end{bmatrix} \\ &= \Phi(t)\rho + Y(t) \end{aligned}$$

which implies $\varepsilon(t,\rho,\eta) = \psi^T(t,\rho)\eta$.
It clearly holds that

$$\begin{aligned}\frac{\partial \varepsilon(t,\rho,\eta)}{\partial \rho} &= \varphi_F(t,\eta)\\ \frac{\partial \varepsilon(t,\rho,\eta)}{\partial \eta} &= \psi(t,\rho) = \Phi(t)\rho + Y(t)\end{aligned}$$

The consecutive steps in the relaxation algorithm can be rewritten as follows with the operator $\tilde{E} = \frac{1}{N}\sum_{t=1}^{N}$.

Step 1 is the least squares problem

$$\begin{aligned}\rho_k &= arg\min_{\rho} V(\rho,\eta_{k-1})\\ &= \left(\tilde{E}\varphi_F(t,\eta_{k-1})\varphi_F^T(t,\eta_{k-1})\right)^{-1}\tilde{E}(\varphi_F(t,\eta_{k-1})H^{-1}(\eta_{k-1})y(t))\\ &= \left(\tilde{E}\varphi_F(t,\eta_{k-1})\varphi_F^T(t,\eta_{k-1})\right)^{-1}\tilde{E}(\varphi_F(t,\eta_{k-1})\left[\varepsilon(t,\rho_{k-1},\eta_{k-1}) - \varphi_F^T\rho_{k-1}\right]\\ &= \rho_{k-1} + \left(\tilde{E}\varphi_F(t,\eta_{k-1})\varphi_F^T(t,\eta_{k-1})\right)^{-1}\tilde{E}(\varphi_F(t,\eta_{k-1})\varepsilon(t,\rho_{k-1},\varphi_{k-1})\\ &= \rho_{k-1} + \left(\tilde{E}\varphi_F(t,\eta_{k-1})\varphi_F^T(t,\eta_{k-1})\right)^{-1}\left.\frac{\partial V(\rho,\eta)}{\partial \rho}\right|_{\rho_{k-1},\eta_{k-1}}\end{aligned}$$

The third equality uses $H^{-1}(\eta_{k-1})y(t) = \varepsilon(t,\rho,\eta_{k-1}) - \varphi_F^T\rho$ for any ρ.

Step 2 is the least square problem

$$\begin{aligned}\eta_k &= arg\min_{\eta} V(\rho_k,\eta)\\ &= \left(\tilde{E}\psi(t,\rho)\psi^T(t,\rho)\right)^{-1}\tilde{E}(\psi(t,\eta_{k-1})v(t,\rho_k))\\ &= \left(\tilde{E}\psi(t,\rho_k)\psi^T(t,\rho_k)\right)^{-1}\tilde{E}(\psi(t,\rho_k)\left[\varepsilon(t,\rho_k,\eta_{k-1}) - \psi^T(t,\rho_k)\eta_{k-1}\right]\\ &= \eta_{k-1} + \left(E\psi(t,\eta_{k-1})\psi^T(t,\eta_{k-1})\right)^{-1}\tilde{E}(\psi(t,\eta_{k-1})\varepsilon(t,\rho_{k-1},\varphi_{k-1})\\ &= \eta_{k-1} + \left(\tilde{E}\psi(t,\eta_{k-1})\psi^T(t,\eta_{k-1})\right)^{-1}\left.\frac{\partial V(\rho,\eta)}{\partial \eta}\right|_{\rho_k,\eta_{k-1}}\end{aligned}$$

The third equality uses $v(t,\rho_k) = \varepsilon(t,\rho_k,\eta) - \psi^T(t,\rho_k)\eta$ for any η.
So we have

$$\begin{bmatrix}\rho_k\\ \eta_k\end{bmatrix} = \begin{bmatrix}\rho_{k-1}\\ \eta_{k-1}\end{bmatrix} + \begin{bmatrix}\tilde{E}(\varphi_F\varphi_F^T)^{-1} & 0\\ 0 & \tilde{E}(\psi\psi^T)^{-1}\end{bmatrix}\begin{bmatrix}\left.\frac{\partial V(\rho,\eta)}{\partial \rho}\right|_{\rho_{k-1},\eta_{k-1}}\\ \left.\frac{\partial V(\rho,\eta)}{\partial \eta}\right|_{\rho_k,\eta_{k-1}}\end{bmatrix}$$

Due to the sufficiency of excitation condition it holds that $\tilde{E}\psi\psi^T > 0$ and $\tilde{E}\varphi\varphi^T > 0$. Moreover, the value of the cost function is decreased in each iteration. To proof convergence we try to rewrite the algorithm to the form

$$\begin{bmatrix} \rho_k \\ \eta_k \end{bmatrix} = \begin{bmatrix} \rho_{k-1} \\ \eta_{k-1} \end{bmatrix} + \begin{bmatrix} F_1 & 0 \\ F_{21} & F_2 \end{bmatrix} \begin{bmatrix} \frac{\partial V(\rho,\eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}} \\ \frac{\partial V(\rho,\eta)}{\partial \eta}\Big|_{\rho_{k-1},\eta_{k-1}} \end{bmatrix}$$

with a positive definite matrix on the right hand side and $V(\rho_{k-1}, \eta_k) < V(\rho_{k-1}, \eta_{k-1})$. This is a type of algorithm that is studied well in literature. For a detailed convergence proof see e.g. [85] (chapter 7, especially p. 215 for the global convergence result of steepest descent methods. Also chapter 6 p. 187 for the important global convergence theorem) and [97] (section 10.5 p. 314 and further). The minor difference is that the *sufficient decrease requirement* is guaranteed here by the structure of the algorithm instead of a line search.

To bring the algorithm to this form we rewrite

$$\frac{\partial V}{\partial \eta}\Big|_{\rho_k,\eta_{k-1}} = \tilde{E}\varepsilon(\rho_k, \eta_{k-1}) \frac{\partial \varepsilon(t, \rho_k, \eta)}{\partial \eta}\Big|_{\rho_k,\eta_{k-1}}$$

using

$$\rho_k = \rho_{k-1} + \tilde{E}(\varphi_F \varphi_F^T)^{-1} \frac{\partial V(\rho, \eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}}$$

This yields

$$\begin{aligned} \varepsilon(t, \rho_k, \eta_{k-1}) &= y_F(t) - \varphi_F^T(t, \eta_{k-1})\rho_k && (8.2) \\ &= y_F(t) - \varphi_F(t, \eta_{k-1})\rho_{k-1} - \varphi_F(t, \eta_{k-1})\tilde{E}(\varphi_F\varphi_F^T)^{-1} \frac{\partial V(\rho, \eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}} && (8.3) \\ &= \varepsilon(t, \rho_{k-1}, \eta_{k-1}) - \varphi_F(t, \eta_{k-1})E(\varphi_F\varphi_F^T)^{-1} \frac{\partial V(\rho, \eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}} && (8.4) \end{aligned}$$

$$\begin{aligned} \frac{\partial \varepsilon(t, \rho_k, \eta)}{\partial \eta}\Big|_{\rho_{k-1},\eta_{k-1}} &= \psi(t, \rho) = Y(t) + \Phi(t)\rho_k \\ &= Y(t) + \Phi(t) \left(\rho_{k-1} + \tilde{E}(\varphi_F\varphi_F^T)^{-1} \frac{\partial V(\rho, \eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}} \right) \\ &= \frac{\partial \varepsilon(t, \rho_k, \eta)}{\partial \eta}\Big|_{\rho_{k-1},\eta_{k-1}} + \Phi(t)\tilde{E}(\varphi_F\varphi_F^T)^{-1} \frac{\partial V(\rho, \eta)}{\partial \rho}\Big|_{\rho_{k-1},\eta_{k-1}} \end{aligned}$$

This yields

$$\frac{\partial V}{\partial \eta}\Big|_{\rho_k,\eta_{k-1}} = \tilde{E}\varepsilon(\rho_k, \eta_{k-1}) \frac{\partial \varepsilon(t, \rho_k, \eta)}{\partial \eta}\Big|_{\rho_k,\eta_{k-1}}$$

$$
\begin{aligned}
&= \tilde{E}\left(\varepsilon(t,\rho_{k-1},\eta_{k-1}) - \varphi_F(t,\eta_{k-1})\tilde{E}(\varphi_F\varphi_F^T)^{-1}\left.\frac{\partial V(\rho,\eta)}{\partial\rho}\right|_{\rho_{k-1},\eta_{k-1}}\right)\cdot \\
&\quad\cdot\left(\left.\frac{\partial\varepsilon(t,\rho_k,\eta)}{\partial\eta}\right|_{\rho_{k-1},\eta_{k-1}} + \Phi(t)\tilde{E}(\varphi_F\varphi_F^T)^{-1}\left.\frac{\partial V(\rho,\eta)}{\partial\rho}\right|_{\rho_{k-1},\eta_{k-1}}\right) \\
&= \tilde{E}\varepsilon(\rho_{k-1},\eta_{k-1})\left.\frac{\partial\varepsilon(t,\rho_{k-1},\eta)}{\partial\eta}\right|_{\rho_{k-1},\eta_{k-1}} + \\
&\quad\tilde{E}\left[\varepsilon(\rho_{k-1},\eta_{k-1}))\Phi(t)E(\varphi_F\varphi_F^T)^{-1} - \left.\frac{\partial\varepsilon(t,\rho_{k-1},\eta)}{\partial\eta}\right|_{\rho_{k-1},\eta_{k-1}}\varphi_F(t,\eta_{k-1})\tilde{E}(\varphi_F\varphi_F^T)^{-1} + \right. \\
&\quad\left.-\varphi_F(t,\eta_{k-1})\tilde{E}(\varphi_F\varphi_F^T)^{-1}\left.\frac{\partial V(\rho,\eta)}{\partial\rho}\right|_{\rho_{k-1},\eta_{k-1}}\right]\left.\frac{\partial V(\rho,\eta)}{\partial\rho}\right|_{\rho_{k-1},\eta_{k-1}} \\
&= \left.\frac{\partial V}{\partial\eta}\right|_{\rho_{k-1},\eta_{k-1}} + \bar{F}_{12}\left.\frac{\partial V(\rho,\eta)}{\partial\rho}\right|_{\rho_{k-1},\eta_{k-1}}
\end{aligned}
$$

Resulting in

$$
\begin{aligned}
F_1 &= \tilde{E}(\varphi_F\varphi_F^T)^{-1} > 0 \\
F_2 &= \tilde{E}(\psi\psi^T)^{-1} > 0 \\
F_{12} &= \tilde{E}(\psi\psi^T)^{-1}\bar{F}_{12}
\end{aligned}
$$

Hence, the algorithm has the desired format. For algorithms of this form it is proven in e.g. [85][97] that it converges and that it converges to a point that has a gradient equal to zero

$$\frac{\partial V(\rho,\eta)}{\partial\rho} = 0, \frac{\partial V(\rho,\eta)}{\partial\rho} = 0$$

if the iteration starts sufficiently close to a stationary point of the cost function, the iteration converges to this point. It is important to note that there are no guarantees that this point is a minimizer because the second order conditions may not be satisfied.

Proof of proposition 3.4.3

For the general case regarded in proposition 3.4.2 it cannot be proven that the second order optimality conditions hold. However, under the sufficiency of excitation and the dynamics of the system contained in the model set $G_0 \in \mathcal{G}$ it can be shown that for $N \to \infty$ the Hessian is positive definite which implies that the convergence point is a minimizer. The Hessian $\dfrac{\partial^2 V}{\partial\theta\partial\theta^T}$ of the cost function can be analyzed as follows. Rewriting yields:

$$\frac{\partial V(\rho,\eta)}{\partial\theta} = \frac{-2}{N}\sum_{t=1}^{N}\varepsilon(t,\rho,\eta)\begin{bmatrix}\varphi_F(t,\eta)\\ \psi(t,\rho)\end{bmatrix} \tag{8.5}$$

it follows that

$$\frac{\partial^2 V_N}{\partial\rho\partial\rho^T} = \frac{2}{N}\sum_{t=1}^{N}\varphi_F(t,\eta)\varphi_F^T(t,\eta) \tag{8.6}$$

$$\frac{\partial^2 V_N}{\partial\eta\partial\eta^T} = \frac{2}{N}\sum_{t=1}^{N}\psi(t,\rho)\psi^T(t,\rho). \tag{8.7}$$

These two block terms appear on the (block) diagonal of the Hessian matrix. For analyzing the off-diagonal block terms we consider:

$$\frac{\partial^2 V_N}{\partial\rho_i\partial\eta_j} = \tag{8.8}$$

$$= \frac{-2}{N}\frac{\partial}{\partial\eta_j}\sum_{t=1}^{N}\varepsilon(t,\rho,\eta)\cdot H^{-1}(q,\eta)f_i(q)u(t) \tag{8.9}$$

$$\begin{aligned} &= \frac{-2}{N}\sum_{t=1}^{N}[\varepsilon(t,\rho,\eta)\cdot w_j(q)f_i(q)u(t) \\ &- w_j(q)[y(t)-G(q,\rho)u(t)]\cdot H^{-1}(q,\eta)f_i(q)u(t) \\ &= \frac{-2}{N}\sum_{t=1}^{N}H^{-1}(q,\eta)[y(t)-G(q,\rho)u(t)]\cdot w_j(q)f_i(q)u(t) \\ &- w_j(q)[y(t)-G(q,\rho)u(t)]\cdot H^{-1}(q,\eta)f_i(q)u(t) \end{aligned}$$

For $N\to\infty$ and in expectation this expression tends to zero if $G_0\in\mathcal{G}$. This can be seen as follows. In that case $\hat{G}\to G_0$ which implies that $y(t)-G(q,\rho)u(t)$ tends to the disturbance $v(t)$ which is uncorrelated with the input $u(t)$.
This shows that the Hessian is (asymptotically) block diagonal and positive definite. Therefore, the points where the gradient vanishes are minimizers.

List of symbols

Sets

$\mathbb{C}$	Set of complex numbers
$\mathcal{G}$	Set of parametrized deterministic models
H_2	Set of all complex valued functions which are analytic outside the unit disk and squared integrable over the unit circle
H_∞	Set of all complex valued functions which are analytic and uniformly bounded outside the unit disc and on the unit circle
l_1	Set of absolute summable sequences
l_2	Set of square summable sequences
l_∞	Set of sequences that are uniformly bounded
$\mathcal{M}$	Set of parametrized deterministic models and noise models
$\mathbb{R}$	Set of real numbers
RH_2	Set of all real rational functions in H_2
$\mathcal{U}$	Set of input signal with a free parametrization, input space
$\mathcal{U}_r$	Reduced input space

Abbreviations

AR	Auto Regressive model structure
ARE	Algebraic Riccati Equation
ARMAX	Auto Regressive Moving Average with eXogeneous input
ARX	Auto Regressive with eXongeous input
BJ	Box Jenkins model structure
DARE	Discrete Algebraic Riccati Equation
FDLTI	Finite Dimensional Linear Time Invariant
FIR	Finite Impulse Response
GLS	Generalized Least Squares
LQ	Linear Quadratic
LQR	Linear Quadratic Regulator

LQG	Linear Quadratic Gaussian
LTI	Linear Time Invariant
MIMO	Multi Input Multi Output
MISO	Multi Input Single Output
MPC	Model Predictive Control
OE	Output Error model structure
ORTFIR	ORThogonal basis function Finite Impulse Response
(P)RBS	(Pseudo) Random Binary Sequence
QP	Quadratic Programming
RDE	Riccati Difference Equation
sp	Span
w.p.	With Probability

Plant, model and controller

C	Arbitrary controller
F_i	Multivariable basis function
f_i	Scalar basis function
G_0	(Unknown) Plant
G	Arbitrary plant model
$\hat{G}$	Estimated plant model
H_0	(Unknown) noise filter
H	Arbitrary disturbance model
$\hat{H}$	Estimated disturbance model
$\mathcal{S}$	System

Signals

d	disturbance signal
ε	prediction error
e	white noise singal
r	reference signal
u	plant input signal
v	unmeasurable disturbance
w	measurable disturbance
y	measured output
z	performance output

Miscellaneous symbols and abbreviations

A, B, C, D	State space matrices
G_b	Inner function for basis generation
I_n	Identity matrix of order n
$im\{X\}$	Image space or column span of matrix X
λ	Lagrange multiplier
$\lambda(X)$	Eigenvalues of the matrix X
$\sigma(X)$	Singular values of the matrix X
$\kappa(X)$	Condition number of the matrix X
Θ	Free parameter matrix
θ, ρ, η	Free parameter vector
$\hat{\theta}$	Estimated parameter
θ^*	Asymptotic parameter value
Φ	Regression matrix
φ	Regression vector
$\mathcal{N}(\mu, P)$	Gaussian distribution with mean value μ and covariance (matrix) P

Bibliography

[1] F. Allgöwer, T.A. Badgwell, J.S. Qin, and J.B. Rawlings. Nonlinear predictive control and moving horizon estimation - an introductory overview. In P.M. Frank, editor, *Advances in Control, Highlights of ECC'99,* Springer, pages 391–449, 1999.

[2] E.D. Andersen and Y.Ye. On a homogeneous algorithm for the monotone complementarity problem. *Math. Programming*, 84(2):375–399, 1999.

[3] B.D.O. Anderson and J.B. Moore. *Optimal Control, Linear Quadratic Methods.* Prentice-Hall Inc., Englewood Cliffs, NJ, 1989.

[4] E. Anderson and K. Anderson. Exploiting parallel hardware in solving optimization problems. *SIAM News*, 32(4), 1999.

[5] A.J.J.M. Ariaans. *Structure in Practical Model Error Bounds.* PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 1997.

[6] Inc. Aspen Technology. Webpagina. *www.aspentec.com*, 1999.

[7] K.J. Åstrom and B. Wittenmark. *Computer Controlled Systems – Theory and Practice.* Prenctice-Hall Inc., 2nd edition, 1990.

[8] A. Bemporad, A. Casavola, and E. Mosca. Nonlinear control of constrained linear systems via predictive reference management. *IEEE Trans. Autom. Control*, 42(3):340–349, 1997.

[9] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Technical report AUT99-16, ETH Zürich*, 1999.

[10] R.R. Bitmead, M. Gevers, and V. Wertz. *Adaptive Optimal Control-The Thinking Man's GPC.* Prentice-Hall Inc., Englewood Cliffs, NJ, 1990.

[11] A. Bjorck. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, PA, 1997.

[12] B.D. Bojanov, H.A. Hakopian, and A.A. Sahakian. *Spline Functions and Multivariate Interpolations,* Mathematics and its applications Vol. 248. Kluwer, Dordrecht, 1993.

[13] S. Boyd and C. Barrat. *Linear Controller Design, Limits of Performance*. Prentice-Hall, 1991.

[14] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in Systems and Control Theory*. SIAM, Philadelphia, PA, 1994.

[15] P.J. Campo and M. Morari. ∞-norm formulation of model predictive control problems. *Proc. American Contr. Conf., Seattle, Washington*, pages pp 339–343, 1986.

[16] P.J. Campo and M. Morari. Robust control of processes subject to saturation nonlinearities. *Comp. Chem. Engng.*, 14(4):343–358, 1990.

[17] H. Chen. Stability and robustness considerations in nonlinear model predictive control. *Fortschr.-Ber. VDI Reihe 8, Nr. 674 VDI Verlag, Düsseldorf*, 1997.

[18] L. Chisis and E. Mosca. Stabilizing I-O receding horizon control of CARMA plants. *IEEE Transactions on Automatic Control*, 39(3):614–618, 1994.

[19] D.W. Clarke. Advances in model-based predictive control. In: *Advances in Model-Based Predictive Control, Oxford Univ. Press*, pages 3–21, 1994.

[20] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part i. the basic algorithm. *Automatica*, 33(2):137–148, 1987.

[21] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part ii. extensions and interpretations. *Automatica*, 23(2):149–160, 1987.

[22] C.R. Cutler. Dynamic matrix control of imbalanced systems. *ISA Transactions*, 21(1):1–6, 1982.

[23] C.R. Cutler and R.B. Hawkins. Application of a large predictive multivariable controller to a hydrocracker second stage reactor. *Proc. American Contr. Conf.,Atlanta, GA*, pages 284–291, 1988.

[24] C.R. Cutler and B.L. Ramaker. Dynamic matrix control - a computer control algorithm. *Proc. Automatic Control Conf.,Paper WP5-B*, 1980.

[25] I. Daubechies. *Ten Lectures on Wavelets,* volume 61 of CBMS Lecture notes. SIAM, Philadelphia, 1992.

[26] D.K. De Vries and P.M.J. Van den Hof. Frequency domain identification with generalized orthonormal basis functions. *IEEE Trans. Autom. Control*, 43(3), March 1998. Also in Proc. 34th IEEE Conf. Decision and Control, New Orleans, LA, pp. 1240-1245.

[27] A.C. den Brinker, F.P.A. Benders, and T.A.M. Oliveira e Silva. Optimality conditions for truncated Kautz series. *IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing*, 43(2):117–122, February 1996.

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

[29] E.T. van Donkelaar, O.H. Bosgra, and P.M.J. Van den Hof. Model predictive control with generalized input parametrization. In: Bosgra et al. (Eds.), *Sel. Topics in Identification, Modelling and Control,* Delft University Press, 11:69–78, 1998.

[30] E.T. van Donkelaar, O.H. Bosgra, and P.M.J. Van den Hof. Model predictive control with generalized input parametrization. *Proc. of European Control Conference, Karlsruhe, Germany. Paper code CP-12*, 1999.

[31] E.T. van Donkelaar, O.H. Bosgra, and P.M.J. Van den Hof. Model predictive control with on-line input parametrization. *Proc. of Conference on Decision and Control 1999, Phoenix AZ, United States*, 1999.

[32] E.T. van Donkelaar and P.M.J. Van den Hof. Analysis of closed-loop identification with a tailor-made parametrization. *European Journal on Control*, 6(1), 2000.

[33] E.T. van Donkelaar, P.S.C. Heuberger, and P.M.J. Van den Hof. Identification of a fluidized catalytic cracking unit: an orthonormal basis function approach. In: O.H. Bosgra *et al.* (Eds.), *Sel. Topics in Id., Mod. and Contr.,* Delft Univ. Press, 10:1–8, 1997.

[34] E.T. van Donkelaar, P.S.C. Heuberger, and P.M.J. Van den Hof. Identification of a fluidized catalytic cracking unit: an orthonormal basis function approach. *Proceedings Amer. Contr. Conf.*, pages 1914–1917, 1998.

[35] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Trans. Autom. Control*, 34(8):831–847, 1989.

[36] N.S. Draper and R.C. Van Nostrand. Ridge regression and james-stein estimation: review and comments. *Technometrics*, 21(8):451–466, 1979.

[37] G.A. Dumont, Y. Fu, and A.L. Elshafi. Orthonormal functions in identification and adaptive control. In *Proc. IFAC Intern. Symp. Intelligent Tuning and Adaptive Control*, Singapore, 1990.

[38] A. Elshafei, G.A. Dumont, and A. Elnagger. "Adaptive GPC based Laguerre-filters modelling". *Automatica*, 30:19913–1920, 1994.

[39] D. Enns. *Model Reduction for Control Systems Design*. PhD thesis, Department of Aeronautics and astronautics, Stanford University, California, 1984.

[40] R. Fletcher. *Practical Methods of Optimization*. Wiley, Chichester, 1987.

[41] J.B. Froisy. Model predictive control: past, present and future. *ISA Transactions 22*, pages 235–243, 1994.

[42] J.B. Froisy and J. Richalet. Industrial applications of IDCOM. In T.J. McAvoy, Y. Arkun and E. Zafiriou (editors), *Model-Based Proces Control of the 1988 IFAC Workshop, Georgia, USA*, pages 233–244, 1986.

[43] C.E. Garcia and M. Morari. Internal model control 1. a unifying review and some new results. *Ind. Eng. Chem Process Des. Dev.*, 21:308–323, 1982.

[44] C.E. Garcia and M. Morari. Internal model control 2. design procedure for multivariable systems. *Ind. Eng. Chem Process Des. Dev.*, 24:472–484, 1985.

[45] C.E. Garcia and M. Morari. Internal model control 3. multivariable control law computation and tuning guidelines. *Ind. Eng. Chem Process Des. Dev.*, 24:484–494, 1985.

[46] C.E. Garcia and M. Morari. Model predictive control: theory and practice - a survey. *Automatica*, 25(3):335–348, 1989.

[47] H. Genceli and M. Nikolaou. Robust stability analysis of constrained l_1-norm model predictive control design. *AIChE Journal*, 39(12):1954–1959, 1993.

[48] M.R. Gevers. Towards a joint design of identification and control? In: *Essays on Control: Perspectives in the Theory and its Applications.* H.L. Trentelman and J.C. Willems (Eds.). Birkhäuser, Boston., pages 111–115, 1993.

[49] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, New York, 1981.

[50] K. Godfrey. *Perturbation Signals for System Identification*. Prentic-Hall Inc., 1993.

[51] G.H. Golub and C.F. van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, 3rd edition, 1996.

[52] V. Gopal and L.T. Biegler. Large scale inequality constrained optimization and control. *IEEE Constrol Systems Magazine*, 18(6):59–68, 1998.

[53] P. Grossdidier, B. Froisy, and H. Hammann. The IDCOM-M controller. In T.J. McAvoy, Y. Arkun and E. Zafiriou (editors), *Model Based Process Control of the 1988 IFAC Workshop, Georgia, USA*, pages 31–36, 1988.

[54] R.G. Hakvoort and P.M.J. Van den Hof. Identification of probabilistic uncertainty regions by explicit evaluation of bias and variance errors. *IEEE Trans. Autom. Control*, 42(11), November 1997.

[55] A.J. Helmicki, C.A. Jacobson, and C.N. Nett. Identification in $\mathcal{H}_\infty$: linear algorithms. In *Proc. American Control Conf.*, pages 2418–2423, San Diego, CA, 1990.

[56] P. S. C. Heuberger, P. M. J. Van den Hof, and O. H. Bosgra. A generalized orthonormal basis for linear dynamical systems. In *Proceedings of the 32nd Conference on Decision and Control*, volume 3, pages 2850–2855, San Antonio, Texas, 1993. IEEE.

[57] P.S.C. Heuberger. *On Approximate System Identification With System Based Orthonormal Functions*. PhD thesis, Delft University of Technology, 1991.

[58] P.S.C. Heuberger and P.M.J. Van den Hof. The Hambo transform: A signal and system transform induced by generalized orthonormal basis functions. In *Preprints of the* 13*-th IFAC World Congress*, volume I, pages 357–362, San Francisco, CA, 1996.

[59] P.S.C. Heuberger, P.M.J. Van den Hof, and O.H. Bosgra. A generalized orthonormal basis for linear dynamical systems. *IEEE Transactions on Automatic Control*, 40(3):451–465, March 1995.

[60] H. Hjalmarsson, M.R. Gevers, and F. de Bruyne. For model based control design, closed loop identification gives better performance. *Automatica*, 12:1659–1673, 1996.

[61] B.L. Ho and R.E. Kalman. Effective construction of linear state-variable models from input/output functions. *Regelungstechnik*, 14(12):545–592, 1966.

[62] H. Jaddu and E. Shimemura. Computation of optimal control trajectories using Chebychev polynomials. *Optimal Control, Applications and Methods*, 20(1):21–42, 1999.

[63] E.A. Jonckheere and L.M. Silverman. A new set of invariants for linear systems - application to reduced order compensator design. *IEEE Trans. Autom. Control*, 28(10):953–964, 1983.

[64] R.E. Kalman. Contributions to the theory of optimal control. *Bull. Soc. Math. Mex.*, 5:102–119, 1960.

[65] W.H. Kautz. Network synthesis for specified transient response. Technical Report 209, Massachusetts Institute of Technology, Research Laboratory of Electronics, April 1952.

[66] W.H. Kautz. Transient synthesis in the time domain. *IRE Transactions on Circuit Theory*, 1:29–39, 1954.

[67] S.S. Keerthi and E.G. Gilbert. Optimal infinite-horizon feedback law for a general class of constrained discrete-time systems: stability and moving-horizon approximations. *J. Optim. Theory Appl.*, 57(2):265–293, 1988.

[68] P. Kokotovic, A. Bensoussan, and G. Blankenship. *Singular Perturbations and Asymptotic Analysis in Control Systems*. Lecture Notes in Control and Information Sciences, Springer-Verlag, 1986.

[69] M.V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

[70] E. Kreyszig. *Introductory Functional Analysis with Applications*. John Wiley & Sons, New York, 1978.

[71] S Kung. A new identification and model reduction algorithm via singular value decompositions. *Proc. 12th Asilomar conf. Circuits Syst. and Computers, Pacific Grove, CA*, pages 705–714, 1978.

[72] H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, New York, 1972.

[73] W.H. Kwon and D.G. Byun. Receding horizon tracking control as a predictive control and its stability properties. *Int. J. Control*, 50(5):1807–1824, 1989.

[74] W.H. Kwon and A.E. Pearson. A modified quadratic cost problem and feedback stabilization of a linear system. *IEEE Transactions on Automatic Control*, AC-22(5):838–842, 1977.

[75] W.E. Larimore. System identification of feedback and "causality" structure using canonical variate analysis. Proceedings: *IFAC Symposium on System Identification,* Fukuoka, Japan, pages 1101–1106, 1997.

[76] H.L. Lee, M. Morari, and C.E. Garcia. State-space interpretation of model predictive control. *Automatica*, 30(4):707–717, 1994.

[77] J.H. Lee and Z.H. Yu. Tuning of model predictive controllers for robust performance. *Comp. Chem. Engng.*, 18(1):15–37, 1994.

[78] S. Li, K.Y. Lim, and D.G. Fischer. A state space formulation for model predictive control. *AIChE journal*, 35(2):241–249, 1989.

[79] W. Li and J. Swetits. A new algorithm for solving strictly convex quadratic programs. *SIAM J. Optim.*, 7(3):595–619, 1997.

[80] P. Lindskog. *Methods, Algorithms and Tools for System Identification Based on Prior Knowledge*. PhD thesis, Linköping University, May 1996.

[81] L. Ljung. *System Identification. Theory for the user*. Prentice-Hall, Englewood Cliffs, NJ, 1987.

[82] L. Ljung and T. McKelvey. A least squares interpretation of sub-space methods for system identification. *Proc. 35th IEEE Conf. Dec. and Control,* Kobe, Japan, pages 335–342, 1996.

[83] L. Ljung and Z.D. Yuan. Asymptotic properties of black-box identification of transfer functions. *IEEE Trans. Autom. Control*, 30:514–530, 1985.

[84] D.G. Luenberger. *Optimization by Vector Space Methods*. John Wiley, New York, 1969.

[85] D.G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley Pub. Comp., Reading, Massachusetts, 1984.

[86] J.M. Maciejowski. Reconfiguring control systems by optimization. *Proc. 4 th European Control Conference, Brussels*, 1997.

[87] P. Marquis and J.P. Broustail. SMOC, a bridge between state space and model predictive controllers: application to automation of a hydrotreating unit. In T.J. McAvoy, Y. Arkun and E. Zafiriou (editors), *Model-Based Process Control of the 1988 IFAC Workshop, Georgia, USA*, pages 37–45, 1988.

[88] D.Q. Mayne and W.R. Schroeder. Nonlinear control of constrained linear systems. *Int. J. Control*, 60(5):1035–1043, 1994.

[89] R.C. McFarlane, R.C. Reineman, J.F. Bartee, and C. Georgakis. Dynamic simulator for a model IV fluid catalytic cracking unit. *Computers Chem. Engng.*, 17(3):275–300, 1993.

[90] R.K. Mehra, R. Rouhani, J. Eterno, J. Richalet, and A. Rault. Model algoritmic control (MAC); review and recent developments. *Eng. Foundation Conf. Chem. Proc. Contr. II, Sea Island, GA*, pages 287–310, 1982.

[91] R.K. Mehra, R. Rouhani, A. Rault, and J.G. Reid. Model algoritmic control: theoretical results on robustness. *Proc. American Contr. Conf., Denver, Collorado*, pages pp 387–392, 1979.

[92] B.C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, February 1981.

[93] M. Morari and E. Zafiriou. *Robust Process Control*. Prentice-Hall, Englewood Cliffs, NJ, 1989.

[94] A.M. Morshedi, C.R. Cutler, and T.A. Skrovanek. Optimal solution of dynamic matrix control with linear programming techniques (LDMC). *Proc. American Contr. Conf., Boston, Massachusetts*, pages pp 199–208, 1985.

[95] K.R. Muske and J.B. Rawlings. Linear model predictive control of unstable processes. *J. Proc. Cont.*, 3(2):85–96, 1993.

[96] K.R. Muske and J.B. Rawlings. Model predictive control with linear models. *AIChE Journal*, 39(2):262–287, 1993.

[97] S.G. Nash and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill Inc., USA, 1996.

[98] Y. Nesterov and A. Nemirovski. *Interior Point Algorithms in Convex Programming*. SIAM, Piladelphia, PA, 1994.

[99] A. Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review*, 40:636–666, 1998.

[100] V. Nevistić. Optimal control: a review. *Technical report AUT 97-05, Automatic Control Laboratory, ETH Zürich, Switzerland*, 1997.

[101] V. Nevistić and M. Morari. Constrained control of feedback linearizable systems. In: *Proc. 3rd European Control Conference ECC'95, Rome*, pages 1726–1731, 1995.

[102] B. Ninness and J. Goméz. Asymptotic analysis of MIMO system estimates by the use of orthonormal bases. In *Proceedings of the 13th IFAC World Congress, San Francisco, USA*, pages 363–368, 1996.

[103] B. Ninness, J. Goméz, and S. Weller. MIMO system identification using orthogonal basis functions. In *Proceedings of the 34th Conf. Dec. Control, New Orleans, LA*, pages 703–708, 1995.

[104] B. Ninness and F. Gustafsson. A unifying construction of orthonormal bases for system identifcation. *IEEE Transactions on Automatic Control*, 42(4):515–521, April 1997.

[105] B. Ninness and H. Hjalmarsson. Modelling of random processes using orthonormal bases. Technical Report EE9618, 1996.

[106] B. Ninness and H. Hjalmarsson. Model structures and numerical properties of normal equations. Technical Report EE9801, 1998.

[107] B. Ninness, H. Hjalmarsson, and F. Gustafsson. Bias and variance analysis of estimation using fixed denominator model structures. Technical Report EE9617, 1996.

[108] B.M. Ninness and J.C. Gómez. Asymptotic analysis of mimo systems estimates by the use of othonormal bases. In *Preprints of the* 13*-th IFAC World Congress*, volume I, pages 363–368, San Francisco, CA, 1996.

[109] B.M. Ninness and G.C. Goodwin. Estimation of model quality. *Automatica*, 31(12):1771–1797, December 1995.

[110] B.M. Ninness and F. Gustafsson. A unifying construction of orthonormal bases for system identification. *IEEE Transactions on Automatic Control*, 42(4):515–521, April 1997.

[111] B.M. Ninness, H. Hjalmarsson, and F. Gustafsson. On the fundamental role of orthonormal bases in system identification. *IEEE Trans. Autom. Control*, 47(8), 1999.

[112] T. Oliveira e Silva. Optimality conditions for truncated Laguerre networks. *IEEE Transactions on Signal Processing*, 42(9):2528–2530, September 1994.

[113] T. Oliveira e Silva. Optimality conditions for truncated Kautz networks with two periodically repeating complex conjugate poles. *IEEE Transactions on Automatic Control*, 40(2):342–346, February 1995.

[114] T. Oliveira e Silva. A n-width result for the generalized orthonormal basis function model. In *Preprints of the* 13*-th IFAC World Congress*, volume I, pages 375–380, San Francisco, CA, USA, 1996.

[115] T. Oliveira e Silva. Stationarity conditions for the L^2 error surface of the generalized orthonormal basis functions lattice filter. *Signal Processing*, 56(3):233–253, February 1997.

[116] A.V. Oppenheim, A.S. Willsky, and I.T. Young. *Signals and systems*. Prentice-Hall International, Inc., London, 1983.

[117] P. Overschee. *Subspace identification, Theory - Implementation - Application*. PhD Thesis. Katolieke Universiteit Leuven, Leuven, Belgium, 1995.

[118] R.S. Patwardhan, S. Lakshminarayanan, and S.L. Shah. Constrained nonlinear mpc using Hammerstein and Wiener models: PLS framework. *AICHE JOURNAL*, 44(7):1611–1622, 1998.

[119] A. Pinkus. *n-Widths in approximation theory*. Springer-Verlag, New York, 1985.

[120] R. Pintelon, P. Guillaume, Y. Rolain, J. Schoukens, and H. Van Hamme. Parametric identification of transfer functions in the frequency domain - a survey. *IEEE Trans. on Autom. Control*, 39:2245–2260, 1994.

[121] R. Pintelon, J. Schoukens, and G. Vandersteen. Frequency domain system identification using arbitrary signals. *IEEE Transactions on Automatic Control*, 42(12):1717–1720, December 1997.

[122] E. Polak and T.H. Yang. Moving horizon control of linear systems with input saturation and plant uncertainty. part 1. robustness. *Int. J. Control*, 58(3):613–638, 1993.

[123] K. Poolla, P. Kargonekar, A. Tikku, J. Krause, and K. Nagpal. A time-domain approach to model validation. *IEEE Trans. Autom. Control*, 39:951–959, 1994.

[124] K.R. Popper. *Conjectures and Refutations: The Growth of Scientific Knowledge*. Routledge, London, 1963.

[125] M. Pottmann and D.E. Seborg. A nonlinear predictive control strategy based on radial basis function models. *Computers Chem. Engng.*, 21(9):965–980, 1997.

[126] J.A. Primbs and V. Nevistic. Constrained finite receding horizon linear quadratic control. *Technical Memorandum No. CIT-CDS 97-002,* CalTech, Pasadena, California, 1997.

[127] S.J. Qin and T.A. Badgwell. An overview of industrial model predictive control technology. *AIChE CPC IV*, 1997.

[128] J.B. Rawlings and K.R. Muske. The stability of constrained receding horizon control. *IEEE Trans. Autom. Control*, AC-38(10):1512–1516, 1993.

[129] J.B. Rawlings and K.R. Muske. Nonlinear model predictive control: A tutorial and survey. In: *Advanced Control of Chemical Processes.* IFAC Symposium, Kyoto, Japan, pages 203–214, 1994.

[130] J. Richalet. Industrial applications of model based predictive control. *Automatica*, 29(5):1251–1274, 1993.

[131] J. Richalet, S.A. El Ata-Doss, C. Arber, H.B. Kuntze, A. Jacubasch, and W. Schill. Predictive functional control application to fast and accurate robots. *Proc. 10th IFAC world congres, Munich, Germany*, 1987.

[132] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heuristic control: applications to industrial processes. *Automatica*, 14:413–428, 1978.

[133] N.L. Ricker. Model predictive control with state estimation. *Ind.Eng.Chem.Res.*, 29:374–382, 1990.

[134] N.L. Ricker, T. Subrahmanian, and T. Sim. Case studies of model-predictive control in pulp and paper production. *In T.J. McAvoy, Y. Arkun and E. Zafiriou (editors), Model based proces control of the 1988 IFAC Workshop, Georgia, USA*, pages 13–22, 1988.

[135] R.A. Roberts and C.T. Mullis. *Digital Signal Processing*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1987.

[136] J.A. Rossiter, B. Kouvaritakis, and J.R. Gossner. Feasibility and stability results for constrained stable generalized predictive control. *Automatica*, 31(6):863–877, 1995.

[137] J.A. Rossiter, M.J. Rice, J. Schuurmans, and B. Kouvaritakis. A computationally efficient constrained predictive control law. *Proceedings 1998 American Contr. Conf., Philadelphia, PA*, 1998.

[138] C. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via LMI optimization. *IEEE Trans. Autom. Control*, 42(7):896–911, 1997.

[139] P.O.M. Scokaert and J.B. Rawlings. On infeasibilities in model predictive control. In: *Chemical Proces Control - V, Assessment and new directions for research,* Tahoe City, CA, USA, 1996.

[140] P.O.M. Scokaert and J.B. Rawlings. Constrained linear quadratic regulation. *IEEE Trans. Automat. Contr.*, 43(8):1163–1169, 1998.

[141] P.O.M. Scokaert and J.B. Rawlings. Constrained lqr regulation. *IEEE Trans. Autom. Control*, AC-43(8):1163–1169, 1998.

[142] P.O.M. Scokaert, J.B. Rawlings, and E.S. Meadows. Discrete-time stability with perturbations: application to model predictive control. *Automatica*, 33(3):463–470, 1997.

[143] T. Söderström. Convergence properties of the generalized least squares identification method. *Automatica*, 10:617–626, 1974.

[144] T. Söderström. *Discrete-time stochastic systems: estimation and control*. Prentice-Hall Int., Hemper Hempstead, Hertfordshire, 1994.

[145] T. Söderström and P. Stoïca. *System identification*. Prentice-Hall Int., Hemper Hempstead, Hertfordshire, 1989.

[146] H.J. Sussman and J.C. Willems. 300 years of optimal control: from the brachystochrone to the maximum principle. *IEEE Control Systems Magazine*, 17:32–44, 1997.

[147] G. Szegö. *Orthogonal Polynomials*, volume 23 of *Colloquium Publications*. American Mathematical Society, Providence, Rhode Island, 4th edition, 1978.

[148] F.Z. Tatrai, P.A. Lant, P.L. Lee, I.T. Cameron, and R.B. Newell. Model reduction for regulatory control: an fccu case study. *Trans. IChemE*, 72(Part A):402–407, 1994.

[149] A.N. Tikhonov. *Solutions of ill-posed problems*. Scripta series in mathematics, Washington, Winston, 1977.

[150] H.J.A.F. Tulleken. Generalized binary noise test-signal concept for improved identification experiment design. *Automatica*, 26(1):37–49, 1990.

[151] P.M.J. Van den Hof, P.S.C. Heuberger, and J. Bokor. System identification with generalized orthonormal basis functions. *Automatica*, 31(12):1821–1834, December 1995.

[152] P.M.J. Van den Hof and R.J.P. Schrama. An indirect method for transfer function estimation from closed loop data. *Automatica*, 29(6):1523–1527, 1993.

[153] P.M.J. Van den Hof and R.J.P. Schrama. Identification and control - closed loop issues. *Automatica*, 31(12):1751–1770, 1995.

[154] P.M.J. Van den Hof, R.J.P. Schrama, R.A. de Callafon, and O.H. Bosgra. Identification of normalised coprime plant factors from closed-loop experimental data. *European Journal of Control*, 1(1):62–74, 1995.

[155] J.B. Van Helmont, A.J.J. Van der Weiden, and H. Anneveld. *Design of optimal controller for a coal fired Benson boiler based on a modified approximate realization algorithm*. Elsevier Publ. Comp., London, 1990.

[156] V. Vassiliadis. *Computational solution of dynamic optimization problems with general differential-algebraic equations*. PhD thesis, University of London, London, UK, 1993.

[157] M. Verhaegen and P. Dewilde. Subspace model identification, part 1 and 2. *Int. J. Contr.*, 56(5):1187–1210 and 1211–1241, 1993.

[158] M. Viberg. Subspace-based methods for the identification of linear time-invriant systems. *Automatica*, 31(12):1835–1851, 1995.

[159] B. Wahlberg. Model reductions of high-order estimated models: the asymptotic ML approach. *International Journal of Control*, 49(1):169–192, 1989.

[160] B. Wahlberg. System identification using Laguerre models. *IEEE Transactions on Automatic Control*, 36(5):551–562, May 1991.

[161] B. Wahlberg. System identification using Kautz models. *IEEE Transactions on Automatic Control*, 39(6):1276–1282, June 1994.

[162] B. Wahlberg and P.M. Mäkilä. On approximation of stable linear dynamical systems using Laguerre and Kautz functions. *Automatica*, 32(5):693–708, May 1996.

[163] J.C. Willems. From time series to linear system - part i. finite dimensional linear time invariant systems. *Automatica*, 1986.

[164] J.C. Willems. From time series to linear system - part ii. exact modelling. *Automatica*, 1986.

[165] J.C. Willems. From time series to linear system - part iii. approximate modelling. *Automatica*, 1987.

[166] J.C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Automatic Control*, 1991.

[167] P.M.R. Wortelboer. *Frequency Weighted Balanced Reduction of Closed-Loop Mechanical Servo Systems: Theory and Tools*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 1993.

[168] S.J. Wright. Applying new optimization algorithms to model predictive control. *Conf.Chemical Process Control V, Lake Tahoe, CA*, pages 147–155, 1996.

[169] S.J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, 1997.

[170] Fu Y. and Dumont G.A. An optimum time scale for discrete laguerre network. *IEEE Transactions on Automatic Control*, 38(6):934–938, June 1993.

[171] E. Zafiriou. Robust model predictive control of processes with hard constraints. *Computers Chem. Engng.*, 14(4):359–371, 1990.

[172] G. Zames. Feedback and optimal sensitivity: model reference transformations, multplicative seminorms and approximative inverses. *IEEE Trans. Automatic Control*, 26(2):301–320, 1981.

[173] A. Zheng, R. Balakrishnan, and M. Morari. Constrained stabilization of discrete -time systems. *Int. J. Robust and Nonlinear Contr.*, 5:461–485, 1995.

[174] A. Zheng and M. Morari. Stability of model predictive control with mixed constraints. *IEEE Trans. Autom. Control*, 40(10):1818–1823, 1995.

[175] K. Zhou, J.C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice-Hall Inc., Upper Saddle River, New Jersey, 1996.

[176] Y.C. Zhu. Black-box identification of MIMO transfer functions: asymptotic properties of prediction error models. *Int. J. Adaptive Control and Signal processing*, 3:357–373, 1989.

[177] Y.C. Zhu. Multivariable process identification for MPC: the asymptotic method and its applications. *J. Proc. Contrl.*, 8(2):101–115, 1998.

[178] Y.C. Zhu and A.C.P.M. Backx. *Identification of Multivariable Industrial Processes - For Simulation, Diagnosis and Control*. Advances in industrial control. Springer Verlag, 1993.

Samenvatting

Verbetering van de efficiëntie van identificatie en model predictive conrol van industriële processen

Een flexibele lineaire parametrizatie aanpak

Een optimalisatieprobleem dient zowel in model predictive control (MPC) als in systeemidentificatie (ID) opgelost te worden. In MPC dient een optimaal stuursignaal over een bepaalde tijdhorizon in de toekomst bepaald te worden, in ID is een optimaal model van de systeemdynamica gewenst. In beide optimalisatieproblemen is een intelligente keuze van de vrijheidsgraden van groot belang. Beide optimalisatieproblemen zijn convex voor kwadratische kostenfunctie, wat gebruikelijk is in beide velden, en een lineare parametrizatie van respectievelijk het stuursignaal en het model. Er is grote vrijheid in de keuze van de lineaire parametrizatie. In dit proefschrift wordt onderzocht in hoeverre een linear parametrizatie bij kan dragen aan de verbetering van de efficientie van MPC en ID.
In predictiefout ID dient een niet-convex optimalizatieprobleem opgelost te worden om een optimaal model te vinden voor de systeem- en verstoringsdynamica. Het probleem is niet-convex door de parametrizatie van het model. Het optimalisatieprobleem is over het algemeen niet oplosbaar voor industriële processen, die vaak veel ingangen en uitgangen hebben. Aan de andere kant levert een lineare parametrizatie van het model een convex optimalisatieprobleem, maar dit kan leiden tot overdimensionering, dat wil zeggen dat veel parameters nodig zijn voor een bepaalde nauwkeurigheid wat een negatief effect heeft op de variantie van de schatting.
In dit proefschrift wordt onderzocht hoe het optimale predictiefout model, die het resultaat is van een niet-convexe optimalisatie, benaderd kan worden in termen van systeemgebaseerde basisfuncties, zoals Laguerre, twee-parameter Kautz en gegeneraliseerde basisfuncties. Zo wordt slechts een beperkt aantal parameters gebruikt waardoor een model verkregen wordt met een kleine bias en variantie. Ook dient de procedure een beperkte rekentijd te hebben, gezien de grootschaligheid van industriële identificatieproblemen.
Systeem-gebaseerde orthogonale basisfuncties kunnen gebruikt worden als bouwstenen voor multivariabele modelstructuren, die gebruik maken van voorkennis omtrent systeemdynamica en structuur. Hoe nauwkeuriger de voorkennis, hoe hoger de nauwkeurigheid van het model en hoe lager de rekencomplexiteit. Indien alleen de systeemoverdracht geschat wordt, is het probleem een eenvoudig lineair kleinste kwadraten probleem.
Ook kunnen modelstructuren gedefinieerd worden waarin ook de verstoringsoverdracht meegeschat wordt. Een nieuwe modelstructuur, waarbij systeem- en verstoringsoverdracht onafhankelijk zijn geparametrizeerd, wordt voorgesteld en geanalyseerd. Deze modelstructuur is bilineair in de parametervector. Hiervoor wordt een alternatief optimalisatiealgoritme voorgesteld dat onder milde condities convergeert naar het globale

optimum van de kostfunctie.
De overdimensionering van de modellen zijn afhankelijk van de hoeveelheid en nauwkeurigheid van de voorkennis. Met een iteratief schema van hoge orde identificatie met orthogonale basisfuncties en modelreductie kan het aantal vrije parameters, die nodig zijn voor een zekere nauwkeurigheid, systematisch verlaagd worden. Zo kan het globale optimum van het niet-convexe optimalisatieprobleem met stijgende nauwkeurigheid benaderd worden. Een grondige convergentieanalyse van dit iteratieve schema is (nog) niet beschikbaar, maar simulatieresultaten zijn veelbelovend.
In MPC dient on-line een optimaal ingangstraject berekend te worden dat voldoet aan alle beperkingen van het systeem. MPC is een van de weinige regelstrategieën die systematisch omgaat met deze beperkingen. De on-line optimalisatie is tijdrovend, waardoor deze technologie voornamelijk toegepast wordt op systemen met een grote bemonstertijd, zoals (petro)chemische processen. Door de grote rekentijd is MPC nog niet geschikt voor snelle systemen zoals consumenten-elektronica, vliegtuigen, autos en telecommunicatienetwerken, terwijl ook deze systemen beperkingen hebben waarvan het wenselijk is dat zij door de regeling gerespecteerd worden.
De rekentijd van MPC kan, buiten de verbetering van hardware en optimalisatiesoftware, verkleind worden door de keuze van een klein aantal vrijheidsgraden in het ingangstraject. De keuze voor de parametrizatie dient zo te zijn dat een hoge performance gegarandeerd kan worden met slechts weinig vrije parameters.
In dit proefschrift worden twee nieuwe aanpakken voorgesteld om de parametrizatie van het ingangssignaal in MPC te kiezen. De eerste aanpak gebruikt de observatie dat de ruimte met alle ingangstrajecten zonder beperkingen een dimensie heeft, die gelijk is aan de modelorde. Voor MPC met een oneindige horizon wordt deze ruimte gegenereerd door een lineair dynamisch systeem. Een verdere verkleining van het aantal vrijheidsgraden kan verkregen worden door dit lineaire systeem te reduceren met standaard reduktietechnieken die aangepast worden voor de gebruikte kostfunctie van de regelaar. Met deze aanpak kan het verlies in performance gekwantificeerd worden aan de hand van de indicatoren van de gebruikte reductiemethode. Deze systeem-gebaseerde parametrizatie levert een efficiënte parametrizatie op voor situaties waarin de beperkingen van het systeem een ondergeschikte rol spelen. Daar waar de beperkingen een grote rol spelen kan een aanzienlijk verlies aan performance optreden. Daarom is een tweede aanpak ontwikkeld die gebruik maakt van voorkennis van de beperkingen op het systeem om een parametrizatie te construeren. Deze aanpak kan beschouwd worden als een uitbreiding op een quadratisch programmeerroutine die snel een laagdimensionale deelruimte selecteert waarin de optimale oplossing zich bevindt. Deze procedure kan de rekentijd aanzienlijk verkleinen.
De voorgestelde aanpak voor ID en MPC is toegepast op een multivariable niet-lineair simulatiemodel van een industriëel proces. Dit geeft een illustratie van de te verwachten verbetering in efficiëntie.

Curriculum vitae

Edwin Teunis van Donkelaar was born on July 25, 1971 in Ede, The Netherlands.

1983-1989	Pre-university education (VWO) at Marnix College, Ede, The Netherlands.
1989-1995	M.Sc. student with Mechanical Engineering at Deflt University of Technology. Graduated in the Mechanical Engineering Systems and Control Group on identification of model uncertainty of a Philips compact disc mechanism.
1995-2000	Ph.D. student with the Mechanical Engineering Systems and Control Group Delft University of Technology, sponsored by the Dutch Science Foundation (STW).
2000-	Designer at the control group of ASM Lithography B.V., Veldhoven