Julian Straus

Optimal Operation of Integrated Chemical Processes

With Application to the Ammonia Synthesis

Thesis for the degree of philosophiae doctor
Trondheim, August 2018

Norwegian University of Science and Technology
The Faculty of Natural Sciences
Department of Chemical Engineering

NTNU
Innovation and Creativity

NTNU
Norwegian University of Science and Technology

Thesis for the degree of philosophiae doctor

The Faculty of Natural Sciences
Department of Chemical Engineering

ISBN 978-82-326-3234-3 (printed version)
ISBN 978-82-326-3235-0 (electronic version)
ISSN 1503-8181

Doctoral theses at NTNU, 2018:222

1st edition, August 2018

Printed by Xerox

Dedicated to my parents.

Summary

Chemical processes have to be operated at their economic optimum to remain competitive. This is called optimal operation. Optimal operation is frequently incorporated using a real-time optimization approach. In this approach, a model of the process is updated using plant and disturbance measurement and subsequently optimized to calculate the setpoints of the controller.

The competition in the bulk chemical industry furthermore dictates the necessity of heat and mass integration. As a consequence, it is more and more difficult to obtain a model of the overall process which can be used in real-time optimization. This thesis is therefore applying and developing methods to achieve optimal operation in the case of integrated chemical processes.

If it is not possible to obtain a detailed model for real-time optimization of the overall process, it is only natural to try to achieve optimal operation for subprocesses. This can be achieved either through the application of real-time optimization for the respective subprocesses or through process control. The first part of this thesis is investigating different approaches to obtain optimal operation for said subprocesses. Economic nonlinear model predictive control is one approach based on online dynamic optimization. In addition to converging to an optimal operation point, economic nonlinear model predictive control follows the optimal trajectory to this point. Due to the potential complicated optimization problem and problems associated with plant-model mismatch, self-optimizing control in itself and in a hierarchical combination with extremum-seeking control is subsequently applied to the same case study. When disturbances occur, self-optimizing control is keeping the operation close to the optimum whereas extremum-seeking control adjusts the setpoints to the self-optimizing variables to remove the steady-state loss of self-optimizing control. This allows achieving optimal operation without the necessity of a detailed model and reduces the impact of plant-model mismatch and the solution time of the optimization problem in economic nonlinear model predictive control. Feedback real-time optimization as a novel alternative transforms the optimization problem of conventional real-time optimization to a control problem. This allows a fast response

to disturbances and removes problems associated with the cost measurement and gradient estimation in extremum-seeking control.

The second part of this thesis is developing methods for surrogate model generation. Surrogate models are computational cheap regression models of computational expensive detailed models. They can be used in the context of real-time optimization to reduce the computational load to solve the optimization problem. The separation of the initial process model into subprocesses results in models that are computational cheaper to solve. Surrogate models are subsequently fitted to the subprocesses and combined into a surrogate model flowsheet. The optimization is then performed using the surrogate model flowsheet. Using surrogate models, it is possible to perform a variable transformation. Partial least squares regression allows a reduction in the number of independent variables through the calculation of new, latent variables. The application of self-optimizing variables in the generation of surrogate models results in a simplified surface of the detailed model. The simpler surface requires then fewer points to achieve a satisfactory fit of the surrogate model. Partial least squares regression can be used as a termination criterion in sampling without the need of fitting a surrogate model at each sampling iteration step as well. This results in a reduction of the computational load in sampling for surrogate model generation. The surrogate model flowsheet can then be used in real-time optimization.

Acknowledgement

This Ph.D. thesis would not have been possible without the help of many people. First and foremost, I would like to thank my supervisor, Sigurd Skogestad, for offering me the possibility to pursue a Ph.D. in his research group. Without him, I would not have the possibility to come to the beautiful city of Trondheim and conduct a Ph.D. in the field of process systems engineering. I learned a lot in the last 3.5 years about process modelling and control. I would also like to thank my co-supervisor Johannes Jäschke for his support. This brings me to the third big contributor of the thesis; thanks to Yara International ASA for part-financing this thesis and giving me the chance to work on a challenging but interesting topic within the fertilizer industry. Special thanks goes to Dr. Knut Wiig Mathisen, Dr. Bjørn Glemmestad, Guro Mauset, and Stian Dreyer Jonskås for constructive telephone conferences and help in the modelling phase as well as providing peculiarities of the process itself.

In the Department of Chemical Engineering at NTNU, I would like to thank my former project and master students Martin Bland, Kun Wang, Petter Bjørnstad Markussen, Halvor Aarnes Krogh, Harro Bonnowitz, and Rebecca Gullberg for contributing parts of the thesis. The colleagues I worked with enriched my life in Trondheim a lot. A special thanks goes to my office mates, Adriana, Pablo, Mandar, and Cristina as well as the exchange Ph.D. students Daniela, Thomas, and Prathak. We had always a nice and productive atmosphere in the office. With the other colleagues from the Process Systems Engineering Group, Vinicius, Vlad, Chriss, Pedro, Bahareh, Tamal, Timur, Dinesh, Adriaen, Eka, Christoph, Tobias, Cansu, and Sigve, I had always a good relationship with you and we enjoyed events outside of the university to free our mind of research.

Finally I would like to thank my parents in supporting me during my entire studies. Despite the longer distance and less frequent visits, they supported me in the decision to move to Norway.

Julian Straus
Trondheim, August 2018

Contents

List of Figures

List of Tables

Abbreviations

ALAMO	Automated learning of algebraic models for optimization
CSTR	Continuous stirred-tank reactor
CV	Controlled variable
EKF	Extended Kalman filter
ESC	Extremum-seeking control
F-RTO	Feedback steady-state real-time optimization
LHS	Latin hypercube sampling
MIQP	Mixed integer quadratic programming
MV	Manipulated variable
NCO	Necessary conditions of optimality
(E-)NMPC	(Economic) Nonlinear model predictive control
PI(D)	Proportional-integral(-derivative)
PLS(R)	Partial least square (regression)
(D)RTO	(Dynamic) Real-time optimization
RMSE	Root mean squared error
SIMC	Simple internal model control
SOC	Self-optimizing control

Part I

Introduction and Preliminaries

Chapter 1

Introduction

The following chapter outlines the motivation for the research conducted in this project. Furthermore, it gives the scope and the structure of the thesis. It finalizes with the main contributions of the thesis and the publications written in the course of the Ph.D. studies.

1.1 Motivation for and Scope of this Thesis

Chemical processes in mature industries become more and more integrated. El-Halwagi [26] defines process integration as "a holistic approach to process design, retrofitting, and operation, which emphasizes the unity of the process". This implies that excess energy and mass is utilized within the process. As a result, many recycle streams are present. The reason for energy and mass integration is the ever increasing competition, environmental constraints, and small profit margins in the chemical industry. A downside of this integration is that it is difficult to achieve optimal operation as it leads to complicated optimization problems. The available software for simulating chemical processes complicates the optimization further.

Chemical processes are frequently modelled using flowsheeting software. This allows the utilization of detailed thermodynamics and model equations. The available software can be separated into two categories: sequential-modular and equation-oriented simulator [12]. Sequential-modular simulators treat the individual unit operation independently as self-containing blocks including the thermodynamic calculation. As a result, solving a unit operation is fast and simple. If a recycle is present, this approach unfortunately requires the introduction of tear streams, which have to be converged on a higher level. In the case of an integrated plant, several mass and energy recycles are present. This leads to a poor convergence for solving the flowsheet. Consequently, it is difficult to use this type of simulators in optimization. Equation-oriented simulators try to avoid con-

vergence problems given by the recycles by combining all equations in one system of nonlinear equations. Hence, the recycle streams are solved simultaneously with the balances and thermodynamic calculations of the process. This increases the usefulness of the simulators in optimization. However, it is difficult to initialize the system of equations. Consequently, it is difficult to use flowsheeting software in real-time applications.

As a first alternative, the infeasible path algorithm developed by Biegler and Hughes [11] tried to circumvent problems associated with recycle streams. They achieved this by moving the convergence of the recycle streams to the optimization layer. In the case of a large number of recycles, this can be however complicated due to the large number of independent variables.

A second alternative can be seen in optimal operation of subprocesses. This approach achieves optimal operation for parts of the processes with the aim that the overall process is then at the optimal operation point as well if each subprocess is at its respective optimum.

A third alternative is given by the application of surrogate models for unit operations or subprocesses to simplify the complete flowsheet. Surrogate models (also known as response surface models, metamodels, or reduced order models) can be seen as input-output relationships for given sampled data. The simplification can be achieved through the substitution of computationally expensive unit operation by surrogate models.

Based on the state-of-the-art approaches for optimal operation, the scope of this thesis is to investigate and develop novel approaches for achieving optimal operation of integrated chemical processes. This includes both optimal operation of parts of the process and optimization of the overall process through the application of surrogate models.

1.2 Structure and Summary

This thesis is structured into four parts.

Part I provides the introduction into the thesis. Chapter 2 gives an overview of the ammonia synthesis process and elaborates on its advantage as case study. Chapter 3 introduces the field of optimal operation of chemical processes and discusses the state of the art methods.

Part II covers the development and application of optimal control methods for parts of chemical processes and consists of six chapters. A heat-integrated ammonia reactor serves as case study in all chapters. The basic structure of the case study and the model is explained in Appendix A. Adjustments to the case study are explained in the respective chapters. Chapter 4 formulates and solves a steady-state optimization problem for

this case study. Subsequently, economic nonlinear model predictive control as method for achieving optimal operation is applied in Chapter 5. As economic nonlinear model predictive control may not be applicable due to plant-model mismatch, computational expense, and stability concerns, Chapter 6 investigates the utilization of self-optimizing control for the case study. The reactor is incorporated into a simplified recycle system resulting in a change of the considered disturbances, and hence, the requirement of a set point change. Chapter 7 combines self-optimizing and extremum-seeking control to achieve this set point change of the controlled variable. A new method to convert the optimizing controller of economic nonlinear model predictive control into a feedback control problem is applied in Chapter 8. The second part concludes with a summary and comparison of the developed and/or studied methods in Chapter 9.

Part III introduces surrogate models for optimization and consists of five chapters. Chapter 10 proposes a framework for the optimization of integrated chemical processes. Approaches for flowsheet splitting and substitution of variables are proposed, which are subsequently applied in the following chapters. As surrogate models may struggle with a large number of independent variables, Chapter 11 introduces a three-step procedure for variable reduction and the generation of simplified surrogate models based on partial least squares regression (PLSR). This procedure can be considered as gray-box modelling due to the incorporation of process knowledge. Chapter 12 investigates the influence of the sampling space and independent variable definition on the developed variable reduction procedure. Chapter 13 combines the concepts of surrogate modelling and self-optimizing control. The aim behind this approach is to achieve a simplified response surface while potentially reducing the number of independent variables. A novel termination criterion for sampling based on PLSR is presented in Chapter 14. Furthermore, the surrogate models are combined with the initial model and the comparison of the optimization results are presented in this chapter.

Part IV concludes this thesis with a summary and discusses possible future research directions based on the presented work.

1.3 Main Contributions

This thesis contributes to research in two areas: the development of new methods for optimal operation of subprocesses and the development of a surrogate model-based framework for optimization of integrated plants.

The first main contribution of this thesis is the analysis of the impact of dependent disturbances on the calculation of self-optimizing variables. The study shows the impact of neglecting recycle streams in calculating self-optimizing variables and the necessity to adjust setpoints and the weighting matrices.

The combination of self-optimizing control and extremum-seeking control is the second main contribution. This allows the adjustment of the setpoint in the case of persistence disturbances or plant-model mismatch while maintaining fast disturbance rejection. Through the combination with extremum-seeking control, this is achieved in a model-free approach.

The third main contribution is the development of a procedure for optimization of integrated processes based on surrogate models. The procedure includes a new gray-box model structure for surrogate models that incorporates exact mass balances to achieve mass consistency. Two approaches are developed to improve the surrogate model performance. The first approach uses PLSR to calculate latent variables as linear combinations of the original independent variables. This reduces the number of independent variables. The second approach merges the concepts of self-optimizing control and surrogate modelling. This results in a simpler, flatter response surface with respect to the independent variables.

The fourth main contribution is the development of a new termination criterion for sampling for surrogate model generation. Contrary to other sampling methods, this procedure does not require the fitting of surrogate models at each sampling iteration. Consequently, the computational load is reduced. This sampling procedure is based on PLSR.

1.4 Publications

During the course of the Ph.D. studies, the following publications were submitted or accepted. The chapters of the thesis itself are based on these publications, but not limited to them.

1.4.1 Journal Articles

J. Straus, D. Krishnamoorthy, and S. Skogestad. On combining self-optimizing control and extremum-seeking control - Applied to an ammonia reactor case study. Submitted to the *Journal of Process Control*, 2018 (Chapter 7).

J. Straus and S. Skogestad. Surrogate model generation using self-optimizing variables. Submitted to *Computers & Chemical Engineering*, 2018 (Chapter 13).

J. Straus and S. Skogestad. Sampling for surrogate model generation using partial least squares regression. Submitted to *Computers & Chemical Engineering*, 2018 (Chapter 14).

1.4.2 Peer-Reviewed Conference Articles

J. Straus and S. Skogestad. Self-optimizing control in chemical recycle processes. Presented at *10th IFAC Symposium on Advanced Control of Chemical Processes*, Shenyang, 2018 (Chapter 6).

H. Bonnowitz, J. Straus, D. Krishnamoorthy, E. Jahanshahi, and S. Skogestad. Control of the steady-state gradient of an ammonia reactor using transient measurements. In A. Friedl, J. Klemeš, S. Radl, P. Verbanov, and T. Wallek, editors, *28th European Symposium on Computer Aided Process Engineering*, volume 43 of *Computer Aided Chemical Engineering*, pages 1111 – 1116. Elsevier, 2018 (Chapter 8).

J. Straus and S. Skogestad. Use of latent variables to reduce the dimension of surrogate models. In A. Espuña, M. Graells, and L. Puigjaner, editors, *27th European Symposium on Computer Aided Process Engineering*, volume 40 of *Computer Aided Chemical Engineering*, pages 445 – 450. Elsevier, 2017 (Chapter 12).

J. Straus and S. Skogestad. Economic NMPC for heat-integrated chemical reactors. In *2017 21st International Conference on Process Control (PC)*, pages 309–314, June 2017 (Chapters 4 and 5).

J. Straus and S. Skogestad. Variable reduction for surrogate modelling. In *Proceedings of Foundations of Computer-Aided Process Operations 2017*, Tucson, AZ, USA, 2017 (Chapter 11).

J. Straus and S. Skogestad. Minimizing the complexity of surrogate models for optimization. In Z. Kravanja and M. Bogataj, editors, *26th European Symposium on Computer Aided Process Engineering*, volume 38 of *Computer Aided Chemical Engineering*, pages 289 – 294. Elsevier, 2016 (Chapter 10).

1.4.3 Conference Abstracts and Presentations

J.Straus and S. Skogestad. Surrogate model generation using the concepts of self-optimizing control. In *Proceedings of the 21st Nordic Process Control Workshop*, Turku, Finland, 2018.

D. Krishnamoorthy, J. Straus, and S. Skogestad. Combining self optimizing control and extremum seeking control - applied to an ammonia reactor case study. In *Proceedings of AIChE Annual Meeting 2017*, Minneapolis, MN, USA, 2017.

J.Straus and S. Skogestad. Surrogate subsystem modelling of chemical processes. In *Proceedings of the 20th Nordic Process Control Workshop*, Sigtuna, Sweden, 2016.

Chapter 2

The Ammonia Process

Nitrogen containing chemicals are one of the most important industrial products. They include among others fertilizers and explosives as well as the majority of fine chemicals. Initially synthesized with nitrates as starting materials, it was already predicted by William Crookes in 1898 [45] that the natural reserves of nitrates will soon be outgrown by the demand. Hence, the fixation of atmospherical nitrogen as a new nitrogen source is crucial for feeding the world population. To this end, Fritz Haber and Carl Bosch developed the catalytic high-pressure synthesis of ammonia in the second decade of the 20th century [44]. The process allows the relatively cheap production of ammonia according to

$$3\,H_2 + 2\,N_2 \rightleftharpoons 2\,NH_3 \tag{2.1}$$

Since its introduction, 90 % of the production of ammonia is based on the Haber-Bosch process [5] and reached an estimated 150×10^6 t worldwide in 2017 [101].

Due to the large production volume, even small reductions in the production cost can provide an advantage over competitors. Hence, optimal design and operation conditions are crucial for achieving a competitive advantage. Consequently, mass and energy integration are applied extensively in modern ammonia plants. This integration leads to difficulties in modelling and optimization, and thus in achieving optimal operation.

The production of ammonia can be split into two sections,

1. synthesis gas (hydrogen and nitrogen) production and
2. ammonia production.

In the following, both sections will be briefly explained with a focus on the process integration in the ammonia production section.

2.1 Synthesis Gas Production

The synthesis gas production received most attention since the introduction of the process. As a result, several different feedstocks can be used. Depending on the chosen feedstock, different processing methods are chosen. Table 2.1 gives an overview of the different feedstocks and the required energy and corresponding CO_2 emissions. The chosen feedstock is depending on its availability at the site of the plant. Roughly one third of the energy requirement (and hence CO_2 emissions) is related to the burning of fuel and two thirds to the production of hydrogen [49]. Ammonia plants in Europe and North America are using today almost exclusively natural gas as feedstock whereas China uses mostly coal due to the abundant available reserves. This section focuses hence on hydrogen production using natural gas.

Natural gas, predominantly methane (CH_4), is the best hydrogen source as it possesses the highest hydrogen to carbon ration. Consequently, the produced hydrogen to carbon dioxide ratio is lowest. The first step is steam reforming in the primary reformer, in which hydrogen as well as carbon monoxide and carbon dioxide are produced;

$$CH_4 + H_2O \rightleftharpoons 3\,H_2 + CO \tag{2.2}$$

$$CH_4 + 2\,H_2O \rightleftharpoons 4\,H_2 + CO_2 \tag{2.3}$$

Air is then added in the secondary reformer resulting in the production of water, CO, and CO_2 through partial oxidation. Further hydrogen is obtained through the water-gas shift equilibrium reaction

$$H_2O + CO \rightleftharpoons H_2 + CO_2 \tag{2.4}$$

The majority of the carbon dioxide is then removed using absorption towers. The resulting process gas consists of hydrogen and nitrogen with a molar ratio of three. In addition, argon and methane are frequently present as inert gases.

Table 2.1: Summary of different feedstocks with corresponding energy requirement and CO_2 emissions [49].

Feedstock	Process	Energy [GJ/t NH_3]	CO_2 emissions [t/t NH_3]
Natural gas	Steam Reforming	28	1.6
Naphta	Steam Reforming	35	2.5
Heavy fuel oil	Partial oxidation	38	3.0
Coal	Partial oxidation	42	3.8
Water	Electrolysis	34[1]	0.0[2]

[1] With an energy efficiency of 100 %, *e.g.* wind energy, water energy, or photovoltaics.
[2] If the electricity is produced carbon neutral.

The electrolysis of water as alternative to fossil feedstocks was already used in the 20th century [103]. With the promotion of renewable energy sources and potentially stricter environmental regulations, it may be again a promising hydrogen source for ammonia plants in the future.

2.2 Ammonia Production

The ammonia production consists in total of four sections:

1. synthesis gas makeup and compression;
2. reaction section with the ammonia reactor;
3. separation section based on ambient or cooled temperatures;
4. refrigeration section providing the cooling for the separation section and potentially the compression section.

Each section can contain energy and mass recycles within the section and to the other sections. Due to the thermodynamical limitations in the ammonia synthesis, and hence a reduced conversion *per pass*, it is necessary to recycle the majority of the process gas. This results in an overall mass recycle corresponding to 80 % of the feed flow to the reactor.

2.2.1 Synthesis Gas Makeup and Compression

The task of the synthesis gas makeup and compression section is to prepare the synthesis gas for the reaction. This includes the compression to 150-250 bar and the removal of remaining carbon dioxide, oxygen, and water. These chemical components are present from the hydrogen production and are catalyst poisons that can lead to catalyst deactivation.

The removal can be achieved using molecular sieves or washing with liquid nitrogen or ammonia. The latter is facilitated through the high solubility of water and carbon dioxide in ammonia. The washing with liquid ammonia can be as well incorporated into the separation section, if the separation section is between the compressor and the reactor. As an alternative, it is possible to use a so-called *cold-box* before or between the compressors. In this cold-box, energy integration is used to cool the synthesis gas to low temperatures. Through the addition of produced liquid ammonia, it is possible to remove the undesired catalyst poisons. As a second task, this section compresses and mixes the recycles gas from the separation section.

Integration in this section is mostly incorporated through the main mass recycle as well as the removal of the remaining carbon dioxide and water. It is connected to the reaction

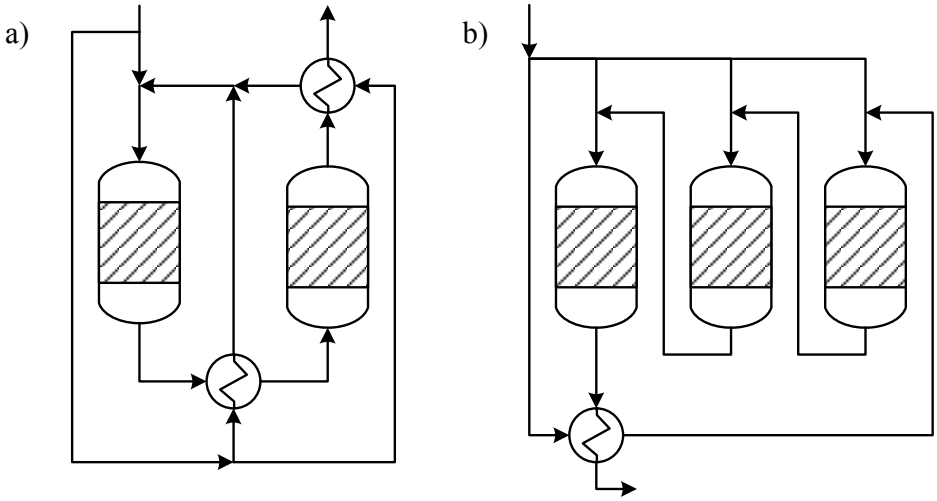


Figure 2.1: a) A reactor with interstage heat exchanger [98] and b) a quench flow reactor [75] as examples for two heat-integrated ammonia reactors.

section and the separation section. As the compressors heat the synthesis gas, the makeup section is frequently as well connected to the refrigeration section to cool the synthesis gas in-between the stages of the compressor to improve the compressor efficiency.

2.2.2 Reaction Section

The reaction section is the core of the ammonia production. It includes the reactor and heat integration within the reactor and the section for utilizing of the reaction energy.

Heat-integrated chemical reactors are generally applied in the case of exothermic reactions to increase on one hand the feed temperature to the first bed. This avoids the necessity of external heating streams. On the other hand, the inlet temperature of the subsequent beds should be reduced. This reduction of the inlet temperature increases the conversion in the beds by shifting the thermodynamic equilibrium. As a result, the conversion *per pass* is increased. Another example for this type of reactor is the methanol synthesis reactor [87].

The heating of the feed to the first bed and cooling of the feed streams of subsequent beds can be achieved through several reactor configurations. Figure 2.1 shows the concepts of quench flows and interstage cooling as examples for two frequent reactor configurations. Quench flows correspond to the cooling of the feed to a reactor bed through the addi-

tion of fresh feed. In the case of interstage heat exchangers, the reactor includes a heat exchanger in-between the beds. The main difference between the two reactor systems is that the complete feed is fed to the first bed in the case of interstage heat exchangers, whereas quench flows have a reduced flow through the first beds. In general, it is possible to combine interstage cooling with quench flows. The generated heat of the reactor section is furthermore utilized for producing high pressure steam that can be used in the steam turbines for compression of the synthesis gas.

The reactor section contains several internal mass and energy recycles. These are mainly located inside the reactor and are necessary for an increased conversion *per pass*. There is generally a heat-integration between the inlet and outlet of the reaction section to avoid external heaters. It is connected to the synthesis gas makeup and separation section through the overall mass recycles.

2.2.3 Separation Section

The separation section follows the reaction section. It separates ammonia from the reactants (hydrogen and nitrogen) and inert gases (frequently methane and argon). This is achieved through condensation of ammonia at reduced temperatures. Depending on the pressure in the synthesis loop, different refrigerants have to be used. In high pressure synthesis loops ($\sim$ 450 bar), water and air cooling may be sufficient to condense ammonia. In medium pressure synthesis loops, ammonia itself is frequently used as refrigerant. In this case, a refrigeration section has to be incorporated. The condensed ammonia is after the separation flashed at 20 bar to remove the dissolved gases.

In addition to the separation of ammonia, this section includes a purge stream. This stream is necessary to prevent the accumulation of inert gases in the synthesis gas loop and has a major influence on the total flow in the mass recycle.

Integration within the separation section is given through heat integration for cooling ammonia. Furthermore, it is connected to the refrigeration section through several heat exchangers and the synthesis gas makeup and reaction section through the overall gas recycle.

2.2.4 Refrigeration Section

The refrigeration section is necessary in the case of medium pressure synthesis loops. It consists of a refrigeration loop. Frequently, ammonia is used as refrigerant. It provides the cooling of the effluent of the reactor section and potentially in-between the compressor stages. This is achieved through evaporation of ammonia at different pressure levels. The refrigeration section can be designed as a closed system, in which ammonia is circulated, and as an open system. This means that the produced ammonia is fed to and removed from the refrigeration section.

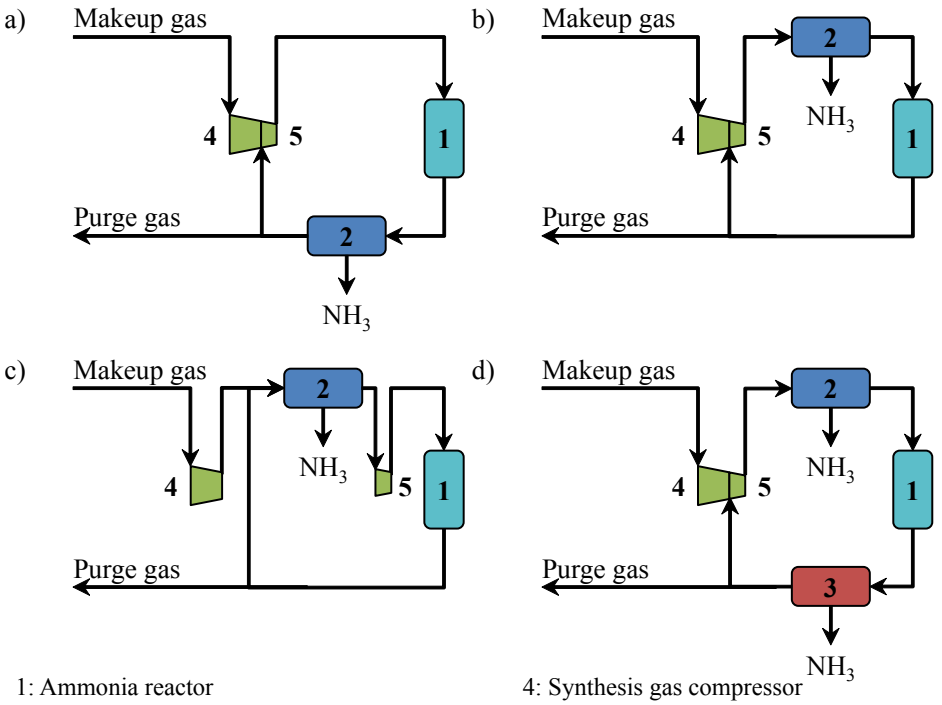
As the refrigeration section is either a closed or an open system, several recycle streams exist inside it. Optimal operation of refrigeration loops is still an ongoing research area in itself. In addition, it is integrated with the separation section and potentially the synthesis gas makeup section.

2.2.5 Combined Ammonia Production

The position of the introduced sections can differ depending on the configuration of the ammonia synthesis loop. Figure 2.2 shows several utilized configurations for the synthesis loop [6]. The refrigeration section is not included in these plots and is in general connected to the *ammonia recovery at reduced temperature*. Configuration a) is energetically most favourable. The separation of ammonia and the purge stream before the recycle compressor reduces the duty of the recycle compressor. Furthermore, the purge stream is located after the separation to avoid purging ammonia and fresh feed. If catalyst poisons are still present in the makeup gas, it may be necessary to separate the ammonia after the addition of the makeup gas as shown in configuration b)-d). This can be done before (configuration c)) or after the recycle compressor (configuration b) and d)). As a disadvantage, the purge stream has the same ammonia concentration as the reactor outlet. This can be circumvented in high pressure systems through the separation of a part of the produced ammonia at ambient temperature as in configuration d).

Retrofitting of existing plants and purification of the makeup gas aims at configurations close to a). The aforementioned *cold-box* is one approach to reduce the energy consumption in the compressors and avoid purging ammonia.

Each of the subprocesses of the ammonia synthesis loop contains several internal mass and energy integration. In addition, all sections are connected through the overall mass recycle and further heat recycles. This corresponds to several nested and adjacent recycle loops. The overall mass recycle corresponds to ~ 80 % of the feed flow to the reactor. Small changes in the different subprocesses can consequently result in large changes in the other subprocesses. As a result, the overall ammonia production is difficult to model and optimize. Hence, it is a good case study for this thesis.



1: Ammonia reactor
2: Ammonia recovery at reduced temperature
3: Ammonia recovery at ambient temperature
4: Synthesis gas compressor
5: Recycle compressor

Figure 2.2: Different configurations for the synthesis loop of the ammonia process, derived from [6].

Chapter 3

Optimal Operation - State of the Art

As mentioned in the introduction, this thesis investigates the optimal operation of integrated processes. But what does optimal operation mean? This chapter will give an overview of optimal operation and current practice in implementing it. Section 3.1 explains the general concept of optimal operation. Section 3.2 explains the main ideas behind self-optimizing control as a means for selecting controlled variables. Section 3.3 covers model-free methods whereas Section 3.4 introduces online optimization-based methods.

3.1 General Concepts

Each chemical plant has an economic profit or cost function corresponding to its economic performance. The aim is then to minimize the operating cost for the production of a certain amount of product. Optimal operation tries to achieve this through control of the process. Hence, it is useful to look at the control structure.

For simplicity, the control structure is frequently divided in multiple layers [31]. A typical control hierarchy is shown in Figure 3.1 [90]. It consists of the overall *scheduling* layer, which defines how much should be produced when and where. The operation targets are in this layer defined in a time scale of weeks and it is generally not automated or uses simple linear models. In the context of the ammonia production, the *site-wide optimization* layer then optimizes the ammonia process in combination with the nitric acid and fertilizer production with a time scale of a day. It frequently uses a simplified steady-state model of the processes. The *local optimization* as layer below looks into the different processes like the production of synthesis gas, ammonia, or nitrous oxides in the Ostwaldt process. As disturbances may change regularly, this layer has a time-scale of hours. The *supervisory* and *regulatory control* layers focus subsequently on the con-

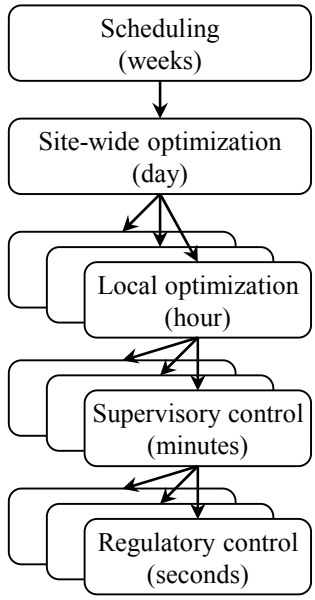


Figure 3.1: Typical control hierarchy of chemical processes, adopted from [90].

trol of the process. Generally, the *supervisory control* layer is operating on a slower time scale than the *regulatory control* layer. The task of the former is to satisfy operational constraints and is generally a multivariable control layer. The aim of the *regulatory control* layer is as the fastest layer the stabilization of the process, *e.g.* an ammonia synthesis reactor.

The layers are hereby connected through the respective controlled variables. For example, the layer *local optimization* calculates a setpoint c_s for the controlled variable c that is optimal for the current disturbances. The *supervisory control* layer then tries to keep the measurement c_m at the setpoint c_s while satisfying operational constraints, until it receives a new setpoint from the *local optimization* layer. Frequently, we assume as well that there is a time scale separation in-between the different layers. That implies that adjustments to the setpoints are immediately effective.

As for all simplifications, the control hierarchy does not represent the practice of all chemical plants. It may serve however as an illustration of the general structure of control systems in the chemical industry. In this context, optimal operation then requires an optimal interplay between the layers in the hierarchy. As scheduling is frequently not automated and does not require detailed models, this thesis will exclude the scheduling

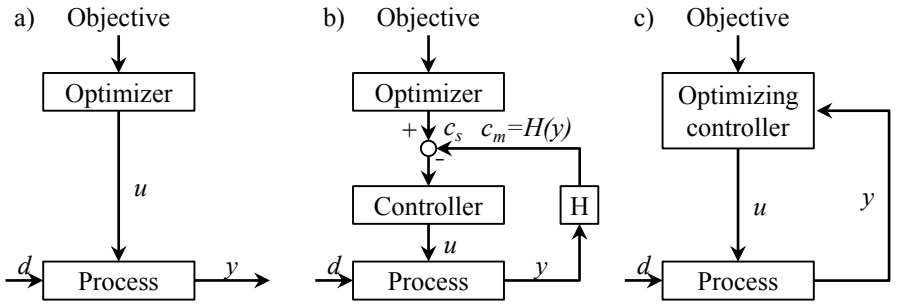


Figure 3.2: Possible implementations of optimal operation, adopted from [90].

layer and focus on the four lower layers.

The overall aim of optimal operation can as well be translated into a mathematical problem. Consider a nonlinear, dynamic model of any chemical process

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t)) \\ \mathbf{0} &\geq \mathbf{h}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t)) \end{aligned} \tag{3.1}$$

in which $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ are the state variables, $\mathbf{d}(t) \in \mathbb{R}^{n_d}$ the disturbance variables, and $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ the degrees of freedom. The dynamic behaviour of the process is described by $\mathbf{f}$. In addition, operational constraints on the states, inputs, and disturbances can be imposed through $\mathbf{h}$. These constraints include normally environmental and product quality constraints as well as bounds on states and inputs. The aim is now to minimize an economic cost function $J_{dyn}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t))$ subject to the constraints of the process model (3.1). This means that we want to manipulate our degrees of freedom to achieve the lowest production cost (or equivalently the highest profit).

This can be then rewritten as dynamic optimization problem for an infinite time horizon:

$$\begin{aligned} \min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot)} \quad & \int_0^\infty J_{dyn}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t)) \, \mathrm{d}t \\ \text{s.t.} \quad & \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t)) \qquad \forall t \in [0, \infty) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t)) \qquad \forall t \in [0, \infty) \\ & \mathbf{x}(0) = \mathbf{x}_0 \end{aligned} \tag{3.2}$$

The solution of this nonlinear problem results then in optimal operation. This includes the optimal trajectory of the inputs. How can the mathematical optimization strategy then be implemented in practice?

Figure 3.2 shows three possible approaches for the implementation of optimal operation [90].

The three approaches are

a) Open-loop implementation: the optimal trajectories are calculated offline beforehand. Hence, feedback is not incorporated and disturbances result in deviation from the optimal operation.

b) Closed-loop implementation with a control layer: introduction of controllers whose setpoints are given by the optimization problem. In this approach, the controller adjusts the manipulated variables if the disturbances change.

c) Closed-loop implementation without a control layer: The optimization problem is an optimizer and controller in one. This implies that feedback exists and the setpoints of the manipulated variables are adjusted with changing disturbances at the optimization level.

These approaches can be either implemented on a local or site-wide level. The block *Process* corresponds to the process as seen by the optimization layer. Hence, it may include stabilizing controllers and is not necessarily uncontrolled. In this situation, the setpoints to the controllers are then decided by the optimization layer.

3.2 Variable Selection - Self-Optimizing Control

Controlled variables are the links of the different layers in a control hierarchy as it was mentioned in the previous section. This leaves an important question as how one should select the controlled variables. Already in 1973, Foss [34] asked in his *Critique of chemical process control theory*:

> Which variables should be measured, which inputs should be manipulated, and what links should be made between these two sets? This problem is considered by many to be the most important problem encountered by designers of chemical process control systems.

Based on this question, many scientists worked on control structure design and variable selection [2, 46, 51, 54, 72, 77, 90, 107]. Obviously the best controlled variables are those that we can keep at a constant setpoint. In this case, it is not necessary to have frequent reoptimizations. These variables are called *self-optimizing variables* and their application as controlled variables results in *self-optimizing control*. Skogestad [90] writes about self-optimizing control:

> Self-optimizing control is when we can achieve an acceptable loss L with constant setpoint values for the controlled variables $\mathbf{c}$ (without the need to reoptimize when disturbances occur)

Self-optimizing control itself is not a controller design, but a control structure philosophy. Its aim is the selection of controlled variables and it corresponds to approach b) in Figure 3.2.

How should these variable be selected? According to Skogestad [90], a self-optimizing variable should have the following properties

1. The optimal value of the controlled variables should be insensitive to disturbances so that the setpoint error is small.
2. The chosen controlled variables should be easy to measure and control so that the implementation error is small.
3. The gain from the input u to the controlled variable c should be large. This corresponds to a flat optimum with respect to c.
4. The controlled variables $\mathbf{c}$ should not be closely related in the case of several self-optimizing variables.

This can be as well posed in a mathematical way. As self-optimizing control is looking at the steady-state loss, optimization problem (3.2) has to be rephrased as a steady-state optimization problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J_{SS}(\mathbf{x},\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{f}(\mathbf{x},\mathbf{d},\mathbf{u}) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x},\mathbf{d},\mathbf{u}) \end{aligned} \tag{3.3}$$

with a steady-state cost function J_{SS}. In the case of disturbances $\mathbf{d}$, an optimal controller would use a setpoint $\mathbf{u}^{opt}(\mathbf{d})$ resulting in an optimal cost function $J_{SS}^{opt}(\mathbf{x}^{opt}(\mathbf{d}),\mathbf{d},\mathbf{u}^{opt}(\mathbf{d}))$. As this is frequently not possible, a steady-state loss L exists for a chosen controlled variable:

$$L = J_{SS}(\mathbf{x},\mathbf{d},\mathbf{u}) - J_{SS}^{opt}\left(\mathbf{x}^{opt}(\mathbf{d}),\mathbf{d},\mathbf{u}^{opt}(\mathbf{d})\right) \tag{3.4}$$

The aim of self-optimizing control is then to choose controlled variables, which minimize this loss. It is straight forward to notice that the ideal self-optimizing variable is the gradient of the cost function with respect to the manipulated variables, $\mathbf{J_u}$. This follows from the first-order necessary conditions of optimality given by the Karush-Kuhn-Tucker (KKT) conditions [79].

As it is frequently not possible to implement the gradient as controlled variable, the majority of the methods select a linear combination of measurements $\mathbf{y}_m = \mathbf{y} + \mathbf{n}^{\mathbf{y}}$ using

a selection matrix $\mathbf{H}$ according to

$$\mathbf{c} = \mathbf{H}\mathbf{y}_m \tag{3.5}$$

Furthermore, these methods are local methods. Hence, the plant model from the input and disturbances to the measurements is linearized at the current operating point

$$\mathbf{y} = \mathbf{G}^{\mathbf{y}}\mathbf{u} + \mathbf{G}_{\mathbf{d}}^{\mathbf{y}}\mathbf{d} \tag{3.6}$$

The cost function J_{SS} in optimization problem (3.3) can be approximated around the nominal point ($\mathbf{x}^*$, $\mathbf{d}^*$, $\mathbf{u}^*$) as a second-order Taylor expansion [46].

$$\begin{aligned} J_{SS}(\mathbf{x},\mathbf{d},\mathbf{u}) \approx & J_{SS}(\mathbf{x}^*,\mathbf{d}^*,\mathbf{u}^*) + \begin{bmatrix} \mathbf{J}_{SS,\mathbf{u}} \\ \mathbf{J}_{SS,\mathbf{d}} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{d} \end{bmatrix} \\ & + \frac{1}{2} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{d} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \mathbf{J}_{SS,\mathbf{uu}} & \mathbf{J}_{SS,\mathbf{ud}} \\ \mathbf{J}_{SS,\mathbf{ud}}^{\mathsf{T}} & \mathbf{J}_{SS,\mathbf{dd}} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{d} \end{bmatrix} \end{aligned} \tag{3.7}$$

with $\Delta\mathbf{d} = \mathbf{d} - \mathbf{d}^*$ and $\Delta\mathbf{u} = \mathbf{u} - \mathbf{u}^*$. Note, that $\mathbf{J}_{SS,\mathbf{u}}$, $\mathbf{J}_{SS,\mathbf{d}}$, $\mathbf{J}_{SS,\mathbf{uu}}$, $\mathbf{J}_{SS,\mathbf{ud}}$, and $\mathbf{J}_{SS,\mathbf{dd}}$ are evaluated at the nominal point ($\mathbf{x}^*, \mathbf{d}^*$, $\mathbf{u}^*$). If the nominal point is an extremum ($\mathbf{J}_{SS,\mathbf{u}} = \mathbf{0}$) and for a given disturbance ($\Delta\mathbf{d} = \mathbf{0}$), it is possible to express the loss in Eq. 3.4 as

$$L = \frac{1}{2}\left(\mathbf{u} - \mathbf{u}^{opt}(\mathbf{d})\right)^{\mathsf{T}} \mathbf{J}_{SS,\mathbf{uu}} \left(\mathbf{u} - \mathbf{u}^{opt}(\mathbf{d})\right) \tag{3.8}$$

Using the linear model (3.6) and the loss expression (3.8), several methods were developed for selecting measurements (and combination of measurements) for self-optimizing control. A concise summary of these methods can be found in [53].

Self-optimizing control will be used in Chapters 6, 7, and 13. These chapters will explain the applied methods for calculating the optimal selection matrix $\mathbf{H}$ in more detail.

3.3 Model-Free Methods

Several different measurement-based alternatives for achieving optimal operation have been developed that avoid to solve the optimization problem (3.2) by simply transforming it into a feedback control problem. These methods are classified under direct input adaptation-based methods [16] and require a good measurement or estimation of the steady-state cost J_{SS}. Such methods are computationally cheap to implement since optimization is done *via* feedback. A further advantage is that they are model free. That implies that plant-model mismatch does not affect the methods and it is not necessary to develop a detailed model of the process. Extremum-seeking control (ESC) and necessary conditions of optimality (NCO) tracking belong to such methods. A good classification of the different methods available can be found in [16] and [96].

3.3.1 Extremum-Seeking Control

Extremum-seeking control is a fairly old method and dates back to the 1920s [99]. However, it received increased attention after the proof of stability by Krstić and Wang [62] in 2000. The concept of extremum-seeking control is to use input excitation (dither) and the resulting change in the cost to estimate the gradient of the cost function. It can be as well applied if the dependency of the states and disturbances on the cost function is known [43]. ESC requires a time-scale separation in-between the plant dynamics and the dither as well as in-between the dither and the control action. This is necessary as it allows to consider the system as a static map. Therefore, it is a rather slow method.

The traditional approach to ESC is to use a sinusoidal for plant excitation and an estimation of the gradient through the combination of a high- and low-pass filter [62]. Hunnekens et al. [48] developed a dither free approach based on least squares estimation of the gradient to remove one time-scale separation. This method estimates the gradient using a buffer of past input usage and past values of the cost. Similarly, Chioua et al. [18] reported improvements in convergence through a recursive least squares estimation with forgetting factor.

Most studies on extremum-seeking control investigate the convergence from a suboptimal to the optimal operation point. Hence, disturbances are not considered. Krishnamoorthy et al. [60], Marinkov et al. [66], and Marinkov et al. [67] reported problems in the estimation of the gradient when disturbances occur.

Extremum-seeking control, with a focus on least squares estimation of the gradient, and its combination with self-optimizing control is covered in more detail in Chapter 7. A new method to handle unmeasured disturbances is proposed as well.

3.3.2 NCO Tracking

NCO tracking is similar to extremum-seeking control in the respect that it does not require a model, but a measurement of the cost function. It does however differ as the majority of the publications on NCO tracking use a discrete update of the input [35, 42, 50], which is given by a Newton step with reduced step size. The gradient is estimated using finite differences. An advantage of NCO tracking over ESC is the possibility to track and adopt to changes in the active constraints [97].

3.4 Online Optimization Based Methods

Online optimization-based methods rely on solving the optimization problem as steady-state or dynamic problem. The former is known as steady-state real-time optimization (SRTO) whereas the latter is called economic nonlinear model predictive control (E-NMPC) or dynamic real-time optimization (DRTO). This implies that they can be used in the local and the side-wide optimization layer in Figure 3.1. Darby et al. [21] and Engell [28] give extensive reviews of RTO, whereas Ellis et al. [27] explains E-NMPC in detail.

3.4.1 Real-Time Optimization

Conventional real-time optimization solves the steady-state problem (3.3). This results in optimal steady-state setpoints, which are given to the lower level control layers. The idea behind this approach is that the majority of chemical processes have a steady-state optimum. Steady-state models, which can be used for optimization, are generally developed during the design face. The development of dynamic models, which accurately represent the process, is however more complicated. As result, conventional RTO gives a transient loss as steady-state optimization is used. This is however deemed acceptable as it is computational cheaper to optimize a steady-state problem than a dynamic problem.

The steps in RTO are given by [78]:

1. **Steady-state detection**
   The majority of the data reconciliation and parameter estimation methods are based on steady-state models. Hence, it is necessary to detect, whether the condition of steady state is fulfilled or not.
2. **Data reconciliation**
   As data is general suspect to errors, data reconciliation adjusts measured and potentially estimates unmeasured states.
3. **Parameter estimation and model adaptation**
   Based on the reconciled data, this step updates the model parameters.
4. **Steady-state optimization**
   Using the steady-state optimization problem (3.3), new setpoints for the controlled variables are calculated.
5. **Setpoint update**
   The calculated setpoints are checked for plausibility and *e.g.* feasibility in a lower level model predictive controller, before transferred to control layer.

One major drawback of RTO is the requirement of steady state before it can be used. This is reflected in the long period, in which the optimization problem is solved. If a

disturbance occurs, the low level controllers will first regulate the plant to their previous setpoints. Once all control loops have settled, it is possible to start the aforementioned steps and update the setpoints. Hence, the period of solving optimization problem (3.2) generally ranges from 4-8 h or even only once a day [28]. Improvements are therefore mostly located in the data reconciliation step. Data reconciliation can use linear or nonlinear, static or dynamic models. Câmara et al. [19] give an extensive overview of different methods and their application in data reconciliation. Especially the application of dynamic models for data reconciliation allows a reduced period for RTO updates.

Conventional real-time optimization is not applied in this thesis. The development of surrogate models for integrated chemical processes in Part III aims however at the application of the developed models in a RTO environment.

3.4.2 Economic Nonlinear Model Predictive Control

As an alternative to conventional RTO, economic model predictive control (E-NMPC) and dynamic real-time optimization (DRTO) are attracting more and more research in recent time. The idea behind both methods is similar, that is using the dynamic model to predict the optimal trajectories. DRTO has however a lower level control layer and corresponds hence to control structure b) in Figure 3.2. Furthermore, it only has economic terms in the objective function.

E-NMPC developed from nonlinear model predictive control. The exchange of the control cost function to an economic cost function is only a small step. This results in the integration of the controller and the optimizer in one optimizing controller similar to Figure 3.2 c). Hence, as the optimizer is as well controlling the process, it is necessary to implement E-NMPC at a higher sampling rate than DRTO. Especially in the case of transient process, E-NMPC is advantageous compared to RTO as it is not possible to achieve steady state. This includes *e.g.* batch processes.

E-NMPC solves the following dynamic optimization problem at sampling step τ_k repeatedly with a given sampling interval $\Delta t = \tau_{k+1} - \tau_k$ and time horizon $\tau_N = N\Delta\tau_k$.

$$\begin{aligned} \min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot)} \quad & \int_0^{\tau_N} J_{dyn}(\mathbf{x}(t),\mathbf{d}(t),\mathbf{u}(t))\ \mathrm{d}t \\ \text{s.t.} \quad & \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t),\mathbf{d}(t),\mathbf{u}(t)) \qquad \forall t \in [0,\tau_N) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x}(t),\mathbf{d}(t),\mathbf{u}(t)) \qquad \forall t \in [0,\tau_N) \\ & \mathbf{x}(0) = \mathbf{x}(\tau_k) \end{aligned} \tag{3.9}$$

The initial states $\mathbf{x}(0)$ and disturbances have to be measured and/or estimated accordingly. The solution to the problem then describes the optimal trajectory with respect to the dynamic cost function. However, only the first value of the input function is imple-

mented. Feedback is achieved as the optimization problem is solved repeatedly. There are however three points that have to be addressed.

First of all, the formulation of the optimization problem (3.2) does not incorporate feasibility. The optimization problem has to be both both feasible at the initial starting point as well as recursively at each sampling point. Consequently, soft constraints have to be used with exact penalties in the cost function [88].

Secondly, even if feasibility is assumed, stability of the closed-loop system is not yet considered. Several approaches exist to guarantee stability of the closed-loop system. A first approach is to consider an infinite horizon as in problem (3.2) [106]. This results in a modified cost function

$$\int_0^{\infty} e^{-\rho t} J_{dyn}(\mathbf{x}(t), \mathbf{d}(t), \mathbf{u}(t), t)\ \mathrm{d}t \tag{3.10}$$

with ρ as a discount factor to incorporate the present value of money. Furthermore, the time was transformed to achieve a finite time horizon. A second approach introduces terminal constraints. This includes the introduction of a terminal penalty cost $V_f(\mathbf{x}(t_{max}))$ to the purely economic cost function J_{dyn} to guarantee stability or terminal constraints on the states

$$\mathbf{x}(t_{max}) \in \mathbb{X}_f \tag{3.11}$$

for a terminal region $\mathbb{X}_f$. A third approach introduces Lyapunov constraints. Faulwasser et al. [29] provide a broad overview of different approaches to guarantee stability in E-NMPC, albeit under assumptions.

As a third point, closed-loop performance cannot be guaranteed in general even if stability and feasibility are guaranteed. Ellis et al. [27] show this with a simple example.

Economic nonlinear model predictive control is applied to an ammonia reactor in Chapter 5.

Part II

Optimal Operation of Subprocesses

Chapter 4

Steady-State Optimum of a Heat-Integrated Ammonia Reactor

Heat-integrated chemical reactors are frequently utilized for exothermic reactions. Section 2.2 introduced two types of chemical reactors used in reactions which are limited by the thermodynamic equilibrium. Due to the reluctance of industry to use automated control for these types of reactors, limit cycle behaviour can occur in the case of disturbances like a pressure or temperature drop with potential extinction of the chemical reaction [75]. In order to counteract this behaviour, operation points away from the optimal conditions have to be chosen. This results in a constant economic loss. Nonetheless, the system stability is not guaranteed for large disturbances.

Morud and Skogestad [75] showed that this behaviour is caused in the case of a three-bed quench flow reactor by a combination of positive feedback by the preheater and an inverse response of the reactor outlet temperature to a step change in the reactor inlet temperature. For the same case study, Naess et al. [76] proposed a controller based on the inlet temperature of the respective beds with an additional split range controller for controlling the ratio between the flow through the heat exchanger and quench flow 1. However, no dynamic simulation results showing the performance of this control structure were presented.

Similar problems were reported by Rovaglio et al. [86] for a different ammonia reactor configuration. The investigated configuration consisted of two beds with an interstage heat exchanger. They considered in addition the incorporation of the reactor into the overall synthesis loop. In this case, the feedback through the recycle results in limit cycle behaviour for disturbances in which the reactor itself (without recycle) would not show limit cycle behaviour. As a result, they concluded that this behaviour is intrin-

sically linked to exothermic reactions in heat-integrated reactors. A stabilizing control structure was proposed and dynamic simulations showed the ability of the proposed control structure to prevent limit-cycle behaviour.

However, the aim of the proposed control structures for both reactors was stabilization. Optimal operation of the reactor in the presence of the disturbances was not considered As we are interested in optimal operation of subprocesses, this chapter will first present the steady-state optimization of the ammonia reactor. The model for the ammonia reactor as described by Morud and Skogestad [75] will serve as a starting point for the subsequent evaluation of different control strategies to achieve optimal operation in subunits. A detailed model description can be found in Appendix A.

This chapter is structured as follows. In Section 4.1, the optimization problem will be posed and simplified based on the definition of the system. It furthermore includes the steady-state results with the reactor profiles highlighting the improved bed utilization. Section 4.2 visualizes problems which are present due to the change of operation conditions in dynamic simulations and the resulting implications on operating the reactor system with fixed split ratios.

4.1 Problem Statement and Results

The constraints for the steady-state system are defined by Eq. (A.19) with $\dot{\mathbf{x}} = \mathbf{0}$. The number of discrete volumes in each reactor bed is chosen to be $n = 10$ as in this case, the actual diffusion is cancelled by *numerical* diffusion [75]. This results in a nonlinear problem with 30 algebraic and 30 dynamic states as well as 3 decision variables. The aim is to maximize the rate of extent of reaction of ammonia which is defined as

$$\dot{\xi} = \dot{m}_{in} \left(w_{\mathrm{NH_3},30} - w_{\mathrm{NH_3},in} \right) \quad \text{in [kg NH}_3\text{/ s]} \tag{4.1}$$

The sole maximization of the rate of extent of reaction furthermore reduces the cost in the separation and synthesis gas make-up section through a reduction of the recycle stream and hence a reduced compressor power in the recycle compressor. A higher rate of extent of reaction additionally increases the outlet temperature of the system. The increased outlet temperature can then be utilized to produce high-pressure steam for the reformer as this process requires a large amount of energy due to its endothermic nature. For a given feed, the cost function can be simplified to the outlet mass fraction of ammonia $w_{\mathrm{NH_3},30}$. This results in the following nonlinear problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{z},\mathbf{u}} \quad & -w_{\mathrm{NH_3},30} \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{f}(\mathbf{x},\mathbf{z},\mathbf{d},\mathbf{u}) \\ & \mathbf{0} = \mathbf{g}(\mathbf{x},\mathbf{z},\mathbf{d},\mathbf{u}) \\ & 0 \geq h(\mathbf{u}) \end{aligned} \tag{4.2}$$

Table 4.1: Results of the steady-state optimization.

	Split ratio reactor 1	Split ratio reactor 2	Split ratio reactor 3	$\dot{\xi}$ [kg NH_3/s]
Nominal	0.2302	0.1389	0.1270	16.2147
Optimal	0.2124	0.3079	0.2958	18.1797

The results of the steady-state optimization and a comparison to the current operation point are given in Table 4.1. As only the produced ammonia is interesting (*vide supra*), the rate of the rate of extent of reaction as defined in Eq. (4.1) is used for comparison. The change in produced ammonia corresponds to a 12 % increase compared to the nominal case. The temperature and concentration profiles of the optimized system as well as the original system are given in Figure 4.1. Especially the split ratios to reactor beds 2 and 3 were increased, as this indicates a reduction in temperature at the inlet of the respective bed which results in a lower equilibrium temperature at the reactor outlet, and hence, higher conversion. The split ratio to bed 1 is slightly reduced. However, the flow through the heat-exchanger is reduced as well from 0.5039 to 0.1839. This results to an overall lower inlet temperature in bed 1.

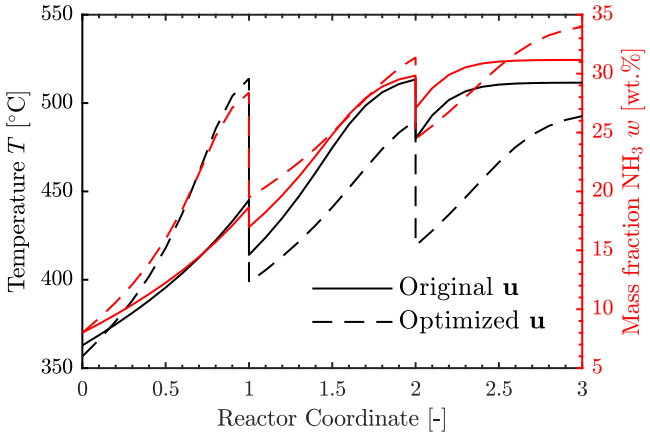


Figure 4.1: Reactor profile for the original and optimal split ratios **u** with nominal inlet conditions ($\dot{m}_{in} = 70$ kg/s, $p_{in} = 200$ bar, $T_{in} = 200$ °C, and $w_{NH_3,in} = 8$ wt%) and steady-state.

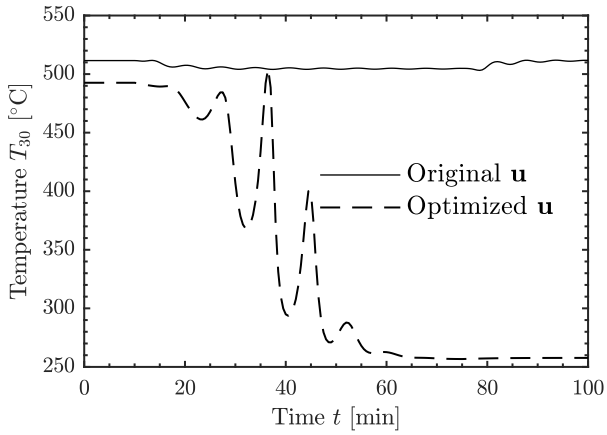


Figure 4.2: Outlet temperature of bed 3 with a pressure drop of $\Delta p_{in} = -20$ bar at $t = 10$ min and back to nominal conditions at $t = 75$ min with a constant input $\mathbf{u}$ corresponding to manual operation (open loop).

4.2 Discussion and Conclusion

The steady-state optimization improves the reactor utilization in the system. This is directly visible in Figure 4.1. Half of reactor bed 3 is not utilized with the nominal split ratios. This is indicated by the flat temperature and concentration profiles at the end of the reactor in the nominal case. A further improvement is given by the larger residence time in the first two reactor beds which corresponds as well to an improved utilization of the catalyst bed.

However, the problem in this heat-integrated reactors is the occurrence of limit-cycle behaviour or reactor extinction (*vide supra*), and hence, dynamic simulations with an input disturbance were conducted to compare the response of the operation point to disturbances. Figure 4.2 shows a pressure disturbance of $\Delta p_{in} = -20$ bar occurring after $t = 10$ min and the return to nominal conditions at $t = 75$ min. As we can see, the optimized split ratios decrease the potential of the system to reject disturbances. This can be explained by a reduced inlet temperature of the first bed, which results in complete extinction of bed 1, and therefore, the reactor.

Hence, automated control is necessary for operating at the optimum operation conditions. The following chapters will look into different controller designs to achieve optimal operation of the ammonia reactor.

Chapter 5

Economic NMPC for a Heat-Integrated Ammonia Reactor

As shown in Chapter 4, it is necessary to have automated control if the optimal operation point should be implemented. Using a simple feedback controller as suggested by Naess et al. [76] and Morud and Skogestad [75] would work for stabilizing the reactor at the optimal operation point. However, it would not result in optimal operation in the case of disturbances. In order to achieve optimal operation, this chapter investigates the application of economic nonlinear model predictive control (E-NMPC) for the ammonia reactor presented in Chapter 4. E-NMPC combines the controller and optimizer into a single optimizing controller. This removes the required waiting time for steady state and allows the setpoint to the controllers to follow the optimal trajectory. It is explained in Chaper 3.

This chapter is organized as follows. Section 5.1 defines the optimal control problem based on the nonlinear steady state problem derived in Chapter 4 with the respective tuning parameter of the E-NMPC. Section 5.2 shows the performance of the controller in the case of start-up from the initial operating point as well as disturbance rejection in the case of input disturbances in the mass flow rate $\dot{m}_{in}$, pressure p_{in}, temperature T_{in}, and ammonia mass fraction $w_{\mathrm{NH_3},in}$. Section 5.3 and Section 5.4 conclude this chapter with a discussion of the limitations and alternatives to the proposed control structure. The model, the state variables, and the used algorithms are described in Appendix A.

5.1 Problem Statement and Tuning Parameters

The core of the optimal control problem is given by the steady-state optimization problem (4.1). As a modification, a penalty term for input usage is introduced. This results in the following cost function at each step of the moving horizon

$$J_{dyn}(\mathbf{z}(t),\mathbf{u}(t)) = -w_{\mathrm{NH_3},30} + \dot{\mathbf{u}}^{\mathsf{T}}\mathbf{R}\dot{\mathbf{u}} \tag{5.1}$$

This corresponds to an optimal control problem given by

$$\begin{aligned}
\min_{\mathbf{x}(\cdot),\mathbf{z}(\cdot),\mathbf{u}(\cdot)} \quad & \int_0^{t_{max}} -w_{\mathrm{NH_3},30}(t) + \dot{\mathbf{u}}(t)^{\mathsf{T}}\mathbf{R}\dot{\mathbf{u}}(t)\ \mathrm{d}t \\
\text{s.t.} \quad & \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t),\mathbf{z}(t),\mathbf{d}(t),\mathbf{u}(t)), && t \in [0,t_{max}] \\
& \mathbf{0} = \mathbf{g}(\mathbf{x}(t),\mathbf{z}(t),\mathbf{d}(t),\mathbf{u}(t)), && t \in [0,t_{max}] \\
& \mathbf{0} \geq h(\mathbf{u}(t)), && t \in [0,t_{max}]
\end{aligned} \tag{5.2}$$

Neither terminal regions, nor a terminal cost is introduced. Hence, stability is not guaranteed for this controller. The parameters shown in Table 5.1 are used as tuning parameters of the NMPC. Morud shows [75], that the time the temperature wave requires to move through the reactor is given by roughly 350 s. Hence, it is necessary, that the NMPC horizon is at least 350 s long to capture the process dynamics. Due to the change in the input variables, the residence time in the first two reactors is increased and hence, the interval should be even longer. As the interval in the beginning of the NMPC should be accurate, input blocking is applied using an increasing block length for the input of

$$\begin{bmatrix}1 & 1 & 1 & 2 & 2 & 2 & 5 & 6\end{bmatrix} t_{block,NMPC} \tag{5.3}$$

in which the input is not changed. This corresponds to a total time frame of 600 s. The advantage of input blocking is the reduced computational expense of the optimization problem. The tuning matrix $\mathbf{R}$ was chosen to limit the input movement and to avoid oscillatory behaviour while maintaining fast settlement to the new setpoint.

Table 5.1: Tuning parameters for the NMPC optimization.

Parameter	Value	Unit
Integrator step length t_{int}	1	s
Input movement penalty $\mathbf{R}$	$\mathrm{diag}\left(\begin{bmatrix}20 & 20 & 20\end{bmatrix}\right)$	-
Sampling time $t_{samp,NMPC}$	30	s
Block time $t_{block,NMPC}$	30	s
Horizon	20	-

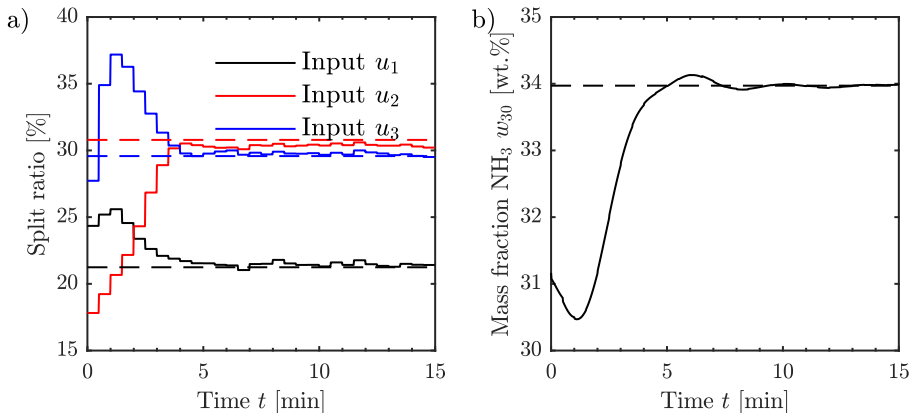


Figure 5.1: Response of the split ratios a) and the ammonia mass fraction b) during startup of the NMPC at nominal conditions ($\dot{m}_{in} = 70$ kg/s, $p_{in} = 200$ bar, $T_{in} = 200$ °C, and $w_{in} = 8$ wt.%).

Full state knowledge is assumed to simplify the calculations. Furthermore, it is assumed that the NMPC calculation is instantaneous. The investigated disturbances are input disturbances and assumed to have a measurement error given by Gaussian white noise with a standard deviation of $\sigma = 0.5\%$.

5.2 Results

During the start-up of a plant, E-NMPC is generally turned off as the model is fitted in the operation range as well as additional equipment like heaters may be used. Hence, the start-up of the controller from the nominal conditions was investigated in a first step to evaluate the possibility of switching from manual to automatic control. The results are plotted in Figure 5.1. The dashed lines in the following figures correspond to the optimal value for the given input conditions obtained in the steady-state analysis. We can directly see an inverse response of the outlet mass fraction of ammonia. The E-NMPC controller is able to reach the optimal setpoint within five minutes and settles within the first ten minutes to the optimal value given in Table 4.1. Small oscillations around the optimal concentration are hereby caused by the Gaussian white noise.

To evaluate the performance of the control structure on disturbance rejection, simulations with disturbance changes in all inlet variables were conducted. Here, one-directed disturbances are applied as disturbances in the other direction may lead to sub-optimal behaviour, but not to limit-cycle behaviour or reactor extinction. This can be explained

by an increased rate of extent of reaction resulting in an increased outlet temperature of reactor bed 3, T_{30}, and hence, an increased inlet temperature of bed 1, T_0.

Disturbances in the inlet flowrate $\dot{m}_{in}$ as well as the pressure of the system p_{in} are presented in Figure 5.2. This flowrate change increase and pressure drop is generally quite large. Flowrate increases can occur during the operation if problems with the purge control are present. They can lead to reactor extinction due to a reduced residence time and hence reduced rate of extent of reaction. Similarly, pressure drops can occur if problems with the compressor trains exist and result in the same problem as in the case of an increased inlet flowrate due to a reduction in the reaction rate and in the equilibrium concentration. As shown in Figure 4.2, already a pressure drop of $\Delta p_{in} = -20$ bar leads to unstable performance in manual mode using the optimized input values without adjustment. The application of E-NMPC however gives in the case of both disturbances close-to-optimal behaviour. The settling time of the optimized outlet mass fraction of ammonia corresponds to about 10 min as it was already the case in the startup of the controller. Both disturbances result in an increased inlet temperature T_0.

Disturbances in the inlet temperature T_{in} as well as the inlet mass fraction $w_{\mathrm{NH_3},in}$ are presented in Figure 5.3. Temperature reductions in the inlet can be present if the preheating control is faulty and can result in a lower inlet temperature of the first bed. Concentration increases on the other hand can occur of the ammonia separation temperature is too high and the ammonia concentration in the recycle increases. Similarly to the pressure and flowrate disturbance, the temperature drop and ammonia mass fraction increase can be handled by the controller. Again, the settling time to the nominal optimum is around 10 min. The inlet temperature of the first bed is increased for all disturbances showing the reduced rate of extent of reaction in the beds and hence reduced heat of reaction. Additionally, the variation in the outlet temperature is way smaller than the variation in the inlet temperature due to the equilibrium. This can be explained by the fact, that an outlet temperature of around 490 °C corresponds to the best achievable equilibrium concentration for the reactor configuration and nominal operation conditions.

5.3 Discussion

The rejection of all disturbances which would lead in the case of manual operations to limit-cycle behaviour or extinction of the reactor shows the performance of the chosen controller configuration. It is interesting to note that for all disturbances, the quench flows are reduced compared to the optimal values without disturbances. This reduction increases the preheating of the feed to the first bed and increases the inlet temperature to avoid limit cycle behaviour or reactor extinction. It is furthermore similar to the original split ratios as shown in Chapter 4.

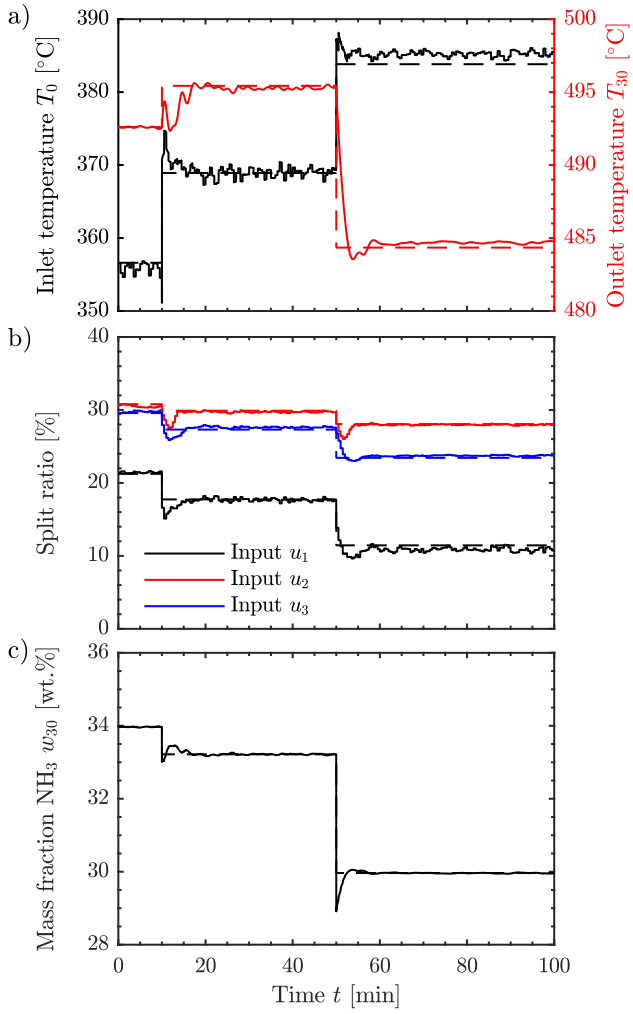


Figure 5.2: Response of the inlet (T_0) and outlet (T_{30}) temperature a), the split ratios u_i b), and the ammonia mass fraction at the outlet w_{30} with start at nominal conditions and as disturbance an inlet flowrate increase of $\Delta\dot{m}_{in} = 15$ kg/s at $t = 10$ min and back to nominal flow rate at $t = 50$ min with a simultaneous pressure drop of $\Delta p_{in} = -50$ bar.

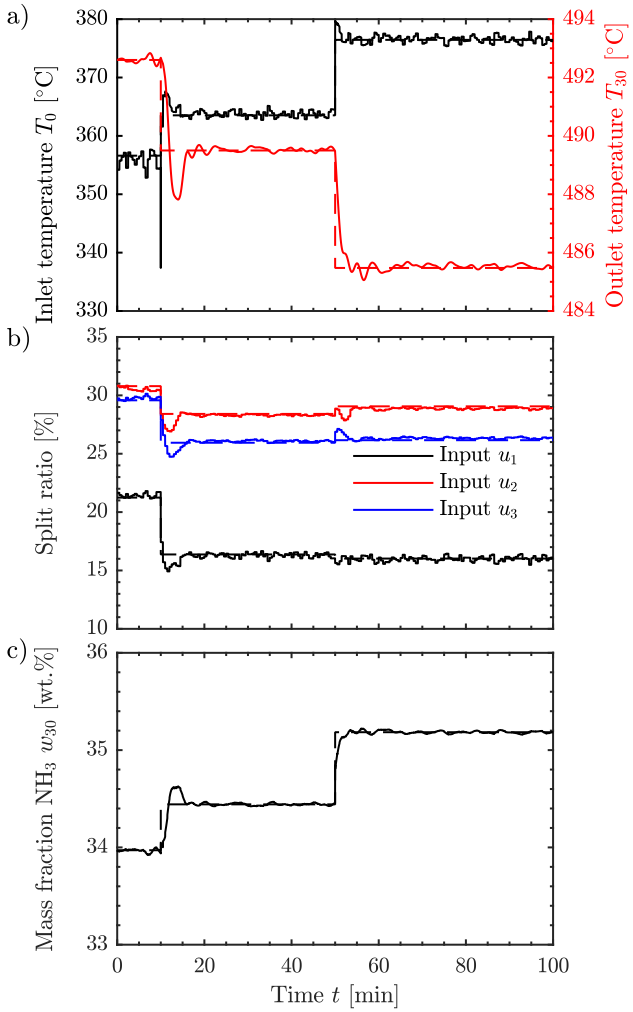


Figure 5.3: Response of the inlet (T_0) and outlet (T_{30}) temperature a), the split ratios u_i b), and the ammonia mass fraction at the outlet w_{30} with start at nominal conditions and as disturbance a temperature drop of $\Delta T_{in} = -30$ °C at $t = 10$ min and back to nominal inlet temperature at $t = 50$ min with a simultaneous mass fraction increase of $\Delta w_{in} = 4$ wt.%.

The assumption of full state feedback is generally not possible to achieve in practice where state estimators have to be used. Temperature measurements can be implemented at least at the inlet and outlet of each bed. However, in the case where the states at the time step 0 of the controller are assumed to be at their respective nominal optimum, the E-NMPC is able to achieve close to optimal conditions. As the median of the optimization time is around 2.5 s and the maximum 6 s on an Intel® Core™ i5-6600K, the assumption of instantaneous calculations can be seen valid with a sampling time of $t_{NMPC} = 30$ s.

5.4 Conclusion

An economic nonlinear model predictive controller was introduced for the control of the split ratios in an ammonia synthesis reactor as an example of heat-integrated chemical reactors. The application of the controller yields the optimal conversion of ammonia in the case of drastic input disturbances which would lead in manual operation to reactor extinction. It has to be noted, that these disturbances are generally not encountered in day-to-day operation and are only occurring in severe plant failures as it was the case for the reported limit-cycle [75].

The tuning of this controller was performed by trial-and-error, and hence, potential improvements of the performance of the controller are possible. The big advantage of utilizing E-NMPC for subsystems of chemical processes is given by the ability to adjust to feedback (both negative and positive) from the recycle the reactor is incorporated in. This allows to operate at optimal conversion *per pass* independently of the feed and hence may allow optimal operation for integrated chemical processes which do not allow to determine a steady-state optimum. Stability is not guaranteed for the investigated case as neither terminal constraints nor a penalty cost were introduced.

As an alternative to E-NMPC, it is also possible to use real-time-optimization (RTO) with setpoint tracking NMPC. Setpoint tracking NMPC without an RTO layer may however not be feasible to implement for this system as, depending on the magnitude of a disturbance, defined setpoints for temperatures may not be reached. This would then require the introduction of soft constraints as outlined in Chapter 3. Similarly, the application of the control structure of Naess et al. [76] can result in input saturation in the case of large disturbances.

Chapter 6

Self-Optimizing Control in Chemical Recycle Systems

Chapter 5 investigated the application of economic nonlinear model predictive control for the ammonia reactor. Due to the simplified model and the made assumptions, it may not be feasible to apply the proposed controller to a real ammonia reactor.

As an alternative, self-optimizing control may be utilized as outlined in Chapter 3. Here, the starting point for selecting a good control structure is to optimize the process for various disturbances. The aim is to find a simple way of implementing optimal operation, that is, a simple control structure with a small loss. Frequently, it is difficult to obtain a detailed process model that can be used for optimization, especially for systems that incorporate mass and energy recycle.

Applying optimization locally, however, results in a scenario where the considered disturbances may be dependent on the selected input variables through the recycle, resulting in a feedback. Furthermore, the cost function may be different in the overall flowsheet and the submodels. The submodel operation point does not necessarily correspond to the true optimum including the recycle loop as well. Therefore, the application of self-optimizing control to individual submodels of a large process can result in a situation, in which the selected measurement combination is not optimal.

The aim of this chapter is to investigate, how the dependency of disturbances may influence the theoretical performance of a self-optimizing control structure. Section 6.1 recapitulates SOC with focus on the applied exact local method [46], whereas Section 6.2 looks into the effect of dependent disturbances in the calculation of the optimal selection matrix $\mathbf{H}$. Section 6.3 investigates the influence of the feedback on a case study representing an ammonia reactor with a simplified recycle loop.

6.1 Self-Optimizing Control

Self-optimizing control (SOC) is the selection of controlled variables **c**, which when kept constant in the case of a disturbance, result in an acceptable economic loss [90]. The starting point is a steady-state optimization problem given by

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J(\mathbf{x},\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}(\mathbf{x},\mathbf{d},\mathbf{u}) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x},\mathbf{d},\mathbf{u}) \end{aligned} \tag{6.1}$$

in which $\mathbf{x} \in \mathbb{R}^{n_x}$ denote the state variables, $\mathbf{d} \in \mathbb{R}^{n_d}$ the disturbance variables, and $\mathbf{u} \in \mathbb{R}^{n_u}$ the steady-state degrees of freedom. The process model itself is given by $\mathbf{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_g}$ whereas $\mathbf{h} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_h}$ denote the operational constraints given by the process. The cost function $J : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ describes an economic cost of the system.

For given disturbances **d**, we assume that there exists an input $\mathbf{u}^{opt}(\mathbf{d})$ that minimizes the optimization problem (6.1). If different values than the optimal input $\mathbf{u}^{opt}$ are chosen for the manipulated variables **u**, there will be a steady-state loss

$$L = J(\mathbf{x},\mathbf{d},\mathbf{u}) - J\left(\mathbf{x}^{opt}(\mathbf{d}),\mathbf{d},\mathbf{u}^{opt}(\mathbf{d})\right) \tag{6.2}$$

The aim of self-optimizing control is then to find controlled variables **c**, which when kept constant give a **u** that minimize this loss for expected disturbances.

One direct solution to self-optimizing control is to control the gradient of the cost function J with respect to the inputs **u** ($\mathbf{J_u}$) to 0 as this would imply that the cost function is always at an extremum. The corresponding model-free approach of controlling the measured gradient to zero is called extremum seeking control and dates back to 1922 [99]. However, in general, the gradient cannot be measured. In certain cases it is possible to express the gradient of the cost function as a direct function of the measurements and control it to 0 [52].

As it is frequently not possible to obtain the gradient of the cost function as a simple expression of the measurements $\mathbf{y} \in \mathbb{R}^{n_y}$

$$\mathbf{y} = \mathbf{h_y}(\mathbf{x},\mathbf{d},\mathbf{u}) \tag{6.3}$$

it is necessary to define the controlled variables **c** as a function of the available measurements as

$$\mathbf{c} = \mathbf{h_c}(\mathbf{y}) \tag{6.4}$$

in which $\mathbf{h_c} : \mathbb{R}^{n_y} \rightarrow \mathbb{R}^{n_c}$ may be a function of any type. Frequently, linear measurement combinations with $\mathbf{H} \in \mathbb{R}^{n_c \times n_y}$ are used resulting in

$$\mathbf{c} = \mathbf{H}\mathbf{y} \tag{6.5}$$

6.1.1 Linearization of the Process Model and Cost Function

The majority of the self-optimizing control methods are based on a local analysis at the nominal optimal operation point. This results in a linearization of the measurements

$$\mathbf{y} = \mathbf{G}^{\mathbf{y}}\mathbf{u} + \mathbf{G}_{\mathbf{d}}^{\mathbf{y}}\mathbf{d} \tag{6.6}$$

where $\mathbf{G}^{\mathbf{y}} \in \mathbb{R}^{n_y \times n_u}$ and $\mathbf{G}_{\mathbf{d}}^{\mathbf{y}} \in \mathbb{R}^{n_y \times n_d}$ are the process and disturbance gain matrices, respectively. The cost is approximated through a second order Taylor expansion around the nominal operation point ($\mathbf{x}^*$, $\mathbf{d}^*$, $\mathbf{u}^*$)

$$\begin{aligned} J_{SS}(\mathbf{x},\mathbf{d},\mathbf{u}) \approx & J_{SS}(\mathbf{x}^*,\mathbf{d}^*,\mathbf{u}^*) + \begin{bmatrix}\mathbf{J}_{SS,\mathbf{u}}\\ \mathbf{J}_{SS,\mathbf{d}}\end{bmatrix}^{\mathsf{T}} \begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{d}\end{bmatrix} \\ & + \frac{1}{2}\begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{d}\end{bmatrix}^{\mathsf{T}} \begin{bmatrix}\mathbf{J}_{SS,\mathbf{uu}} & \mathbf{J}_{SS,\mathbf{ud}}\\ \mathbf{J}_{SS,\mathbf{ud}}^{\mathsf{T}} & \mathbf{J}_{SS,\mathbf{dd}}\end{bmatrix} \begin{bmatrix}\Delta\mathbf{u}\\ \Delta\mathbf{d}\end{bmatrix} \end{aligned} \tag{6.7}$$

with $\Delta\mathbf{d} = \mathbf{d} - \mathbf{d}^*$ and $\Delta\mathbf{u} = \mathbf{u} - \mathbf{u}^*$. Note, that the derivatives $\mathbf{J}_{SS,\mathbf{u}}$, $\mathbf{J}_{SS,\mathbf{d}}$, $\mathbf{J}_{SS,\mathbf{uu}}$, $\mathbf{J}_{SS,\mathbf{ud}}$, and $\mathbf{J}_{SS,\mathbf{dd}}$ are evaluated at the nominal point ($\mathbf{x}^*$, $\mathbf{d}^*$, $\mathbf{u}^*$). Combining Eq. (6.2) with Eq. (6.7) and utilizing that $\mathbf{J}_{SS,\mathbf{u}} = 0$ at the optimum, we can calculate the loss for disturbances $\mathbf{d} = \mathbf{d}^*$ as

$$L = \frac{1}{2}\left(\mathbf{u} - \mathbf{u}^{opt}(\mathbf{d})\right)^{\mathsf{T}} \mathbf{J}_{SS,\mathbf{uu}} \left(\mathbf{u} - \mathbf{u}^{opt}(\mathbf{d})\right) \tag{6.8}$$

6.1.2 Calculation of the Selection Matrix H

Several methods exist to obtain optimal measurement combinations, $\mathbf{c} = \mathbf{H}\mathbf{y}$. The reader is referred to Jäschke et al. [53] for a concise review of the different methods, which can be utilized. In this study, the exact local method as developed by Halvorsen et al. [46] and simplified by Yelchuru and Skogestad [107] is utilized. In order to make a statement about the loss, Halvorsen et al. [46] introduced diagonal scaling matrices for the disturbances $\mathbf{W}_{\mathbf{d}}$ and measurement errors $\mathbf{W}_{\mathbf{n}^y}$ as

$$\Delta\mathbf{d} = \mathbf{W}_{\mathbf{d}}\mathbf{d}'; \qquad \mathbf{n}^{\mathbf{y}} = \mathbf{W}_{\mathbf{n}^y}\mathbf{n}^{\mathbf{y}'} \tag{6.9}$$

in which the vectors $\mathbf{d}'$ and $\mathbf{n}^{\mathbf{y}'}$ are assumed to satisfy

$$\left\| \begin{bmatrix}\mathbf{d}'\\ \mathbf{n}^{\mathbf{y}'}\end{bmatrix} \right\|_2 \leq 1 \tag{6.10}$$

For a given selection matrix $\mathbf{H}$, the linearized model (6.6), and the general loss expression (6.8), it is possible to derive the worst-case loss [46] and the average expected loss [55] as

$$L_{WC}(\mathbf{H}) = \frac{1}{2}\bar{\sigma}(\mathbf{M})^2 \tag{6.11}$$

$$L_{avg}(\mathbf{H}) = \frac{1}{2}\left\|\mathbf{M}\right\|_F^2 \tag{6.12}$$

in which the loss matrix $\mathbf{M}$ is shown to be

$$\mathbf{M} = \mathbf{J}_{\mathbf{uu}}^{1/2} (\mathbf{H}\mathbf{G}^{\mathbf{y}})^{-1} \mathbf{H}\mathbf{Y} \tag{6.13}$$

with

$$\mathbf{Y} = \begin{bmatrix} \mathbf{F}\mathbf{W}_{\mathbf{d}} & \mathbf{W}_{\mathbf{n}^{\mathbf{y}}} \end{bmatrix} \tag{6.14}$$

The optimal sensitivity matrix for the measurements $\mathbf{F}$ can be obtained numerically or calculated from the linearized model [46]

$$\mathbf{F} = \frac{\partial \mathbf{y}^{opt}}{\partial \mathbf{d}} \tag{6.15}$$

$$= -\left(\mathbf{G}^{\mathbf{y}} \mathbf{J}_{SS,\mathbf{uu}}^{-1} \mathbf{J}_{SS,\mathbf{ud}} - \mathbf{G}_{\mathbf{d}}^{\mathbf{y}} \right) \tag{6.16}$$

The optimal measurement combination $\mathbf{H}$ can now be calculated as the solution, which minimizes the average (6.12) and worst case (6.11) loss. Both these optimization problems have the same optimal solution [55], which can be obtained by solving

$$\min_{\mathbf{H}} \left\| \mathbf{J}_{\mathbf{uu}}{}^{1/2} (\mathbf{H}\mathbf{G}^{\mathbf{y}})^{-1} \mathbf{H}\mathbf{Y} \right\|_F \tag{6.17}$$

The analytical solution to this problem was first described by Alstad et al. [3] and later simplified by Yelchuru and Skogestad [107] to

$$\mathbf{H}^{\mathsf{T}} = \left(\mathbf{Y}\mathbf{Y}^{\mathsf{T}} \right)^{-1} \mathbf{G}^{\mathbf{y}} \tag{6.18}$$

From Eqs. (6.18) and (6.14), we can see that the required model information is $\mathbf{G}^{\mathbf{y}}$ and $\mathbf{F}$, where the latter can be calculated using Eq. (6.16). In practice, if a nonlinear process model is utilized, it is simpler to calculate $\mathbf{F}$ numerically from Eq. (6.15) and using finte differences. Similarly, the loss L can be calculated using the nonlinear model and optimization problem (6.1).

6.2 Dependent Disturbances

Consider the block diagram in Figure 6.1, where *Local plant* represents our submodel (ammonia reactor in our case study) and *Remaining plant* represents the neglected part of the process (the recycle in our case).

The first question is now: Assume that we optimize our *Local plant* with a fixed value of $\mathbf{d}_0$, that is, we neglect the effect $\mathbf{u}$ has on $\mathbf{d}_0$ through, for example, the recycle. Is this acceptable? Of course, the answer is generally no

The second question is: Assume now that we find controlled variables (that is, find $\mathbf{H}_0$) based on considering our *Local plant*. Is this acceptable? Again, the answer is generally

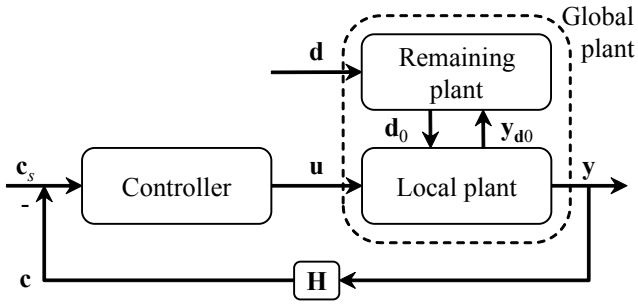


Figure 6.1: Visualization of the dependency of local disturbances $\mathbf{d}_0$ on the inputs $\mathbf{u}$, measurements $\mathbf{y}$, and the independent disturbances $\mathbf{d}$.

no, but in practice the answer may be "yes" if the local cost function is the same as the overall one. To better understand this, let us consider how the matrices used to find $\mathbf{H}$ in Eqs. (6.13) and (6.14) may change.

To see the difference between $\mathbf{G}^{\mathbf{y}}$ (based on the overall plant) and $\mathbf{G}_0^{\mathbf{y}}$ (based on the local plant), we can look at the total differential,

$$\begin{aligned}\mathbf{G}^{\mathbf{y}} &\triangleq \left(\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}\mathbf{u}}\right)_{\mathbf{d}} \\ &= \left(\frac{\partial \mathbf{y}}{\partial \mathbf{u}}\right)_{\mathbf{d}_0} + \frac{\partial \mathbf{y}}{\partial \mathbf{d}_0}\frac{\partial \mathbf{d}_0}{\partial \mathbf{y}_{\mathbf{d}0}}\frac{\mathrm{d}\mathbf{y}_{\mathbf{d}0}}{\mathrm{d}\mathbf{u}} \\ &= \mathbf{G}_0^{\mathbf{y}} \qquad + \mathbf{G}_{\mathbf{d}0}^{\mathbf{y}}\ \mathbf{G}_{\mathbf{y}_{\mathbf{d}0}}^{\mathbf{d}0}\mathbf{G}_{\mathbf{u}}^{\mathbf{y}_{\mathbf{d}0}}\end{aligned} \tag{6.19}$$

with $\mathbf{y}_{d0}$ corresponding to the outlet variables of the local plant, which affect the neglected part, see Figure 6.1. In our case, these are the outlet flow, pressure, temperature, and composition. The gain $\mathbf{G}_{\mathbf{y}_{\mathbf{d}0}}^{\mathbf{d}0}$ is the previously neglected feedback and can be obtained from the submodel of the remaining plant. The gain $\mathbf{G}_{\mathbf{u}}^{\mathbf{y}_{\mathbf{d}0}}$ corresponds to the change in the outlet variables with changing input.

A similar analysis can be conducted for the Hessian of the cost function $\mathbf{J}_{\mathbf{uu}}$ and the disturbance gain $\mathbf{G}_{\mathbf{d}}^{\mathbf{y}}$.

6.3 Case Study - Ammonia Synthesis Loop

The core of the case study is the three-bed ammonia reactor used in Chapter 5 in the application of economic nonlinear model predictive control. In this model, the disturbances ($\mathbf{d}_0$) are the inlet variables to the system

$$\mathbf{d}_0 = \begin{bmatrix} \dot{m}_{Feed0} & p_{Feed0} & T_{Feed0} & w_{\mathrm{NH_3},Feed0} \end{bmatrix} \tag{6.20}$$

There exist 3 input variables ($\mathbf{u}$), which correspond to the split ratios to the three reactor beds. The cost function for the ammonia reactor is to maximize the rate of extent of reaction $\dot{\xi}$

$$\begin{aligned} J &= -\dot{\xi} \\ &= -\dot{m}_{Feed0} \left(w_{\mathrm{NH_3},Rea} - w_{\mathrm{NH_3},Feed0} \right) \end{aligned} \tag{6.21}$$

As the reaction is limited by the thermodynamical equilibrium, a recycle is necessary to utilize the unreacted hydrogen and nitrogen. The reactor is connected to the recycle through the inlet stream $\mathbf{d}_0$ and the outlet stream $\mathbf{y}_{\mathbf{d}0}$. This recycle stream ($\mathbf{d}_r$) corresponds to 75 % of the mass of the feed to the reactor. Hence, the impact of the dependency of the neglected remaining plant is expected to be large in this case study.

The model including the recycle is depicted in Figure 6.2. In the recycle system, the actual disturbances (which usually are the true disturbances) are the inlet values to the new system:

$$\mathbf{d} = \begin{bmatrix} \dot{m}_{Feed} & p_{Feed} & T_{Feed} & w_{\mathrm{NH_3},Feed} \end{bmatrix}^{\mathsf{T}} \tag{6.22}$$

Note, that $\mathbf{d}_0$ is dependent on both $\mathbf{d}$ and $\mathbf{y}_{\mathbf{d}0}$ (through $\mathbf{d}_r$).

6.3.1 Model Description

The recycle adds the following assumptions to the model:

- hydrogen and nitrogen are fed as a stoichiometric mixture and no inerts are present in the feed, resulting in neglecting a purge flow;
- the feed to the system determines the pressure in the reactor and the inlet temperature to the reactor system;
- the reactor system operates at constant pressure with a pressure drop after the reactor system;
- the compressor operates with a fixed efficiency of $\eta = 80$ % and is considered to be isothermal as the compression ratio is smaller than 1.1 in the synthesis loop;
- the separation is defined *via* a (fixed) separation coefficient $\alpha = 0.25$ and only ammonia is separated. This corresponds to a splitter for ammonia.

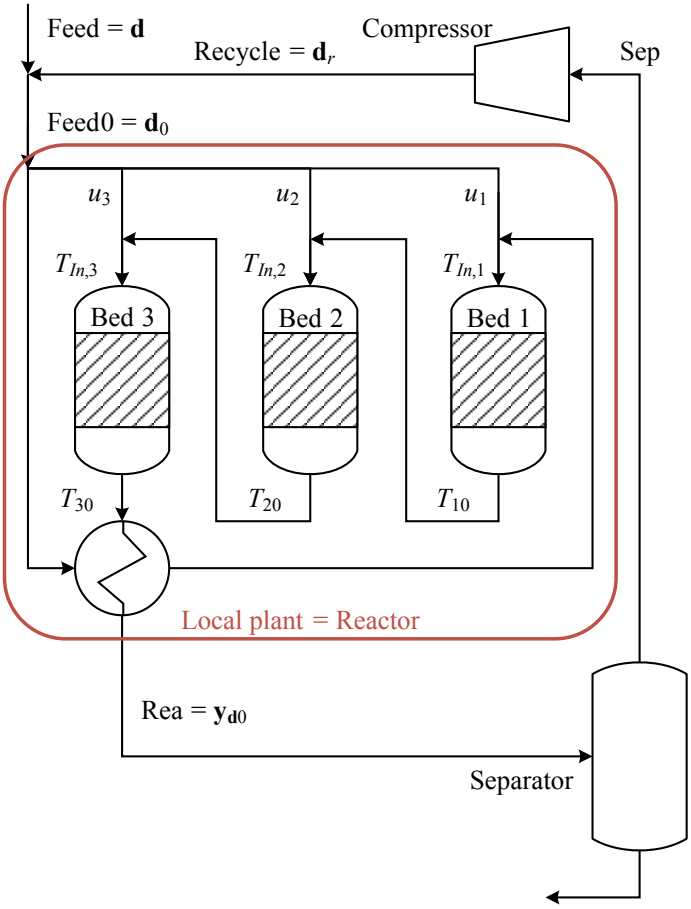


Figure 6.2: Heat-integrated three-bed reactor system incorporated into a simple recycle system consisting of a separator and a recycle compressor.

Based on the assumptions, the separation of ammonia is then calculated as

$$\dot{m}_{Sep} w_{\mathrm{NH_3},Sep} = \alpha \dot{m}_{Rea} w_{\mathrm{NH_3},Rea} \tag{6.23}$$

Additionally, a model equation similar to a valve coefficient has to be added for the pressure drop after the separator

$$0 = \dot{n}_{Sep} - k\sqrt{p_{Feed} - p_{Sep}} \tag{6.24}$$

Table 6.1: Nominal (optimal) inlet conditions for the reactor.

Recycle	$\dot{m}_{Feed0}$ [kg/s]	p_{Feed0} [bar]	T_{Feed0} [°C]	$w_{NH_3,Feed0}$ [wt.%]
Without	70.0	200	250	8.0
With	61.8	200	250	8.3

with a given pressure drop coefficient k (kmol/(s$\cdot\sqrt{\text{bar}}$)). The compressor duty of an isothermal compressor is (*e.g.* Skogestad [92])

$$W = \frac{\dot{n}_{Sep} R T_{Feed}}{\eta} \ln\left(\frac{p_{Feed}}{p_{Sep}}\right) \tag{6.25}$$

As there is no purge flow and the product is pure ammonia, all of the feed has to be converted. The system will therefore operate with a constant rate of extent of reaction, and hence, it cannot be used anymore as cost function as it was the case in the local reactor system. Instead, the new economic cost function corresponds to minimizing the compressor duty of the recycle loop, *i.e.*

$$J = W \tag{6.26}$$

As mentioned beforehand, this change in cost function may affect SOC variables defined for the reactor system. The new cost function aims at minimizing the flow within the recycle. This corresponds to minimizing the feed flow to the reactor while maintaining a constant rate of extent of reaction. It can be seen as equivalent to the old cost function where the aim is to maximize the rate of extent of reaction for a given feed. Alternatively, maximizing the conversion *per pass* can be used in both cases as it is equivalent to $\dot{\xi}$ for a fixed feed and in addition minimizes the recycle flow.

The optimization was performed using CasADi [4] with IPOPT [102].

Let us first consider the first question in Section 6.2; is it possible to optimize the reactor neglecting the recycle? With the new cost function and the modified system, the optimal nominal inlet conditions of the reactor are given in Table 6.1. Unsurprisingly, it is not possible to neglect the recycle in the optimization. Especially the reactor inlet mass flow $\dot{m}_{Feed0}$ changes a lot due to the recycle. This is caused by a positive feedback. A higher conversion *per pass* corresponds to more ammonia produced and separated, and hence, a lower recycle flowrate. This in turn increases the residence time in the beds, and hence, increases the conversion *per pass*. The ammonia mass fraction experiences negative feedback due to the assumption of a constant split factor. Hence, its value changes only by a small value. Due to the aforementioned assumptions, the inlet pressure and temperature of the system are the same with and without the recycle stream.

6.3.2 Application of SOC

This brings us to the second question in Section 6.2. Are the calculated controlled variables $\mathbf{c}_0 = \mathbf{H}_0\mathbf{y}$ based on considering only the reactor a valid choice?

To this end, we apply the exact local method as explained in Section 6.1.2 to both only the reactor (*local plant*) and to the reactor+recycle (*global plant).* In order to reduce the number of measurements utilized and pair the controlled variables close to the inputs, each reactor bed is treated individually and the exact local method is applied to the inlet temperature and the outlet temperature of the respective reactor bed; *i.e.*

$$\mathbf{y}_i = \begin{bmatrix} T_{In,i} \\ T_{i0} \end{bmatrix} \qquad i = 1,2,3 \tag{6.27}$$

This results in the combination of two measurements and corresponds additionally to selecting measurements that have a high gain from the input to the respective measurements.

The scaling matrices for the disturbance and measurement error in Eq. (6.9) are given by

$$\mathbf{W_d} = \text{diag}\left(\begin{bmatrix} 5 & 20 & 20 & 1 \end{bmatrix}\right) \tag{6.28}$$

$$\mathbf{W_{n^y},_i} = \text{diag}\left(\begin{bmatrix} 4 & 4 \end{bmatrix}\right) \qquad i = 1,2,3 \tag{6.29}$$

This implies that the actual optimal operation point with recycle does not fulfill requirement (6.10).

Utilizing the initial model of the reactor without recycle and cost function (6.21), we achieve the following combinations of self-optimizing control variables ($\mathbf{H}_0$)

$$\begin{aligned} c_{1,0} &= 0.053\ T_{In,1} + T_{10} \\ c_{2,0} &= 0.329\ T_{In,2} + T_{20} \\ c_{3,0} &= 1.311\ T_{In,3} + T_{30} \end{aligned} \tag{6.30}$$

whereas, if we incorporate the recycle in the calculation of our SOC variables and use cost function (6.26), we get ($\mathbf{H}$)

$$\begin{aligned} c_1 &= -0.288\ T_{In,1} + T_{10} \\ c_2 &= -0.161\ T_{In,2} + T_{20} \\ c_3 &= \ \ 0.940\ T_{In,3} + T_{30} \end{aligned} \tag{6.31}$$

Comparing the optimal selection matrices (6.30) and (6.31), we can directly see that there are changes in the SOC variables. The most important measurement (T_{10}, T_{20}) in the first 2 self-optimizing variables c_i remains the same, however the weights change.

This can be partly explained by an increase in the process gains $\mathbf{G}_1^{\mathbf{y}}$, $\mathbf{G}_2^{\mathbf{y}}$, and $\mathbf{G}_3^{\mathbf{y}}$ corresponding to the gains from u_i to $\mathbf{y}_i$ by around 15% in average:

$$\begin{aligned} \mathbf{G}_{1,0}^{\mathbf{y}} &= -\begin{bmatrix} 576 \\ 1071 \end{bmatrix}, & \mathbf{G}_1^{\mathbf{y}} &= -\begin{bmatrix} 667 \\ 1283 \end{bmatrix} \\ \mathbf{G}_{2,0}^{\mathbf{y}} &= -\begin{bmatrix} 603 \\ 800 \end{bmatrix}, & \mathbf{G}_2^{\mathbf{y}} &= -\begin{bmatrix} 703 \\ 948 \end{bmatrix} \\ \mathbf{G}_{3,0}^{\mathbf{y}} &= -\begin{bmatrix} 563 \\ 229 \end{bmatrix}, & \mathbf{G}_3^{\mathbf{y}} &= -\begin{bmatrix} 656 \\ 253 \end{bmatrix} \end{aligned} \tag{6.32}$$

The changes in the optimal sensitivity matrices $\mathbf{F}_i$ are even more pronounced, especially for the two disturbances with different values in the nominal optimal case; the inlet flowrate $\dot{m}_{Feed0}$ ($\dot{m}_{Feed}$) and the inlet mass fraction $w_{\mathrm{NH_3},Feed0}$ ($w_{\mathrm{NH_3},Feed}$).

$$\begin{aligned} \mathbf{F}_{1,0} &= \begin{bmatrix} 0.91 & 0.35 \\ -0.505 & 0.085 \\ -0.27 & -0.14 \\ 535.4 & -95.7 \end{bmatrix}^{\mathsf{T}}, & \mathbf{F}_1 &= \begin{bmatrix} 4.13 & 1.79 \\ -0.561 & 0.004 \\ -0.28 & -0.12 \\ 134.9 & -34.4 \end{bmatrix}^{\mathsf{T}} \\ \mathbf{F}_{2,0} &= \begin{bmatrix} 0.73 & 0.25 \\ -0.31 & 0.13 \\ -0.186 & 0.001 \\ 242.8 & -144.4 \end{bmatrix}^{\mathsf{T}}, & \mathbf{F}_2 &= \begin{bmatrix} 3.44 & 1.14 \\ -0.40 & 0.06 \\ -0.18 & 0.02 \\ 57.8 & -47.3 \end{bmatrix}^{\mathsf{T}} \\ \mathbf{F}_{3,0} &= \begin{bmatrix} 0.59 & 0.20 \\ -0.20 & 0.15 \\ -0.04 & 0.12 \\ 116.6 & -181.5 \end{bmatrix}^{\mathsf{T}}, & \mathbf{F}_3 &= \begin{bmatrix} 2.84 & 1.05 \\ -0.28 & 0.09 \\ -0.02 & 0.14 \\ 23.94 & -57.1 \end{bmatrix}^{\mathsf{T}} \end{aligned} \tag{6.33}$$

Based on these findings, it can be concluded that the linearization (6.6) (not surprisingly) is no longer valid if a recycle is introduced. This can be caused by the change in the optimal inlet flowrate of the reactor as shown in Table 6.1. $\dot{m}_{Feed0}$ is reduced by 12 % and should be outside the linear range of the nonlinear model.

It is possible to verify whether the linearization error is caused by the different inlet to the model or the feedback by changing the inlet to the model to the inlet calculated by the model with recycle. The plant gains $\mathbf{G}_i^{\mathbf{y}}$ are in this situation similar to the ones with recycle. Furthermore, $\mathbf{J_{uu}}$ is similar except for a scalar multiplier. This can be explained by the total differential (6.19). In the case of the ammonia reactor with a maximized rate of extent of reaction, $w_{\mathrm{NH_3},Rea}$ is maximized. In addition, T_{Rea} is maximized as well whereas p_{Rea} and $\dot{m}_{Rea}$ are unaffected due to mass conservation and the assumption of constant feed pressure to the reactor. Hence, in our case $\mathbf{G_u^{yd0}} = \mathbf{0}$ and

$$\mathbf{G}^{\mathbf{y}} = \mathbf{G}_0^{\mathbf{y}} \tag{6.34}$$

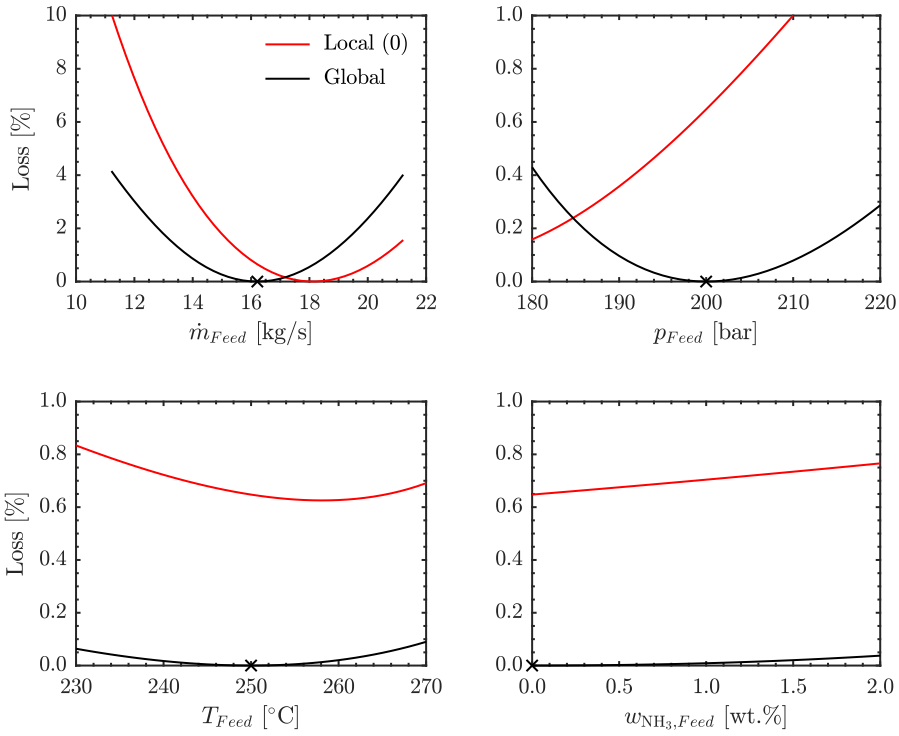


Figure 6.3: Loss as a function of the disturbance for both cases ($\mathbf{H}_0$ and $\mathbf{H}$). The setpoints for the local selection matrices $\mathbf{H}_{i,0}$ are not adjusted to optimal setpoints of the global recycle system.

This special behaviour occurs, if the outlet variables are equivalent to the cost function.

The optimal sensitivity matrices change however due the neglected dependency of $\mathbf{d}_0$ on $\mathbf{y}_{\mathbf{d}0}$ (and hence $\mathbf{u}$) through changes in $\mathbf{G}_{\mathbf{d}}^{\mathbf{y}}$ and $\mathbf{J}_{SS,\mathbf{ud}}$. This explains the changes in the selection matrices $\mathbf{H}_i$, see (6.30) and (6.31).

6.3.3 Loss Calculation

In order to evaluate the performance of both CV selections, $\mathbf{H}_{i,0}$ in (6.30) and $\mathbf{H}_i$ in (6.31), the loss as defined in Eq. (6.2) with cost function (6.26) and the (nonlinear) model including the recycle was calculated. The setpoints for the controller in the problem without recycle were given by the optimal setpoints without recycle. The comparison of

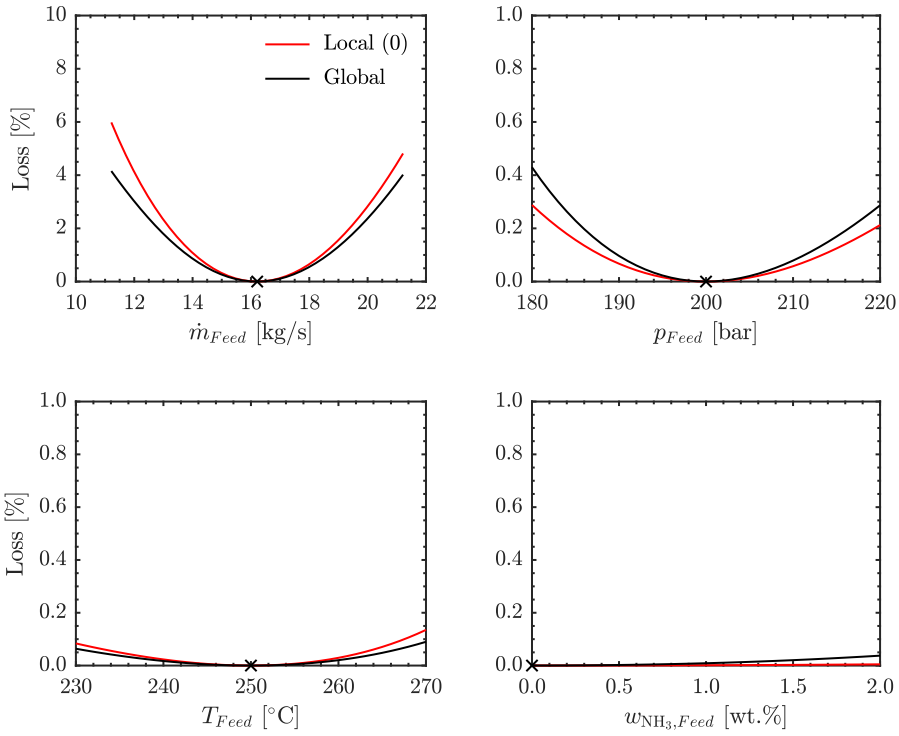


Figure 6.4: Loss as a function of the disturbance for both cases ($\mathbf{H}_0$ and $\mathbf{H}$). The setpoints for local selection matrices $\mathbf{H}_{i,0}$ are adjusted to optimal setpoints of the global recycle system.

both losses is shown in Figure 6.3. As can be seen from the red curves in Figure 6.3, there is a loss even at the nominal point. This is not necessarily caused by a poor $\mathbf{H}_0$ matrix, but by a non-optimal operating point.

Hence, the setpoint for the SOC variables should be adjusted to the new nominal optimum, in which the recycle is considered. The new loss calculations are shown in Figure 6.4. It is interesting to note, that the differences are surprisingly small. For an inlet pressure disturbance and mass flow disturbance, the loss is smaller for $\mathbf{H}$, whereas the loss is higher for $\mathbf{H}$ than for $\mathbf{H}_0$ for an inlet pressure and ammonia mass fraction disturbance.

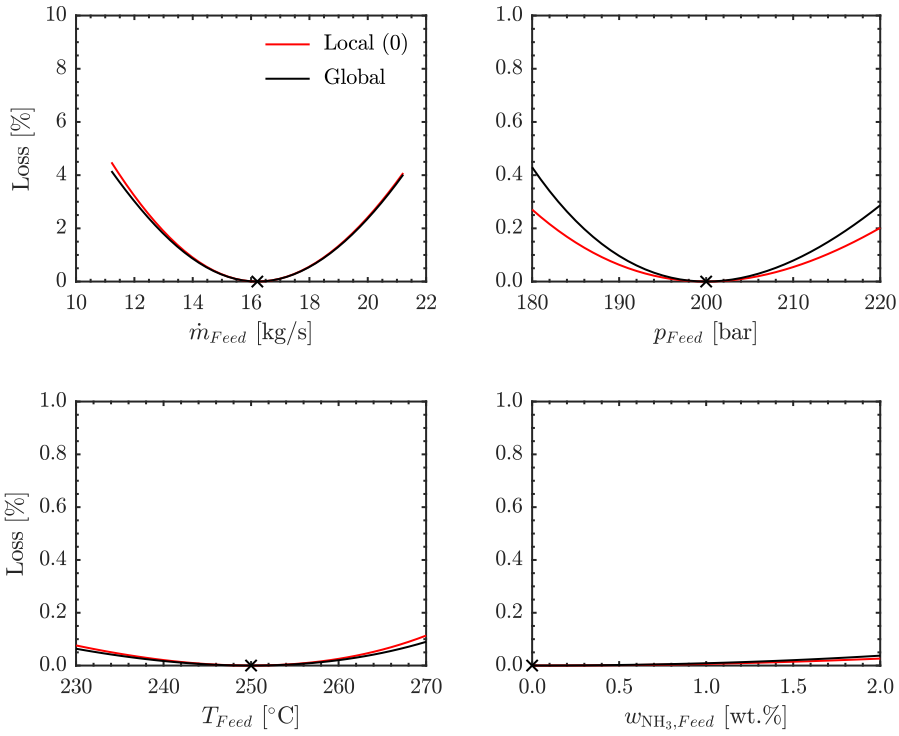


Figure 6.5: Loss as a function of the disturbance for both cases ($\mathbf{H}_{0,2}$ and $\mathbf{H}$). The setpoints for local selection matrices $\mathbf{H}_{i,0,2}$ are adjusted to optimal setpoints of the global recycle system and the weighting matrix $\mathbf{W_d}$ changed.

Both $\mathbf{H}_0$ and $\mathbf{H}$ use the same weighting matrices (6.9). As the reactor inlet mass flow $\dot{m}_{Feed0}$ is varying between 42 kg/s and 84 kg/s for a flowrate disturbance, we can directly see the incorrect weighting of the inlet mass flow. Changing the value of the mass flow disturbance in the weighting matrix to 20 kg/s results in new controlled variables ($\mathbf{H}_{0,2}$)

$$\begin{aligned} c_{1,0,2} &= -0.181\ T_{In,1}+T_{10} \\ c_{2,0,2} &= -0.053\ T_{In,2}+T_{20} \\ c_{3,0,2} &= \quad 0.971\ T_{In,3}+T_{30} \end{aligned} \tag{6.35}$$

which are more similar to (6.31). The corresponding loss is depicted in Figure 6.5. We can directly see that the difference in the loss is marginal, especially for $\dot{m}_{Feed}$, which had the largest loss in Figure 6.4. This is not surprising as the the optimal selection matrix $\mathbf{H}_{0,2}$ (6.35) is close to $\mathbf{H}$ (6.31).

6.3.4 Discussion

It has to be highlighted that in this specific case study, it was possible to define a cost function in the system without recycle, which corresponds to the cost function in the system with recycle. This is not necessarily the case for all submodels of recycle systems. If one would consider the case of a detailed separation section, the aim would be to minimize the cooling costs for a given feed. This feed would also represent some of the disturbances to the model. An unconstrained optimal solution would be given by no cooling and hence no separation. Therefore, separation requirements are needed, either on the separated product or through assigning cost values to all connection streams. Hence, the optimal point would be based on these separation requirements. On the other hand, the total model does not need constraints on the separation as separating no product would result in no profit.

From the definition of the loss in Eq. (6.2), it is obvious that there is a constant loss at the nominal operation point if the setpoint for the self-optimizing variables is not adjusted. Recall that the starting point of this investigation is that it is however too complicated to optimize the overall model, and hence, to calculate the true optimal setpoint. Therefore, a model-free approach, *e.g.* extremum-seeking control or necessary conditions of optimality tracking [35], should be used on top of self-optimizing control for calculating the optimal setpoint.

Neglecting the feedback through a recycle will result in overestimation of the loss in the case of negative feedback (ammonia mass fraction) and underestimation of the loss in case of positive feedback (mass flow). Adjusting the scaling matrices to account for the feedback will reduce the loss in the case of disturbances.

6.4 Conclusion

The dependency of considered disturbances on the input (and measurements) changes the optimal selection matrix in the application of self-optimizing control. This is the case even if the actual values of the disturbances, and hence, the feed to the submodel are unchanged.

The loss is in the investigated case study similar if the setpoints to the controllers and the disturbance weighting matrix $\mathbf{W_d}$ are adjusted. This cannot be generalized and is depending on the neglected dependencies.

Chapter 7

Combining Self-Optimizing Control and Extremum-Seeking Control

Self-optimizing control is an important concept to reduce the loss if disturbances are present. Chapter 6 showed however that it is necessary to update the setpoints to the controlled variables if dependencies of the disturbances are neglected. As it may be prohibitive to obtain a model of the plant for this adjustment, it can be necessary to use a model free method to update said setpoints. Jäschke and Skogestad [50] successfully combined self-optimizing control with NCO tracking in a hierarchical structure and demonstrated that the measurement based optimization techniques and the model based self-optimizing concepts are complementary. As a result, NCO tracking adjusted the setpoints of self-optimizing control to remove the steady-state loss whereas self-optimizing control reduced the input usage of NCO tracking. Both methods converged to the new setpoint at a similar rate. However, the gradient was estimated using finite differences which gave relatively poor NCO tracking. The authors suggested that more advanced gradient estimation and input adaptation methods may give a better overall performance as a future research direction.

A second model free method for achieving optimal operation is extremum-seeking control. As outlined in Chapter 3, extremum-seeking control aims at driving the gradient of the cost function to zero. The use of extremum-seeking control on top of self-optimizing control was briefly discussed by Keating and Alleyne [57] using the classical extremum-seeking method. However, the authors only considered measured disturbances for a single-input single-output system and limited themselves to a predefined optimal selection matrix $\mathbf{H}$. Additionally, based on the simulation results presented, Keating and Alleyne [57] did not consider a clear time scale separation between the extremum-seeking and self-optimizing controllers.

In this chapter, the work from Jäschke and Skogestad [50] and Keating and Alleyne [57] is extended. It is structured as follows. Section 7.1 briefly recapitulates the ideas of self-optimizing control as described in [90] and [46] and extremum-seeking control as described in [8] and [48]. Section 7.2 describes the framework in which we combine the two methods in a hierarchical structure. We then exemplify the proposed method using an ammonia synthesis reactor in Section 7.3. The results are discussed in Section 7.4 before concluding the paper in Section 7.5.

7.1 Background

Consider a process where the optimal operation of the process can be formulated as,

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J(\mathbf{x},\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \\ & \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{d},\mathbf{u}) \end{aligned} \tag{7.1}$$

where $J \in \mathbb{R}$ is a scalar cost, $\mathbf{x} \in \mathbb{R}^{n_x}$ denote the state variables, $\mathbf{u} \in \mathbb{R}^{n_u}$ denotes the vector of manipulated variables and $\mathbf{d} \in \mathbb{R}^{n_d}$ denotes the vector of disturbances and $\mathbf{f}$ denotes the n_x independent model equations. Note that we have no inequality constraints. More precisely, we assume that any active constraints are satisfied and the n_u manipulated variables $\mathbf{u}$ are the remaining unconstrained degrees of freedom, which are available for optimization [90].

We use the steady-state model equations $\mathbf{f}(\mathbf{x},\mathbf{d},\mathbf{u}) = \mathbf{0}$ to formally eliminate the states, $\mathbf{x} = \mathbf{l}(\mathbf{d},\mathbf{u})$. The steady-state cost can then be expressed just in terms of the inputs $\mathbf{u}$ and disturbances $\mathbf{d}$,

$$J_{ss}(\mathbf{u},\mathbf{d}) = J\left(\mathbf{l}(\mathbf{d},\mathbf{u}),\mathbf{d},\mathbf{u}\right) \tag{7.2}$$

The steady-state version of the optimization problem (7.1) is then equivalent to

$$\min_{\mathbf{u}} \quad J_{ss}(\mathbf{u},\mathbf{d}) \tag{7.3}$$

This says that the input $\mathbf{u}$ should be manipulated to optimize the steady state performance for any given disturbance $\mathbf{d}$. We make the following additional assumptions:

Assumption 1. *There exists* $\mathbf{u} = \mathbf{u}^{opt}$ *such that,*

$$\frac{\partial J_{ss}}{\partial \mathbf{u}}(\mathbf{u}^{opt},\mathbf{d}) = \mathbf{J}_{\mathbf{u}^{opt}} = \mathbf{0} \tag{7.4}$$

$$\frac{\partial^2 J_{ss}}{\partial \mathbf{u}^2}(\mathbf{u}^{opt},\mathbf{d}) = \mathbf{J}_{\mathbf{u}\mathbf{u}^{opt}} > \mathbf{0} \tag{7.5}$$

7.1.1 Self-Optimizing Control

Self-optimizing control is a strategy of selecting an optimal measurement combination $\mathbf{c}$ as controlled variables, such that the impact of known but unmeasured disturbances $\mathbf{d}$ on the optimal operation is minimized. This is achieved by using the system model offline to compute an optimal measurement combination. The ideal self-optimizing variable would be to control the gradient $\mathbf{J_u}$ to a constant setpoint of 0. However, in most applications, the gradient cannot be measured. A simple alternative is to identify a controlled variable $\mathbf{c} \in \mathbb{R}^{n_c}$ (with $n_c = n_u$)as a function of the available measurements $\mathbf{y} \in \mathbb{R}^{n_y}$. The simplest approach would be to select a linear combination of measurements given by,

$$\mathbf{c} = \mathbf{H}\mathbf{y}_m \tag{7.6}$$

where, $\mathbf{y}_m = \mathbf{y} + \mathbf{n}^y$ is the vector of available measurements corrupted by measurement noise $\mathbf{n}^y$ and $\mathbf{H} \in \mathbb{R}^{n_c \times n_y}$ is the measurement selection matrix. In addition to finding $\mathbf{H}$, we must also find the optimal setpoint $\mathbf{c}_s$.

Several approaches can be used to calculate the optimal measurement combination $\mathbf{c} = \mathbf{H}\mathbf{y}$. The reader is referred to [53] for a comprehensive review of the different self-optimizing control methods. Most of the self-optimizing control approaches are based on local linearization around the nominal optimal point. In this paper, we consider the exact local method as introduced in [46] and further developed in [3] and [107]. In this method, the optimization problem (7.3) is approximated by a quadratic approximation and a linearized model. Let the linearized measurement model be represented by,

$$\mathbf{y} = \mathbf{G^y}\mathbf{u} + \mathbf{G_d^y}\mathbf{d} \tag{7.7}$$

where $\mathbf{G^y} \in \mathbb{R}^{n_y \times n_u}$ and $\mathbf{G_d^y} \in \mathbb{R}^{n_y \times n_d}$ are the gain matrices from $\mathbf{u}$ to $\mathbf{y}$ and $\mathbf{d}$ to $\mathbf{y}$ respectively. The optimal selection matrix $\mathbf{H}$, which with constant setpoints $\mathbf{c}_s$ for $\mathbf{c}$, minimizes the loss in J_{ss} with respect to the expected disturbances and measurement noise, is then given by the expression,

$$\mathbf{H}^\mathsf{T} = (\mathbf{Y}\mathbf{Y}^\mathsf{T})^{-1}\mathbf{G^y} \tag{7.8}$$

where,

$$\mathbf{Y} = \begin{bmatrix} \mathbf{F}\mathbf{W_d} & \mathbf{W_{n^y}} \end{bmatrix} \tag{7.9}$$

and $\mathbf{W_d}$ and $\mathbf{W_{n^y}}$ are diagonal scaling matrices for the expected magnitudes of the disturbance and the measurement noise, respectively. $\mathbf{F} = \frac{\partial y^{opt}}{\partial d}$ is the optimal sensitivity matrix, which describes how the optimal measurements change with the disturbance. The optimal sensitivity matrix may be determined analytically using

$$\mathbf{F} = -(\mathbf{G^y}{\mathbf{J_{uu}}}^{-1}\mathbf{J_{ud}} - \mathbf{G_d^y}) \tag{7.10}$$

or it may also be determined numerically by perturbing the disturbances and re-solving the optimization problem as described in [2].

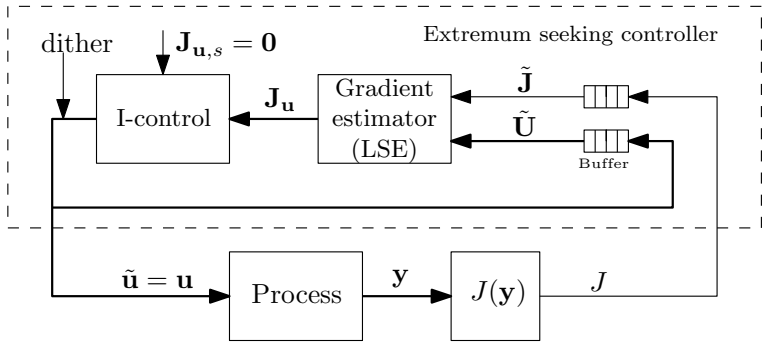


Figure 7.1: Block diagram of the least squares based extremum-seeking controller.

As seen from the equations above, the optimal selection matrix is based on the plant model $\mathbf{G}^y$ and the optimal sensitivity matrix $\mathbf{F}$ for the expected disturbances. Due to the linearization around the nominal optimal point, the controlled variables combination is only locally valid around this nominal optimal point. If a disturbance moves the process far from the nominal optimal point, the local model approximation may be poor, resulting in higher steady-state loss as shown in Chapter 6. Over time, as the plant model mismatch increases, the increase in the loss may no longer be acceptable. This requires re-optimization and computation of new optimal setpoints $\mathbf{c}_s$. Additionally, any unmodelled disturbances that are not accounted for in the optimal sensitivity matrix cannot be handled efficiently.

7.1.2 Extremum-Seeking Control

Extremum-seeking control is a model-free adaptive control method, where the steady-state performance of the system is optimized purely based on measuring the cost. The objective is to drive the estimated steady-state gradient of the cost $\mathbf{J}_\mathbf{u}$ to zero. The main advantage of extremum-seeking control compared to many other real time optimizers is that no plant model is required. This enables extremum-seeking control to optimize the performance of complex systems where the process model is not known accurately. The main disadvantages are that it requires that the cost function is measured and that the convergence can be very slow.

Unlike self-optimizing control, which is based on local linearization of the model around the nominal operating point, extremum-seeking control is based on local linearization of the measured cost around the current operating point. The input and the cost measurements are used to continuously estimate the steady-state gradient $\mathbf{J}_\mathbf{u}$ around the current operating point. The estimated gradient is then controlled to a setpoint of zero.

There are different ways of estimating the gradient based on the input and cost measurements. The classical approach is based on exciting the system with a sinusoidal signal and using a correlation based on high-pass and low-pass filters to retrieve the steady-state gradient information [62]. An alternative extremum-seeking scheme method was proposed in [48], where a linear least squares estimation method was used to estimate the steady-state gradient, which allows for a more general class of excitation signals [17]. The least squares method is also simple to implement and has fewer tuning parameters than the classical method. The least squares approach also provides a natural platform for multivariable systems. Improved performance using a recursive least squares approach was also reported in [18]. Therefore, we proposed to use the least squares based extremum-seeking control in the rest of the paper.

In this work, we extend the least squares based gradient estimation presented in [48] to a multivariable system. The goal is to estimate the gradient from the inputs $\tilde{\mathbf{u}}$ to the measured cost J . In the least squares based extremum-seeking control, the last N samples of data is used to fit a local linear cost model of the form,

$$J = \mathbf{J}_{\tilde{\mathbf{u}}}^{\mathsf{T}} \tilde{\mathbf{u}} + m \tag{7.11}$$

where $\mathbf{J}_{\tilde{\mathbf{u}}} \in \mathbb{R}^{n_u}$ is the vector of gradients from $\tilde{\mathbf{u}}$ to J and $m \in \mathbb{R}$ is the bias.

At the current sample time k, let $\tilde{\mathbf{J}} = [J_k \ \cdots \ J_{k-N+1}]^{\mathsf{T}} \in \mathbb{R}^N$ be the vector of the last N samples of the measured cost and $\tilde{\mathbf{U}} = [\tilde{\mathbf{u}}_k \ \cdots \ \tilde{\mathbf{u}}_{k-N+1}]^{\mathsf{T}} \in \mathbb{R}^{N \times n_u}$ be the vector of the last N samples of the input. A moving window of fixed length N is then used to estimate the gradient using the linear least squares method [64]

$$\hat{\theta} = \arg\min_{\theta} \left\| \tilde{\mathbf{J}} - \Phi^{\mathsf{T}} \theta \right\|_2^2 \tag{7.12}$$

where $\theta \in \mathbb{R}^{(n_u+1)}$ is the vector of parameters to be estimated and is given by

$$\theta = \begin{bmatrix} \mathbf{J}_{\tilde{\mathbf{u}}}^{\mathsf{T}} & m \end{bmatrix}^{\mathsf{T}} \tag{7.13}$$

and $\Phi \in \mathbb{R}^{N(n_u+1)}$ is the regressor vector given by

$$\Phi = \begin{bmatrix} \tilde{\mathbf{u}}_k^{\mathsf{T}} & 1 \\ \tilde{\mathbf{u}}_{k-1}^{\mathsf{T}} & 1 \\ \vdots & \vdots \\ \tilde{\mathbf{u}}_{k-N+1}^{\mathsf{T}} & 1 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{U}} & \mathbf{1} \end{bmatrix}^{\mathsf{T}} \tag{7.14}$$

The analytical solution to the least squares problem is given by, [64]

$$\hat{\theta} = \left[\Phi^{\mathsf{T}} \Phi \right]^{-1} \Phi^{\mathsf{T}} \tilde{\mathbf{J}} \tag{7.15}$$

Note that in theory, it is not necessary to use a dither signal when this approach is used, but for practical purposes it is recommended, and in our case study we use a sinusoidal dither signal with a sufficiently small amplitude.

Once the gradient $\mathbf{J}_{\tilde{\mathbf{u}}}$ is estimated, n_u integral controllers can be used to drive the gradients to zero using as degrees of freedom $\tilde{\mathbf{u}}$ (setpoints to the lower level controllers). The integral controller in general can be written as,

$$\frac{\mathrm{d}\tilde{\mathbf{u}}}{\mathrm{d}t} = \mathbf{K}_I \hat{\mathbf{J}}_{\tilde{\mathbf{u}}} \tag{7.16}$$

where, $\mathbf{K}_I \in \mathbb{R}^{n_u \times n_u}$ is the gain matrix. However, in many cases, we use decentralized control where $\mathbf{K}_I$ is diagonal. This is the case in our case study.

In order to estimate the static gradient $\hat{\mathbf{J}}_{\tilde{\mathbf{u}}}$ using dynamic data, the adaptation gain $\mathbf{K}_I$ must be chosen small enough such that the time-scale of the gradient estimation is slower than that of the system dynamics. Since the linear model assumption is valid only locally around the current operating point, the gradient can be estimated using only the past few samples of data. It was shown in [48] that the least squares based extremum-seeking control is stable and that the error is small for a sufficiently small adaptation gain $\mathbf{K}_I$ and sample size N.

Thus, the extremum-seeking control is based on the local linearization around the current operating point. Using the cost measurements, the gradient from the inputs to the cost is estimated and driven to its optimum. Since the gradient estimation relies entirely on the cost measurements, it requires accurate cost measurements. The convergence to the optimum may also be slow for a dynamic process.

7.2 Proposed Method

In this chapter, an hierarchical implementation with separate optimization and control layers proposed in [46] as shown in Figure 7.2 is proposed. Due to the time-scale separation required between the optimization and control layers [90], the extremum-seeking scheme fits better in the slow optimization layer and thus replaces the conventional RTO. Self-optimizing control is proposed to be in the faster layer below and tracks the updated setpoint given by the extremum-seeking controller. In other words, the extremum-seeking controller uses the measured cost J to compute the setpoints $\mathbf{c}_s$, which are provided to the self-optimizing controller. The controller output from the extremum-seeking controller is $\tilde{\mathbf{u}} = \mathbf{c}_s$ in (7.11)-(7.15)[1].

It may be argued that the self-optimizing control layer is redundant since an extremum-seeking scheme can directly manipulate the process to optimize the objective function.

[1] In the case where extremum-seeking controller controls the plant directly, $\tilde{\mathbf{u}} = \mathbf{u}$

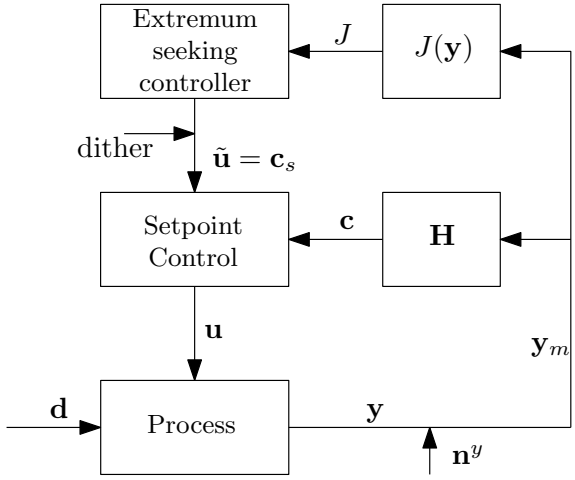


Figure 7.2: Hierarchical implementation of combined self-optimizing control and extremum-seeking control. The extremum-seeking controller used in this paper is shown in Figure7.1. The setpoint controller is a simple PID controller.

However, by using a purely data-driven approach, any *a-priori* knowledge about the system and the effect of disturbances are completely ignored. In addition, the extremum-seeking controller does not make use of measurements besides the cost measurements. Hence the convergence to the optimum is slow following a disturbance. The proposed hierarchical combination of extremum-seeking control and self-optimizing control avoids the shortcomings of the extremum-seeking scheme and improves the convergence to the optimum. This is primarily due to a faster initial reaction of the self-optimizing layer to known (modelled) disturbances. Following a disturbance, the self-optimizing control quickly brings the operation point to the near-optimal region, and on a slower timescale, the extremum-seeking control reduces any loss associated with the self-optimizing control.

At the same time, the self-optimizing control can benefit significantly from an extremum-seeking layer above it. As mentioned earlier, the extremum-seeking layer handles the plant-model mismatch and unmodelled disturbances and reduces any steady-state loss by adjusting the setpoint of the optimal measurement combination. This avoids costly re-optimization and redesign of the controllers.

In summary, we use the knowledge about the system to stay in the near-optimal region using self-optimizing control in the presence of disturbances. The extremum-seeking

Table 7.1: Properties of self optimizing control and extremum-seeking control.

Self-optimizing control	Extremum-seeking control
offline model required	model free
fast rejection of disturbances	slow rejection of disturbances
local linearization around nominal optimal point	local linearization around current operating point
handles unmeasured but expected disturbances	handles unmeasured and unexpected disturbances
needs no cost measurement	requires measurement of cost

control helps to reduce the losses due to plant-model mismatch, handle any unexpected disturbances, and fine tunes the optimal operating point. The key properties of the two methods are summarized and compared in Table 7.1, which shows that self-optimizing control and extremum-seeking control are complementary rather than competing.

Improvements in Gradient Estimation

Although data-driven methods such as extremum-seeking control can handle unmodelled disturbances on a longer timescale, abrupt changes in disturbances may temporarily cause erroneous gradient estimation, especially with the least squares gradient estimation method used in this paper. This may result in undesired manipulations by the extremum-seeking controller during the transients as motivated in [60]. Some modifications to improve the disturbance rejection in extremum seeking has been proposed in [60], [66], and [67], which all require the disturbances to be measured. The least-square based extremum-seeking control presented in this paper can be easily modified to handle measured disturbances, by adding the measured disturbances as a part of the regressor Φ in (7.15). However abrupt changes in unmeasured disturbances still pose issues with erroneous gradient estimation. A natural way to curb the effect of the inaccurate gradient estimate following an abrupt change in the disturbance, is to bound the magnitude of the individual gradients ($\hat{J}_{\tilde{u}_i}$) in (7.16) to a value $J_{u_i,max}$

$$\left|\hat{J}_{\tilde{u}_i}\right| \leq J_{u_i,max} \tag{7.17}$$

This approach is used in our case study as illustrated in Figure 7.4.

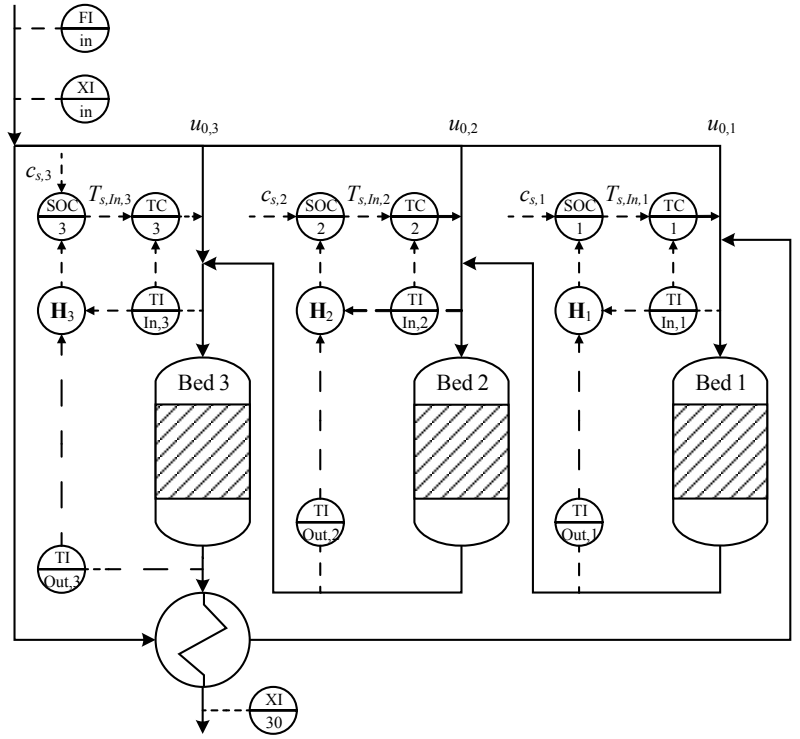


Figure 7.3: Flowsheet of the reactor case study, modified from [75] to include the proposed control structure.

7.3 Case Study - Ammonia Synthesis Reactor

The investigated case study is a three-bed ammonia reactor with heat integration as in the previous chapters. A flowsheet, including the control structure for the proposed method, can be found in Figure 7.3. Economic nonlinear model predictive control was able to react to disturbances and bring the reactor to its steady-state optimum. In order to avoid repeated numerical optimization and to handle disturbances and plant-model mismatch more efficiently, self-optimizing control, extremum-seeking control and a combination the both approaches is considered in this chapter. The incorporation of the reactor into the ammonia synthesis loop requires adjustments to the setpoints of the self-optimizing controllers as shown in Chapter 6. This adjustment can be achieved through extremum-seeking control.

The objective is to maximize the rate of extent of reaction $\dot{\xi}$ for a given feed, that is

$$J = \dot{\xi} \tag{7.18}$$

$$= \dot{m}_{in}\left(w_{\mathrm{NH_3},30} - w_{\mathrm{NH_3},in}\right) \tag{7.19}$$

In the context of the overall process, this corresponds to minimizing the recycle feed and hence minimizes the recycle compressor cost as well as the cooling cost in the separation section.

7.3.1 Summary of the Model

The model consists of three sequential reactor beds and one heat exchanger. The inlet stream to the reactor system (denoted by subscript *in*) is split into four streams; one quench flow to each bed and a preheated flow to the first reactor bed. The quench split ratios correspond to the three manipulated variables $\mathbf{u}_0 = \begin{bmatrix} u_{0,1} & u_{0,2} & u_{0,3} \end{bmatrix}^{\mathsf{T}}$. The three reactor beds are discretized into a cascade of continously stirred tank reactors (CSTR). We use the Temkin-Pyzhev kinetic expression for the reaction rate. The heat-exchanger is modelled using the number of transfer units (NTU) method. The resulting model without controllers corresponds to a differential-algebraic system with $\mathbf{x} \in \mathbb{R}^{30}$ as dynamic state variables corresponding to the temperatures in the beds, $\mathbf{z} \in \mathbb{R}^{30}$ as algebraic state variables corresponding to the ammonia mass fractions in the beds, and $\mathbf{u}_0 \in \mathbb{R}^3$ as manipulated variables. A detailed model description can be found in Appendix A. The system is modelled using CasADi [4]. The nominal optimal point and the optimal sensitivity matrix $\mathbf{F}$ for self-optimizing control were computed using the IPOPT nonlinear problem solver [102]. The plant model was simulated using IDAS [47].

7.3.2 Controller Design

The potential instability in case of disturbances as described in [75] requires a stabilizing *slave* control layer. Otherwise, large disturbance would result in limit-cycle behaviour or even reactor extinction. It was shown in Chapter 4, that if the reactor is operated close to the nominal optimum and without control, the reactor extinction may result even from small disturbances compared to the large disturbances investigated by [75]. Hence, also for the case when extremum-seeking control is utilized without self-optimizing control, a stabilizing *slave* control layer is required resulting in a cascade controller for all investigated control structures. This furthermore reduces the coupling between the self-optimizing controllers. The slave temperature loops as well as the master self-optimizing control loops were tuned using the SIMC rules [93]. These controllers were directly included into the differential-algebraic model increasing the number of differential variables by three for the extremum-seeking control or six for the combined self-optimizing and extremum-seeking control. The extremum-seeking controllers were implemented in discrete time resulting in a discrete-continuous representation.

Table 7.2: PI tuning parameters and of the temperature and SOC controllers in Figure 7.3.

		Input (MV)	Output (CV)	K_p [-]	K_I [s^{-1}]
Slave	TC 1	$u_{0,1}$	$T_{In,1}$	[-]	-2.1×10^{-4}
	TC 2	$u_{0,2}$	$T_{In,2}$	[-]	-2.7×10^{-4}
	TC 3	$u_{0,3}$	$T_{In,3}$	[-]	-4.2×10^{-4}
Master	SOC 1	$T_{s,In,1}$	c_1	0.169	0.76×10^{-3}
	SOC 2	$T_{s,In,2}$	c_2	0.209	1.25×10^{-3}
	SOC 3	$T_{s,In,3}$	c_3	1.000	5.0×10^{-2}

Slave Temperature Controller Pairing and Tuning

Slave temperature controllers are introduced in all control structures studied in this paper. The slave controllers use the splits (bypass) $u_{0,i}$ to control the corresponding bed inlet temperature. This is a pure mixing process with instantaneous dynamics, and an integrating controller is recommended [93]. The desired closed loop time constant for the three controllers was chosen to be $\tau_c = 10$ s. The resulting integral gain K_I for the three temperature loop controllers can be found in Table 7.2.

SOC Controller Pairing and Tuning

The SOC controllers give the setpoints to the respective slave temperature controllers. The measurements $\mathbf{y}$ for self-optimizing control are selected to be the inlet and outlet temperature of each reactor bed; *i.e.*

$$\mathbf{y}_i = \begin{bmatrix} T_{In,i} \\ T_{Out,i} \end{bmatrix} \qquad i = 1,2,3 \tag{7.20}$$

Hence, only two measurements were used for the calculation of $\mathbf{H}_i$ in (7.8). This local treatment of each bed does not necessarily result in overall optimal selection matrices $\mathbf{H}_i$. It would be possible to increase the number of measurements, *e.g.* using all 6 measurements for the calculation of the selection matrices. This will reduce somewhat the steady-state loss in self-optimizing control [107]. However, it may also lead to undesired dynamic behaviour through coupling and delays in the self-optimizing variables $\mathbf{c}$.

The disturbances are the inlet conditions;

$$\mathbf{d} = \begin{bmatrix} \dot{m}_{in} & p_{in} & T_{in} & w_{\mathrm{NH_3},in} \end{bmatrix}^{\mathsf{T}} \tag{7.21}$$

The chosen scaling matrices in (7.9) are

$$\mathbf{W_d} = \begin{bmatrix} 7 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 0.008 \end{bmatrix} \tag{7.22}$$

$$\mathbf{W_{n^y}} = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \tag{7.23}$$

The expected disturbance magnitudes ($\mathbf{W_d}$) are 10 % of the nominal value whereas $\mathbf{W_{n^y}}$ corresponds to a measurement noise of 4 K for each measurement. The resulting (scaled) selection matrices $\mathbf{H}_i$ from (7.8) are then given by

$$\mathbf{H}_1 = \begin{bmatrix} -0.03 \\ 1.00 \end{bmatrix} \qquad \mathbf{H}_2 = \begin{bmatrix} 0.23 \\ 1.00 \end{bmatrix} \qquad \mathbf{H}_3 = \begin{bmatrix} 1.00 \\ 0.74 \end{bmatrix} \tag{7.24}$$

To get a fast response in the self-optimizing control layer, because it will be combined with an upper extremum-seeking layer, the closed-loop time constant τ_c for each of the three controllers is equal to its respective time delay. The resulting PI parameters (K_p and K_I) can be found in Table 7.2.

Extremum-Seeking Controllers Tuning

The upper layer in the control structure in Figure 7.2 consists of the three extremum-seeking controllers. These slow integral controllers give the setpoints to either the base layer temperature ($\mathbf{T}_{s,In}$, denoted T+ESC) or the self-optimizing controllers ($\mathbf{c}_s$, denoted T+SOC+ESC). The estimation of the gradient $\hat{\theta}$ according to (7.15) is performed using $\tilde{\mathbf{u}}$ as the setpoint of the respective slave controller (T or SOC). It is assumed that the disturbances are unmeasured. Hence, θ is given by

$$\theta = \begin{bmatrix} \mathbf{J}_{\tilde{\mathbf{u}}}^{\mathsf{T}} & m \end{bmatrix}^{\mathsf{T}} \tag{7.25}$$

where m is the present value of the cost and $\mathbf{J}_{\tilde{\mathbf{u}}}$ the gradient. As the disturbances are not corrected for, this will result in wrong gradient estimation when disturbances occur. One way to rectify this problem is to temporarily turn off the extremum-seeking controllers when this happens. It may, however, not be obvious that a disturbance is occuring. Hence, instead the gradients $\hat{J}_{\tilde{u}_i}$ are bounded as shown in (7.17).

The tuning of extremum-seeking controllers depends on multiple factors. We need to choose the number of past measurements N, the periods and amplitudes of the sinusoidal dithers, as well as the integral gains. All these parameters have an influence on each other resulting in a difficult selection. The parameters were chosen based on trial and error to

Table 7.3: Controller tuning parameters for the extremum-seeking controllers in the case of only temperature controllers (T) and also self-optimizing control (SOC) as the setpoint control layer.

	Controller	Amplitude [K]	Period [h]	$J_{u_i,max}$ [kg s^{-1} K^{-1}]	K_I [s^{-1}]
T	ESC 1	1.0	2.0	0.3	0.05
	ESC 2	1.0	2.5	0.225	0.05
	ESC 3	1.0	3.0	0.225	0.05
SOC	ESC 1	1.0	2.0	0.3	0.08
	ESC 2	1.0	2.5	0.45	0.04
	ESC 3	1.0	3.0	0.45	0.04

achieve satisfactory performance and are given in Table 7.3. The time horizon for the past measurements was chosen to be 1 hour in all cases. This corresponds to $N = 240$ samples with a chosen integrator step time of $t_{int} = 15$ s. Equal effort for both T+ESC and T+SOC+ESC tuning was attempted to achieve comparable performance.

7.3.3 Results

In order to compare the proposed methods, two disturbances were investigated; a disturbance in the inlet mass flow rate $\dot{m}_{in}$, corresponding to a modelled disturbance in self-optimizing control, and an unmodelled disturbance in the reaction rate r. These disturbances were chosen as they correspond to the largest losses for the self-optimizing control structure (not shown). Hence, the improvement using extremum-seeking control is most pronounced. In addition, both disturbances would result in reactor extinction, if the stabilizing temperature controllers would not be present. The integrated loss (cost difference),

$$J_{int}(t) = \int_0^t \left[\dot{\xi}_{opt,SS}(t') - \dot{\xi}(t') \right] \mathrm{d}t' \tag{7.26}$$

is used to compare the proposed methods.

First, consider a $+20$ % step change in the inlet mass flow rate to see the impact of the bounds (7.17) on the gradient estimates $\hat{J}_{\tilde{u}_i}$. Figure 7.4 shows the bounds and the gradient estimate for gradient 1 as well as the corresponding manipulated variable, $c_{s,1}$. As we can see, the gradient estimate at the time of the disturbance ($t = 3$ h) is indeed outside the respective bound. At $t = 3.025$ h it reaches a minimum value of $\hat{J}_{\tilde{u}_i} = -93.42$ kg s^{-1} K^{-1}, a value 300 times as large as the bound. The estimate is within the bound 1 h after the occurrence of the disturbance ($t = 4$ h). This corresponds to the time horizon of the past measurements according to (7.15). If no bounds are introduced, the wrong estimate

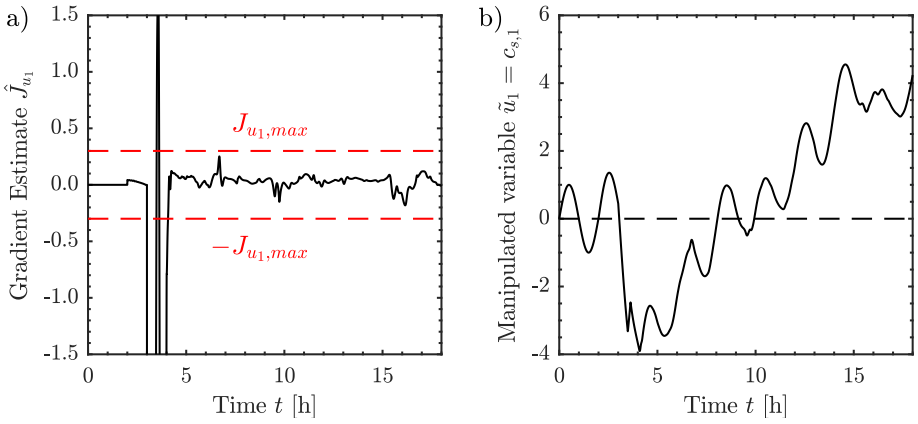


Figure 7.4: a) Gradient estimate for ESC controller 1 and b) controller output $c_{s,1}$.

of the gradient would require a very small integrator gain for the extremum-seeking controller resulting in a very slow convergence.

The cost $J = \dot{\xi}$ and the integrated loss (7.26) are shown in Figure 7.5. The cases with extremum-seeking control settle to the new optimum in contrast to pure self-optimizing control. The combination of self-optimizing control and extremum-seeking control gives a large reduced loss in produced tons of ammonia. As seen in Figure 7.5, this reduction

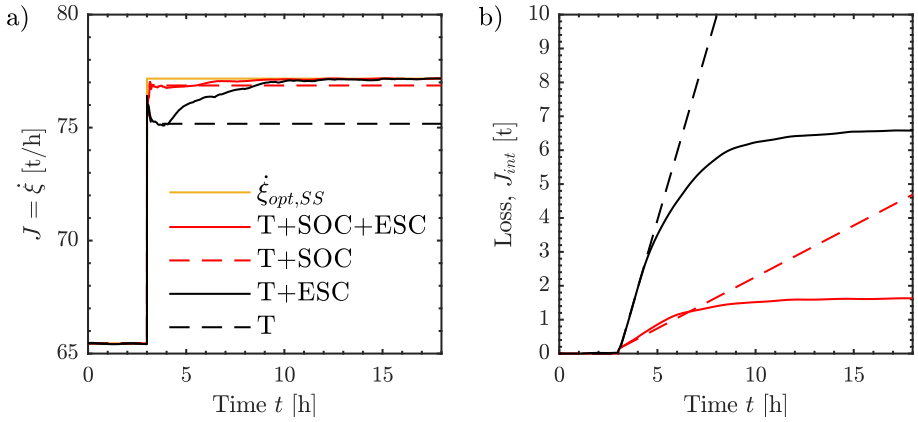


Figure 7.5: Response of a) rate of extent of reaction $\dot{\xi}$ and b) integrated loss to a +20 % disturbance in inlet mass flow rate, $\Delta\dot{m}_{in} = +54$ t/h, at $t = 3$ h.

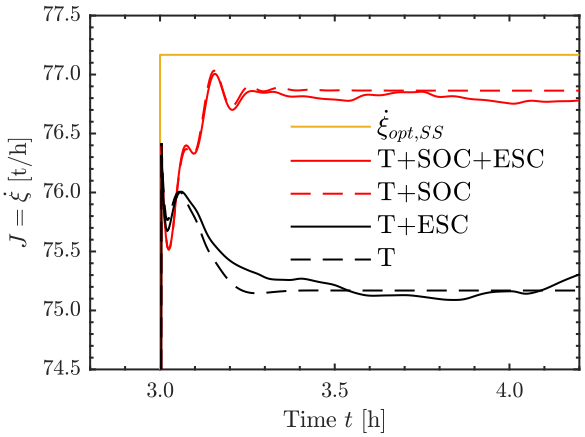


Figure 7.6: Closeup of Figure 7.5 a) at the time when the disturbance occurs.

corresponds to 4.95 t ammonia in the investigated time-frame of 18 hours. One could argue that this is caused by suboptimal tuning parameters in the pure extremum-seeking control. By taking a look at the time the disturbance is occurring, we claim that this is not the case. Figure 7.6 shows the response in the cost function for the first 1.2 hours after the disturbance occurs. From this figure, it can be clearly seen that both ESCs (solid lines) initially follow their respective slave controllers, before deviating when the ESCs start changing the setpoints to the slave controllers. Both ESC control structures are in fact moving initially in the wrong direction, that is, to a reduced rate of extent of reaction. This can be explained by the past measurements, before the disturbance, which are still used at this point. One approach to circumvent this behaviour is to use a smaller time horizon (smaller N). This results on the other hand in a drift away from the optimal setpoint on a long time scale. Hence, it is preferable to have a slightly suboptimal initial performance.

A disturbance in the reaction rate r is an unmodelled disturbance. This implies that it is not considered in the calculation of the optimal selection matrices according to (7.8). It can be considered a plant-model mismatch. The simulation results for a -20 % step change in the reaction rate r is shown in Figure 7.7. Similarly to a disturbance in the inlet mass flow $\dot{m}_{in}$, the control structure based on the proposed method with both self-optimizing and extremum-seeking control settles to the new optimum after 7 hours whereas extremum-seeking control alone requires around 13 hours. During the time the controllers require to settle to the new optimum, the loss is reduced in the proposed con-

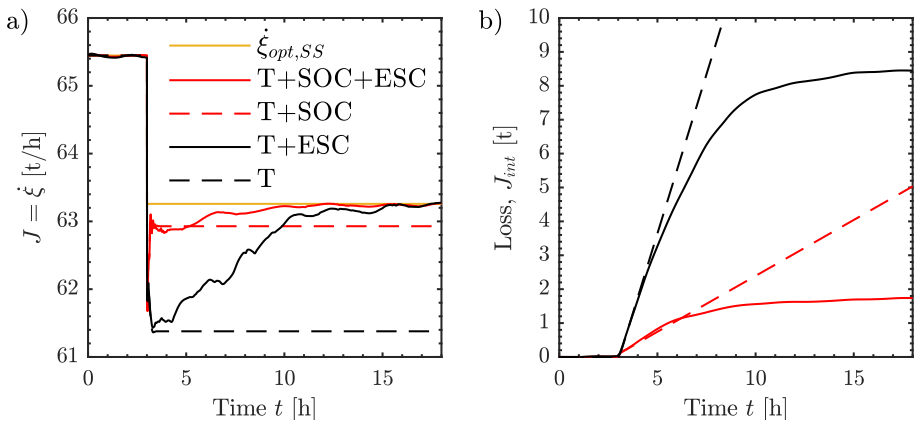


Figure 7.7: Response of a) rate of extent of reaction $\dot{\xi}$ and b) integrated loss to a -20 % disturbance in pre-exponential factors of the Arrhenius equations at $t = 3$ h.

trol structure with SOC. Over 18 hours, the proposed control structure has a reduced loss of 6.71 tons of produced ammonia. Here it has to be noted, that despite this disturbance was not included in the design phase, the self-optimizing control structure has a reduced loss. This can be explained by general favourable properties of self-optimizing feedback with regard to disturbances and plant-model mismatch.

7.4 Discussion

As shown, the hierarchical combination of self-optimizing control with extremum-seeking control improves the rejection of disturbances. This is caused by the (fast) rejection of the disturbance through self-optimizing control combined with the final adjustment of the setpoints $\mathbf{c}_s$ by the extremum-seeking controllers. Is it still possible to speak of self-optimizing control in the context, when the setpoint is adjusted? Yes, the idea of self-optimizing control is to allow for less frequent changes in the setpoint. Skogestad [90] speaks in his original paper on self-optimizing control explicitly of the possibility to adjust the setpoint for SOC variables using an optimizing layer. This is especially important considering the incorporation of the reactor into the synthesis loop in which the recycle is neglected as shown in Chapter 6. The proposed method adjusts then the setpoints, if it is not possible to solve an optimization problem for the overall process

Self-optimizing control is model-based. However, the proposed method is less reliant on the accuracy of the model than many other model-based approaches, since it only uses

the model offline for the calculation of the optimal selection matrices $\mathbf{H}_i$. The setpoints of the self-optimizing controllers are handled by the extremum-seeking controllers in a model-free approach.

There remains however one limitation to the proposed methodology; it is necessary to measure (or estimate) the cost function for the extremum-seeking controller. This is similar to the sole application of extremum-seeking control and can be seen as an inherent limitation of this type of methods.

7.5 Conclusion

It was shown that extremum-seeking control and self-optimizing control are complementary rather than competing. By combining self-optimizing control and extremum-seeking control, the convergence to the optimum is improved and it is possible to handle a wider class of uncertainty than each of the methods individually. Using a three bed ammonia reactor case study, it was demonstrated that the combined system can handle unmeasured and unexpected disturbances and at the same time correct for plant model mismatch.

Chapter 8

Feedback Steady-State Real-Time Optimization

Economic NMPC as described in Chapter 5 handles the dynamic process behaviour, operational constraints, and leads to the optimal inputs for this multivariable processes. Nevertheless, solving the optimization problem for a large-scale problem is computationally expensive and can potentially lead to computational delays. Furthermore, the modelling and controller tuning is challenging to ensure good performance over time [94].

The application of self-optimizing control in Chapter 6 showed, that it is possible to keep the reactor close to optimum. Therefore, it can be used for close to optimal operation while waiting for the steady-state [90]. The implementation is very fast and simple, but in case of unknown or large disturbances, it was shown that the setpoints need to be updated using some other approach, *e.g.* a data-based approach like extremum-seeking control as shown in Chapter 7. Closely related approaches are the *hill-climbing* controller of Shinskey used recently by Kumar and Kaistha [63] and the NCO-tracking approach of Bonvin and coauthors [35]. Their main advantage is that they are model free. The main challenge in these methods is the accurate estimation of the steady-state gradient from dynamic measurements. This normally requires constant excitations that are slow enough such that the dynamic system can be approximated as a static map [62]. As a result the convergence to the optimum is usually very slow. In the presence of abrupt disturbances, extremum-seeking control also causes unwanted deviations as discussed in Chapter 7 and Krishnamoorthy et al. [60].

In this chapter, a new model-based dynamic gradient estimation [61] is applied to drive the process to optimal operation. In contrast to standard extremum-seeking control, the exact steady-state gradients is estimated based on the dynamic model of the process

and hence no excitations are required. In addition, there is no need to measure the cost directly for the proposed method. Moreover, reoptimization is done by feedback control and solving the optimization problem is not necessary.

This chapter starts with the explanation of the utilized estimator and the proposed method in Section 8.1. Section 8.2 formulates then the problem and the model. The results are presented in Section 8.3, including a comparison with the combination of extremum-seeking control and self-optimizing control (Chapter 7).

8.1 Steady-state Gradient Control Using Transient Measurements

We consider a process that can be modelled as a nonlinear dynamic system of the form

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{d}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{h}_{\mathbf{y}}(\mathbf{x}, \mathbf{d}, \mathbf{u}) \end{aligned} \tag{8.1}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$, $\mathbf{u} \in \mathbb{R}^{n_u}$, $\mathbf{d} \in \mathbb{R}^{n_d}$, and $\mathbf{y} \in \mathbb{R}^{n_y}$ are the states, available control inputs, disturbances, and measurements. The cost, which we want to minimize, does not need to be directly measured, but is instead given by

$$J = h_J(\mathbf{x}, \mathbf{d}, \mathbf{u}) \tag{8.2}$$

with $h_J : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \to \mathbb{R}$. In the proposed method, a state estimator such as an extended Kalman filter (EKF) [89] is first applied to estimate the states $\mathbf{x}$ of the system by using the measurements and the dynamic model, given in Eq. (8.1). This is different to Chapter 5 were full state knowledge was assumed.

8.1.1 Extended Kalman Filter

For a discrete-time extended Kalman filter, it is necessary to rewrite the model as

$$\mathbf{x}_k = \mathbf{f}_{EKF,k}(\mathbf{x}_{k-1}, \mathbf{d}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{n}^{\mathbf{x}} \tag{8.3}$$

$$\mathbf{y}_{k,meas} = \mathbf{h}_{\mathbf{y}}(\mathbf{x}_{k-1}, \mathbf{d}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{n}^{\mathbf{y}} \tag{8.4}$$

in which $\mathbf{n}^{\mathbf{x}} \sim \mathcal{N}(0, \mathbf{Q}_k)$ is the process noise with covariance $\mathbf{Q}_k$ and $\mathbf{n}^{\mathbf{y}} \sim \mathcal{N}(0, \mathbf{R}_k)$ is the measurement noise with covariance $\mathbf{R}_k$. Reformulating Eq. (8.1) using the forward Euler method, we can write

$$\mathbf{f}_{EKF,k} = \mathbf{x}_{k-1} + \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{d}_{k-1}, \mathbf{u}_{k-1})\Delta t \tag{8.5}$$

in which Δt corresponds to the discrete-time step. At each step k, it is first necessary to linearize the system around the last previous operating point given by

$$\mathbf{F}_k = \left.\frac{\partial \mathbf{f}_{EKF,k}}{\partial \mathbf{x}_{k-1}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{d}_{k-1},\mathbf{u}_{k-1}} \tag{8.6}$$

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}_\mathbf{y}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{d}_{k-1},\mathbf{u}_{k-1}} \tag{8.7}$$

The prediction step of the extended Kalman filter for the state ($\hat{\mathbf{x}}$) and covariance ($\mathbf{P}$) estimates is described as

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}_{EKF,k}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{d}_{k-1}, \mathbf{u}_{k-1}) \tag{8.8}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^\mathsf{T} + \mathbf{Q}_k \tag{8.9}$$

Using the prediction of the covariance, the Kalman gain $\mathbf{K}_k$ is calculated as

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^\mathsf{T} \left(\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\mathsf{T} + \mathbf{R}_k\right)^{-1} \tag{8.10}$$

and used in combination with the measurements $\mathbf{y}_{k,meas}$ to update the estimates of the states and the covariance

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \left(\mathbf{y}_{k,meas} - \mathbf{h}_y \left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_{k-1}, \mathbf{d}_{k-1}\right)\right) \tag{8.11}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|k-1} \tag{8.12}$$

As it can be seen in the equations of the extended Kalman filter, it is necessary to measure the disturbances. In the case of unmeasured disturbances and unknown parameters, it was suggested [58] to estimate these simultaneously with the states. Therefore, we can rewrite the model given in Eqs. (8.1) to include a differential equation for the disturbances

$$\dot{\mathbf{d}} = \mathbf{w}_\mathbf{d} \tag{8.13}$$

$\mathbf{w}_\mathbf{d}$ is given by a small artificial noise term. The resulting augmented systems with states

$$\mathbf{x}' = \begin{bmatrix} \mathbf{x} \\ \mathbf{d} \end{bmatrix} \tag{8.14}$$

can then be formulated as

$$\begin{aligned} \dot{\mathbf{x}}' &= \begin{bmatrix} \mathbf{f}'(\mathbf{x}', \mathbf{u}) \\ \mathbf{w}_\mathbf{d} \end{bmatrix} \\ \mathbf{y} &= \mathbf{h}'_\mathbf{y}(\mathbf{x}', \mathbf{u}) \end{aligned} \tag{8.15}$$

The models $\mathbf{f}'$ and $\mathbf{h}'_\mathbf{y}$ are the same as the ones given in Eq. (8.1). However, due to the change in notation with $\mathbf{x}'$, it seemed reasonable to rename them. Using the model

described in Eq. (8.15), it is possible to estimate the states and disturbances *via* Eqs. (8.6) to (8.12). If certain disturbances are measured, it is as well possible to only estimate some of the disturbances. It is however necessary, that observability conditions hold. This can be tested using *e.g.* the observability matrix, the observability Gramian, or the output pole vectors. This is explained in detail in [94].

8.1.2 Estimation of the Gradient of the Steady State System

The cost function described in Eq. (8.2) is at each time linearized around the current operating point. This results in a local linear dynamic model for the states $\mathbf{x}$ and the cost j as a function of the input $\mathbf{u}$.

$$\begin{aligned} \dot{\Delta\mathbf{x}} &= \mathbf{A}\Delta\mathbf{x} + \mathbf{B}\Delta\mathbf{u} \\ \Delta J &= \mathbf{C}\Delta\mathbf{x} + \mathbf{D}\Delta\mathbf{u} \end{aligned} \tag{8.16}$$

The system matrices of the state-space representation are hereby evaluated at the current operating point through the estimates of the states and measurements of the disturbances.

$$\mathbf{A} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}},\mathbf{d},\mathbf{u}} \qquad \mathbf{B} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\hat{\mathbf{x}},\mathbf{d},\mathbf{u}}$$
$$\mathbf{C} = \left.\frac{\partial h_J}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}},\mathbf{d},\mathbf{u}} \qquad \mathbf{D} = \left.\frac{\partial h_J}{\partial \mathbf{u}}\right|_{\hat{\mathbf{x}},\mathbf{d},\mathbf{u}}$$

In order to obtain the steady-state gradient, we can set $\dot{\Delta\mathbf{x}} = 0$ and derive in deviation variables

$$\Delta J = \left(-\mathbf{C}\mathbf{A}^{-1}\mathbf{B} + \mathbf{D}\right)\Delta\mathbf{u} \tag{8.17}$$

which, since $\Delta J = \mathbf{J}_\mathbf{u}\Delta\mathbf{u}$, gives the following estimate or prediction of the steady-state gradient

$$\hat{\mathbf{J}}_\mathbf{u} = -\mathbf{C}\mathbf{A}^{-1}\mathbf{B} + \mathbf{D} \tag{8.18}$$

We want to drive the system to an optimal steady-state where $\mathbf{J}_{\mathbf{u},s} = \mathbf{0}$, so even if the system is not at steady-state, we can use feedback control with $\mathbf{y} = \hat{\mathbf{J}}_\mathbf{u}$ as *measurements* to drive the system to the optimal steady-state and by that satisfying the necessary conditions of optimality [61]. Any feedback controller, such as a PI controller, can be used to bring the gradient to zero. It is important to note that by using a nonlinear state estimator and a dynamic model for estimating the steady-state gradient $\mathbf{J}_\mathbf{u}$, we can use transient measurements, without the need to wait for steady-state, as in traditional RTO.

The scheme of the proposed method is illustrated in Figure 8.1. Although an extended Kalman filter was used for state estimation, this is not a necessity. In general, any observer may be used for the state estimation.

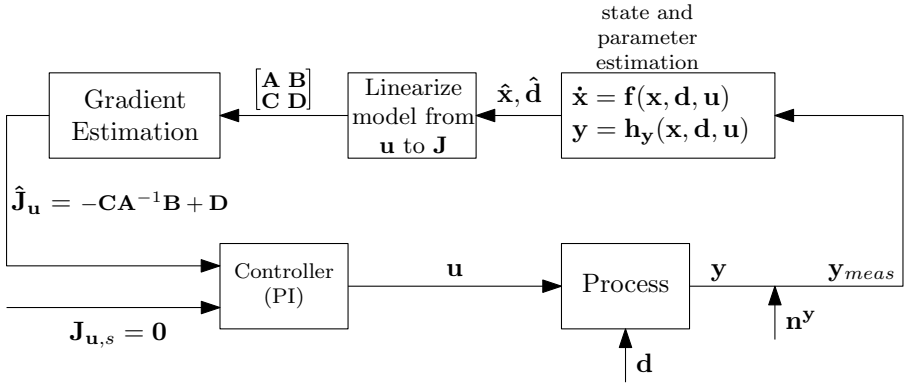


Figure 8.1: Block diagram of the proposed method.

8.2 Adaptation of the Model and Problem Formulation

The model of the ammonia reactor and all the model assumptions are explained in detail in Appendix A. The 3 split ratios $\mathbf{u}_0 = \begin{bmatrix} u_{0,1} & u_{0,2} & u_{0,3} \end{bmatrix}^\mathsf{T}$ are controlled by local temperature controllers. This is necessary for stabilizing the process for the proposed procedure and was already applied in Chapter 7. The temperature controllers are incorporated into the model in continous time increasing the number of states by 3. This leads to $\mathbf{u} = \begin{bmatrix} T_{s,In,1} & T_{s,In,2} & T_{s,In,3} \end{bmatrix}^\mathsf{T}$ for the proposed procedure. The temperature controllers are modelled as single-input single-output integrator controllers, as the response can be approximated as a proportional process. The parameters of the temperature controller can be found in Table 7.2. The required differential equations for the integrated error $\mathbf{e}_{int}$ in the temperature controllers are given by

$$\frac{\mathrm{d}e_{int,j}}{\mathrm{d}t} = T_{s,In,j} - T_{In,j} \qquad \forall i = 1,2,3 \tag{8.19}$$

The SIMC rules [93] were applied for the slave controllers tuning.

In contrast to Chapter 5, full state knowledge is not assumed in this chapter. The state estimation is performed using an EKF as described in Section 8.1.1. The measurement set for state estimation is given by the inlet and outlet temperature of each reactor as well as the outlet temperature of the heat exchanger (see Figure 8.2). This corresponds to

$$\mathbf{y} = \begin{bmatrix} T_{HEx} & T_{In,1} & T_{Out,1} & T_{In,2} & T_{Out,2} & T_{In,3} & T_{Out,3} \end{bmatrix}^\mathsf{T} \tag{8.20}$$

To this end, the model was reformulated as a system of ordinary differential equations. As shown in Appendix A, each reactor bed in the model consists of n discrete volumes,

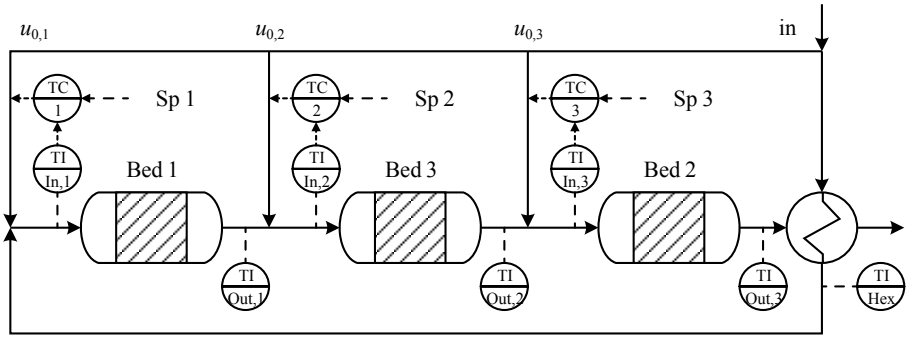


Figure 8.2: Heat-integrated 3 bed ammonia synthesis reactor with cascade control. The setpoint of the slave temperature loop is given by the proposed method.

which can be modelled as a CSTR cascade. This leads for each reactor bed to a total of $2n$ state variables per time step. So far, the ammonia weight fractions $w_{\mathrm{NH}_3,j}$ were considered as algebraic variables. For any CSTR reactor j in the CSTR cascade, the differential equations for the ammonia weight fractions $w_{\mathrm{NH}_3,j}$ can be formulated as seen in Eq. (8.21), in which $\alpha = 0.33$ represents the bed void fraction, $\rho_g = 50$ kg/m^3 the density of the gas, and $V_j = V_{bed}/n$ the volume of each CSTR reactor j [74].

$$\frac{\mathrm{d}w_{\mathrm{NH}_3,j}}{\mathrm{d}t} = \frac{\dot{m}_{j-1} w_{\mathrm{NH}_3,j-1} - \dot{m}_j w_{\mathrm{NH}_3,j} + m_{cat,j} r_{NH_3,j}}{V_j \rho_g \alpha} \tag{8.21}$$

To summarize, we can write, $\mathbf{x} \in \mathbb{R}^{6n+3}$, $\mathbf{u} \in \mathbb{R}^3$ in the system, given in Eqs. (8.1).

The catalyst activity is changing over time in real plants. This is in general difficult or impossible to measure and leads to a plant-model mismatch. To take into account industrial applicability, we assume, that a change in the catalyst activity is not measured, but included in the model as an uncertain parameter. Hence, the states and the uncertain parameter are combined to the augmented states with $d = [a_{cat}]$. This results in an augmented system given by Eq. (8.15).

To optimize the operation, we want to maximize the (mass) rate of extent of reaction.

$$\dot{\xi} = \dot{m}_{in}(w_{\mathrm{NH}_3,out,3} - w_{\mathrm{NH}_3,in}) \tag{8.22}$$

This results in a cost function $J = -\dot{\xi}$. In this case, a cascade control is used, where the master controllers drive the three gradients to zero by giving new set points to three slave control loops. The EKF and the proposed method were implemented in discrete time. The controllers of the proposed method are single-input single-output controllers.

Table 8.1: PI tuning parameters and of the temperature and SOC controllers in Figure 8.2.

		Input (MV)	Output (CV)	K_p [-]	K_I [s^{-1}]
Slave	TC 1	$u_{0,1}$	$T_{In,1}$	[-]	-2.1×10^{-4}
	TC 2	$u_{0,2}$	$T_{In,1}$	[-]	-2.7×10^{-4}
	TC 3	$u_{0,3}$	$T_{In,1}$	[-]	-4.2×10^{-4}
Master	SOC 1	$T_{s,In,1}$	$\hat{J}_{u,1}$	65	1.6
	SOC 2	$T_{s,In,2}$	$\hat{J}_{u,1}$	61	3.4
	SOC 3	$T_{s,In,3}$	$\hat{J}_{u,1}$	80	4.45

The proportional (K_p) and integral (K_I) gain of the controllers can be found Table 8.1. The continuous time process model, given in Eq. (8.1) and Appendix A, was modelled using CasADi [4] and integrated with CVODES, which is part of the SUNDIALS package [47].

8.3 Performance Analysis

In the following section, we consider a disturbance in the feed flow and a plant-model mismatch, given by a mismatch in the catalyst activity. In all cases, we have three inner stabilizing temperature loops as indicated by the letter "T" on the plots. In addition, the results are compared to pure self-optimizing control (SOC) and self-optimizing control with extremum-seeking control (SOC+ESC) in the optimization layer. The latter corresponds to the results presented in Chapter 7. The controller parameters for both self-optimizing control and extremum-seeking control as optimization layer for self-optimizing control can be found in Table 7.2. The integrated cost difference (loss) J_{int} is used for comparison of the different methods

$$J_{int}(t) = \int_0^t \left[\dot{\xi}_{opt,SS}(t') - \dot{\xi}(t') \right] \mathrm{d}t' \tag{8.23}$$

First we simulate a disturbance change in the inlet flowrate $\dot{m}_{in}$ to evaluate the performance of the control structure. The results for an increase in the feed flowrate of $\Delta\dot{m}_{in} = 15$ kg/s at time $t = 1$ h are presented in Figure 8.3. The new proposed method gives fast disturbance rejection and settles down at the new optimal operation after about 30 min, as seen Figure 8.3 a). SOC is equally fast, but it does not quite reach the new optimum due to its constant setpoint approach. This leads to a continous increase in the integrated loss for SOC as seen in Figure 8.3 b). If we compare the proposed method to ESC as optimizing layer on top of SOC, the proposed method is much faster and therefore causes a lower integrated cost difference of $J_{int}(t_{end}) = 0.1$ t. This is because

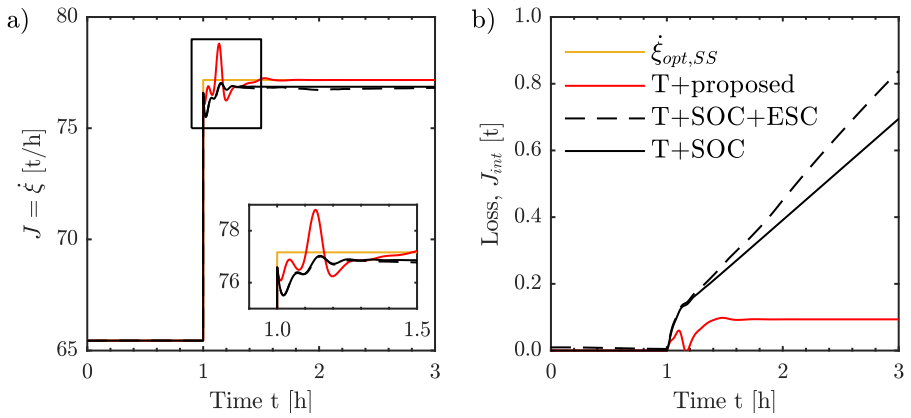


Figure 8.3: Responses of the alternative methods in a) the rate of extent of reaction and b) the integrated loss to a disturbance in the feed flowrate of $\Delta\dot{m}_{in} = 15$ kg/s at time $t = 1$ h. $\dot{\xi}_{opt,SS}$ represents the steady-state optimal extent of reaction.

the data-based gradient estimation takes longer time for accurate gradient estimation as shown in Chapter 7 and the controller gain has to be small to satisfy stability. Extremum-seeking control does not settle to the steady-state optimum in the investigated time frame as well.

In the second simulation, we consider plant-model mismatch. The results for a decreased catalyst activity Δa_{cat} by 20 % at time $t = 1$ h, which normally occurs slowly over a longer period of time, are presented in Figure 8.4. Therefore, the activity of the catalyst, or more specifically the pre-exponential factor of the Arrhenius equation, spontaneously changed between the model used for the simulations and the model for the state estimation. The simulation shows that the proposed method is performing well even in the presence of a plant-model mismatch. This is because we are able to estimate the real value of the catalyst activity using the augmented EKF framework presented above. About 1.5 minutes after the activity change, the mismatch as well as the states are estimated correctly. This good performance requires however the incorporation of the catalyst rate as disturbance into the model. The proposed method is much faster than T+SOC+ESC, which in turn results in a lower total loss as seen in Figure 8.4 b). Again, SOC is equally fast, but the real optimum is not reached. The proposed method causes a total integrated cost difference of about $J_{int}(t_{end}) = 0.12$ t of ammonia for the considered case with plant-model mismatch.

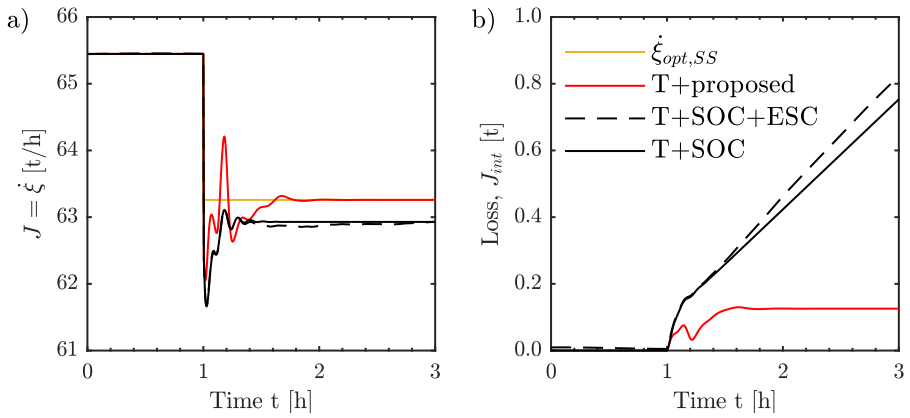


Figure 8.4: Responses of the alternative methods in a) rate of extent of reaction and b) the integrated loss to a plant-model mismatch of $\Delta a_{cat} = -20$ % at time $t = 1$ h. $\dot{\xi}_{opt,SS}$ represents the steady-state optimal extent of reaction.

The reactor settles to the new steady-state optimum in around 30 min. This is one order of magnitude faster than the extremum-seeking approach. However, it is still three times slower than economic nonlinear model predictive control.

8.4 Conclusion

A new method of utilizing transient measurements and a dynamic estimator to estimate the steady-state gradient was applied to the ammonia reactor case study. This allows the usage of a simple PI controller for driving the process to its optimal operation. The method outperforms comparable methods. The industrial applicability is conceivable due to the usage of only seven measurements of the process besides the used dynamic model. An extended Kalman filter allows the estimation of the gradients of the steady-state system, even in case of plant-model mismatch by including unmeasured but modelled parameters in the estimator.

Chapter 9

Summary of the Different Methods

Part II investigated different methods for optimal operation of subprocesses of chemical processes. The utilized case study in all chapters were given by the ammonia reactor as described Appendix A. The investigated methods for optimal operation are given by

1. economic nonlinear model predictive control (E-NMPC) in Chapter 5;
2. self-optimizing control (SOC) in Chapters 6 and 7;
3. extremum-seeking control (ESC) with and without self-optimizing control as lower layer in Chapter 7;
4. feedback steady-state real-time optimization (F-RTO) in Chapter 8.

Table 9.1 gives a summary of the key properties of the different methods, which will

Table 9.1: Comparison of the different methods investigated in Part II.

	E-NMPC	SOC	ESC	F-RTO
Model usage	online	offline	none	online
Optimization	online	offline	none	none
Measurement of Cost	no	no	yes	no
Perturbation of the Plant	no	no	yes	no
Convergence	fast	fast	slow	fast
Optimality	yes[1]	near	yes	yes[1]
Constraint Handling	yes	yes[2]	no	no

[1] when there is not plant-model mismatch.
[2] SOC can handle constraints, albeit the implementation can be difficult if the number of constraint regions is large [53].

be elaborated further in the following sections. E-NMPC is on the one hand an optimizing controller, in which a dynamic optimization problem is solved. On the other hand, SOC, ESC, and F-RTO are feedback methods for achieving the steady-state optimum. This chapter illustrates the advantages and disadvantages of the different methods with respect to optimal operation.

9.1 Economic Model Predictive Control

Economic Model Predictive Control can be seen as a benchmark for analysis of different methods. Among the considered methods, E-NMPC is the only one that takes into account the optimal trajectory to the new steady-state optimum as it solves a dynamic optimization problem. It is in addition a true multivariable controller compared to the other methods presented in this part. As a result, it can handle coupling between the controlled variables without the introduction of decoupling. A further advantage is the possibility to respond to a change in active constraints. A change in the active constraint set may occur *e.g.* by saturation of manipulated variables. These correspond to the split ratios in the ammonia reactor. If one of the split ratios is 0 (or alternatively the sum is 1), single-input single-output controllers as used in the three other approaches may saturate. This results in loss of control for the respective controlled variable and can lead to limit-cycle or reactor extinction. E-NMPC takes this into account when calculating the optimal trajectories. The investigated case study did not have a change in active constraint. This may however happen in the case of other subprocesses with more inequality constraints.

However, there are as well major drawbacks in the application of E-NMPC. As it is model-based, every plant-model mismatch results in suboptimal performance. This requires that the mismatch of the model is constantly monitored and the model needs to be maintained. Furthermore, the states (and disturbances) of the system have to be either measured or estimated. This was not implemented in the case study as full state and disturbance knowledge was assumed. The application of an extended Kalman filter is however possible as shown in Chapter 8. The major drawback of E-NMPC is however given by the computational cost of the optimization problem. Despite the possibility to satisfy stability under certain assumptions [29], the solution time of the nonlinear problem may prohibit its application. Especially in the case of large-scale system, the sampling rate is restricted by the time to solve the optimization problem. Due to the computational delay, instabilities and performance degradation were observed in the case of NMPC [30]. The ever increasing computational power may reduce the issues associated to the computational delay in the future. But with current technology, limitations still exist. In addition, the integration of the optimizer and controller can pose problems in its practical application. If the optimizer fails and produces arbitrary results, the whole control structure will break down.

9.2 Self-Optimizing Control

Self-optimizing control is a strategy that moves the optimization problem from online to offline calculations. This is achieved through the control of variables $\mathbf{c} = \mathbf{h}(\mathbf{y})$ whose setpoints are less sensitive to disturbances. Note, that self-optimizing control is not a type of controller (like *e.g.* MPC or PID), but a philosophy of which variables should be controlled with the unconstrained degrees of freedom. The type of controller is not important, and hence, linear or nonlinear MPC can be used to achieve multivariable control. Manum and Skogestad [65] developed a method for active set changes in self-optimizing control based on results from explicit model predictive control as well.

As self-optimizing control is based on a linearization around the nominal operation point, persistent disturbances will lead to a constant steady-state loss as it was reported in Chapter 6. In the case of plant-model mismatch, this steady-state loss will be present even at the nominal operation point and it may be necessary to compute a new setpoint. Furthermore, if self-optimizing control is computed for subprocesses, neglecting dependencies of the disturbances on the manipulated variables may result in wrong optimal selection matrices. As shown in Chapter 6, it is however possible to achieve similar performance in this case, if the setpoint and the disturbance weighing matrix are updated.

9.3 Extremum-Seeking Control

Compared to all other investigated methods, extremum-seeking control does not require a model. Instead, it fits a linear model from the inputs to the cost function. This is a major advantage as plant-model mismatch cannot occur. As a consequence, ESC can even be applied in cases, where the model is too complicated for online optimization. If combined with self-optimizing control, ESC can be used in the context of real-time optimization to adjust the setpoints to the self-optimizing control layer as shown in Chapter 7. This approach combines the fast response of self-optimizing control with achieving true optimality using extremum-seeking control.

Extremum-seeking control requires the measurement (or estimation) of the cost. This may not be possible in all cases. In the investigated case-study, the cost consists of two concentrations, which are complicated to measure online. It requires furthermore a constant excitation of the plant through the dither in order to estimate the gradient correctly. This can be a limiting factor, as operators may be reluctant to disturb plants regularly. Both the dither and plant dynamics require a time-scale separation so that it is possible to assume the plant as a static map. As a result, the convergence to the steady-state optimum is slow. It is one order of magnitude slower than the other investigated methods. Least-square extremum-seeking controllers have as well problems, if large disturbances are present. As they use past measurements for the estimation of gradients, these mea-

surements correspond to the system without disturbance. Hence, the gradient estimation is incorrect at the point when the disturbance occurs. The proposed limitation on the change in the manipulated variables reduces the problems associated with large disturbances. It does however not account for the wrong gradient estimation, and hence, the convergence to the new optimum is still slow.

9.4 Feedback Steady-State Real-Time Optimization

The control of steady-state gradients using transient measurements has several advantages compared to the other methods. Similarly to ESC, it does not require the solution to an online nonlinear problem. Instead, the optimization problem is transformed into a feedback problem through a local linearisation around the current operating point. Opposite to ESC, F-RTO does however allow the utilization of transient measurement data. This improves the convergence rate to the steady-state optimum. This method does not require offline optimization either. Instead, the model is used for deriving linearized expressions for the gradients $\mathbf{J_u}$ as a function of the states and disturbances. Compared to self-optimizing control, it settles to the true steady-state optimum. As the gradient is not estimated using past measurements, it furthermore has the advantage that it can react fast to step disturbances.

As the method is model-based (like E-NMPC and SOC), the performance degrades in the case of plant-model mismatch. Additionally, this method requires accurate expressions for the continuous time-variant state-space representation.

9.5 Conclusion

The different applied methods have each their own advantages and disadvantages as described in the previous sections. In general, there is not a single method, which is ideal for all subprocesses. The applicability of the different methods is given by the availability of detailed models and measurements.

There is however one common problem in all applied methods; they require a local cost function, which corresponds or is close to the cost function of the overall plant. If this is not the case, then it can result in suboptimal performance of the overall process. That may not be possible in all situations. Frequently, there is an economic trade-off between the costs in the different subprocesses. If one considers for example the synthesis gas makeup section, the real economic cost function for this subprocess would consists of the interstage cooling and compressor duty. Hence, a local economic cost function would correspond to minimizing the outlet pressure and it would be required to introduce constraints, *e.g.* on the outlet pressure. Alternatively, cost values are assigned to the different stream values. The outlet pressure has an effect on the following reaction

section as well. Increasing the pressure results in an increase in the reaction rate and simultaneously shifting the equilibrium concentrations towards ammonia as it can be seen in the reaction rate expression (A.2). This implies that the optimal pressure may not correspond to the constraint defined in the synthesis gas makeup subprocess. Hence, it is necessary to find a cost function, which corresponds to the cost function of the overall plant as it is the case in the extent of reaction.

As an alternative, it is as well possible to extend the subprocess. This can lead to large optimization problems, which may become too complicated for a real-time application in E-NPMC and even for the offline optimization in self-optimizing control. If on the other hand simplified models are used, plant-model mismatch may become a prohibitive factor.

Part III develops a method for optimization of integrated plants through the introduction of surrogate models. The procedure can be applied in cases, where it is not possible to obtain a detailed model for optimization. This addresses both the issues with complicated steady-state models and the utilization of simplified models.

Part III

Optimal Operation through Introduction of Surrogate Models

Chapter 10

A Framework for Surrogate Model-Based Optimization

In chemical engineering, processes are frequently modelled using flowsheeting software. With sequential-modular simulation packages, like Aspen Plus®, Aspen Hysys®, SimSci PRO/II, or UniSim Design Suite, numerical problems may arise especially when we have large recycles. Furthermore, certain unit operations like reactors or columns may be computationally expensive to solve. In large integrated flowsheets, sequential-modular solver have on the one hand problems with convergence due to the recycles. Equation-oriented solvers are on the other hand difficult to initialize. Consequently, it can be necessary to simplify the model if it should be used in optimization. As an alternative, surrogate models may be introduced for individual units or combination of units with incorporated recycle streams resulting in reduced computational cost for solving the overall flowsheet.

Surrogate models, frequently called response surfaces or reduced order models, are an emerging field with many applications [32]. They are simplified mathematical representations of complex models. They can be seen as input-output mapping of a nonlinear models and are in this respect similar to lookup tables. Their application reduces the computational cost. Bhosekar and Ierapetritou [9] give a detailed overview of the application of surrogate models in the field of process systems engineering. One application field is multi-scale modeling [13, 56]. Surrogate models are, for example, in this approach used to include computational fluid dynamics calculations in process simulations. A second emerging field for surrogate models is the optimization of black-box models [15, 25, 33, 40, 84]. Commercial process simulators generally do not provide derivative information. This reduces their applicability in optimization. However, the fitting of surrogate models allows to use derivative-based optimization algorithms.

Caballero and Grossmann [15] developed an *algorithm for the use of surrogate models in modular flowsheet optimization*. In this approach, the surrogate model substitutes *noisy* and/or computational expensive models. By a *noisy* model we mean that the output from the model may vary because of numerical issues, for example, dependencies in the initial values. Their surrogate model was given by Kriging models [59] and can comprise several unit operations. This approach was also applied to a sour-water stripping plant [84]. Through the Kriging model structure, it is possible to account for *noisy* data.

As an alternative to Kriging models and with the aim of using the surrogate model in optimization routines, Cozad et al. [20] developed the ALAMO methodology. In this methodology, the surrogate model is based on a selection of basis functions and the model quality is improved through error maximization sampling. The advantage of the ALAMO approach is the simplicity of the basis functions and the easy availability of derivative information of the surrogate model. However, it is necessary to fit a surrogate model after each sampling iteration of the algorithm.

Eason and Biegler [25] developed a trust region framework for the substitution of computational extensive models in the context of optimization. This framework combines the sampling, fitting of surrogate models and optimization into a single algorithm.

The chosen basis functions for the surrogate model affects the achievable accuracy of the surrogate model to represent the nonlinear response surface. Common basis functions include B-splines [41], Kriging models [15, 25, 59, 84], individual chosen basis function [20], and artificial neural networks [24]. Davis et al. [22] provide an overview of the different methods and compare their performance on 47 challenge functions.

So far, the developed methods focus on the substitution of individual *noisy* or computational expensive unit operations. Problems may then still arise if many recycle streams are present. The approach presented in this chapter develops therefore a framework which uses surrogate models of subprocesses for the optimization of the overall process. This chapter is structured as follows. Section 10.1 introduces a procedure of splitting the overall process into subprocesses and introduces methods for variables selection. Section 10.2 gives two examples, the first for the splitting into subprocesses and the second for the selection of controlled variables.

10.1 Optimization through Separation and Surrogate Modelling

A general nonlinear problem utilizing a commercial flowsheet simulator is defined as

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J(\mathbf{x},\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}(\mathbf{x},\mathbf{d},\mathbf{u}) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x},\mathbf{d},\mathbf{u}) \end{aligned} \tag{10.1}$$

in which $J(\mathbf{x},\mathbf{d},\mathbf{u})$ is the objective function, $\mathbf{g}(\mathbf{x},\mathbf{d},\mathbf{u})$ the equality constraints defined through the steady-state model of the simulator and $\mathbf{h}(\mathbf{x},\mathbf{d},\mathbf{u})$ the inequality constraints which can be imposed on the states $\mathbf{x}$ or inputs $\mathbf{u}$. The inequality constraints are in general linear and not part of the mathematical equations defined in the flowsheet simulator. In addition, disturbance variables $\mathbf{d}$ can be present. If a surrogate model of the whole process should be designed, we would need an increased amount of function evaluations to develop the surrogate model as shown in example 3 of Caballero and Grossmann [15], in which the approach using two surrogate models reduces the number of model evaluations.

10.1.1 Separation into Subprocesses

To circumvent this problem, we can split the equality and inequality constraints in n subprocesses, each with its respective states $\mathbf{x}_i$, inputs $\mathbf{u}_i$, and disturbances $\mathbf{d}_i$. The newly defined optimization problem is then given by

$$\begin{aligned} \min_{\mathbf{x}_c,\mathbf{u}} \quad & J(\mathbf{x}_c,\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}_i(\mathbf{x}_i,\mathbf{d}_i,\mathbf{u}_i) \qquad i \in 1,\ldots,n \\ & \mathbf{0} \geq \mathbf{h}_i(\mathbf{x}_i,\mathbf{d}_i,\mathbf{u}_i) \qquad i \in 1,\ldots,n \end{aligned} \tag{10.2}$$

The overall dimension of the combined vectors of states $\mathbf{x}_c = \begin{bmatrix}\mathbf{x}_1^\mathsf{T} & \mathbf{x}_2^\mathsf{T} & \cdots & \mathbf{x}_n^\mathsf{T}\end{bmatrix}^\mathsf{T}$ is, due to interactions between the subprocesses, higher than $\dim(\mathbf{x})$. The subscript c indicates the combined vector, whereas $\dim(\mathbf{u}_c) = \dim(\mathbf{u})$ and $\dim(\mathbf{d}_c) = \dim(\mathbf{d})$. The subprocesses require, due to their interaction, additional linear equality constraints defined as

$$\mathbf{g}_{i,k} = \mathbf{z}_{i,k} - \mathbf{y}_{i,k} = \mathbf{0} \qquad i = 1,\ldots,n,\ \forall k \neq i \tag{10.3}$$

in which $\mathbf{z}_{i,k}$ corresponds to the input connection variables of submodel k and $\mathbf{y}_{i,k}$ to the output connection variables of submodel i. Hence, the introduction of subprocesses increases concurrently the number of states and equality constraints. However, it allows the combination of rigorous and surrogate modelling and simplifies in general the optimization, as $\dim(\mathbf{x}_i) < \dim(\mathbf{x})$.

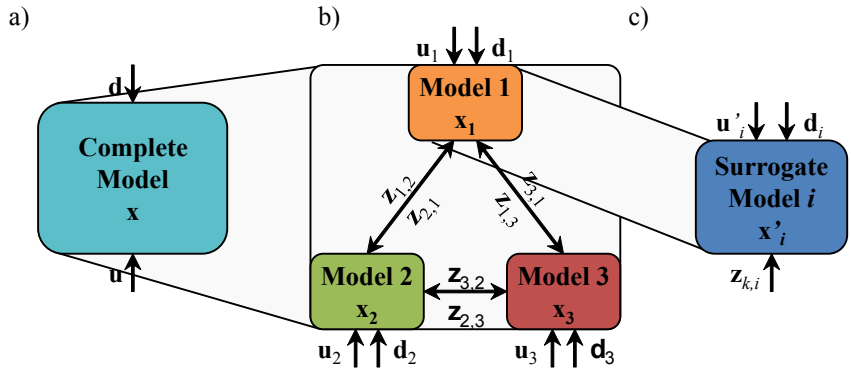


Figure 10.1: a) Complete model with inputs $\mathbf{u}$, states $\mathbf{x}$, and disturbances $\mathbf{d}$, b) its split into 3 subprocesses with the respective inputs $\mathbf{u}_i$, states $\mathbf{x}_i$, and disturbances $\mathbf{d}_i$ and c) the derived surrogate model with inputs $\mathbf{u}'_i$, states $\mathbf{x}'_i$, and disturbances $\mathbf{d}'_i$.

The approach of separation is frequently used in the case of complicated processes, in which only parts are optimized as shown by Araújo and Skogestad [7]. The ammonia synthesis gas loop was in this article split, *e.g.*, from the the synthesis gas preparation and make-up and the refrigeration loop. The approach of Biegler and Hughes [11] can also be seen as a special case of this procedure, as it allows the optimization routine to handle the recycle streams instead of the process simulator and hence introduces additional states and equality constraints. Figure 10.1 as an example shows the splitting of a process (a)) into 3 interacting models (b)). In this case, we would have as additional equality constraints (and as measure of interaction through $\dim(\mathbf{z}_{i,k})$)

$$\begin{aligned} \mathbf{g}_{1,2} &= \mathbf{z}_{1,2} - \mathbf{y}_{1,2} = \mathbf{0} & \mathbf{g}_{2,1} &= \mathbf{z}_{2,1} - \mathbf{y}_{2,1} = \mathbf{0} \\ \mathbf{g}_{1,3} &= \mathbf{z}_{1,3} - \mathbf{y}_{1,3} = \mathbf{0} & \mathbf{g}_{3,1} &= \mathbf{z}_{3,1} - \mathbf{y}_{3,1} = \mathbf{0} \\ \mathbf{g}_{2,3} &= \mathbf{z}_{2,3} - \mathbf{y}_{2,3} = \mathbf{0} & \mathbf{g}_{3,2} &= \mathbf{z}_{3,2} - \mathbf{y}_{3,2} = \mathbf{0} \end{aligned} \tag{10.4}$$

10.1.2 Creation of Surrogate Models

Through the introduction of a low-complexity surrogate model $\mathbf{g}'_i(\mathbf{x}'_i, \mathbf{d}_i, \mathbf{u}'_i)$ for the initial submodel $\mathbf{g}_i(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i)$, we can reduce the number of states $(\dim(\mathbf{x}'_i) \ll \dim(\mathbf{x}_i))$ and hence simplify the problem further. In fact, a surrogate model i only needs as states the interaction states $\mathbf{z}_{k,i}$ and $\mathbf{y}_{i,k}$ as surrogate models can be seen as input-output relationships in which the original states $\mathbf{x}_i$ are not important. Within a surrogate model, the respective inequality constraints are normally treated as linear inequalities, eliminated (never active), or as equalities (always active). We however have to include the original states which have an impact on the cost function in the nonlinear problem (10.2).

This does not require additional flowsheet evaluations. The basis function of the surrogate model is not important for this procedure. In addition, surrogate models allow us to obtain directly derivatives, which reduces the computational expense and noise introduced through numerical differentiation. Furthermore, a change of decision variables $\mathbf{u}_i'$ (satisfying $\dim(\mathbf{u}_i') = \dim(\mathbf{u}_i)$) can be performed simplifying the problem further. However, the selection of the new decision variables has to be performed carefully and the number of decision variables in the surrogate models should be limited due to the exponential dependency of the the number of flowsheet evaluations to the number of independent variables (independent variables to submodel i include $\mathbf{d}_i$, $\mathbf{u}_i$, and $\mathbf{z}_{k,i}$).

Reduction of the Number of Independent Variables and Use of Linear Relationships

There are several heuristic approaches for reducing the number of independent variables. In some cases, the optimal setpoint (CVs) is constant and we may eliminate the variations in this CV from the surrogate model, that is, one degree of freedom u may be eliminated from the surrogate model. This applies if we are controlling an active constraint (CVs = constraint limit) or for an unconstrained problem if $\mathbf{u}_i' = \mathbf{J}_{\mathbf{u}_i}$ (the gradient of J with respect to the degree of freedom $\mathbf{u}_i$). More generally, we may not be able to eliminate a variable, but we can assume that its effect on the output is linear, which gives the same results in terms of simplifying the surrogate model. This can include connection states $\mathbf{z}_{i,k}$ and independent variables $\mathbf{u}_i'$. In addition, plant knowledge can be used for the reduction. For example, if we define a surrogate model for a reaction section and we know that inerts are present, we can define a linear input-output relationship ($\mathbf{y}_{i,out} = \mathbf{z}_{in,i}$) for the inerts in these streams. If, on the other hand, we want to define a surrogate model for the separation section, we can directly say, that for the input coming from the reaction section, the reacting species are depending on each other, *e.g.* through the rate of extent of reaction $\dot{\xi}$ and cannot be treated independently. In the case of a submodel without reaction or phase change we can also directly observe a linear relationship between the input and output in the molar (mass) flowrates $\dot{n}_i$ ($\dot{m}_i$). However, the composition can still play a role for the output temperature and pressure as the thermodynamical calculations are composition dependent. In summary, whether these linear relationships can be applied or not, is defined by the nonlinearity of our flowsheet.

Independent Variable Selection using SOC

One approach to select new independent variables is the application of the concepts of self-optimizing control developed by Skogestad [91]. This concept tries to select controlled variables which minimize the economic loss with respect to disturbances. The remaining controlled variables, with tight control of constrained independent variables, are then selected as

$$\mathbf{c}_i = \mathbf{H}_i \mathbf{y}_i \tag{10.5}$$

in which the selection matrix $\mathbf{H}_i$ can be obtained using the exact local method or the nullspace method [3]. Both methods require a linear model of the process and multiple optimization of the flowsheet. As we would like to define models for optimization, this procedure cannot be implemented directly for our overall model $\mathbf{g}$. We can however implement this procedure for the optimization of subprocesses $\mathbf{g}_i$, where the decision variables $\mathbf{u}_i'$ are selected while constructing the surrogate model. We then choose $\mathbf{u}_i' = \mathbf{c}_i$. This would require new, local cost functions J_i for the respective subprocesses and leads to the following optimization problems

$$\begin{aligned} \min_{\mathbf{x}_i, \mathbf{u}_i} \quad & J_i\left(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i\right) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}_i\left(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i\right) \\ & \mathbf{0} \geq \mathbf{h}_i\left(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i\right) \end{aligned} \tag{10.6}$$

The setpoint for the new decision variables $\mathbf{u}_i'$ is then calculated in the optimization problem (10.2). This approach has furthermore the advantage, that the controlled variables are defined within the surrogate model, as otherwise states which could be the perfect choice for SOC may be omitted in the surrogate models. This approach will be discussed in detail in Chapter 13.

Independent Variable Selection using the Existing Control Structure
Generally, finding the self-optimizing variables requires reoptimization of the system for the expected disturbances, which is actually the end goal for the use of the surrogate models, so they will not be available in most cases, at least not initially. For example, the definition of local cost functions can be in many cases not feasible or introduce additional errors. We may therefore use as alternative the current control strategy of the plant for selecting new independent variables. The idea is that the current control structure is designed to keep its operation close to the optimal in spite of disturbances. Selecting $\mathbf{u}_i'$ as the current controlled variables (CV) for the considered subprocess constrains the sampling domain to the important regions.

10.1.3 Algorithmic Approach

Based on the above concepts of surrogate modelling, process splitting and self-optimizing control, we can define a procedure for solving integrated plants (Algorithm 1). This algorithmic approach can reduce the complexity of integrated flowsheets in the case of recycle streams drastically and allows the subsequent optimization. The final problem (10.2) can then be seen as a combination of sequential-modular and equation oriented [12] operation, which utilizes the advantages of both approaches based on the existing control structure.

Algorithm 1 Surrogate Model Optimzation

1: Define Optimization Problem (10.1).
2: Split of $\mathbf{g}(\mathbf{x},\mathbf{d},\mathbf{u})$ into n $\mathbf{g}_i(\mathbf{x}_i,\mathbf{d}_i,\mathbf{u}_i)$.
3: **for** i in $1:n$ **do**
4: **if** J_i exists **then**
5: Define $\mathbf{u}_i'$ using SOC and Optimization Problem (10.6).
6: **else**
7: Define $\mathbf{u}_i'$ using existing control structure.
8: **end if**
9: Calculate $\mathbf{g}_i'$.
10: **end for**
11: Connect $\mathbf{g}_i'$ and/or $\mathbf{g}_i$.
12: Solve Optimization Problem (10.2).

10.2 Examples and Applications

Two examples will be covered in the following. First, a simplified version of the ammonia synthesis gas loop is used as example for plant separation into three subprocesses. Second, a continuous tank reactor will exemplify the transformation of independent variables.

10.2.1 Ammonia Synthesis Gas Loop

Modern ammonia plants are highly integrated as shown in Chapter 2. The synthesis loop in particular includes several recycle streams in addition to the overall mass recycle. This overall recycle corresponds to up to 75 % of the mass entering the reactor. Solving these type of processes in sequential-modular solvers is difficult and it is next to impossible to use the developed flowsheets for optimization.

However, it is possible to separate the synthesis gas loop into different subprocesses as shown in Figure 10.2. These subprocesses are the synthesis gas makeup, the reaction section and the combined separation/refrigeration section. It is possible for each submodel to define surrogate models. The high amount of connection variables ($z_{i,k}$ = pressure p, enthalpy h or temperature T, 5 molar flows $\dot{n}_i$ (H_2, N_2, NH_3, Ar, and CH_4)) requires however the definition of simple linear relationships, which is for example possible for all flowrates in non-reacting regions of the process on the one hand (*vide supra*). In the reaction section, it is on the other hand possible to utilize the concept of rate of extent of reaction $\dot{\xi}$. In addition, constraints on cooling water usage are usually active reducing the number of independent variables further.

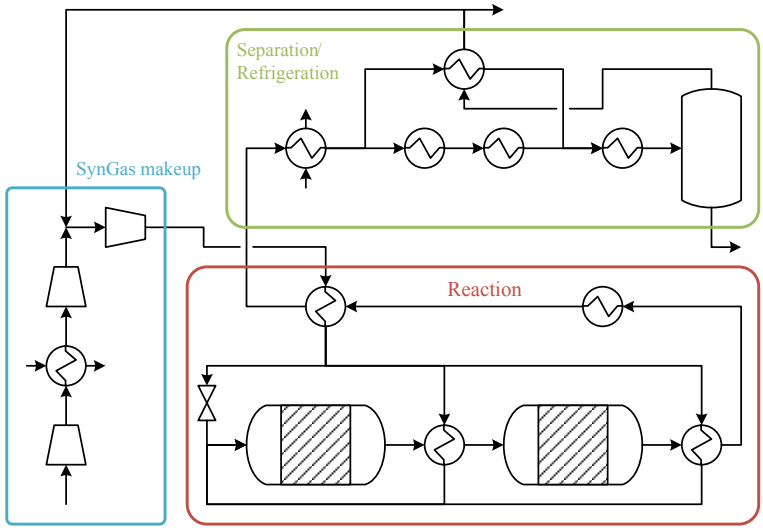


Figure 10.2: The ammonia synthesis gas loop with the three distinctive subprocesses.

10.2.2 Continuous Tank Reactor with Exothermic Reaction

As an example for the application of the change of manipulated variables, consider a continuous tank reactor (see Figure 10.3 with holdup M_R), in which two sequential exothermic reaction take place.

$$\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C} \tag{10.7}$$

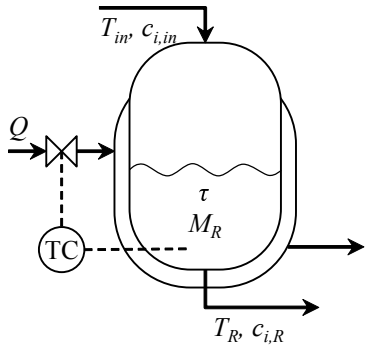


Figure 10.3: Process diagram of a continuous tank reactor.

Table 10.1: Nomenclature of parameters and calculated values.

Variable	Description	Value	Unit
$A_{0,1}$	Arrhenius factor, Reaction 1	5×10^5	
$A_{0,2}$	Arrhenius factor, Reaction 2	5×10^{10}	
$E_{a,1}$	Activation Energy, Reaction 1	45	kJ/mol
$E_{a,2}$	Activation Energy, Reaction 2	105	kJ/mol
R	Universal gas constant	8.314	J/mol/K
τ	Residence time	1800	s
M_R	Reactor hold-up	15	m^3
c_p	Molar heat capacity	10^6	$J/m^3/K$
ΔH_{rx}	Heat of reaction 1	-2×10^4	J/mol A
ΔH_{rx}	Heat of reaction 2	-2×10^5	J/mol B
$c_{A,in}$	Feed concentration A	700	mol/m^3
$c_{B,in}$	Feed concentration B	0	mol/m^3
$c_{C,in}$	Feed concentration C	0	mol/m^3
T_{in}	Feed temperature	20	°C

The aim is in this reaction to produce chemical B, and hence, we would like to maximize the molar fraction x_B. This reactor can be part of a bigger submodel, for which we would like to design a surrogate model. The parameters for the investigated system are given in Table 10.1 As the residence time τ and feed composition $c_{i,in}$ is given, this part of the submodel has one manipulated variable, $u = Q$. The model equations are given by

$$0 = \frac{1}{\tau}(c_{A,in} - c_{A,R}) - r_1 \tag{10.8}$$

$$0 = \frac{1}{\tau}(c_{B,in} - c_{B,R}) + r_1 - r_2 \tag{10.9}$$

$$0 = \frac{1}{\tau}(c_{C,in} - c_{C,R}) + r_2 \tag{10.10}$$

$$0 = \frac{Q}{M_R} + \frac{c_p}{\tau}(T_{in} - T_R) - r_1 \Delta H_{r,1} - r_2 \Delta H_{r,2} \tag{10.11}$$

with

$$r_1 = A_{0,1} \exp\left(-\frac{E_{a,1}}{RT_R}\right) c_{A,R} \tag{10.12}$$

$$r_2 = A_{0,2} \exp\left(-\frac{E_{a,2}}{RT_R}\right) c_{B,R} \tag{10.13}$$

being the reaction rates for reaction 1 and 2.

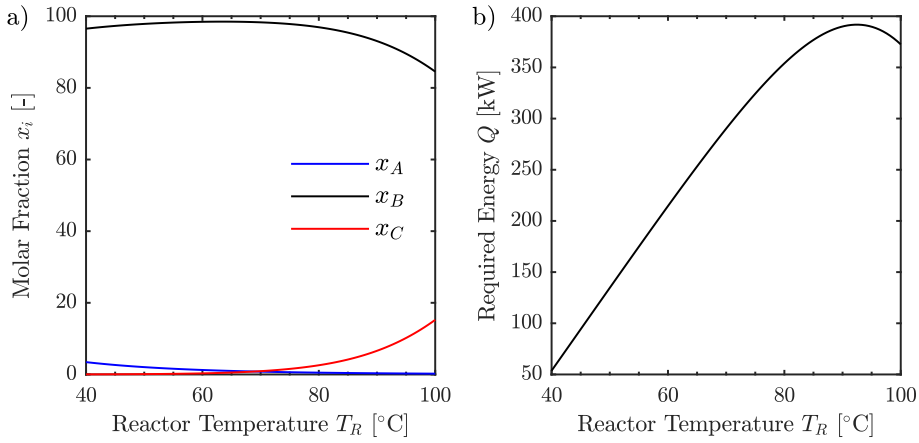


Figure 10.4: a) Molar fraction of chemicals A, B, and C and b) required heating energy Q as a function of the reactor temperature T_R.

Considering Eq. (10.11), it is possible to either define Q (u) or T_R (u') as independent variable while the other is dependent. This allows to impose constraints, *e.g.* maximum heating energy, on both the reactor temperature and the energy. If we would like to define a surrogate model with Q as independent variable, it is possible to end up covering operation regions that are never encountered in practice, or at least are far from the optimum that we want to find. Using instead T_R as independent variable corresponds to a common control structure for this system.

Figure 10.4 shows the molar fraction of chemicals A, B, and C as well the the reactor temperature for this system. For this particular system, we can see that the required energy has a maximum at a reactor temperature above the interesting range. This maximum results on one hand in two solutions for each given energy. On the other hand, certain values for the energy Q do not give a solution. Hence, the variable transformation from Q to T_R simplifies the sampling domain definition and prevents sampling of undesired operation conditions.

10.3 Conclusion

Based on the concepts of process splitting, surrogate modelling, and independent variable transformation, we propose an algorithmic approach to solve flowsheets of integrated chemical plants. Furthermore, this approach allows the combination of rigorous and surrogate models in cases, where detailed models are easily available and the number of independent variables exceeds for parts of the process a practical value.

Two examples of subroutines of the approach are presented. The first example uses the synthesis loop of the ammonia process as an example of splitting the process into subprocesses. The subprocesses are simpler to solve due to a reduced number of recycle streams and can be sampled for the generation of surogate models. The second example demonstrates the utilization of the current control structure for selecting independent variables prior to surrogate model creation. This results in a different sampling domain and may allow simpler solutions of the subprocess.

The following chapters will explain different steps in surrogate model generation in more detail. Chapters 11 and 12 look into the reduction of independent variables. This is especially important for the proposed procedure as due to the connection variables, we generally encounter a large number of independent variables. Chapter 13 extends the idea of using concepts from self-optimizing variables and exemplifies the advantages associated with it. Chapter 14 introduces a novel sampling procedure to avoid both oversampling and fitting of a surrogate model during the sampling procedure.

Chapter 11

Variable Reduction using Partial Least Squares Regression

Chapter 10 proposed a methodology for the optimization of integrated process. This method involves splitting the complete flowsheets into several submodels, fit surrogate models to these submodels and subsequently combining the surrogate models into a system of nonlinear equations which can be optimized. However, due to the connection variables, the number of independent variables (n_u) is generally quite high. This can lead to problems caused by the *dimensional curse* of surrogate models if regular grids are used. The exponential dependency of the surrogate model fitting with n_u independent variables for a 2-point regular grid is given in Eq. (11.1).

$$n_{RG} = 2^{n_u} \tag{11.1}$$

Hence, it is important to keep n_u small, preferably less than four [40].

One way to reduce n_u is given by a factorial sampling plan as developed by Morris [73]. This approach utilizes preliminary simulations, in which it is decided which of the independent variables have an effect on the dependent variables and whether this effect is linear. The insignificant variables may be omitted in the latter design of experiments, whereas linear variables require less sampling points. Additionally, active constraints may be identified and cancelled out to reduce n_u.

Another possibility is to introduce new independent variables (latent variables) $\mathbf{u}'$, which can be, among others, derived *via* partial least squares (PLS) regression. PLS regression is a linear regression tool in which the predicted and observable variables are projected into a new space through the introduction of components. It was developed by Wold et al. [104] to solve the multivariate calibration problem in the case of chemometrics. In

this problem, the number of sampling points is less than the number of independent variables, *i.e.* the number of varied concentrations is smaller than the number of measured frequencies and an optimal combination of measurements for concentration regression has to be found. Similarly, it was applied in the analysis of genomic data [14].

Based on the mentioned previous applications of PLS regression, it seems to be a reasonable tool for the reduction of the number of independent variables. Through the incorporation of PLS regression, a 3-step procedure for surrogate model fitting is developed. The principles of PLS regression are explained in Section 11.1. This method incorporates process knowledge and will be explained in Section 11.2. It is subsequently applied in Section 11.3 to a pipe model, which can have an arbitrary number of independent variables and to the reaction section of a ammonia synthesis loop in Section 11.4 which serves as a case study for integrated chemical processes. It has to be noted, that the procedure itself is not limited to partial least squares regression, but can also utilize for example dimensionless numbers as well for the reduction of independent variables.

11.1 Background - Partial Least Squares Regression

In many applications, the number of independent variables n_u, *e.g.* spectroscopy frequencies and genes, exceed the number of samples n_p, which results in problems with classical multivariate regression models. Furthermore, problems may arise in the multivariate regression, if independent variables are noisy or strongly correlated. To this end, Wold et al. [104] developed partial least squares regression (PLSR). A detailed review of PLSR can be found in [14] and [105]. The former explains various algorithms for the calculation of the latent variables.

The aim of PLSR is a variable reduction in the independent variables resulting in new latent variables. Hence, it is also called Projection to Latent Structures [105]. PLSR is similar to Principal Component Regression (PCR) [68]. It does, however, in contrast to PCR, consider in the calculation of the latent variables their impact on the dependent variables. The variable reduction is given through the transformation of the original independent variable space $\mathbf{U} \in \mathbb{R}^{n_p \times n_u}$ into a space of $n_{u'}$ latent variables $\mathbf{U}' \in \mathbb{R}^{n_p \times n_{u'}}$ through the loads $\mathbf{P} \in \mathbb{R}^{n_u \times n_{u'}}$. Similarly, the dependent variable space $\mathbf{Y} \in \mathbb{R}^{n_p \times n_y}$ is transformed into a space of latent variables $\mathbf{Y}' \in \mathbb{R}^{n_p \times n_{u'}}$ through the loads $\mathbf{Q} \in \mathbb{R}^{n_y \times n_{u'}}$. This is mathematically given by

$$\mathbf{U} = \mathbf{U}'\mathbf{P}^\mathsf{T} + \mathbf{E} \tag{11.2}$$

$$\mathbf{Y} = \mathbf{Y}'\mathbf{Q}^\mathsf{T} + \mathbf{F} \tag{11.3}$$

in which $\mathbf{E}$ and $\mathbf{F}$ are error matrices. From Eq. (11.2), we can derive the following equation for a variable transformation through neglecting the error term $\mathbf{E}$.

$$\mathbf{U}' = \mathbf{U}\mathbf{W} \tag{11.4}$$

with the weight matrix

$$\mathbf{W} = \left(\mathbf{P}^{\mathsf{T}}\right)^{+} \tag{11.5}$$

Here, $\left(\mathbf{P}^{\mathsf{T}}\right)^{+}$ corresponds to the right Moore-Penrose inverse as described in [82]. The load matrices $\mathbf{P}$ and $\mathbf{Q}$ are hereby calculated to maximize the covariance between $\mathbf{U}'$ and $\mathbf{Y}'$.

Several algorithms exist for computing $\mathbf{W}$. An overview is given by Boulesteix and Strimmer [14]. In this thesis, the *Statistically Inspired Modification of PLS* algorithm (SIMPLS) [23] is used. It obtains the weights for each component $i = 1, \ldots, n_{u'}$ sequentially according to

$$\mathbf{w}_i = \arg\max_{\mathbf{w}} \mathbf{w}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}\mathbf{Y}\mathbf{Y}^{\mathsf{T}}\mathbf{U}\mathbf{w} \tag{11.6}$$

with the following constraints

$$\begin{aligned} \mathbf{w}_i^{\mathsf{T}}\mathbf{w}_i &= 1 \\ \mathbf{w}_i^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}\mathbf{U}\mathbf{w}_j &= 0 \qquad \forall j = 1, \ldots, i-1 \end{aligned} \tag{11.7}$$

$\mathbf{w}_i$ denotes the columns of the weight matrix $\mathbf{W}$. It gives the coefficients of the original variables in the calculation of the new latent variables. The first constraint normalizes the weights, whereas the second constraint results in orthogonality of the latent variables.

Depending on the implemented algorithm (*e.g. plsregress* in MATLAB and *simpls* in R [14]), $\mathbf{u}_i'$ corresponding to a column of $\mathbf{U}'$ may have a length of 1, *i.e.*

$$\mathbf{u}_i'^{\mathsf{T}}\mathbf{u}_i' = 1 \tag{11.8}$$

This is contrary to the constraints (11.7). The proposed method however utilizes weights $\mathbf{w}_i$ with unit length. Hence, it requires the transformation of the weights $\mathbf{w}_i$ to have unit length.

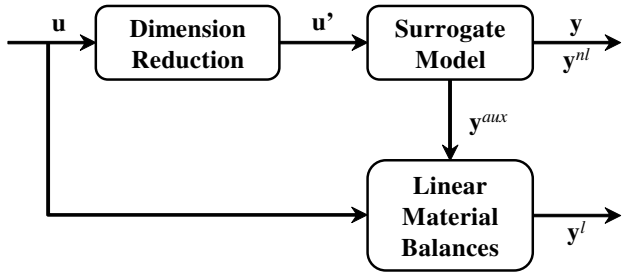


Figure 11.1: Proposed new model structure including the surrogate model.

11.2 Procedure for Surrogate Model Fitting with Dimension Reduction

The overall procedure to reduce n_u consists of in total three steps

1. introduction of linear material balance relationships;
2. independent variable dimension reduction through PLS regression;
3. fitting of the surrogate model to the new independent variables $\mathbf{u}'$.

The content of each of the three steps will be explained in the following subsections. As a result of this procedure, a new model structure is introduced. This structure is visualized in Figure 11.1. This methodology requires the initial sampling of a certain number of points n_p to perform the PLS regression resulting in a sampled space given by $\mathbf{U}$. We propose to incorporate the corner points of a regular grid of the sampling space in order to not extrapolate data within the investigated sampling space and add additional points through Latin hypercube sampling or orthogonal sampling to guarantee a proper distribution of the points. This is however limited in the case of a large number of independent variables u as in the case of 20 independent variables, Eq. (11.1) already gives $n_{RG} = 1048576$ points. In this case, pure Latin hypercube sampling may be sufficient.

11.2.1 Definition of Linear Relationships

Linear input-output relationships can be always defined for mass balances and in certain cases for the energy and force balances. This can be reasoned by the knowledge of the flowsheet topology in the case of production optimization. However, the application of linear relationships may require the introduction of auxiliary variables $\mathbf{y}_{aux}$. In the case of a reaction within the submodel, the rate of extent of reaction $\dot{\xi}$ in combination with the stoichometric factors ν_i allows the reduction of the number of surrogate models to

be fitted. In the case of multiple chemical reactions, it is as well possible to calculate several rate of extent of reactions $\dot{\xi}$. If, on the other hand, a separation takes place in the submodel or a split is present, the mass balances can be introduced *via* separation coefficients **p**. The introduction of linear relationships hence reduces the number of surrogate models which have to be fitted.

In addition, the introduction of linear mass balances results in mass consistency. If this step would not be included, the combination of surrogate models could lead to creation or removal of mass due to model inaccuracy, rendering their application doubtful. Hence, the new model structure can be considered as grey-box modelling through the combination of process knowledge and surrogate models.

11.2.2 Dimension Reduction

As mentioned, the application of PLS regression yields as a result linear combinations of the initial independent variables, which represent the nonlinear output variables $\mathbf{y}_{nl}$ and/or $\mathbf{y}_{aux}$ for the given sampled data best through the introduction of weights **W**. It is important to mention, that a PLS regression should be performed for each of the output variables $\mathbf{y}_{nl}$ and $\mathbf{y}_{aux}$. This corresponds to multiple univariate response cases. Otherwise, components are chosen with a trade-off for fitting the dependent variables to the independent variables. This results in a individual weight $\mathbf{W}_k$ for each dependent variable y_k. The new independent variables for each dependent variable k are then given by

$$\mathbf{u}'_k = \mathbf{u}\mathbf{W}_k \tag{11.9}$$

An additional advantage of the application of PLS regression is that it gives an overview about the influence of the independent variables **u** on the derived nonlinear output values $\mathbf{y}_{nl}$ and $\mathbf{y}_{aux}$. This can be utilized for the addition of points to the sampling domain in the relevant direction, but will not be elaborated further in this chapter. The linear combinations of the components defined are hereby independent of the total number of components. This means, that the linear combination of the first component will be the same if $n_{u'} = 1$ or $n_{u'} = n_u$ as it is shown as well in Section 11.1. Therefore, it is useful to perform the PLS regression directly for $n_{u'} = n_u$ components and only use the first j components for the definition of the surrogate model in the subsequent fitting. Before applying PLS regression, it is additional advantages to perform variable transformations for the independent variables. If it is for example known, that the partial pressure of components or the total flow play a crucial role, it is useful to redefine the matrix for the sampled space **U** in terms of total flow $\dot{n}$ and mole fractions $\mathbf{x}_i$ or partial pressures $\mathbf{p}_i$. This will be further elaborated in Chapter 12.

The SIMPLS algorithm used by MATLAB for PLS regression is strongly depending on the scaling of the variables. Hence, it is crucial to scale the sampled space appropriately before performing PLS regression. If the scaling is not performed properly, the first component will point towards the sampling space instead of capturing the true component. In the following, the standard score will be applied for scaling the sample space $\mathbf{U}$ which is defined as

$$\mathbf{U}_{scaled} = (\mathbf{U} - \mu_{\mathbf{U}}) \circ \sigma_{\mathbf{U}}^{-1} \tag{11.10}$$

where $\mu_{\mathbf{U}}$ is the mean value and $\sigma_{\mathbf{U}}$ the standard deviation in the matrix $\mathbf{U}$ with respect to each of the independent variables $\mathbf{u}$. Using the standard score, we scale the input matrix $\mathbf{U}$ in way that we assume the variance of each independent variable is equal. However, in cases where we would like to preserve the changes in the independent variables, the scaled matrix $\mathbf{U}_{scaled}$ can be further adjusted using a scaling matrix $\mathbf{S}_{\mathbf{U}}$, for example, corresponding to the percentage change in the sampling space.

The scaling is then required in the model structure shown in Figure 11.1 when the surrogate model is combined with the other surrogate models.

11.2.3 Surrogate Model Fitting

The surrogate models are fitted to the new independent variables $\mathbf{u}'$ defined as linear combinations of the original independent variables $\mathbf{u}$ corresponding to

$$\mathbf{g}' : \mathbf{u}' \mapsto \mathbf{y}^{surr} \tag{11.11}$$

with $\mathbf{y}^{surr} = \begin{bmatrix} \mathbf{y}_{nl}^{\mathsf{T}} & \mathbf{y}_{aux}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$. The fitting of the surrogate model is an iterative procedure in which the number of components, $n_{u'}$ is increased until a fitting criteria is fulfilled. Alternatively, the explained variance per component in the response ($\mathbf{y}_{nl}$ and/or $\mathbf{y}_{aux}$) can be utilized as a starting point. The type of surrogate model is not important for this procedure. For example, artificial neural networks, splines, Kriging models, or polynomials can be applied. However, due to the introduction of new independent variables, it is necessary that the surrogate model basis functions do not require a regular grid as a regular grid will not exist after variable transformation through PLS.

11.2.4 Algorithmic Approach

Algorithm 2 summarizes the steps outlined above.

Algorithm 2 Procedure for independent variable reduction.

1: Define sampling domain **U** of the problem.
2: Sample training and validation space.
3: Define linear relationships if possible.
4: **for** $k = 1$ to $n_{y,nl} + n_{y,aux}$ **do**
5: Perform PLS regression with $n_{u',k} = n_u$.
6: **while** $\varepsilon_j > threshold$ **do**
7: Fit surrogate model $\mathbf{g}'$ to $n_{u',k} = j$.
8: $\mathbf{y}_{sm,k} = \mathbf{g}'(\mathbf{u}_{val}, n_{u',j})$.
9: $\varepsilon_k = \frac{|\mathbf{y}_{val,k} - \mathbf{y}_{sm,k}|}{\mathbf{y}_{val,k}}$.
10: $j = j + 1$.
11: **end while**
12: **end for**

11.3 Example 1 - Simple Pipe Model

A pipe model is used as a proof of concept model. The model gives the pressure drop over a pipe as a function of the independent variables inlet pressure p_{in}, temperature T_{in}, and component molar flows $\dot{n}_{i,in}$. The total number of independent variables n_u is hence given by $n_u = 2 + n_{gas}$ in which n_{gas} is the number of chemicals in the gas stream. This value can be varied to increase the number of independent variables in a simple manner.

11.3.1 Model

The model itself consists of an isothermal pressure drop given in Eq. (11.12)

$$p_{in}^2 - p_{out}^2 = 4f\frac{L}{D}\frac{RT_{in}\overline{M}}{A^2}\dot{n}_{tot,in}^2 \tag{11.12}$$

Based on step 1 in the procedure, we can introduce as linear balances the constant temperature assumption and the mass balances

$$T_{in} = T_{out} \tag{11.13}$$

$$\dot{n}_{i,in} = \dot{n}_{i,out} \quad \text{for } i = 1...n_{gas} \tag{11.14}$$

This leaves as a nonlinear relationship the calculation of the outlet pressure. Hence, one surrogate model has to be defined. Simulations with 3, 5, and 8 chemical components are performed to demonstrate the procedure. The sampled space is given by a 2-point regular grid with an additional 100 (1000 and 5000 respectively for 5 and 8 chemicals) points defined as a Latin hypercube. This corresponds in each case to about 2.5 points in a regular grid. The flowrates are varied with ± 20 % around the nominal point, the inlet

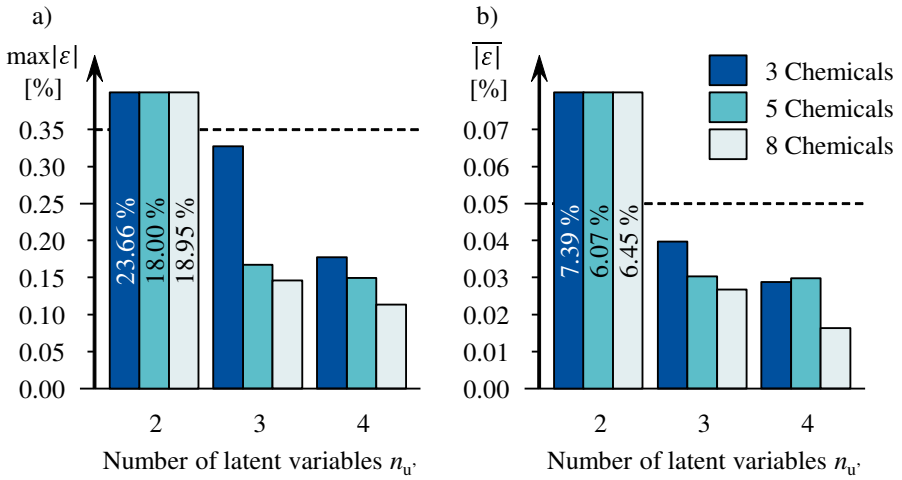


Figure 11.2: a) Maximum and b) mean relative error for the surrogate model of the outlet pressure p_{out} as a function of the number of PLS components $n_{u'}$ in the simple pipe model for a varying number of independent variables.

temperature by ±5 °C, the inlet pressure by ±3 bar. As outlined in Section 11.2, it is possible to perform a variable transformation before applying PLSR. As the total flow rate $\dot{n}_{tot,in}$ and the molar fractions x_i are used in Eq. (11.12), it is reasonable to use these values as well in the application of PLSR. After performing PLS regression, a 1-layer cascade-forward neural network with 5 hidden neurons was fitted using the new independent variables defined *via* PLS regression and the performance of the surrogate model was evaluated with 10^4 points sampled as a Latin hypercube with the same bounds as in the sampling space. The advantage of neural networks is the simple implementation for multivariable regression within MATLAB. Alternative approach, like Kriging models, would require the implementation of the fitting in MATLAB.

11.3.2 Results of the Reduction in Independent Variables

From Eq. (11.12), we can directly see that four independent variables, p_{in}, T_{in}, $\overline{M}$, and $\dot{n}$, are sufficient for the full characterization of the system and it is not necessary to know the exact composition of our gas stream as long as we now the average molar mass $\overline{M}$. As the PLS components are always taking into account the previous, unchanged linear combinations, it has to be noted, that a similar performance cannot be expected.

A PLS regression with 2, 3, and 4 latent variables gives the results in Figure 11.2. It can be seen that the number of variable reduction through PLS allows as little as 3 inde-

pendent latent variables. Increasing the number of latent variables to 4 only marginally improves the performance of the surrogate model fitting. This is confirmed by the explained variance through PLS regression for the response variable p_{out}; from 2 to 3 components, it is increased from 77.71 % to 99.56 % whereas the increase to 4 components only has an influence on the explained variance in the predictor variable matrix U_{scaled}. Analogous results can be found in the case of 5 and 8 chemicals. The increased accuracy for 5 and 8 chemicals with a similar number of components defined *via* PLS regression is given by the increased number of points the surrogate model is fitted to, as the regular grid for the initial independent variables u is exponentially increasing with the number of independent variables as shown in Eq. (11.1). Increasing the sampling space in the case of 3 chemicals to the same number as points as in the case of 8 chemicals results in similar relative errors, confirming this reasoning.

11.4 Example 2 - Reaction Section of the Ammonia Synthesis Loop

The reaction section of an ammonia synthesis loop is used as second case study. The used submodel is elaborated in Appendix B. The submodel consists of 10 independent variables, the inlet pressure p_{in}, temperature T_{in}, and mole flows $\dot{n}_{i,in}$ as well as the outlet temperature of heat exchanger 4 $T_{HEx4,out}$ and the split ratios to the valve n_{Val} and heat exchanger 3 n_{HEx3}.

The flowrates $(H_2, N_2, NH_3, Ar, CH_4)$ are varied with $-\begin{bmatrix}12.5 & 15 & 50 & 40 & 40\end{bmatrix}$ % and $+\begin{bmatrix}12.5 & 15 & 100 & 50 & 50\end{bmatrix}$ % around the nominal point, the outlet temperature of heat exchanger 4 by ± 10 °C, the inlet pressure of the system by ± 6 bar, the inlet temperature of the system by ± 20 %, and the split ratios by ± 3 and ± 10 percentage points respectively.

The sampled domain is given by a 2-point regular grid and 5000 additional points defined as a Latin hypercube to improve the fitting of nonlinearities in the system. The fitted surrogate models are 3-layer cascade-forward neural networks with 2, 5, and 5 hidden neurons in the layers respectively. It has to be mentioned, that the neural network structure was not optimized with respect to the different dependent variables $\mathbf{y}^{surr}$. In addition, the sampling space was chosen too small for the fitting of a nonlinear model to a regular grid as it corresponds to 2.39 points for each independent variable. The validation space was given by a Latin hypercube of 10^4 points.

11.4.1 Results of the Reduction in Independent Variables

In step 1 of the proposed procedure, linear relationships for the mass balances are introduces using the rate of extent of reaction $\dot{\xi}$ as

$$\dot{n}_{i,out} = \dot{n}_{i,in} + \nu_i \dot{\xi} \tag{11.15}$$

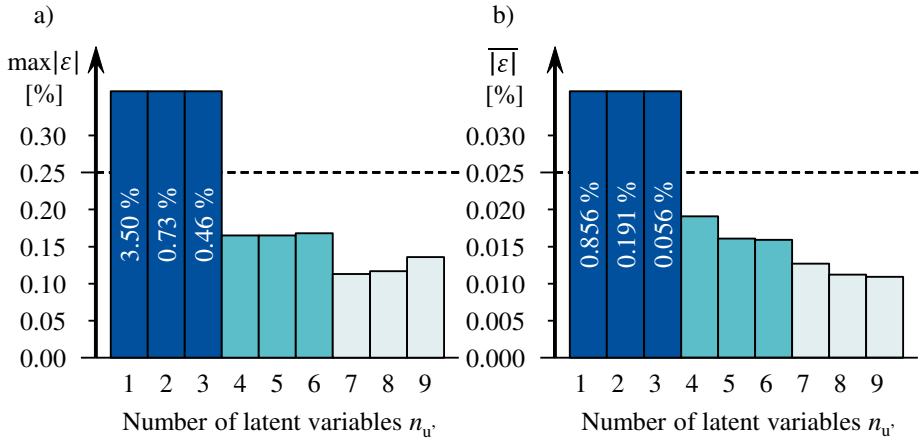


Figure 11.3: a) Maximum and b) mean relative error for the surrogate model of the outlet pressure p_{out} as a function of the number of PLS components $n_{u'}$ for the reaction section of the ammonia synthesis loop.

This leaves nonlinear relationships for p_{out} and T_{out} ($\mathbf{y}_{nl}$), as well as $\dot{\xi}$ (y_{aux}) as dependent variables $\mathbf{y}^{surr}$. Hence, 3 surrogate models have to be fitted in total. Compared to the pipe model, it is this time not possible to define the minimum number of latent variables ($n_{u',min}$) necessary to fit a surrogate model to accurately predict the outlet pressure p_{out}, the outlet temperature T_{out}, and the rate of extent of reaction $\dot{\xi}$. In this situation, it is useful to start at a minimum value for the number of components of $n_{u'} = 5$ and continue in a positive or negative reaction, depending on the fit of the surrogate model. From experience it is expected, that it can be beneficial to describe the problem in terms of a total flow $\dot{n}_{,in}$ and mole fractions $x_{i,in}$ for PLS regression instead of using the mole flows $\dot{n}_{i,in}$. In order to fulfill that the numbers of independent variables remain the same, one mole fraction has to be left out, in this case the mole fraction of hydrogen as this is the highest mole fraction within the system.

The results for the outlet pressure p_{out} can be found in Figure 11.3. From this Figure, we see that the outlet pressure of the system can be accurately described by four or more components obtained *via* PLS regression. In absolute values, the maximum and mean error for four components are given by 0.2 bar and 0.02 bar respectively at a nominal outlet pressure of 129.78 bar. Here, it is interesting to note that the explained variance in the response p_{out} is increasing from one to four components from 96.9 % to 99.94 %, which corresponds to the improved fit of the surrogate model shown in Figure 11.3.

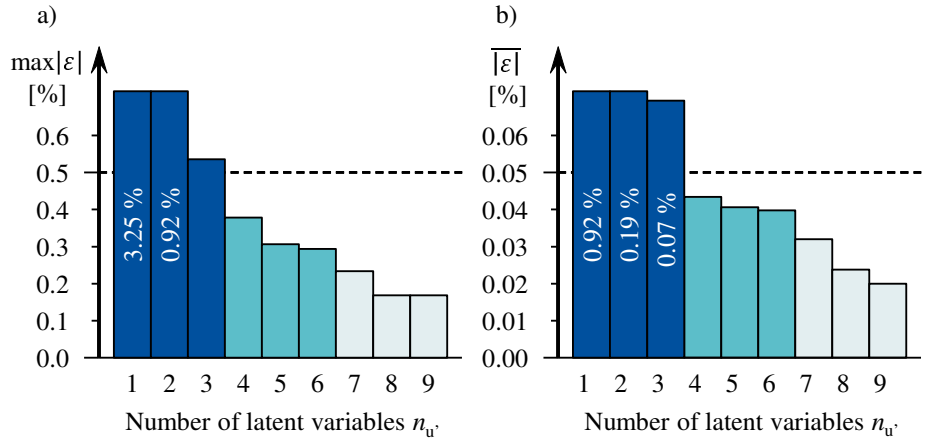


Figure 11.4: a) Maximum and b) mean relative error for the surrogate model of the outlet temperature T_{out} as a function of the number of PLS components $n_{u'}$ for the reaction section of the ammonia synthesis loop.

Similar to the outlet pressure p_{out}, the outlet temperature T_{out} can be adequately described with four or more PLS component as shown in Figure 11.4. In general, the maximum and mean relative error is higher than in the case of the outlet pressure. However, the maximum and mean error correspond to only 0.20 °C and 0.02 °C respectively. Analogous to p_{out}, a drastic improvement can be found by increasing the number of PLS components from 1 to 4. The improvement in the explained variance in the response T_{out} is increasing in these steps as well from 99.83 % to 99.99 % showing that the explained variance can be used for analyzing results, but not for the prediction of the accuracy of the model fit. Otherwise, one would conclude that one component would be sufficient.

Unlike the outlet pressure and temperature, the rate of extent of reaction $\dot{\xi}$ does not result in a similar good fitting as it can be seen in Figure 11.5. This can be explained by the influence of all independent variables in the first four components defined *via* PLS indicating the difficulty to find linear combinations. This is also visible in the increase in the explained variance in the response $\dot{\xi}$ from 82.01 % with $n_{u'} = 1$ to 97.89 % with $n_{u'} = 5$. This finding correlates with the improve of the fit as it was in the case of the pressure and temperature. The maximum and mean relative error using 5 PLS components corresponds hereby to an error of 7.72 mol/s and 0.86 mol/s respectively whereas the nominal rate of extent of reaction is given by $\dot{\xi} = 413.4$ mol/s. Despite the relatively high error in these calculations, it is possible to apply the rate of extent of reaction surrogate model with 5 PLS components into the procedure described in Chapter 10.

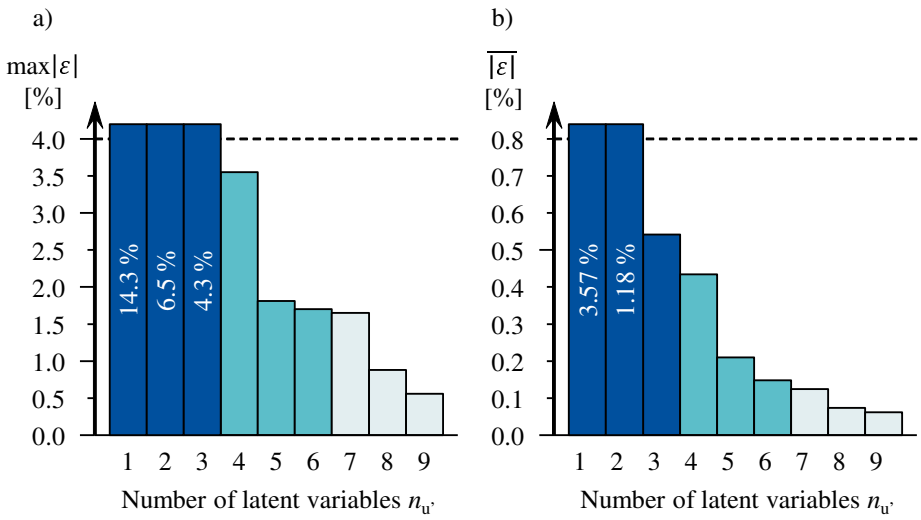


Figure 11.5: a) Maximum and b) mean relative error for the surrogate model of the rate of extent of reaction $\dot{\xi}$ as a function of the number of PLS components $n_{u'}$ for the reaction section of the ammonia synthesis loop.

11.5 Conclusion

The developed three-step procedure was applied to two examples, a pipe model and the reaction section of the ammonia synthesis loop. In both cases, it was possible to obtain surrogate models with high accuracy considering the reduction in the variable space. Incorporation of the surrogate model into a flowsheet consisting of a synthesis-gas make-up section, the reaction section, and a separation section results in a maximum relative error of 0.1 % in all streams.

Chapter 12

Preprocessing of Sampling Data for Partial Least Squares Regression

In the 3-step procedure developed in Chapter 11, the definition of the independent variables **u** was arbitrarily chosen. In both the pipe (Section 11.3) and the reaction section (Section 11.4) of the ammonia synthesis loop case study, the total flow $\dot{n}_{tot,in}$ and the molar fractions $x_{i,in}$ were used for performing partial least squares regression and surrogate model fitting. To this point, process knowledge was not incorporated into the definition of the sampling domain and preprocessing of the sampled data was not investigated. This can potentially lead to improved fit of the surrogate model. Hence, the aim of this chapter is to investigate the influence of preprocessing of sampled data as well as dependency incorporation into the sampling domain on the resulting fit of the surrogate model.

The chapter itself is structured as follows. Section 12.1 explains the different preprocessing and sampling domain definition tasks. Section 12.2 applies the tasks to the reactor section case study whereas section 12.3 summarizes the results and gives recommendation for the preprocessing and domain definition in surrogate model definition.

12.1 Investigation of Sampling Space Definition for Model Fitting

12.1.1 Preprocessing of the Data

The composition in inlet streams is generally defined using an extensive property like the molar or the mass flow. However, chemical reaction kinetics and flash calculations in process simulators are generally given by the intensive variables mole fractions x_i or partial pressure p_i. The utilization of intensive variables will influence the weights defined *via* PLS regression and may hence influence the surrogate model fit. Therefore,

the application of intensive variables will be investigated.

A second question arising is the impact of using differences in pressure and temperature instead of outlet pressure and temperature. The outlet temperature and pressure is general depending on the inlet pressure and temperature respectively. This results in the first component being almost exclusively the inlet pressure or temperature. Through taking the difference of the outlet and inlet condition, it is possible to remove this dependency. This is similar to the extent of reaction as the difference between inlet and outlet molar flow. Hence, the required number of components will be compared between using the outlet variables as dependent variables or their difference.

12.1.2 Introduction of Dependencies of the Independent Variables.

Normally, the sampled space **U** is obtained using upper and lower bounds on each of the independent variables **u** independently. If we consider an inlet stream to a section, the molar flows $\dot{n}_{i,in}$ are generally not independent from each other, as the aim is to keep them close to stoichiometry. This is achieved in the upstream subprocesses through control. The incorporation of a dependency between inlet molar flows reduces the variation in the ratio of the variables as they are not varied independently any more. This can have an effect on independent variables which are depending on ratios and will be hence investigated.

12.2 Case Study - Reaction Section of an Ammonia Process

The influence of preprocessing of the data as well as the sampling domain definition will be analysed using the reaction section of a simplified ammonia synthesis loop as explained in Appendix B with a 5 chemicals feed (hydrogen, nitrogen, and ammonia, as well as argon and methane) corresponding to the reacting and inert chemicals.

12.2.1 Description of the Sampling Domain

The sampled space is given by a two-point regular grid and 5000 points were sampled in addition using Latin hypercube sampling. The flowrates are varied with ± 20 % around the nominal point, the temperatures (both inlet and outlet of heat exchanger 4) by ± 10 °C, the inlet pressure by ± 6 bar, and the split ratios by ± 5 (Valve) and ± 10 percentage points (heat exchanger 3) respectively. The sampling domain is hence different compared to Chapter 11 and the results cannot be compared directly.

The fitted neural network is a 3-layer cascade forward neural network with a layer size of 5, 5, and 5 hidden neurons respectively. The same sampling space was used for all analysis and the resulting surrogate models were validated with 10^4 points randomly

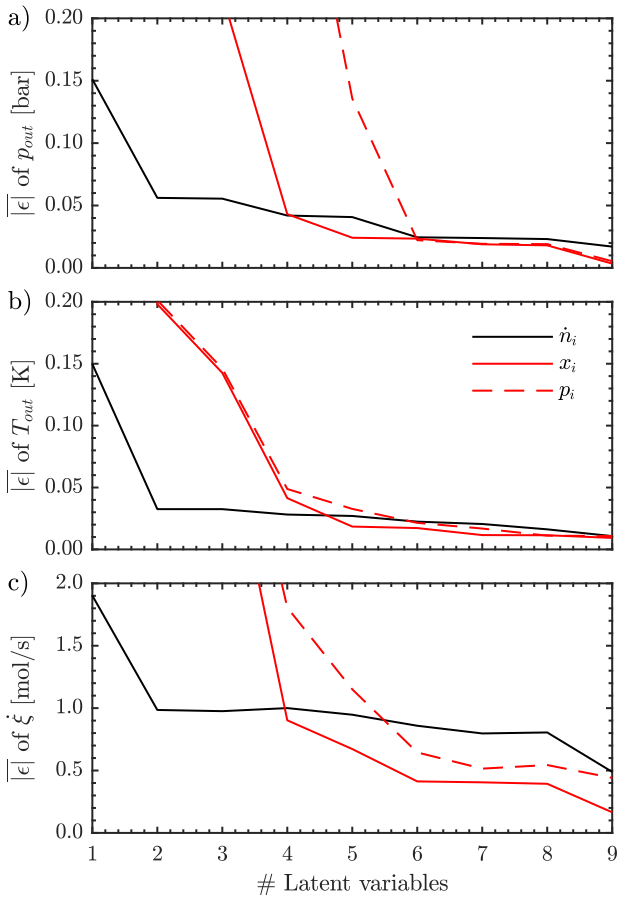


Figure 12.1: Comparison of the preprocessing of the independent variables on the mean absolute error of a) the outlet pressure p_{out}, b) the outlet temperature T_{out}, and c) the extent of reaction ξ as a function of the number of latent variables $n_{u'}$. The independent variables are in molar flow $\dot{n}_i$, mole fraction x_i, and partial pressure p_i.

sampled. The new domain definition in the case of the dependency analysis required newly sampled points.

12.2.2 Preprocessing of the Independent Variables

As described before, the independent variables were represented by the extensive variable molar flow $\dot{n}_i$ as well as the intensive variables mole fraction x_i and partial pressure

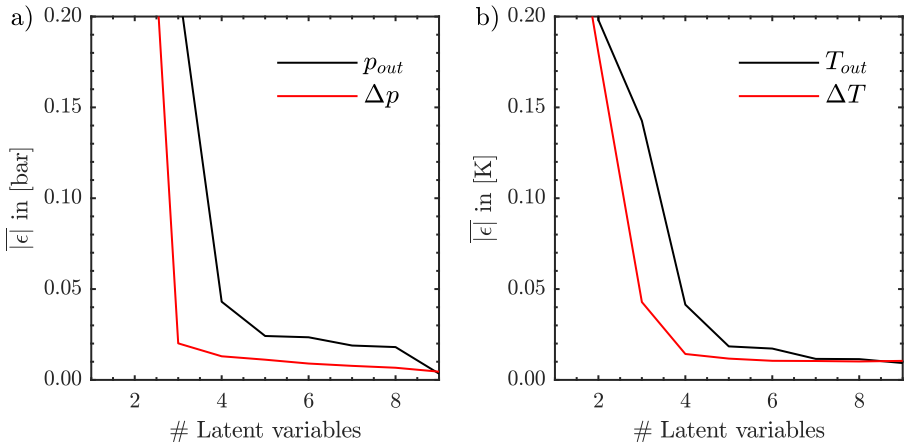


Figure 12.2: Comparison of the mean absolute error of defining the independent variables a) outlet pressure p_{out} and b) outlet temperature T_{out} in pressure drop Δp and temperature change ΔT as function of the number of latent variables $n_{u'}$.

p_i. In the case of mole fractions, the largest mole fraction, x_{H_2}, was left out to account for the imposed total sum of mole fractions. The results of the analysis can be found in Figure 12.1. From this Figure we see that it is best to use extensive variables $\dot{n}_{i,in}$ for three or fewer latent variables, but for more, especially in the case of the extent of reaction, better fit can be achieved using intensive variables. The worse fit of utilizing the intensive variables for less than four latent variables is caused by a smaller explained variance of the first latent variables in the dependent variables. In the case of using the molar flow as independent variable, the first latent variable already explains 99.87 % of the variance in the outlet pressure p_{out}, 99.96 % of the variance in the outlet temperature T_{out}, and 99.69 % of the variance in the extent of reaction ξ. The subsequent latent variables improve the explained variance in the predictor matrix $\mathbf{U}$, but not in the response variables. Contrary to that, using the mole fractions or the partial pressure results in subsequent latent variables explaining as well variance in the response variables. For the following sections, the mole fraction was chosen as independent variable as it combines a reasonable amount of latent variables for all dependent variables with an improved fit for the extent of reaction.

Next, consider the model fit when pressure and temperature differences are used instead of the outlet temperature and pressure. The main influence on the first component for both independent variables T_{out} and p_{out} is given by the respective inlet variable as shown in Chapter 11. As it can be seen in Figure 12.2, using the difference instead of the

absolute value increases the accuracy of the fit for the pressure by a factor of 2 whereas in the case of the temperature the number of necessary latent variables is reduced by one to maintain a similar fit. Here, it is interesting to note that the influence of the inlet pressure and temperature is reduced from being the most crucial independent variable to being less important as given by the loads calculated by PLS regression.

12.2.3 Incorporation of Dependencies in the Sampling Domain Definition

The so far applied individual variation of the molar flows by ± 20 % results in extreme values for the H_2/N_2 ratio of

$$\max\left(\dot{n}_{\mathrm{H_2},in}/\dot{n}_{\mathrm{N_2},in}\right) = 4.7$$
$$\min\left(\dot{n}_{\mathrm{H_2},in}/\dot{n}_{\mathrm{N_2},in}\right) = 2.1$$

However, in practical operations, the ratio will be close to the stoichiometric ratio of 3. Hence, the nitrogen molar flow was varied around the hydrogen molar flow with ± 10 % resulting in an maximum and minimum ratio of 3.5 and 2.86 respectively. The results of this change can be found in Figure 12.3. Due to the definition of the nitrogen molar flow as a function of the hydrogen molar flow, the application of the ratio $\dot{n}_{\mathrm{H_2},in}/\dot{n}_{\mathrm{N_2},in}$ as independent variable instead of the mole fraction $x_{\mathrm{N_2}}$ was investigated as well. In the case of the pressure and temperature difference, the introduction of the dependency results in the same fit. This can be explained by a reduced dependency of both variables on the exact molar composition. However, if the ratio is not used as independent variable, one more component is required for equal good fit for the pressure drop and temperature change. In the case of the the extent of reaction, the introduction of this dependency improves the model fit for 5 to 8 components while remaining the same for a lower number of latent variables. This is valid for both the ratio and the nitrogen mole fraction as independent variable.

12.3 Conclusion

Based on the above presented results, we conclude that preprocessing of data as well as analysing dependencies in the feed to a submodel or between other independent variables has a major influence on the number of latent variables required for a good model fit. If the costs of the fitting of a surrogate model to data is depending on the number of independent variables, it would make sense to use extensive variables as independent variables. If on the other hand, this is not crucial, it is advisable to apply intensive variables as most of the thermodynamical or reaction kinetics equations are functions of intensive variables. Dependent variables should furthermore be defined as differences, as this reduces the influence of the respective inlet variable on the first component defined via PLS regression resulting in an improved fit or alternatively a reduced number of necessary independent variables for a similar fit.

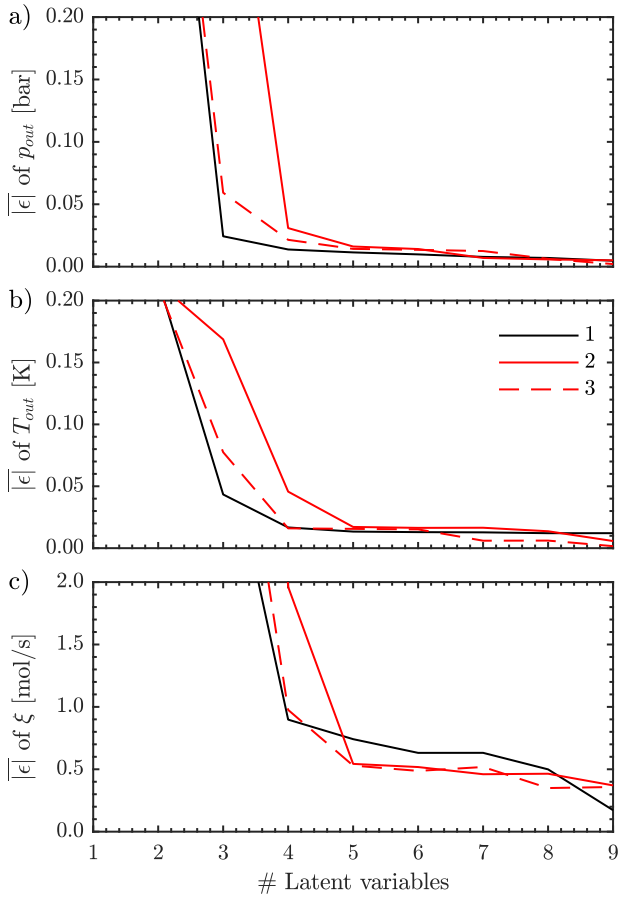


Figure 12.3: Comparison between without incorporation of the hydrogen/nitrogen ratio dependency in the domain definition (1), and with incorporation using the mole fraction of nitrogen (2) and the $\dot{n}_{\mathrm{H}_2,in}/\dot{n}_{\mathrm{N}_2,in}$ ratio as independent variable (3) on the mean absolute error of a) the outlet pressure p_{out}, b) the outlet temperature T_{out}, and c) the extent of reaction ξ as a function of the number of latent variables $n_{u'}$.

Chapter 13

Surrogate Model Generation Using Self-Optimizing Variables

Chapter 11 introduced partial least squares regression (PLSR) for the reduction of the independent variables. Through PLS regression on the sampled space, new latent variables $\mathbf{u}'$ with $\dim(\mathbf{u}') < \dim(\mathbf{u})$ are defined. The independent variables $\mathbf{u}$ for the surrogate model fitting are subsequently the latent variables $\mathbf{u}'$.

Chapter 10 proposed to use the concepts of self-optimizing control (SOC) [90] to reduce the number of independent variables n_u. Self-optimizing control is a philosophy from control theory. The aim is to select controlled variables, which, if kept constant when disturbances occur, give small economic loss. Using SOC, it is possible to reduce the number of independent variables by the number of manipulated variables for each submodel or find variables where a linear model is sufficient. Furthermore, it allows the mapping of the region we are actually interested in. The combination of self-optimizing control and surrogate model generation will hence be investigated in this chapter.

This chapter is structured as follows; Section 13.1 summarizes the application of surrogate models in the context of optimization based on Chapter 10 and introduces the variables used in this chapter. Section 13.2 explains the concepts of self-optimizing control and measurement selection. Section 13.3 discusses how self-optimizing variables can be applied in the generation of surrogate models. Section 13.4 first introduces the ammonia case study and then shows results from the application of self-optimizing control in surrogate model generation. Section 13.5 discusses the applicability of the proposed procedure and addresses limitations and problems if self-optimizing control is utilized.

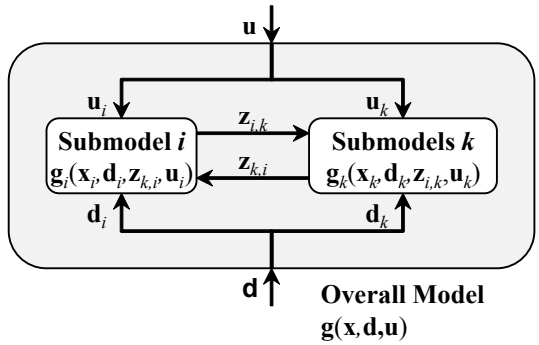


Figure 13.1: Example of a submodel within an overall model.

13.1 Optimization Using Local Surrogate Models

Consider a large-scale steady-state process to be optimized, given by

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J(\mathbf{x},\mathbf{d},\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}(\mathbf{x},\mathbf{d},\mathbf{u}) \\ & \mathbf{0} \geq \mathbf{h}(\mathbf{x},\mathbf{d},\mathbf{u}) \end{aligned} \tag{13.1}$$

where J is a scalar cost function, usually an economic cost, $\mathbf{d} \in \mathbb{R}^{n_d}$ denotes the disturbances, for example feed variables and model parameters, $\mathbf{u} \in \mathbb{R}^{n_u}$ are the independent decision variables, and $\mathbf{x} \in \mathbb{R}^{n_x}$ are the internal model variables. The equality constraints $\mathbf{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ are typically given by the equations in the flowsheeting software. Operational inequality constraints $\mathbf{h} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_h}$ can be imposed on the states $\mathbf{x}$ or inputs $\mathbf{u}$. As the optimization of a large-scale process is generally difficult, the process is split into several submodels given by $\mathbf{g}_i : \mathbb{R}^{n_{x_i}} \times \mathbb{R}^{n_{d_i}} \times \mathbb{R}^{n_{u_i}} \to \mathbb{R}^{n_{x_i}}$ and $\mathbf{h}_i : \mathbb{R}^{n_{x_i}} \times \mathbb{R}^{n_{d_i}} \times \mathbb{R}^{n_{u_i}} \to \mathbb{R}^{n_{h_i}}$. This is exemplified for the distinctive submodel i and the remaining submodels k in Fig. 13.1. Each submodel may have individual manipulated variables $\mathbf{u}_i \in \mathbf{u}$ and disturbances $\mathbf{d}_i \in \mathbf{d}$. It is possible that a disturbance or manipulated variable appears in several submodels. In addition, each submodel has inlet $\mathbf{z}_{k,i} \in \mathbb{R}^{n_{z_{k,i}}}$ and outlet connection variables $\mathbf{z}_{i,k} \in \mathbb{R}^{n_{z_{i,k}}}$. Note that the variables $\mathbf{z}_{i,k}$ are states or outputs for the submodel i they come from, $\mathbf{z}_{i,k} = \mathbf{y}_{i,k}$, whereas they are disturbances for the submodel k they enter. The connection variables and the disturbance variables can be combined into an augmented *disturbance* vector

$$\tilde{\mathbf{d}}_i = \begin{bmatrix} \mathbf{d}_i \\ \mathbf{z}_{k,i} \end{bmatrix} \tag{13.2}$$

Each submodel $\mathbf{g}_i$ may be reformulated as a surrogate model given by

$$\mathbf{g}'_{i,k} : \{\tilde{\mathbf{d}}_i, \mathbf{u}_i\} \mapsto \mathbf{y}_{i,k} \tag{13.3}$$

The total number of independent variables for each submodel is $n_i^{tot} = n_{u_i} + n_{\tilde{d}_i}$. Note that this is an input-output model with no explicit internal states. The reformulated optimization problem in terms of surrogate models then becomes

$$\begin{aligned} \min_{\tilde{\mathbf{d}},\mathbf{u}} \quad & J\left(\tilde{\mathbf{d}},\mathbf{u}\right) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{y}_{i,k} - \mathbf{g}'_{i,k}\left(\tilde{\mathbf{d}}_i,\mathbf{u}_i\right) && i \in 1,\ldots,n, \forall k \neq i \\ & \mathbf{0} = \mathbf{z}_{i,k} - \mathbf{y}_{i,k} && i \in 1,\ldots,n, \forall k \neq i \\ & \mathbf{0} \geq \mathbf{h}_i\left(\tilde{\mathbf{d}}_i,\mathbf{u}_i\right) && i \in 1,\ldots,n \end{aligned} \tag{13.4}$$

The sampling domain for surrogate model generation is given by bounds on the independent variables for each submodel

$$\tilde{\mathbf{d}}_{i,min} \leq \tilde{\mathbf{d}}_i \leq \tilde{\mathbf{d}}_{i,max} \tag{13.5}$$

$$\mathbf{u}_{i,min} \leq \mathbf{u}_i \leq \mathbf{u}_{i,max} \tag{13.6}$$

The sampling may be performed using for example Latin hypercube sampling or regular grid sampling. Depending on the number of independent variables n_i^{tot} and the nonlinearity of the model, this can require a large sampling space. Hence, a reduction in the complexity of the surrogate model may be necessary.

13.2 Previous Results on Self-Optimizing Control

Consider the following optimization problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & J\left(\mathbf{x},\mathbf{d},\mathbf{u}\right) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}\left(\mathbf{x},\mathbf{d},\mathbf{u}\right) \\ & \mathbf{0} \geq \mathbf{h}\left(\mathbf{x},\mathbf{d},\mathbf{u}\right) \end{aligned} \tag{13.7}$$

For example, this could be the local optimization problem, but with subscript i omitted. The aim of self-optimizing control is to identify controlled variables $\mathbf{c}$ which, when kept constant, result in a minimum loss in the presences of disturbances $\mathbf{d}$. Frequently, linear combinations of measurements $\mathbf{y}$ are used

$$\mathbf{c} = \mathbf{H}\mathbf{y} \tag{13.8}$$

where $\mathbf{H} \in \mathbb{R}^{n_u \times n_y}$ is a combination matrix. The question is: how can we identify the optimal selection matrix and correspondingly the self-optimizing variables? A detailed review answering this question can be found in Jäschke et al. [53].

13.2.1 Summary of Self-Optimizing Control Approaches for Obtaining H

For simplicity, the subscript i is dropped in the following explanation of self-optimizing control. The optimal selection matrix $\mathbf{H}$ as introduced in Eq. (13.8) that minimizes $\left|J(\mathbf{c},\mathbf{d}) - J_{opt}(\mathbf{d})\right|$ can be calculated using the nullspace method [2] or the exact-local method [3]. It is given by the solution to the following optimization problem

$$\begin{aligned} \min_{\mathbf{H}} \quad & \left\|\mathbf{HY}\right\|_F \\ \text{s.t.} \quad & \mathbf{HG^y} = \mathbf{J_{uu}^{1/2}} \end{aligned} \tag{13.9}$$

with $\mathbf{G^y} \in \mathbb{R}^{n_y \times n_u}$ representing the measurement gain matrix with respect to the input $\mathbf{u}$. $\mathbf{Y}$ is given by

$$\mathbf{Y} = \begin{bmatrix} \mathbf{FW_d} & \mathbf{W_{n^y}} \end{bmatrix} \tag{13.10}$$

The optimal sensitivity matrix $\mathbf{F} = \frac{\partial \mathbf{y}^{opt}}{\partial \mathbf{d}}$ can be calculated as

$$\mathbf{F} = -\left(\mathbf{G^y J_{uu}^{-1} J_{ud}} - \mathbf{G_d^y}\right) \tag{13.11}$$

where $\mathbf{J_{uu}} \in \mathbb{R}^{n_u \times n_u}$ is the Hessian of the cost function, and $\mathbf{J_{ud}} \in \mathbb{R}^{n_u \times n_d}$ which is the second order derivative of J with respect $\mathbf{u}$ and $\mathbf{d}$. Alternatively, if it is not possible to easily obtain the analytic matrices of the cost function $\mathbf{J_{uu}}$ and $\mathbf{J_{ud}}$, the optimal sensitivity matrix $\mathbf{F}$ can also be calculated using finite differences. This results in n_d additional optimization problems. $\mathbf{W_d}$ and $\mathbf{W_{n^y}}$ are the disturbance and measurement noise scaling matrices given by

$$\Delta\mathbf{d} = \mathbf{W_d d'}; \qquad \mathbf{n^y} = \mathbf{W_{n^y} n^{y'}} \tag{13.12}$$

The vectors $\mathbf{d'}$ and $\mathbf{n^{y'}}$ are assumed to satisfy

$$\left\| \begin{bmatrix} \mathbf{d'} \\ \mathbf{n^{y'}} \end{bmatrix} \right\|_2 \leq 1 \tag{13.13}$$

Thus, $\mathbf{W_d}$ and $\mathbf{W_{n^y}}$ represent the magnitude of the expected variations in $\mathbf{d}$ and $\mathbf{y}$. The solution to problem (13.9) when $\mathbf{H}$ is a full matrix is [107]:

$$\mathbf{H}^\top = \left(\mathbf{YY}^\top\right)^{-1} \mathbf{G}^y \tag{13.14}$$

13.2.2 Measurement Selection for H

It is in general desirable to use few measurements $\mathbf{y}$. In order to select the an optimal subset of measurements n_y, Yelchuru and Skogestad [107] developed a mixed integer quadratic programming approach. It requires the reformulation of the problem given in

Eq. (13.9) in vectorized form:

$$\begin{aligned} \min_{\mathbf{h}_\delta \sigma_\delta} \quad & \mathbf{h}_\delta^\mathsf{T} \mathbf{F}_\delta \mathbf{h}_\delta \\ \text{s.t.} \quad & \mathbf{G}_\delta^{y\,\mathsf{T}} \mathbf{h}_\delta = \mathbf{j}_\delta \\ & \sum_{k=1}^{n_{y^{tot}}} \sigma_k = n_y \end{aligned} \tag{13.15}$$

where $\sigma_k \in \{0,1\}$ with $k = 1 \ldots n_{y^{tot}}$ are binary variables to indicate, whether measurements are used in the selection matrix. The quadratic cost term is given by

$$\mathbf{F}_\delta = \mathbf{Y}_\delta \mathbf{Y}_\delta^\mathsf{T} \tag{13.16}$$

and is block diagonal. The same holds true for $\mathbf{G}_\delta^{y\,\mathsf{T}}$ whereas $\mathbf{h}_\delta$ and $\mathbf{j}_\delta$ are a vectorized form of $\mathbf{H}$ and $\mathbf{J_{uu}}$ respectively. Further constraints have to be imposed on $\mathbf{h}_\delta$ to guarantee that $h_{jk} = 0$ for $\sigma_k = 0$ and input u_j and measurement y_k. In this problem, the big-m approach is chosen. This results in bounds for the entries in the selection matrix $\mathbf{H}$ given by

$$-\begin{bmatrix} m \\ m \\ \vdots \\ m \end{bmatrix} \sigma_k \leq \begin{bmatrix} h_{1k} \\ h_{2k} \\ \vdots \\ h_{n_u k} \end{bmatrix} \leq \begin{bmatrix} m \\ m \\ \vdots \\ m \end{bmatrix} \sigma_k, \qquad \forall k \in 1,2,\ldots,n_{y^{tot}} \tag{13.17}$$

For a detailed description and derivation of the MIQP approach for measurement selection, the reader is referred to Yelchuru and Skogestad [107].

13.3 Surrogate Model Generation Using Self-Optimizing Variables

Consider a detailed model representation

$$\mathbf{g}_i \left(\mathbf{x}_i, \tilde{\mathbf{d}}_i, \mathbf{u}_i \right) = \mathbf{0} \tag{13.18}$$

of submodel i and let $\mathbf{y}_{i,k} = \mathbf{f}_{i,k} \left(\mathbf{x}_i, \tilde{\mathbf{d}}_i, \mathbf{u}_i \right)$ represent the variables we are interested in knowing. To avoid solving the detailed model (13.18) every time, for example during optimization, we want to derive a surrogate model (13.3). To be able to introduce self-optimizing variables $\mathbf{c}_i$ to replace the original independent variables $\mathbf{u}_i$, we assume that we can define a a local cost function $J_i \left(\mathbf{x}_i, \tilde{\mathbf{d}}_i, \mathbf{u}_i \right)$. This cost function should reflect the overall cost J in (13.1) because we are not interested in arbitrary variations in $\mathbf{u}_i$, but in changes along the optimal surface. That is, we want to find instead a surrogate model

$$\mathbf{g}'_{i,k,SOC} : \{ \tilde{\mathbf{d}}_i, \mathbf{c}_i \} \mapsto \mathbf{y}_{i,k} \tag{13.19}$$

where we remain close to the optimal surface for expected variations in $\tilde{\mathbf{d}}_i$ when the new variables $\mathbf{c}_i$ are kept constant (or strictly speaking their setpoints $\mathbf{c}_{i,s}$).

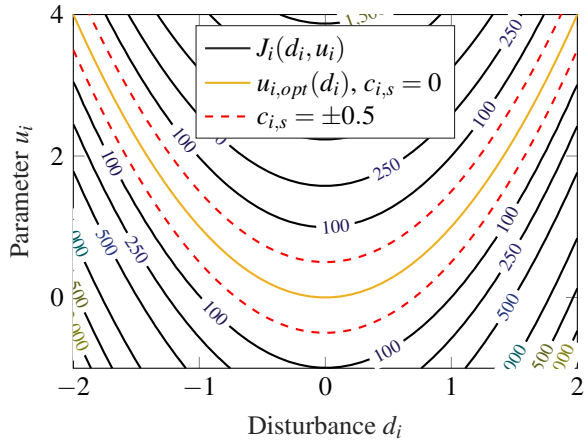


Figure 13.2: Example for mapping the optimal response surface using the Rosenbrock function as case study.

13.3.1 Motivating Example

As a motivating illustration of the concept, consider the Rosenbrock function [85]:

$$J_i(u_i) = (1 - d_i)^2 + 100\left(u_i - d_i^2\right)^2 \tag{13.20}$$

The cost function is in this case also the outlet dependent variable y_i. The contour map for J_i as a function of u_i and d_i is shown in Figure 13.2. For a given disturbance value d_i, it would not make sense to map the whole region for u_i as it includes regions with a high value of the cost function. Instead, it is preferable to map only the region around the optimal input $u_{i,opt}(d_i)$ as given by the yellow line. Note, by introducing

$$c_i = u_i - d_i^2 \tag{13.21}$$

and setting $c_{i,s} = 0$, the cost function is minimized independently of the value of d_i as we indirectly get $u_i = u_{i,opt}(d_i) = d_i^2$. This allows to map along the optimal response surface as shown by the dashed lines. These bounds correspond to $c_i = \pm 0.5$. The close-to-optimal response surface has a simpler structure compared to the complete response surface. Compared to the optimal response surface approach, it is possible to vary the setpoint of the new variable, $c_{i,s}$ as well. The surrogate models according to Eqs. (13.3) and (13.19) are then given by

$$\mathbf{g}_i' : \{d_i, u_i\} \mapsto J_i \tag{13.22}$$

$$\mathbf{g}_{i,SOC}' : \{d_i, c_i\} \mapsto J_i \tag{13.23}$$

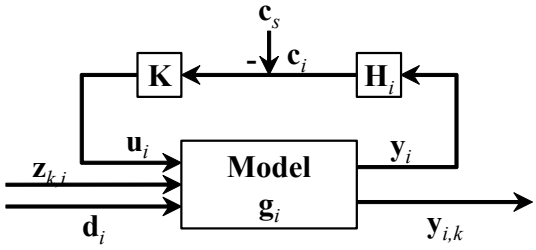


Figure 13.3: Block diagram illustrating the change of independent variables.

in which $\mathbf{g}_i'$ would include interaction terms between u_i and d_i and fourth-order terms in d_i. On the other hand, $\mathbf{g}_{i,SOC}'$ does not include interaction terms and at most second-order terms are needed in the model. Correspondingly, less points have to be sampled due to the simpler structure of the model. Note that this illustrative example does not correspond to the linear self-optimizing variables $\mathbf{c}_i$ as used in this paper. Unfortunately, it is in general difficult to obtain nonlinear self-optimizing variables. Hence, linear combinations of measurements are used as self-optimizing variables as outlined in the next section.

13.3.2 Prcoedure for Selecting Self-Osptimizing Variables

Let the *measurements*

$$\mathbf{y}_i = \begin{bmatrix} \mathbf{x}_i \\ \tilde{\mathbf{d}}_i \\ \mathbf{u}_i \end{bmatrix} \tag{13.24}$$

represent all the system variables and assume that we use self-optimizing control ideas to find a linear measurement combination

$$\mathbf{c}_i = \mathbf{H}_i \mathbf{y}_i \tag{13.25}$$

with $\mathbf{c}_i \in \mathbb{R}^{n_{u_i}}$ are the local self-optimizing variables and $\mathbf{y}_i \in \mathbb{R}^{n_{y_i}}$ are the selected measurements to replace the independent variables $\mathbf{u}_i$. This variable change is illustrated in Figure 13.3 using a block diagram. The controller $\mathbf{K}$ has integral action so that we have perfect control at steady state ($\mathbf{c}_i = \mathbf{c}_{i,s}$). Note that Figure 13.3 is just for illustrating how we can change the independent variables from $\mathbf{u}_i$ to $\mathbf{c}_i$, and there are no dynamics present in the surrogate model. The objective is that with this change in independent variables, the surrogate models become much simpler, for example linear, and in some cases we may even eliminate variables.

To this effect, we define a local cost function $J_i(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i)$ and consider the following local optimization problem

$$\begin{aligned} \min_{\mathbf{x}_i, \mathbf{u}_i} \quad & J_i(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i) \\ \text{s.t.} \quad & \mathbf{0} = \mathbf{g}_i(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i) \\ & \mathbf{0} \geq \mathbf{h}_i(\mathbf{x}_i, \mathbf{d}_i, \mathbf{u}_i) \end{aligned} \tag{13.26}$$

We need to define the expected disturbance set through the weight $\mathbf{W_d}$ and the expected *noise* (caused by numerical errors) in the measurements $\mathbf{y}_i$ through the weight $\mathbf{W_{n^y}}$. The self-optimizing variables are then obtained as the set $\mathbf{c}_i = \mathbf{H}_i \mathbf{y}_i$ which minimizes $\left| J_i\left(\mathbf{c}_i, \tilde{\mathbf{d}}_i\right) - J_{i,opt}\left(\tilde{\mathbf{d}}_i\right) \right|$. If we neglect *noise*, then we may use the nullspace method [2], but in this paper we include *noise* and use the exact local method, [3] as described in more detail in Section 13.2.

Note that we include the inlet connection variables $\mathbf{z}_{k,i}$ as disturbances in the calculation of $\mathbf{H}_i$, that is

$$\tilde{\mathbf{d}}_i = \begin{bmatrix} \mathbf{d}_i \\ \mathbf{z}_{k,i} \end{bmatrix} \tag{13.27}$$

and $n_{\tilde{d}_i} = n_{d_i} + n_{z_{k,i}}$. The calculation of the SOC selection matrix according to optimization problem (13.15) requires the solution to 1 (or $n_{\tilde{d}_i} + 1$ if the optimal sensitivity matrix is calculated using finite differences) nonlinear problem(s) and n_{u_i} mixed integer quadratic problems. Furthermore, n_i^{tot} nonlinear systems of equations have to be solved to obtain the gain matrix $\mathbf{G^y}$ and the disturbance gain matrix $\mathbf{G_d^y}$. The sampling for the calculation of the surrogate model then consists of solving n_p nonlinear systems of equations.

As scaling matrices, we suggest using

$$\mathbf{W_d} = \operatorname{diag}\left(\max\left(\tilde{\mathbf{d}}_i - \tilde{\mathbf{d}}_{i,min}, \tilde{\mathbf{d}}_{i,max} - \tilde{\mathbf{d}}_i\right)\right) \tag{13.28}$$

as this results in minimizing the loss within the surrogate model domain. The *measurement noise* scaling matrix should be set to the expected numerical noise in $\mathbf{y}_i$. If this noise is small compared to the disturbance scaling matrix, we can set the measurement noise scaling matrix to

$$\mathbf{W_{n^y}} = w_{n^y} \operatorname{diag}(\mathbf{1}) \tag{13.29}$$

where w_{n^y} is small and $\mathbf{1}$ is a vector of ones with length n_y. However, two necessities arise for the parameter w_{n^y}

1. w_{n^y} is large enough so that $\mathbf{Y}\mathbf{Y}^\mathsf{T}$ in (13.14) is nonsingular;
2. w_{n^y} should be small compared to the entries of $\mathbf{W_d}$ to reduce the effect of measurement noise in the calculation of the selection matrix $\mathbf{H}$.

It is often preferable to use a block diagonal selection matrix $\mathbf{H}_i$. The advantage of a block diagonal matrix is to reduce the computational load of adjusting the setpoints iteratively in the flowsheeting software. This corresponds to Problem 3 described by Yelchuru and Skogestad [107] and cannot be solved using the MIQP approach of [107] as it violates the convex formulation theorem. However, it is possible to calculate a *local* selection matrix $\mathbf{H}_{i,l}$ for each input $u_{i,l}$ using only measurements in the vicinity of $u_{i,l}$, see 13.2.2 for details. The resulting block diagonal matrix is due to neglecting interactions not optimal, but is sufficient for the subsequent application. A discussion of the scaling matrices and the use of a structured selection matrix $\mathbf{H}_i$ is provided in Section 13.5.2.

In summary, the procedure for utilizing self-optimizing variables in the context of surrogate model generation can be summarized as follows:

1. Set up a nonlinear problem (13.26) for submodel i and identify the connection variables $\mathbf{z}_{k,i}$ and $\mathbf{y}_{i,k}$.
2. Construct the augmented disturbance vector
$$\tilde{\mathbf{d}}_i = \begin{bmatrix} \mathbf{d}_i \\ \mathbf{z}_{k,i} \end{bmatrix}$$
and the measurement vector
$$\mathbf{y}_i = \begin{bmatrix} \mathbf{x}_i \\ \tilde{\mathbf{d}}_i \\ \mathbf{u}_i \end{bmatrix}$$
3. Define the sampling domain and the scaling matrices $\mathbf{W_d}$ and $\mathbf{W_{n^y}}$, for example using (13.28) and (13.29).
4. Solve the nonlinear problem (13.26) for the nominal input variables and calculate the sensitivity matrix $\mathbf{F}$ either using Eq. (13.11) or through finite differences. This requires the solution of 1 or $1+n_{\tilde{d}_i}$ optimization problems, depending on the availability of analytic expressions for $\mathbf{G^y}$, $\mathbf{G^y_d}$, $\mathbf{J_{uu}}$, and $\mathbf{J_{ud}}$.
5. Define the local measurements around the manipulated variables used for the calculation of the self-optimizing variables based on the total measurement $\mathbf{y}_i$.
6. Define the maximum number of measurements used for each manipulated variable u_l.
7. Calculate the optimal selection matrices $\mathbf{H}_{i,l}$ for the different manipulated variables $u_{i,l}$ using the MIQP (13.15).
8. Add the linear equality constraints given by Eq. (13.25) to the model $\mathbf{g}_i$ and sample n_p points.
9. Construct the surrogate models $\mathbf{g}'_{i,k,SOC}$ for the dependent variables $\mathbf{y}_{i,k}$ as given in Eq. (13.19).

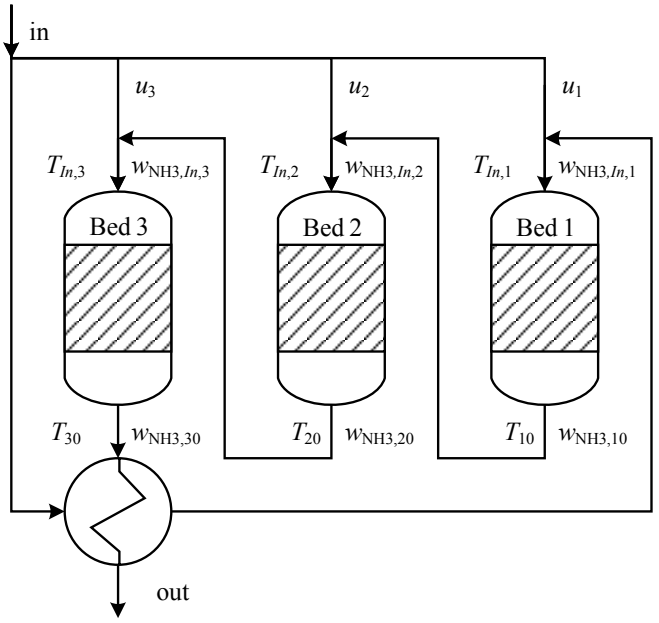


Figure 13.4: Heat-integrated three bed reactor system of the ammonia synthesis gas loop.

13.4 Case Study - Ammonia Synthesis Reactor

The case study is a heat-integrated ammonia reactor as shown in Figure 13.4. This reactor was previously used for stability analysis [75] and the application of several optimal operation methods in Part II. A detailed model description can be found in Appendix A. Chapter 4 showed that small disturbances lead to limit-cycle behaviour and/or reactor extinction at the steady-state optimal operation point. Varying the manipulated variables **u** individually results therefore in creating a response surface that includes undesirable operating regions with reactor extinction and limit cycle behaviour. This results in a complicated response surface and it is necessary to sample a lot of points to achieve a good performance of the surrogate model. Hence, the ammonia reactor can be seen as an excellent case study for the application of the proposed method.

13.4.1 Model Description and Modification

The aim of the reactor is to maximize the conversion *per pass*, which can be expressed in this example as the rate of the extent of reaction ξ [kg/s],

$$\dot{\xi} = \dot{m}_{in}\left(w_{\mathrm{NH_3},30} - w_{\mathrm{NH_3},in}\right) \tag{13.30}$$

Table 13.1: Bounds and units for the connection variables.

	$\dot{m}_{in}$ [kg/s]	p_{in} [bar]	T_{in} [°C]	$w_{\mathrm{NH_3},in}$ [wt.%]	$R_{\mathrm{H_2/N_2},in}$ [-]
Lower Bound	59.5	185	235	7	2.8
Nominal Value	70.0	200	250	8	3.0
Upper Bound	80.5	215	265	9	3.2

where $\dot{m}$ [kg/s] is the mass flow and $w_{\mathrm{NH_3},i}$ the ammonia mass fraction. Correspondingly, the cost function for the optimization problem (13.26), which is posed as minimization problem, is given by

$$J = -\dot{\xi} \tag{13.31}$$

The equality constraints are given by the ammonia mass balance and the energy balance described in Appendix A for each CSTR j in the CSTR cascade used to represent each reactor bed. The number of CSTRs in each bed is $n = 10$.

In order to increase the applicability of the resulting surrogate model, the hydrogen to nitrogen molar ratio is not considered to be fixed anymore. Instead, the molar ratio of hydrogen to nitrogen,

$$R_{\mathrm{H_2/N_2},j} = \frac{\dot{n}_{\mathrm{H_2},j}}{\dot{n}_{\mathrm{N_2},j}} \tag{13.32}$$

in each reaction section j is introduced as an algebraic state. This results in 30 additional algebraic constraints

$$0 = R_{\mathrm{H_2/N_2},j} - \frac{\dot{n}_{\mathrm{H_2},j-1} + r_{\mathrm{H_2},j} m_{cat,j} / M_{\mathrm{H_2}}}{\dot{n}_{\mathrm{N_2},j-1} + r_{\mathrm{N_2},j} m_{cat,j} / M_{\mathrm{N_2}}} \tag{13.33}$$

in which M_i is the respective molar mass and $r_{i,j}$ the reaction rate in [kg i/kg$_{cat}$ h].

For this system, the independent variables are given by the three split ratios

$$\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^\mathsf{T} \tag{13.34}$$

which may be viewed as the real manipulated variables. The five disturbances are the inlet conditions to the system

$$\tilde{\mathbf{d}} = \begin{bmatrix} \dot{m}_{in} & p_{in} & T_{in} & w_{\mathrm{NH_3},in} & R_{\mathrm{H_2/N_2},in} \end{bmatrix}^\mathsf{T} \tag{13.35}$$

These are the connection variables $\mathbf{z}_{k,i}$. The bounds on the disturbance variables are given in Table 13.1. Two output variables have to be fitted in the surrogate model. These are the (mass) rate of the extent of reaction ξ and the outlet temperature T_{out}. The outlet ratio $R_{\mathrm{H_2/N_2},out}$ can be calculated through the respective outlet molar flows $\dot{n}_{i,out}$ which

in turn are calculated from exact mass balances using ξ. This furthermore guarantees mass conservation in the resulting surrogate model. To summarize,

$$\mathbf{y}_{i,k} = \begin{bmatrix} \xi \\ T_{out} \end{bmatrix} \tag{13.36}$$

The system was modelled using CasADi [4] and optimized using IPOPT [102].

13.4.2 Application of SOC

As $n_u = 3$, three SOC variables $\mathbf{c} = \mathbf{H}\mathbf{y}$ have to be obtained. We want to use local variables for each reactor bed to simplify the calculations when using a flowsheet simulator. Hence, the MIQP approach in (13.15) as proposed by Yelchuru and Skogestad [107] is applied individually for each bed resulting in a block diagonal matrix given by

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & 0 & 0 \\ 0 & \mathbf{H}_2 & 0 \\ 0 & 0 & \mathbf{H}_3 \end{bmatrix} \tag{13.37}$$

In order to have a small number of measurements, we consider for each bed $n_y = 1$ and $n_y = 2$. This is compared to a more intuitive control structure, where the the inlet temperatures (*In*) as well as the inlet and outlet temperatures of the respective beds are used (*In,Out*). The scaling matrix $\mathbf{W_d}$ according to Eq. (13.28) and Table 13.1 is

$$\mathbf{W_d} = \mathrm{diag}\left(\begin{bmatrix} 10.5 & 15 & 15 & 1 & 0.2 \end{bmatrix}\right) \tag{13.38}$$

whereas the parameter w_{n^y} in the calculation of $\mathbf{W_{n^y}}$ is selected as $w_{n^y} = 10^{-3}$.

The candidate measurements for the MIQP approach are

$$\mathbf{y}_1 = \begin{bmatrix} T_{In,1} \\ \mathbf{T}_{1:10} \end{bmatrix} \qquad \text{for Bed 1, } u_1 \tag{13.39}$$

$$\mathbf{y}_2 = \begin{bmatrix} T_{In,2} \\ \mathbf{T}_{11:20} \end{bmatrix} \qquad \text{for Bed 2, } u_2 \tag{13.40}$$

$$\mathbf{y}_3 = \begin{bmatrix} T_{In,3} \\ \mathbf{T}_{21:30} \end{bmatrix} \qquad \text{for Bed 3, } u_3 \tag{13.41}$$

Therefore, each of the three selection matrices considers 11 measurements. The mass fraction measurements $\mathbf{w}_{\mathrm{NH_3}}$ would not be viable measurements for control purposes. However, in the case of surrogate model generation, they can still be used. We found that including the mass fractions in the measurements did not change the selected subset of measurements. Hence, the mass fractions are excluded in the candidate measurements.

Table 13.2: Optimal selection matrix for a fixed selection (*In,Out*) as well as the optimal measurement subset for each input and the corresponding optimal selection matrix $\mathbf{H}_i$ with $n_y = 2$ ($MIQP_2$).

		Chosen Variables	Selection Matrix $\mathbf{H}_i$
In,Out	Bed 1	$T_{In,1}$, T_{10}	$\begin{bmatrix} 0.067 & -1.000 \end{bmatrix}$
	Bed 2	$T_{In,2}$, T_{20}	$\begin{bmatrix} 0.098 & 1.000 \end{bmatrix}$
	Bed 3	$T_{In,3}$, T_{30}	$\begin{bmatrix} 1.000 & 0.721 \end{bmatrix}$
$MIQP_2$	Bed 1	T_4, T_6	$\begin{bmatrix} 0.952 & -1.000 \end{bmatrix}$
	Bed 2	$T_{In,2}$, T_{11}	$\begin{bmatrix} 0.982 & -1.000 \end{bmatrix}$
	Bed 3	T_{28}, T_{30}	$\begin{bmatrix} 1.000 & -0.994 \end{bmatrix}$

The optimization problem (13.15) was solved using $m = 100$ in the big-m approach of Eq. (13.17). The solution to the problem with one measurement ($n_y = 1$, $MIQP_1$) for each bed gives as chosen measurements:

$$\text{Bed 1: } T_9 \qquad \text{Bed 2: } T_{18} \qquad \text{Bed 3: } T_{25} \tag{13.42}$$

The solution to the optimization problem (13.15) with $n_y = 2$ is given in Table 13.2 ($MIQP_2$). Similar to the results reported by Yelchuru and Skogestad [107], the chosen measurements change depending on the chosen number of measurements n_y. That means that a measurement which is optimal with only one measurement is not necessarily included with two measurements.

13.4.3 Fitting of the Surrogate Model

The surrogate models are cubic B-splines fitted through the application of the SPLINTER library [41], which requires a regular grid in the independent variables $\mathbf{c}_s$ (which here replace $\mathbf{u}$) and $\tilde{\mathbf{d}}$. In this case study, the overall cost function is minimized by optimizing locally the degrees of freedom ($\mathbf{c} = \mathbf{H}\mathbf{y}$), that is, the local cost J_i is equal to the global cost J, so it is not necessary to include the setpoints $\mathbf{c}_s$ as degrees of freedom as it normally would be. The regular grid is given by four points for each of the varied variable $\tilde{\mathbf{d}}$, $\dot{m}_{in}$, p_{in}, T_{in}, $w_{\mathrm{NH_3},in}$, and $R_{\mathrm{H_2/N_2},in}$. This results in $n_p = 4^5 = 1024$ sampling points. The advantage of using B-splines of order two or higher is that it gives continuity of the first derivative of the surrogate model. This gives advantages for the subsequent optimization. If self-optimizing variables are not used, we would need to consider all variables ($\tilde{\mathbf{d}}$ and $\mathbf{u}$) simultaneously giving $4^8 = 65536$ sampling points. Alternatively, other surrogate model structures like Kriging or the ALAMO approach [20] could be used.

Table 13.3: Estimation error ε with fixing the three SOC variables using different selection matrices $\mathbf{H}$.

$\mathbf{H}$ definition	Rate of Extent of Reaction ξ		Outlet Temperature T_{out}	
	$\max\|\varepsilon\|$	$\overline{\|\varepsilon\|}$	$\max\|\varepsilon\|$	$\overline{\|\varepsilon\|}$
In	4.353 %	0.742 %	8.00 K	1.457 K
In,Out	0.540 %	0.092 %	1.02 K	0.184 K
$MIQP_1$	0.211 %	0.027 %	0.41 K	0.055 K
$MIQP_2$	0.022 %	0.003 %	0.04 K	0.005 K

13.4.4 Evaluation of the Surrogate Model Performance

The resulting surrogate models for the outlet temperature T_{out} and ξ were evaluated using 5000 randomly sampled validation points. These validation points are the optimal response surface for this model. This implies that the surrogate model may theoretically give perfect fit for the self-optimizing variables response surface. However, this is only of minor interest as the aim of the surrogate model is to utilize it in further optimization.

In order to compare the different methods, the maximum absolute error $\max|\varepsilon|$ and the mean absolute error $\overline{|\varepsilon|}$ are calculated with respect to the optimal response surface. The results of the four different combination matrices can be found in Table 13.3. We can see that arbitrarily chosen measurements (*In* and *In,Out*) do not necessarily result in a good surrogate model fit. Using only the three inlet temperatures (*In*) results in a training space with infeasible points. For example, two of the split ratios are negative for

$$\check{\mathbf{d}} = \begin{bmatrix} 80.5 & 185 & 235 & 9 & 2.8 \end{bmatrix}^{\mathsf{T}} \tag{13.43}$$

Adding the outlet temperature of each bed to the selected measurements (*In,Out*) reduces the error by one order of magnitude. Furthermore, all points in the training space are feasible. Selecting only one optimal measurement in each bed ($MIQP_1$, see (13.42)) reduces the error by more than a factor two compared to using two arbitrary measurements (*In,Out*) in each bed. Finally, increasing the number of measurements in the MIQP approach from $n_y = 1$ ($MIQP_1$) to $n_y = 2$ ($MIQP_2$) in each bed gives a further decrease by one order of magnitude. This shows that it is important to select the best measurements. Otherwise, the resulting surface is more complicated and it may be even necessary to reduce the sampling space to avoid sampling infeasible points.

Importantly, the resulting response surface is simple. To show this, the surrogate models were validated using a response surface created through incorporation of constraints (13.25). The comparison to this validation space results in a maximum absolute error for the surrogate model using the inlet temperatures (*In*) of 0.37 % in ξ and 0.7 K in T_{out}. Compared to the optimal response surface, this error is one order of magnitude

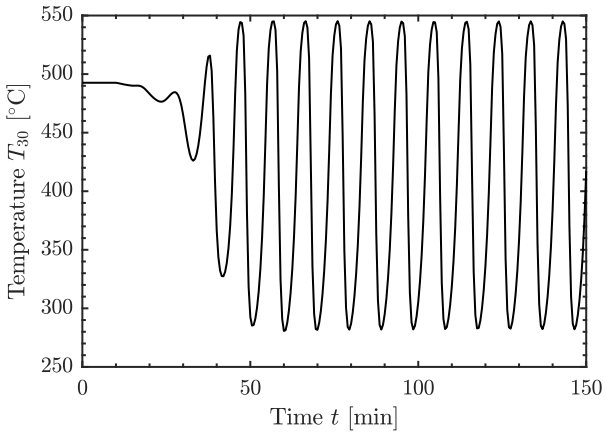


Figure 13.5: Outlet temperature of Bed 3 with a pressure drop of $\Delta p_{in} = -15$ bar at $t = 10$ min with a constant input $\mathbf{u}$ at the optimal point.

smaller. In comparison, with the MIQP approach with $n_y = 2$ ($MIQP_2$), the maximum absolute error is given by 0.001 % in ξ and 0.001 K in T_{out}. This indicates, that the resulting response surface is indeed simpler and it is possible to reduce the number of sampling points. Using instead quadratic B-splines with three points for each variable $\tilde{\mathbf{d}}$, $n_p = 5^3 = 243$ points have to be sampled. The maximum absolute error is then given by 0.01 % in ξ and 0.02 K in T_{out} which is still below the error of using the inlet temperatures with 4 points for each variable.

Sampling the space without variable transformation, *i.e.* selecting $\mathbf{c} = \mathbf{u}$, is not advisable for this case study. First, as already mentioned, it would require the sampling of much more sampling points. In addition, the resulting surface is more complicated. To illustrate this, consider the case when all disturbance variables are at their lower bound (Table 13.1)

$$\tilde{\mathbf{d}} = \begin{bmatrix} 59.5 & 185 & 235 & 7 & 2.8 \end{bmatrix}^{\mathsf{T}} \tag{13.44}$$

and the manipulated variables are fixed at their nominal optimum ($\mathbf{u} = \mathbf{u}_{opt}\left(\tilde{\mathbf{d}}_{nom}\right)$). In this situation, the reactor is extinct. Hence, using the split ratios $\mathbf{u}$ as independent variables would require the mapping of regions in which the reactor is extinct as well as crossing the limit-cycle region [75]. This region is exemplified in Figure 13.5 where the inlet pressure is at its lower bound and the other disturbances at their nominal value. We can see that the system displays limit-cycle behaviour and it is not possible to define a steady-state value for this operating point. However, these regions are not important for the subsequent optimization, and hence, should not be sampled.

13.5 Discussion

13.5.1 Advantages of the Proposed Method

The proposed utilization of self-optimizing variables to map the optimal response surface is a promising new method in the generation of surrogate models. The main advantages are given by

1. A response surface which is close to the optimal response surface but does not require the solution of a large number of nonlinear problems;
2. Potentially a reduced number of sampling points compared to sampling with the original independent variables.

This allows us to sample only regions we are interested in, and to neglect regions that are not encountered in practice. In the case study, it is in fact *not* possible to use the original inputs $\mathbf{u}_i$ (split ratios) as independent variables. Thus, a variable transformation would be required independently of the application of self-optimizing control. For example, one could use the variable transformation utilizing the existing control structure as proposed in Chapter 10. If it is necessary to have surrogate models for other states than the dependent variables $\mathbf{y}_{i,k}$, it is possible to calculate them as well, *e.g.* for the actual split ratios $\mathbf{u}_i$ or for additional potential measurements. In certain cases, as for our case study, it is as well feasible to reduce the number of independent variables.

An alternative to utilizing self-optimizing variables is to directly sample the optimal response surface given by

$$\mathbf{g}'_{i,opt} : \{\tilde{\mathbf{d}}_i\} \mapsto \mathbf{y}_{i,k} \tag{13.45}$$

This approach is however computationally expensive. It would require the solution to n_p nonlinear problems whereas in the application of the proposed method, only $n_{\tilde{d}} + 1$ have to be solved in the calculation of the optimal sensitivity matrix $\mathbf{F}$. In addition, this approach does not allow for having the set points $\mathbf{c}_s$ as degree of freedom for solving the overall optimization problem as it is allowed with surrogate model (13.19).

Certain limitations of the proposed method can be identified and need to be addressed.

13.5.2 Practical Use of the Proposed Method

A first important point is the selection of the disturbance and measurement scaling matrices, $\mathbf{W_d}$ and $\mathbf{W_{n^y}}$. These matrices will influence the performance of the resulting surrogate model. The 2-norm is used for scaling of the disturbances and measurement noises as we can see in Eq. (13.13). This implies that all disturbances and measurement noises may not be at their upper or lower limit simultaneously. In the case of control, this seems reasonable and a detailed discussion for using the 2-norm is given by Halvorsen et al.

[46]. However, this is not the case, if we want to use the self-optimizing variables in the calculation of surrogate models. We actually want to sample these so-called corner points to avoid extrapolation. The best would be to use another norm, for example the 1-norm, but it can be partly circumvented by multiplying $\mathbf{W_d}$ by $\sqrt{n_{\tilde{d}}}$.

A second important point relates to the selection of the *measurements* $\mathbf{y}$ as they influence the loss when disturbances are present. In the case of surrogate model generation, the measurements do not need to be actual measurement as it is in the control application of SOC. Hence, it is possible to extend the measurements to states that are generally not considered as they are hard to measure, *e.g.* concentrations. As a result, the number of possible measurements $n_{y^{tot}}$ can be high. This requires the application of the MIQP approach as developed by [107] and described in 13.2.2. Unfortunately, the MIQP approach for measurement selection does not handle structural zeros in the selection matrix $\mathbf{H}$. The reason is that the the convex reformulation for obtaining the optimization problem (13.9) does not hold in this case as a pre-multiplication of $\mathbf{H}$ with a non-singular matrix $\mathbf{Q}$ will not preserve the structure of $\mathbf{H}$. Consequently, it is necessary to minimize the nonlinear loss expression

$$L = \frac{1}{2} \left\| \mathbf{J}_{\mathbf{uu}}^{1/2} \left(\mathbf{H}\mathbf{G}^{\mathbf{y}} \right)^{-1} \mathbf{H}\mathbf{Y} \right\|_F^2 \tag{13.46}$$

This optimization could be performed using a global non-linear mixed integer optimization solver like BARON [100] or ANTIGONE [71]. Unfortunately, there are no simple methods for solving this problem in a convincing way as highlighted by Jäschke et al. [53]. The development of an approach to include structural zeros is not the scope of this paper and will therefore not be discussed further. When we used alternatively a full selection matrix $\mathbf{H}$, we found that complicated adjustments in the flowsheet solver occurred. The proposed *local* approach and the resulting block diagonal selection matrix does not guarantee the optimal measurement combination in the combined measurement matrix and may lead to cases, where problems may arise. However, it is not possible to generalize when problems may occur and when not.

13.5.3 Number of Independent Variables

It is in general not possible to say, when the application of self-optimizing variables allows a reduction in the number of independent variables. There are however certain conditions, which have to be fulfilled as it is the case in the case study. One prerequisite is that the cost function J_i corresponds to the overall cost function J. In addition, it is necessary that the cost function is flat with respect to the self-optimizing variables as it was already stated by Skogestad [90]. The simpler response surface as aim of the introduction of self-optimizing variables is still likely to hold, independently of whether it is possible to reduce the number of independent variables. Especially if there are many

disturbances (or connection variables), this may give a much simpler surrogate model which requires fewer sampling points to get a desired accuracy, as in the case study.

13.5.4 Application in Flowsheeting Software

The application of self-optimizing variables $\mathbf{c}_i$ in flowsheeting software can be difficult. It requires the use of additional equality constraints which can cause problems because it may require many iterations in sequential-modular simulators. As a result, the computational expense is increased. Hence, we proposed to use a structured selection matrix $\mathbf{H}_i$ with measurements in the vicinity of the respective manipulated variables. The variables $\mathbf{c}_i$ are then more decoupled and it is not necessary to converge the complete flowsheet for each adjustment. This problem is less pronounced in equation-oriented simulators. There, the application of surrogate model-based optimization results in smaller models, and hence, a simpler initialization of the models. The application of self-optimizing variables will then only increase the number of equality constraints.

13.5.5 Local Cost Function

The application of self-optimizing variables requires the definition of a local cost J_i corresponding to the overall cost J. This is possible for the investigated case study as it is general advantageous to maximize the conversion per pass of a chemical reactor. If this is not the case, the self-optimizing variables must be obtained using the overall cost and optimization problem (13.1). In the investigated case study, the introduction of an additional heat-exchanger with external cooling duty would for example complicate the cost function as there is no direct cost linked to the conversion per pass. This brings us back to the starting point of the application of surrogate model generation and is similar to the *hen and egg* problem. If it is not possible to optimize the overall flowsheet, how can we then calculate the self-optimizing variables for each submodel $\mathbf{g}_i$?

One approach to achieve this is the utilization of a simplified overall model for the calculation of the self-optimizing variables. These can then be used in the generation of the surrogate models for the submodels. In addition, this would require the incorporation of the setpoints $\mathbf{c}_s$ of the self-optimizing variables into the sampling domain as they do not correspond to the actual optimal values due to the simplified overall model. The advantage here is the simplified response surface. Another approach, which is is probably better, is to define a reasonable local cost function, for example, based on physical arguments.

13.5.6 Manipulated Variables and Disturbances Affecting Several Submodels

As mentioned in Section 13.1, it possible that disturbances and manipulated variables affect several submodels. This can include, for example, a disturbance in the cooling water temperature or if several compressors are connected to the same turbine shaft. This leads to the question if we can apply the procedure in these cases as well.

On one hand, it is not a problem with a disturbance affecting several submodels. The reason is that the disturbances are included as independent variables in the surrogate model generation (see Eqs. (13.19) and (13.2)). Therefore, the connection of the submodels for optimization will provide to all surrogate models the same disturbance value.

One the other hand, manipulated variables $\mathbf{u}_i^*$, which affect several submodels, have to be handled more carefully. Due to the variable transformation, it is not possible to calculate self-optimizing variables for these manipulated variables in each submodel. Instead, the manipulated variables must be assigned to one submodel. In addition, it is required in this submodel to fit a surrogate model

$$\mathbf{g}'_{u_i} : \{\mathbf{c}_i, \tilde{\mathbf{d}}_i\} \mapsto \mathbf{u}_i^* \tag{13.47}$$

to calculate the value of $\mathbf{u}_i^*$. This value is then used as connection variable in the other submodels that contain $\mathbf{u}_i^*$ as independent variable.

It has to be noted that it is more common in chemical process that disturbances affect several submodels. Furthermore, the developer of the surrogate models can decide to include all unit operations with the same manipulated variables $\mathbf{u}_i^*$ into one submodel. Then, it is not necessary to fit a surrogate model (13.47) to the manipulated variables $\mathbf{u}_i^*$.

13.6 Conclusion

Combining principles from control theory and surrogate modelling, a new method was developed to simplify the structure of surrogate models. The main idea is to replace the original independent variables $\mathbf{u}_i$ by a better set $\mathbf{c}_i$ using the approach of self-optimizing control. This is caused by omitting regions in which the submodel is suboptimal, for example because a reactor is extinct. In addition, it may in some cases result in fewer independent variables. The proposed method is independent of the structure of the surrogate model. Hence, it is possible to combine it with other approaches in the literature for the calculation of surrogate models, *e.g.* Kriging models and the ALAMO approach.

Chapter 14

Sampling for Surrogate Model Using Partial Least Squares Regression

So far, the sampling of points for the surrogate model was based on a fixed number of sampling points n_p. The sampling of points from the detailed model has, in addition to the fitting of the surrogate model, the main computational cost. Hence, the aim of sampling is to sample as few points as possible while achieving satisfactory performance of the surrogate model. The overall concept is called *design of computer experiments*. Garud et al. [38] provide an extensive review of the different sampling approaches. They can be differentiated between predefined (static) and adaptive sampling. In the former, the sampling points are generated and sampled in one iteration, whereas in the latter the performance of the surrogate model affects the placement of the new sampling points.

Predefined (static) sampling is the simplest approach. Monte Carlo sampling [70] is an early method based on pseudo-random numbers. The key idea of Monte Carlo sampling is that the randomness in sampling will result in space filling. This is however not guaranteed and may result in a large number of sampling points n_p.

Hence, space-filling methods are considered instead. The simplest space-filling method is regular grid sampling. It is applied for surrogate modeling [40], but it has an exponential increase in sampling points

$$n_p = n_g^{n_u} \tag{14.1}$$

where n_g is the number of points per dimension in the regular grid. Therefore it is only useful for a small number of independent variables n_u.

Several other methods have been developed to overcome this so-called *curse of dimensionality*. Latin hypercube sampling (LHS) [69] is probably the most popular method

today. It is applied by *e.g.* Ochoa-Estopier et al. [81] for a heat-integrated crude oil distillation system for $n_u = 10$ and $n_p = 3000$. It may however not explore the whole space as shown for a simple 2-dimensional case study by Garud et al. [38]. Independent of the chosen sampling method, static approaches have in addition the inherent problem of selecting how many points to sample. This can lead to undersampling or oversampling.

This can be alleviated by incremental sampling with evaluation of the surrogate model fit. Nuchitprasittichai and Cremaschi [80] use such an incremental approach based on LHS. Surrogate models are fitted after each additional sampling step and the procedure is stopped after reaching a termination criteria based on boots trapping. Quirante and Caballero [84] used the maxmin approach in which the points are placed so that the minimum distance between sampling points is maximized. Depending on the performance of the surrogate model, more points are sampled, again using the maxmin approach.

Adaptive sampling methods were developed as an alternative to the static approaches to overcome the problem of oversampling. They are generally based on two concepts, exploration and exploitation [38]. While the former tries to achieve point placement in regions which are poorly represented in the sampling space, the latter utilizes the fitting of a surrogate model to sample in highly nonlinear regions.

The smart sampling algorithm developed by Garud and co-workers is one of the adaptive sampling methods [37, 39]. Through the application of two metrics, one for exploitation and one for exploration, they identify new optimal points. Cozad et al. [20] developed a combined surrogate model fitting and sampling algorithm which aims at sampling points which have a maximum error with the surrogate model. The resulting surrogate models have a simple structure allowing the easy calculation of derivatives. Eason and Cremaschi [24] combined space filling through incremental LHS with exploitation through jackknifing.

As mentioned, static sampling has the potential problem of under- or oversampling. The incremental approach of Nuchitprasittichai and Cremaschi [80] and Quirante and Caballero [84] alleviates this problem, but adds the cost of fitting a surrogate model at each step . In addition, one also has to make a choice for the basis functions. Therefore, the development of a termination criteria for incremental sampling without the need of fitting a surrogate model is attractive, and the focus of this chapter. One possible approach is to apply partial least squares regression (PLSR), which has a very low computational cost, and use this as a termination criteria.

PLSR is a method from chemometrics, developed for the analysis of high-dimensional data. Chapter 11 and Chapter 12 apply PLSR in the calculation of surrogate models to reduce the number of independent variables n_u in the fitting through the introduction of latent variables. The new latent variables $\mathbf{u}'$ are calculated using the weights $\mathbf{W}$ given

by PLSR. In this chapter, we use this information instead as a termination criteria for sampling without the need to fit a surrogate model. PLSR is explained in Section 11.1.

This chapter is structured as follows; Section 14.1 describes the developed procedure for sampling of surrogate models without the necessity of fitting a surrogate model in each iteration. Section 14.2 illustrates the steps in the procedure using a simple pipe model as motivating example. Section 14.3 applies this method to two case studies, the reaction and the separation sections of a simplified ammonia synthesis reactor. These two submodels are then combined with the original synthesis gas makeup section for the respective submodels and evaluated in comparison to the original model. Section 14.4 then discusses the properties of the proposed method.

14.1 Proposed Sampling Procedure Utilizing PLSR

The idea is to compute the weight matrix $\mathbf{W}$ after each sampling, or after a block n_{add} of samplings, and consider the convergence of $\mathbf{W}^k$. This may be done by monitoring how the norm of the difference

$$\Delta\mathbf{W}^k = \mathbf{W}^k - \mathbf{W}^{k-1} \tag{14.2}$$

depends on the iteration k. The norm, $\left\|\Delta\mathbf{W}^k\right\|$, can be utilized as termination criteria for the sampling procedure. However, $\mathbf{W}$ should only include the weights corresponding to the important latent variables by setting a threshold β. This results in defining

$$\mathbf{W}^k = \begin{bmatrix}\mathbf{w}_1^k & \cdots & \mathbf{w}_{n_s}^k\end{bmatrix} \tag{14.3}$$

where the omitted weight vector $\mathbf{w}_{n_s+1}^k$ explains less than β % of the variance of the dependent variable y. The value n_s corresponds therefore to the number of significant weights.

The initialization of the procedure consists of sampling n_{ini} points. PLSR is then applied to calculate the initial weights $\mathbf{W}^1$. In the subsequent iterative procedure, n_{add} points are sampled at each iteration step k. This corresponds to a so-called *arithmetic sampling*, as defined by Provost et al. [83], and can be written as

$$n_p(k) = n_{ini} + k \cdot n_{add} \tag{14.4}$$

The sampling of the points can be performed using any method, *e.g.* Latin hypercube sampling [69] or Sobol sampling [95]. Subsequently, the weights of the latent variables $\mathbf{W}^k$ are calculated. Using the explained variance, we calculate the difference $\Delta\mathbf{W}^k$ of the significant weights to the previous calculated weights and its norm $\left\|\Delta\mathbf{W}^k\right\|_F$.

Although the norm converges, it can temporarily increase and decrease. This *noise* may terminate the procedure before reaching a satisfactory performance. To avoid a preemptive termination, we propose to average the norm of the last n_f steps resulting in the

Algorithm 3 Sampling procedure.

1: For a given subprocess $\mathbf{g}$ with independent variables $\mathbf{u} \in \mathbb{R}^{n_u}$ and dependent variables $\mathbf{y} \in \mathbb{R}^{n_y}$, define upper and lower bounds for the independent variables.
2: Sample n_{ini} initial points.
3: Select the threshold β and calculate $\mathbf{W}^1$ according to Eq. (14.3).
4: Initialize with $k = 2$.
5: **while** $\left\|\Delta\mathbf{W}^k\right\|_F^{av} > \gamma$ **do**
6: Sample n_{add} additional points.
7: Perform PLS regression.
8: Obtain the number of significant weights n_s using the selected β and calculate $\Delta\mathbf{W}^k$ according to Eq. (14.2) and Eq. (14.3).
9: Calculate the averaged norm $\left\|\Delta\mathbf{W}^k\right\|_F^{av}$ in Eq. (14.5).
10: Set the iteration number $k = k + 1$.
11: **end while**
12: Fit the surrogate models.

calculation of the averaged norm

$$\left\|\Delta\mathbf{W}^k\right\|_F^{av} = \frac{\sum_{l=k-n_f+1}^{k} \left\|\Delta\mathbf{W}^l\right\|_F}{n_f} \tag{14.5}$$

The averaged norm is compared to a threshold γ and, if it is below γ, the iterative procedure is stopped and a surrogate model is fitted to the sampling space. Algorithm 3 summarizes the procedure. The reason behind choosing the Frobenius norm is discussed in Section 14.4. In the case of multiple dependent variables $\mathbf{y}$, it is either possible to perform PLS regression for all dependent variables independently or simultaneously. The former is computationally more expensive. If the latent variables are used to fit the surrogate model, it was advised in Chapter 11 to perform PLS regression independently. However, we are looking at the differences and do not use the latent variables for the fit of the surrogate model. We therefore use the simultaneous approach. This will be further discussed in the case studies in Section 14.3.

Table 14.1: Parameters of the pipe case study.

Parameter	L/D [-]	A [m^2]	f [-]
Value	8.8×10^4	0.2	0.003

14.2 Description of the Sampling Procedure

The sampling procedure will now be explained in detail using the pressure drop over an isothermal pipe, as used in Chapter 11, as motivating example. The independent variables are the inlet pressure p_{in}, the temperature T, and the inlet molar flows $\dot{n}_{in,i}$. The dependent variable is the outlet pressure p_{out}. The model is

$$0 = \left(p_{in}^2 - p_{out}^2\right) - 4f\frac{L}{D}\frac{RT\overline{M}}{A^2}\dot{n}_{in}^2 \tag{14.6}$$

This model allows for changing number of independent variables n_u through changing the number of gas components n_{gas} in the stream. These influence the average molar mass

$$\overline{M} = \frac{\sum_{i=1}^{n_{gas}} M_i \dot{n}_{in,i}}{\dot{n}_{in}} \tag{14.7}$$

and the total flow

$$\dot{n}_{in} = \sum_{i=1}^{n_{gas}} \dot{n}_{in,i} \tag{14.8}$$

We have as input $\mathbf{u} = \begin{bmatrix} p_{in} & T & \dot{\mathbf{n}}_{in}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$. The investigated case has 5 gas components ($i = H_2$, N_2, NH_3, Ar, and CH_4) resulting in $n_u = 7$. One surrogate model has to be fitted for the pressure difference $y = p_{in} - p_{out}$ ($n_y = 1$). Molar fractions x_i are used as independent variables in the fitting of the surrogate model and calculation of the PLSR weights. The data of the pipe are given in Table 14.1. The nominal value and the bounds (lower and upper bound) of the grid can be found in Table 14.2. Table 14.3 gives the parameters for the sampling procedure (Algorithm 3).

Table 14.2: Upper and lower bounds and the nominal value of the independent variables ($\mathbf{u}$) (pipe model).

	p_{in} [bar]	T [°C]	$\dot{n}_{H_2,in}$ [mol/s]	$\dot{n}_{N_2,in}$ [mol/s]	$\dot{n}_{NH_3,in}$ [mol/s]	$\dot{n}_{Ar,in}$ [mol/s]	$\dot{n}_{CH_4,in}$ [mol/s]
Lower Bound	23	0	700	230	50	10	10
Nominal Value	27	10	1400	460	100	20	20
Upper Bound	31	20	2100	690	150	30	30

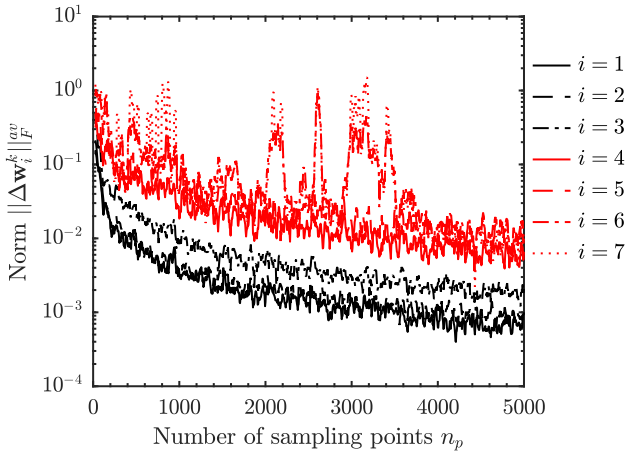


Figure 14.1: Development of the norm of the weights, $\left|\left|\Delta \mathbf{w}_i^k\right|\right|_F^{av}$ (pipe model).

14.2.1 Evaluation of the norm of the weights

We only include the the significant weights $\mathbf{w}_i$ in $\mathbf{W}$, see Eq. (14.3). To understand this better, Figure 14.1 shows the convergence of all the seven weights $\mathbf{w}_i$ for an increasing sampling space $n_p(k)$. Note the log scale for the norm. For illustration purposes, we oversample using 5000 points sampled as a Latin hypercube. PLSR was performed every 5 sampling points ($n_{add} = 5$) after initialization with 25 sampling points. The last 5 calculated norms were used for filtering ($n_f = 5$). The colour code shows the three significant weights (black) and the four weights with an explained variance smaller than $\beta = 2$ % (red). As we can see, all weights are converging. However, it is possible to see a clear difference between the significant and insignificant weights. $\mathbf{w}_1$ and $\mathbf{w}_2$ are similar in convergence and hard to distinguish. $\mathbf{w}_3$ is converging at a slightly slower rate and has a value in-between the significant and insignificant weights. The insignificant weights converge at a much slower rate. Especially $\mathbf{w}_6$ and $\mathbf{w}_7$ experience frequent changes in the norm resulting in noisy bumps, even with the applied filtering.

It has to be noted that the number of significant weights n_s (with $\beta \geq 2$ %) decreases with increasing n_p in this case study. Initially, $\mathbf{w}_{4,k}$ explains between 2 % and 4 % of the

Table 14.3: Tuning parameters of the proposed sampling method (all case studies).

Parameter	n_{ini}	n_{add}	n_f	β	γ
Value	25	5	5	2 %	$n_{add} \times 10^{-2}$

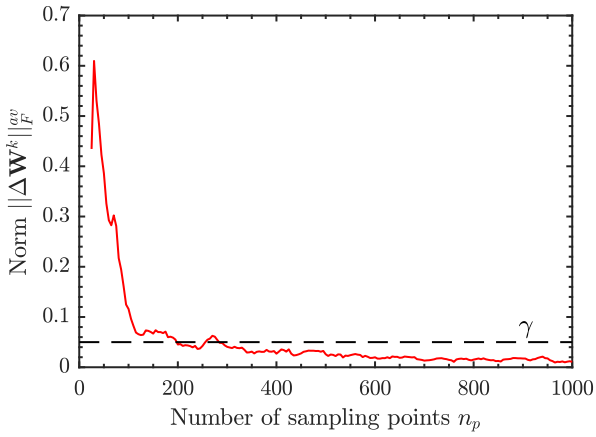


Figure 14.2: Development of the averaged norm of the combined weight matrix of the significant weights, $\left|\left|\Delta \mathbf{W}^k\right|\right|_F^{av}$ (pipe model).

variance in y, so $n_s = 4$. However, it settles to around 0.5 % after around 300 sampled points, giving $n_s = 3$. As a result, the number of significant weights n_s can change in the course of the sampling.

Figure 14.2 shows how the important combined averaged norm of the significant weights $\left|\left|\Delta \mathbf{W}^k\right|\right|_F^{av}$ changes for the first 1000 sampling points, but here using a linear scale for the norm. As we can see, the reduction in the norm is especially pronounced in the first 100 to 150 sampling points and is less pronounced with increasing sampling points. This threshold $\gamma = 0.05$ is reached after 200 sampling points. The norm of the differences of the combined significant weights, $\left|\left|\Delta \mathbf{W}^k\right|\right|_F^{av}$, is less susceptible to the *noise* in the calculation compared to the individual weights shown in Figure 14.1. Hence, it is not necessary to use a large n_f for filtering.

14.2.2 Error of the surrogate model

We found in Figure 14.2 that the significant weights $\mathbf{W}^k$ converge after about 200-300 sampling points. How does this reduction correspond to the accuracy of a fitted surrogate model?

To this end,we investigate the correlation between the norm of the difference, $\left|\left|\Delta \mathbf{W}^k\right|\right|_F^{av}$, and the performance of the surrogate model. The surrogate model structure is a 2-layer cascade forward neural network with 5 hidden neurons in each layer. The surrogate models were fitted after each 5 additional points starting at initially 25 points. After

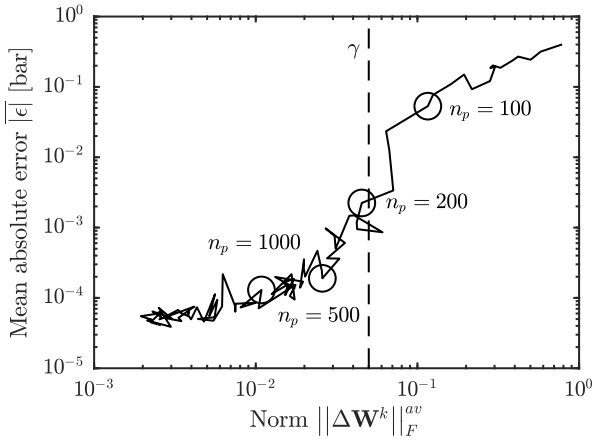


Figure 14.3: Mean absolute error of the surrogate model $\overline{|\varepsilon|}$ as function of the averaged Frobenius norm of the significant weights $\mathbf{W}^k$ (pipe model).

100 sampled points, the interval is increased to every 25 points and to every 100 points after 1000 sampled points. Each time, 10 neural networks were fitted to average the randomness in the initial seed to the neural networks. The dependent variable of the surrogate model fit ($y = p_{in} - p_{out}$) is then calculated as the average of the 10 neural networks. The validation space is given by 10^4 randomly sampled points. Note that the neural networks were not fitted to the latent variables $\mathbf{u}'$, but to the initial independent variables $\mathbf{u}$. This is different to the results reported in Chapters 11 and 12.

Figure 14.3 shows the mean absolute error $\overline{|\varepsilon|}$ of the pressure difference $y = p_{in} - p_{out}$ as a function of $\left\|\mathbf{W}^k\right\|_F^{av}$. The threshold $\gamma = 0.05$ used in the previous section is also shown. From this figure, where we used log-scale for $\overline{|\varepsilon|}$, we see that sampling more than 1000 points does not reduce the error further. Increasing the sampling space above $n_p \approx 300 - 500$ only marginally reduces the error in the fitted neural network. This corresponds to the concept of learning curves as described by Provost et al. [83] which says that an increase in sampling points does not improve the accuracy of the surrogate model. The threshold γ corresponds to the point in which the decrease in the averaged norm $\left\|\Delta\mathbf{W}^k\right\|_F^{av}$ in Figure 14.2 flattens and is at

$$\left\|\Delta\mathbf{W}^k\right\|_F^{av} \approx 0.02\text{-}0.05 \tag{14.9}$$

Since we want to avoid the fitting of the surrogate model (neural networks) during the sampling, this can be used as the termination criteria in the sampling for surrogate model generation

14.2.3 Results of the Applied Procedure

The application of the method with the tuning parameters given in Table 14.3 and Latin hypercube sampling for the calculation of new sampling points results in a termination after 210 sampled points. This is similar to the previous oversampling shown in Figure 14.2. Here, the threshold γ is crossed after 200 sampling points. The resulting surrogate model shows a maximum absolute error of 0.07 bar and an average absolute error of 1×10^{-3} bar. The 3 significant weights explain 94.30 % of the variance in the dependent variable $y = p_{in} - p_{out}$. All 7 weights explain in total only 94.45 % of the variance in the dependent variable due to the nonlinearity of the pipe model. Consequently, the insignificant weights explain combined only 0.15 % of the variance in y. The high maximum absolute error is caused by neglecting the corner points of the independent variables, *i.e.* the points given by constructing a 2-point regular grid using the bounds in Table 14.2. Hence, the surrogate model is extrapolating close to the corners. With $n_u = 7$, it would be possible to incorporate the corner points as they only correspond to $2^7 = 128$ points. However, if $n_u > 10$, the incorporation of the corner points would require a large number of sampled points. In this situation, it is best to apply the surrogate model first. If necessary, it is then possible to add only the corner points in which the subsequent application of the surrogate model is moving. This reduces the points which one has to sample.

14.3 Ammonia Synthesis Loop Case Studies

So far, the method was applied to a single case in which $n_y = 1$. Now, two additional case studies are used for testing the sampling procedure with $n_y > 1$ and evaluate, whether similar conclusion can be drawn. Both case studies are part of the ammonia synthesis loop shown in Figure 14.4. The first case study is the reaction section (marked red) as previously used in Chapters 11 and 12. The second case study is looking at the separation section (marked green) of the same synthesis loop.

The maximum absolute error $\max|\varepsilon|$ and the root mean squared error (RMSE)

$$\mathrm{RMSE}_i = \frac{\sum_1^{n_{val}} \left(y_{surr,i} - y_{val,i}\right)^2}{n_{val}} \tag{14.10}$$

are used to assess the performance of the surrogate models. In addition, the relative error is calculated using the range of the dependent variables of the validation space, $\mathbf{y}_{val}$, *i.e.*

$$\varepsilon_{i,rel} = \frac{\varepsilon_i}{\max \mathbf{y}_{val,i} - \min \mathbf{y}_{val,i}} \tag{14.11}$$

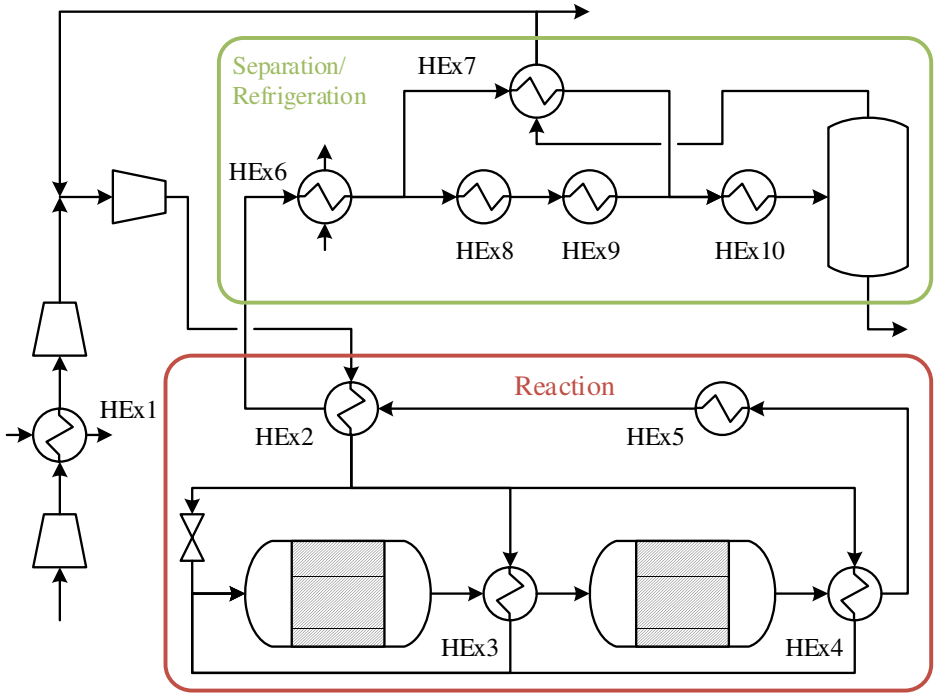


Figure 14.4: Ammonia synthesis loop with the submodels *Reaction Section* and *Separation Section*.

14.3.1 Reaction Section of an Ammonia Synthesis Loop

The reaction section of the ammonia synthesis loop is applied in Chapters 11 and 12 in the introduction of new latent variables $\mathbf{u}'$. It is connected to the compressor train and the separation section through the overall mass recycle. It consists of two consecutive reactor beds with interstage heat integration (HEx3). Furthermore, the reaction heat is used for the generation of high pressure stream (HEx5) and heating the inlet flow to the first bed (HEx2 and HEx4). It is shown in Figure 14.4. Note, that the labeling of the heat exchangers is different than in Chapers 11 and 12.

Model Description

The model has 10 independent variables ($\mathbf{u}$): the inlet pressure p_{in}, the inlet temperature T_{in}, 5 inlet molar flows $\dot{n}_{i,in}$ (H_2, N_2, NH_3, Ar, and CH_4), 2 split ratios, and the outlet temperature of the steam generation heat exchanger 5, $T_{HEx5,out}$. There are 4 dependent variables ($\mathbf{y}$): the pressure drop Δp, the temperature change ΔT, the rate of extent of

reaction $\dot{\xi}$, and the duty of heat exchanger 5, Q_{HEx5}. It is possible to define exact mass balances using $\dot{\xi}$ and the stoichometric coefficients ν_i

$$\dot{n}_{i,out} = \dot{n}_{i,in} + \nu_i \dot{\xi} \tag{14.12}$$

In the previous application of the model for surrogate model generation in Chapters 11 and 12, a 2-point regular grid and 5000 points defining a Latin hypercube were used. This resulted in reasonable errors for the dependent variables Δp, ΔT, and $\dot{\xi} s$ through the introduction of latent variables $\mathbf{u}'$. The duty of the heat exchanger is a new dependent variable. The 2-point regular grid corresponds already to $2^{10} = 1024$ sampling points, but we want to see if we can terminate the sampling with even fewer points.

The upper and lower bounds of the sampling grid can be found in Table 14.4. Mole fractions x_i and the total molar flow $\dot{n}_{in}$ are used as independent variables in the surrogate model generation and application of PLSR instead of the molar flows $\dot{n}_{i,in}$. This requires omitting the mole fraction of hydrogen. Furthermore, the molar ratio H_2/N_2 is used instead of the mole fraction of nitrogen as independent variable in surrogate model fitting.

The surrogate model structure is a two layer cascade forward neural network with 5 hidden neurons in each layer. The validation space consists of $n_{val} = 10^4$ randomly sampled points.

Table 14.4: Upper and lower bounds of the independent variables (**u**) (reaction section).

Variable	Unit	Lower Bound	Upper Bound
p_{in}	[bar]	-10	$+10$
T	[K]	-20	$+20$
$\dot{n}_{\mathrm{H_2},in}$	[%]	-20	$+20$
$\frac{\dot{n}_{\mathrm{H_2},in}}{\dot{n}_{\mathrm{N_2},in}}$	[%]	-10	$+10$
$\dot{n}_{\mathrm{NH_3},in}$	[%]	-20	$+20$
$\dot{n}_{\mathrm{Ar},in}$	[%]	-20	$+20$
$\dot{n}_{\mathrm{CH_4},in}$	[%]	-20	$+20$
$T_{HEx5,out}$	[K]	-20	$+20$
Split Ratio 1	[pp]	-5	$+5$
Split Ratio 2	[pp]	-20	$+20$

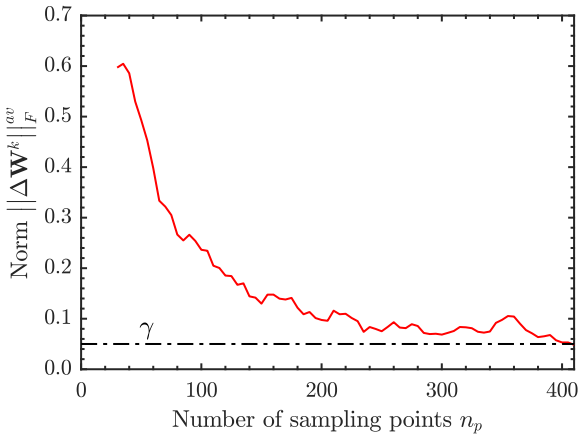


Figure 14.5: Development of the Frobenius norm $\left|\left|\Delta \mathbf{W}^k\right|\right|$ as a function of the number of sampling points with $\gamma = 5 \times 10^{-2}$ (reaction section).

Results

The data for the proposed sampling procedure are the same as in the pipe case study, see Table 14.3. PLSR was applied to all dependent variables simultaneously with $\gamma = 0.05$. The sampling procedure terminated after 410 sampled points. This corresponds to a regular grid with 1.8 points for each dependent variable. Figure 14.5 shows the evaluation of the norm of the significant weights. Similar to the pipe section, we can observe a steep decrease in $\left|\left|\Delta \mathbf{W}^k\right|\right|_F^{av}$ for the first 100 sample points. This decrease is reduced with an increasing sampling space. We have $n_s = 5$ weights in $\mathbf{W}$ which explain more than 98.34 % of the variance in the dependent variables $\mathbf{y}$ after 410 sampling points. $\mathbf{w}_6$ explains only 0.6 % of the variance in $\mathbf{y}$. During the sampling procedure, n_s changed twice in the first 100 points but remained constant at $n_s = 5$ from 100 points onwards.

Repeating the sampling procedure 10 times, results in a mean number of sampling points $\overline{n_p} = 382.5$ with a standard deviation of $s = 27.2$. This shows that the proposed sampling procedure is consistent in its termination. The variation in the number of sampling points is caused by the randomness in the new sampling points. The performance measures $\max|\varepsilon_i|$ and RMSE_i for the four dependent variables ($\mathbf{y}$) can be found in Table 14.5. Again, the corner points were not sampled. This results in extrapolation for certain values of the independent variables $\mathbf{y}$. The maximum absolute normalized error $\max\left|\varepsilon_{i,rel}\right|$ is 0.09 %, 0.26 %, 0.28 %, and 0.27 % for Δp, ΔT, $\dot{\xi}$, and Q_{HEx5} respectively.

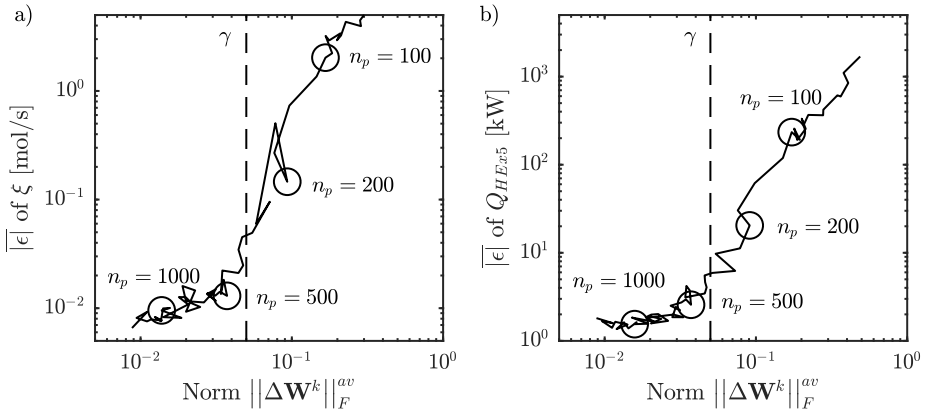


Figure 14.6: Mean absolute error for y_3 and y_4 of the surrogate model $\overline{|\varepsilon|}$ as function of the averaged Frobenius norm of the significant weights $\mathbf{W}^k$ (reaction section).

The chosen threshold

$$\gamma = n_{add} \times 10^{-2} = 5 \times 10^{-2} \tag{14.13}$$

was based on the threshold in the pipe case study. Hence, we want to analyze the the correlation of $\left\|\Delta \mathbf{W}\right\|_F^{av}$ with the surrogate model fit. 2000 points were sampled using Latin hypercube sampling and were used in the following analysis. 10 neural networks were fitted every 5 points from 25 to 100 points, every 25 points to 1000 points and subsequently every 100 points. The used value of the dependent variable in the calculation of the error is the average value of these 10 values. PLSR was applied to all dependent variables simultaneously. Figure 14.6 shows the mean absolute error for the dependent variables $\dot{\xi}$ and $QHEx5$ as a function of $\left\|\Delta \mathbf{W}\right\|_F^{av}$. The two dependent variables correspond to the variables with the highest maximum absolute relative error according to Eq. (14.11). If we compare Figure 14.6 (this case study) to Figure 14.3 (pipe model), we can directly see that the correlation between $\overline{|\varepsilon|}$ and $\left\|\Delta \mathbf{W}\right\|_F^{av}$ is similar. In both cases, increasing the number of sampling points does not improve the fit from a certain point onward and gives a similar threshold γ.

Table 14.5: Results for the dependent variables $\mathbf{y}$ (reaction section).

$\mathbf{y}$	Δp [mbar]	ΔT [mK]	$\dot{\xi}$ [mol/s]	Q_{HEx5} [kW]
$\max \lvert\varepsilon_i\rvert$	8.0	14.8	0.78	85.0
RMSE_i	0.5	1.0	0.03	7.2

14.3.2 Separation Section of an Ammonia Synthesis Loop

The task of the separation section of the ammonia synthesis loop is to separate ammonia from the reaction gas. This is achieved by several sequential and parallel heat exchangers followed by a separator. A heat exchanger using water as coolant (HEx6) cools the gas stream leaving the reaction section before it is split into two parallel heat exchanger trains. The first cooling train uses the gas stream leaving the separator for heat integration (HEx7) whereas the second cooling train uses liquid ammonia as refrigerator in two separate heat exchangers (HEx8 and HEx9). The two streams are subsequently mixed and cooled (HEx10) with liquid ammonia. Ammonia is then separated in a separator in which the liquid stream is considered as product stream and the gas stream is heat-integrated with the first parallel heat exchanger (HEx7).

Model Description

HEx6 and HEx7 are modelled using the Number of Transfer Units Method. HEx8, HEx9, and HEx10 are heat exchangers with fixed outlet temperatures $T_{HEx8,out}$, $T_{HEx9,out}$, and $T_{HEx10,out}$. The duties of the heat exchangers are calculated using the mass enthalpy of the gas streams as a function of the temperature, pressure, and composition. The mass enthalpy was calculated using a surrogate model based on cubic B-splines [41]. This surrogate model was fitted to points sampled in Aspen HYSYS. This is a simplified approach, but rather accurate. The separator is calculating the vapour-liquid equilibrium using Raoult's law for NH_3 and Henry's law for the other gas components [1]. It has to be noted that heat exchangers 8 and 9 are redundant in this model structure as heat exchanger 10 is cooling the stream to a fixed outlet temperature. However, in a real plant, the cooling in heat exchangers 8,9, and 10 is achieved using an ammonia refrigeration loop. The different heat exchangers correspond then to a liquid ammonia refrigerant at different pressure levels.

The separation section has 13 independent variables ($\mathbf{u}$). These are the inlet pressure p_{in}, the inlet temperature T_{in}, 5 molar flows $\dot{n}_{i,in}$ (H_2, N_2, NH_3, Ar, and CH_4), the inlet flow rate $\dot{n}_{H_2O,in}$ and temperature $T_{H_2O,in}$ of the cooling water in HEx6, 1 split ratio, and the outlet temperatures of the heat exchangers $T_{HEx8,out}$, $T_{HEx9,out}$, and $T_{HEx10,out}$. The 12 dependent variables ($\mathbf{y}$) are the stream variables of the two streams leaving the section (Δp, ΔT, and $\dot{n}_i$) corresponding to the product (subscript P) and the recycle (subscript R) stream, the temperature change of the water stream in heat exchanger 6, ΔT_{H_2O} , and the heat duties in the heat exchangers 8, 9, and 10 (Q_{HEx8}, Q_{HEx9}, and Q_{HEx10}. Note, that the temperature difference between the liquid outlet stream and the feed stream as dependent variable can be calculated using the two independent variables T_{in} and $T_{HEx10,out}$ as

$$\Delta T_P = T_{in} - T_{HEx10,out} \tag{14.14}$$

Table 14.6: Upper and lower bounds of the independent variables (**u**) (separation section).

Variable	Unit	Lower Bound	Upper Bound
p_{in}	[bar]	-10	$+10$
T	[K]	-25	$+25$
$\dot{n}_{\mathrm{H_2},in}$	[%]	-15	$+15$
$\frac{\dot{n}_{\mathrm{H_2},in}}{\dot{n}_{\mathrm{N_2},in}}$	[%]	-10	$+10$
$\dot{n}_{\mathrm{NH_3},in}$	[%]	-20	$+20$
$\dot{n}_{\mathrm{Ar},in}$	[%]	-20	$+20$
$\dot{n}_{\mathrm{CH_4},in}$	[%]	-20	$+20$
$\dot{n}_{\mathrm{H_2O},in}$	[%]	-20	$+20$
$T_{\mathrm{H_2O},in}$	[K]	-5	$+5$
$T_{HEx8,out}$	[K]	-4	$+4$
$T_{HEx9,out}$	[K]	-4	$+4$
$T_{HEx10,out}$	[K]	-8	$+8$
Split Ratio	[pp]	-5	$+5$

Chapter 11 proposes to use exact component mass balances to avoid the creation or destruction of mass through the introduction of surrogate models. This can be achieved through defining a separation factor α_i for each chemical component i:

$$\dot{n}_{i,rec} = \alpha_i \dot{n}_{i,in} \tag{14.15}$$

$$\dot{n}_{i,prod} = (1-\alpha_i)\,\dot{n}_{i,in} \tag{14.16}$$

Consequently, 12 surrogate models have to be fitted. The upper and lower bounds of the independent variables can be found in Table 14.6. The parameters used are the same as in the reaction section and for the pipe model. They are given in Table 14.3. The surrogate model structure is a 2-layer cascade forward neural network with 5 hidden neurons in each layer. The validation space consists of $n_{val} = 10^4$ randomly sampled points.

Results

We apply PLSR to all dependent variables **y** simultaneously because with $n_y = 12$, it is computationally expensive to perform PLSR independently. The method terminated after $n_p = 635$, corresponding to 1.6 points in a regular grid. This is a similar number of points in a regular grid as in the reaction section. With $\beta = 2$ %, we find that $n_s = 5$ weights explain 90.46 % of the variance in the dependent variables **y**. The neglected $\mathbf{w}_6$ explains 1.83 % whereas $\mathbf{w}_7$ explains 0.78 %. Figure 14.7 shows the evaluation of the filtered norm for the simultaneous approach. We can see a jump in the norm at about

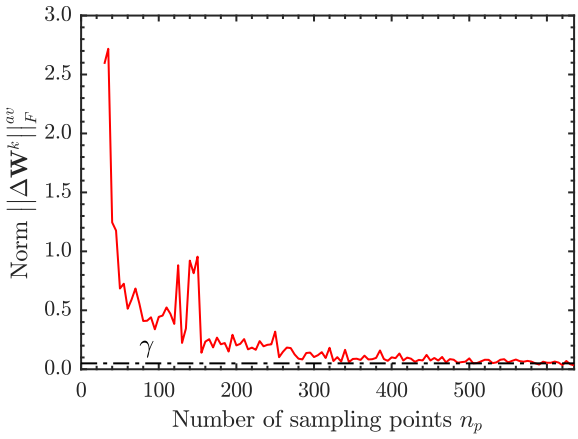


Figure 14.7: Development of the Frobenius norm $\left|\left|\Delta \mathbf{W}^k\right|\right|$ as a function of the number of sampling points with $\gamma = 5 \times 10^{-2}$ (separation section).

100 points corresponding to an increase in n_s from 6 to 7. After 250 sampled points, n_s is further reduced to the final 5 significant weights.

The error performance measures $\max|\varepsilon_i|$ and RMSE_i in the corresponding surrogate model fit can be found in Table 14.7. As the splitting factors for H_2, N_2, Ar, and CH_4 are all around 99 %, it is not useful to calculate the error directly. Hence, their errors are calculated as the error in the recycle stream $\dot{n}_{i,rec}$. Normalizing the error according to Eq. (14.11) results in a maximum absolute normalized error of around 0.1 % for the first 9 dependent variables in Table 14.7. The last three variables (heat exchanger duties) have however a normalized error of around 1 %. This can be explained by the phase change occurring in the heat exchangers through the condensation of ammonia. This phase change is not captured perfectly using the surrogate model approach. Applying the method 10 times gives an average number of sampling points of $n_p = 639.5$ and a standard deviation $s = 51.2$.

Table 14.7: Results for the dependent variables **y** (separation section).

y	Unit ε	max $\lvert\varepsilon_i\rvert$	RMSE$_i$
Δp_R	[mbar]	0.98	0.07
ΔT_R	[K]	0.130	0.013
Δp_P	[mbar]	2.42	0.14
$\Delta T_{\mathrm{H_2O},out}$	[mK]	2.0	0.3
$\alpha_{\mathrm{H_2}}$	[mmol/s]	0.6	0.06
$\alpha_{\mathrm{N_2}}$	[mmol/s]	0.18	0.02
$\alpha_{\mathrm{NH_3}}$	[mmol/s]	54.5	4.6
α_{Ar}	[mmol/s]	0.99	0.09
$\alpha_{\mathrm{CH_4}}$	[mmol/s]	0.65	0.04
Q_{HEx8}	[kW]	231	24
Q_{HEx9}	[kW]	71	7
Q_{HEx10}	[kW]	94	12

14.3.3 Combination of Surrogate Models for Optimization

So far, we fitted individual surrogate models to the reaction and separation section. The validation error of the surrogate model is small. However, in the real process the two models are combined and have a recycle (Figure 14.4), and a good individual fit does not guarantee that the combined model converges to the same optimum as the detail model. The reaction and the separation section are therefore combined with the models of the purge split and the compressor train to form the flowsheet in Figure 14.4 which has 9 operational degree of freedom.

The considered cost function which should be minimized is

$$\begin{aligned} J = &- p_P \dot{n}_P - p_{purge} \dot{n}_{Purge} - p_S Q_{HEx5} \\ &+ p_{feed} \dot{n}_{feed} \\ &+ p_C \left(Q_{Comp1} + Q_{Comp2} + Q_{Comp3}\right) \\ &+ p_{HEx} \left(Q_{HEx8} + Q_{HEx9} + Q_{HEx10}\right) \end{aligned} \tag{14.17}$$

The prices for the feed, product, and purge stream as well as the compressor duties are adopted from [7] with $p_{feed} = 0.704$ \$/kmol, $p_P = 3.4$ \$/kmol, $p_{purge} = 0.0112$ \$/kmol, and $p_C = 0.072$ \$/kW. The heat duty in heat exchanger 5 has a cost term of $p_S = 0.036$ \$/kW whereas the cooling in heat exchangers 8, 9, and 10 has a cost term of $p_{HEx} = 0.027$ \$/kW. The cooling water flow and temperature to heat exchangers 1 and 6 are considered to be at a fixed value.

The operational constraints are given by the bounds in the decision variables for surrogate model generation. In addition, there are bounds on the purge split ratio and the com-

pressor circumferential velocity. The duties of heat exchanger 8 and 9 may be different between the surrogate model and the original model. This is caused by the redundancy of both heat exchangers.

Both the original model and the surrogate-based model are subsequently optimized for a given feed. The results are very similar. 8 degrees of freedom are at constrained operation. The compressor speed is unconstrained and the error with respect to the original model is 0.59 %. The resulting relative error in the cost function is 0.14 %.

14.4 Discussion

The proposed method does not require the fitting of a surrogate model. In this respect, it differs to the ALAMO approach [20], the smart sampling algorithm [37], and the adaptive sampling approach of Eason and Cremaschi [24]. The computational expenses are hence reduced if n_y or n_u is large or the fitting of the surrogate model is computational expensive. A further advantage is that the decision about the surrogate model basis function is separated from the sampling. This allows to choose the best basis function based on the sampled space. There are however certain points which have to be addressed.

14.4.1 Choice of Tuning Parameters

Several tuning parameters have to be decided. The most important tuning parameter is the threshold γ of the norm as it serves as termination criteria. It is possible to continue the procedure by lowering the threshold γ, if one is not satisfied with the performance of the surrogate model. Hence, a higher threshold may be convenient to avoid oversampling. In general, the nonlinearity of the response surface has a high influence on the required threshold. The results of the three case studies indicate however that a threshold of approximately 5×10^{-2} seems to work for several cases, if PLSR is performed after every fifth sampled point. The similar performance can be explained through the constraint of having weights with length 1. This allows the application of the same threshold for several cases.

A second important tuning parameter is the threshold β in the explained variance which is used to select the significant weights $\mathbf{w}_{i,k}$. Depending on the definition of the independent variables, this threshold may exclude the majority of the weights as discussed in Chapter 12. For example, if molar flows are used as independent variables in the pipe case study, then only 1 weight is significant. On the other hand, using mole fraction as independent variables results in the presented 3 significant weights. Furthermore, using a hard bound β may result in constant switching of n_s. This results in drastic changes in the norm as illustrated for $\Delta\mathbf{W}_{k,\dot{\xi}}$ in Figure 14.7. This was weakened in the presented case studies by using the minimum value of n_s of the last two steps. As an alternative, it is possible to choose n_s directly after sampling a certain number of points instead of

Table 14.8: Simultaneous *vs.* individual application of PLSR.

Case Study	Approach	$\overline{n_p}$	s
Reaction	Individual	368	28.7
	Simultaneous	382.5	27.2
Separation	Individual	664	52
	Simultaneous	639.5	51.2

choosing the threshold β. In all case studies, n_s did not change after a certain number of sampled points with a $\left\|\Delta\mathbf{W}\right\|_F^{av}$ still way larger than the threshold γ.

Further tuning parameters are the number of sampled points in each iteration, n_{add}, and the past horizon n_f for averaging the norm. It is advisable to have a small value of n_{add} to avoid problems in the calculation of the differences $\Delta\mathbf{W}_k$. However, if chosen too small, it can be that the sampling space is not properly filled. Provost et al. [83] proposed as an alternative to the arithmetic a geometric sampling approach. The number of sampled points increases with increasing step number in geometric sampling. They showed that the computational load to converge to the required number of sampling points is reduced as the termination criteria does not have to be evaluated as frequently. Applying this approach for the PLSR-based termination criteria can however be problematic as the method is relying on the difference in the loads. As the calculation of the weights is not computationally expensive, at least if applied simultaneously, the arithmetic approach used in this paper seems reasonable.

The past horizon n_f is important to remove problems with oscillatory behaviour of the norm. Oversampling can be the result if it is chosen too high. Hence, it should be limited. The value $n_f = 5$ used in the case studies seems to be a reasonable. It avoids oversampling while preventing preemptive termination of the sampling.

14.4.2 Simultaneous and Individual Application of PLSR

If $n_y > 1$, one has to decide whether PLSR is applied individually to each dependent variable y_i or simultaneously to all dependent variables. We used simultaneously in both case studies. The advantage of applying PLSR individually is that it is possible to see which of the dependent variables require the most sampling points.

Both ammonia case studies were repeated 10 times with individual application of PLSR to compare the performance of the sampling procedure and whether there is a difference if PLSR is applied simultaneously or individually. The resulting average number of sample points and standard deviations can be found in Table 14.8. The difference in the average number of sampling points is not significant in either case study. Hence, we

conclude that is advantageous to apply PLSR simultaneous to all dependent variables. This reduces the computational load in calculating the weights.

14.4.3 Choice of Norm

The choice of the norm is in general not very important. It only has an influence on the defined threshold. The 1-norm will correspond to the the 1-norm of the weight $\mathbf{w}_{n_s,k}$ as the difference is largest in the last significant weight. The contribution from the other weights $\mathbf{w}_{i,k}$ with $i < n_s$ are then neglected. As a result, the termination threshold has to be higher than in the case of other norms. The infinity norm on the other hand calculates the maximum absolute row sum. As the individual weights $\mathbf{w}_{i,k}$ are the columns of the matrix $\mathbf{W}_k$, this approach seems counter intuitive. However, the infinity norm looks at all weights $\mathbf{w}_{i,k}$ with $i \leq n_s$ compared to the 1-norm. The 2-norm and the Frobenius norm incorporate all entries in the difference $\Delta\mathbf{W}_{1:n_s,k}$. The Frobenius norm was eventually chosen due to the similarity of the Frobenius norm to the vector 2-norm and its performance in the application.

14.5 Conclusion

A new method for sampling for surrogate model was introduced. It incorporates a novel termination criteria to predict when sufficient points are sampled. This termination criteria is independent of the surrogate model basis functions and does not require the fitting of a surrogate model at each sampling step. This is advantageous if the fitting of the surrogate model is computational expensive and/or the number of dependent variables, n_y, is large. The case studies showed that the application of the termination criteria allows a reduction in sampling points compared to predefined sampling. For the ammonia process, the combination of the surrogate models with the compressor train of the original model resulted in very good results in the subsequent optimization.

Part IV

Closing Remarks

Chapter 15

Conclusion

The aim of this thesis was to investigate and develop methods for optimal operation of integrated chemical processes. As mentioned in the introduction, this includes both optimal operation of subprocesses and the development of methods for the optimization of integrated processes.

Optimal operation of subprocesses was investigated using an ammonia reactor as case study. This allows the comparison of the different applied approaches. Economic nonlinear model predictive control served as benchmark for the other methods. In its implementation, it is the fastest method to converge to the steady-state optimum while simultaneously considering the optimal trajectory.

The investigation of the effect of dependent disturbances was analyzed in the context of self-optimizing control. Neglecting the dependency of the disturbance on the input and the real disturbance to the system results in a new optimal selection matrix, even if the nominal inlet values to the subprocess are unchanged. However it was shown, that the optimal selection matrix calculated from the subprocess has a similar performance if the calculated setpoints are adjusted. This is necessary as the optimal operating point can be different. Adjusting the disturbance weighting matrix to the actual magnitude of the disturbances reduced the difference in loss further.

The hierarchical combination of self-optimizing control and extremum-seeking control is hence proposed. The self-optimizing controllers provide in this method a fast, close-to-optimal rejection of the disturbances. The extremum-seeking controllers then adjust the setpoints of the self-optimizing controllers to remove the steady-state loss in the case of persistent disturbances. A further advantage of this combined approach is the reduction of the impact of plant-model mismatch as the adjustment of the setpoint is performed using a model-free approach. The hierarchical implementation is possible due to a time

scale separation of both control structures.

Feedback real-time optimization as third investigated method is the translation of the optimization problem in conventional real-time optimization into a feedback problem. This reduces the computational load of the optimizing controller and allows the application in cases where the optimization problem is computational expensive. The application to the ammonia synthesis reactor showed that the method can be as well applied to a multivariable case. Its performance is better than the performance of extremum-seeking control while reducing the computational load compared to economic nonlinear model predictive control. It is however depending on the accuracy of the model, similarly to economic nonlinear model predictive control.

The application of optimal operation of subprocesses is depending on the possibility to define a local cost function. This cost function has to correspond to the overall cost function as otherwise the overall process is not at its optimum. Even if it may be possible to achieve optimal operation for certain subprocess, it remains challenging to achieve optimal operation for all subprocesses due to the problems associated with the local cost function. The application of surrogate models to subprocesses and subsequent optimization of the combined process using the surrogate models can be seen as one solution to this problem. Through the incorporation of recycle streams and computational expensive models in the surrogate models, it is possible to reduce the computational cost of the optimization problem. This may allow the application of optimal operation using detailed models. Several approaches in the field of surrogate model generation were developed.

First, the application of partial least square regression allows a reduction in independent variables through the introduction of latent variables. This is especially useful in the case of a large number of independent variables, as the computational cost of fitting surrogate models increases with an increasing number of independent variables. It reduces however the performance of the resulting surrogate model as information is lost about the previous independent variables. The introduction of exact mass balances in the method prevents the creation or destruction of mass.

Second, the response surface should be as simple as possible. The simpler the response surface, the less points have to be sampled to obtain satisfactory performance of the surrogate model. This is achieved through a variable transformation from the original independent variables to self-optimizing variables. Contrary to their application in self-optimizing control, all state variables can be used in the developed method.

Third, the sampling has a major influence on the performance of the surrogate model. Current sampling procedures may either result in oversampling or require the fitting of a surrogate model at each sampling step. Both can be prohibitive due to the computational expense in the case of a large number of independent variables. Partial least square re-

gression can be used as a computational cheap termination criteria for sampling to avoid oversampling and the fitting of surrogate models at each sampling iteration.

Chapter 16

Future Work

This chapter will discuss future research direction based on the results presented in this thesis. Future work can be conducted on both optimal operation of subprocesses as well as the application of surrogate models in the context of real-time optimization.

16.1 Optimal Operation of Subprocesses

Part II looked at the optimal operation of subprocesses with an ammonia reactor as case study. The studied methods all have their own advantages and disadvantages. The majority of these methods attract constant research interest for improvements in the convergence to the optimum and stability analysis of the resulting controllers. Hence, this will not be covered in detail.

The investigated case study has however one advantageous property, independently of the utilized method. It is possible to define a cost function which corresponds to the overall cost function when integrated in the ammonia synthesis loop. This is obviously not always the case. Chapter 6 discussed potential implications of other subprocesses in the ammonia synthesis loop. However, no direct conclusions can be drawn for other processes.

Hence an open research direction is the development of a method to assess the applicability of optimal operation to subprocesses. This can help in the decision what type of control structure should be applied. Similarly, the development of a method to assign economic costs to the connection streams may allow the utilization of the studied methods. In the context of real-time optimization, the prices assigned to different stream then have to be updated regularly. Both research direction may aid in the decision, whether it is useful to use optimal operation for subprocesses.

16.2 Optimal Operation through Introduction of Surrogate Models

Part III looked at the introduction of surrogate models in the context of optimization. Current research in the field of surrogate modelling is either focusing on the sampling or the utilized basis functions. Both fields have a large impact on the performance of the surrogate model independently of its application. This section will introduce future research directions in addition to the above mentioned research areas.

16.2.1 Selection of Splitting Streams

The splitting of the overall process into subprocesses allows the generation of surrogate models for said subprocesses. The optimization of a combined flowsheet based on the surrogate models reduces the computational costs, if these subprocesses incorporate recycle streams or are complex. The application of surrogate models in the literature is based on process knowledge. Frequently, surrogate models are fitted to *noisy* or complex unit operations.

The question remains which subprocesses or unit operations should be substituted by surrogate models. It is general advisable to split streams which do not have a lot of stream variables. However, there is no methodology for this problem. This reduces the number of independent and dependent variables in surrogate model generation. The development of a concise theory for which unit operations or subprocesses should be substituted is a future research direction The adaptation of partioning and tearing principles from sequential-modular flowsheeting software can be one approach to develop a theory for splitting of a process into subprocesses.

16.2.2 Sample Domain Definition

The definition of the sampling domain is crucial in the generation of surrogate models. In general, it is beneficial to sample only in regions we are interested in. As a result, less points have to be sampled and the response surface may be simpler. Chapter 12 investigated this influence in the case of two reacting species and concluded that it is necessary to identify dependencies in the inlet variables to improve the fit of the surrogate model. Similarly, the introduction of self-optimizing variables in Chapter 13 aims at sampling only the regions we are interested in to obtain a simpler response surface.

However, most variables are still sampled within fixed bounds. This approach neglects dependencies and increases the sampling space unnecessarily. This results frequently in more complicated response surfaces which require more sampling points. For example, if an inlet stream to a surrogate model is coming from a compressor, both the temperature and the pressure are not entirely independent. Similarly, the variation of the inlet molar flows results in the separation section in sampling mole fractions of ammonia which are

impossible to achieve due to the thermodynamical equilibrium. Hence, a future research direction may be the development of a method to improve the incorporation of process knowledge in the definition of the sampling grid. This method should preferably incorporate information obtained from the other subprocesses to simplify its application.

Appendices

Appendix A

Model Description for the Ammonia Reactor of the Brunsbüttel Ammonia Plant

This chapter is describing the model used in Chapters 4, 5, 6, 7, 8, and 13. The process model is similar to the one used by Morud and Skogestad in their analysis of the limit-cycle behaviour [75]. The modifications to the *core* process model as described below are given in the individual chapters. The process itself consists of three sequential reactor beds and is shown in Figure A.1. The feed (denoted by the subscript *in*) is split into four streams given by three split ratios. These split ratios correspond to the manipulated variables $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^{\mathsf{T}}$. One of the streams is heated through the reactor effluent in a heat exchanger to increase the inlet temperature of the first bed whereas the other three streams are *quench* (cooling) streams to the three reactor beds. This results into the positive feedback mentioned by Morud and Skogestad [75].

A.1 Model assumptions

In order to simplify the mathematical model, the following assumptions are made:

- there is no pressure drop in the system;
- the heat capacity of the streams are independent of composition and temperature;
- there is a perfect low level ratio controller controlling the split ratios;
- the change in the split ratios can be assumed to be instantaneous, and hence, dynamics for the valves do not need to be incorporated;

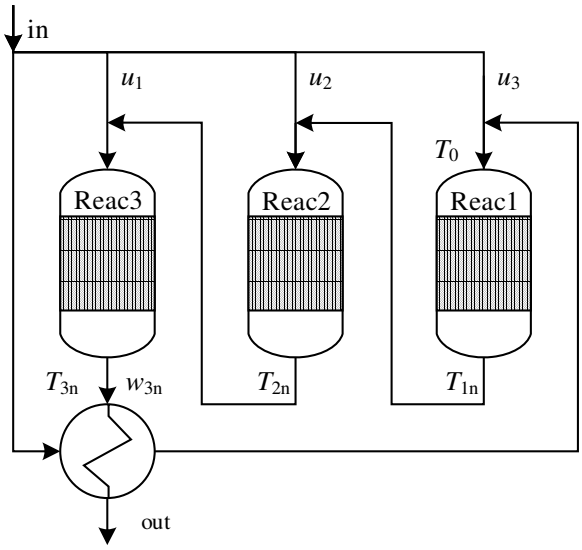


Figure A.1: Heat-integrated 3 bed reactor system of the ammonia synthesis gas loop.

- the reactor beds can be modelled as continuous stirred tank reactor (CSTR) cascade. This corresponds to a discretization of the partial differential equation of a plug-flow reactor along the x-axis;
- there is a time-scale separation between the changes in the concentration and the temperature. Hence, it is possible to assume that the concentration is at steady-state in the investigated reactor section.
- the gas hold-up in the sections of the bed is assumed to be constant;
- the mixing of the streams before the reactor beds is perfect;
- the heat capacity of the process gas is negligible compared to the heat capacity of the catalyst bed.

Based on the above mentioned assumptions, a differential algebraic formulation is proposed in the following sections. The considered disturbances are inlet disturbances

$$\mathbf{d} = \begin{bmatrix} \dot{m}_{in} & p_{in} & T_{in} & w_{\mathrm{NH_3},in} \end{bmatrix}^{\mathsf{T}} \tag{A.1}$$

The differential equations represent the temperature evolution within the sections of each reactor bed whereas the algebraic equations define the concentrations within the sections of the reactor beds. The definition of the states and parameters are given in the Tables A.1 and A.2.

A.2 Reactor model

The reaction rate as a function of the partial pressure $p_{i,j}$ in the CSTR reactor j of the cascade is given by the Temkin-Pyzhev equation as described by Froment [36].

$$r_{\mathrm{N_2},j} = k_{j,1} \frac{p_{\mathrm{N_2},j} p_{\mathrm{N_2},j}^{\frac{3}{2}}}{p_{\mathrm{NH_3},j}} - k_{j,-1} \frac{p_{\mathrm{NH_3},j}}{p_{\mathrm{H_2},j}^{\frac{3}{2}}} \tag{A.2}$$

in which the reaction constants $k_{j,\pm 1}$ are given by the Arrhenius equation:

$$k_{j,\pm 1} = A_{0,\pm 1} e^{-\frac{E_{a,\pm 1}}{R(T_j + 273.15)}} \tag{A.3}$$

It has to be noted, that in this model, the molar ratio of hydrogen to nitrogen is considered to be fixed at 3. This corresponds to the stoichiometric ratio in the reaction. As the only considered concentration state is given by the mass fraction of ammonia, w_i, the partial pressures have to be calculated from the mass fraction of ammonia. This reaction rate is written in [kmol N_2/ m^3_{cat} h] and hence, the reaction rate in [kg NH_3/kg_{cat} h] needed for the mass and temperature balances is then given by

$$r_{\mathrm{NH_3},j} = f r_{\mathrm{N_2},j} \frac{2 \times 17}{\rho_{cat}} \tag{A.4}$$

The reaction rate is multiplied with a factor of $f = 4.75$ to match plant data as explained by Morud and Skogestad [75]. The change in the temperature in each subsection of the reactor is then given by

$$\frac{dT_j}{dt} = \frac{c_{p,gas} \left(\dot{m}_{j-1} T_{j-1} - \dot{m}_j T_j \right) + m_{cat,j} r_{\mathrm{NH_3},j} \Delta H_{rx}}{m_{cat,j} c_{p,cat}} \tag{A.5}$$

whereas the component balances can be written as

$$0 = \dot{m}_{j-1} w_{\mathrm{NH_3},j-1} - \dot{m}_j w_{\mathrm{NH_3},j} + r_{\mathrm{NH_3},j} m_{cat,j} \tag{A.6}$$

Table A.1: Nomenclature of the states and decision variables.

State	Description	Lower bound	Upper bound	Unit
x	Temperatures **T**	200	600	°C
z	Mass fractions NH_3 $\mathbf{w}_{\mathrm{NH_3}}$	0	100	wt.%
u	Split ratios **u**	0	100	%

Table A.2: Nomenclature of parameters and calculated values.

Variable	Description	Value	Unit
$A_{0,+1}$	Arrhenius factor, forward	1.79×10^4	-
$A_{0,-1}$	Arrhenius factor, backward	2.57×10^{16}	-
$E_{a,+1}$	Activation Energy, forward	87,090	J/mol
$E_{a,-1}$	Activation Energy, backward	198,464	J/mol
R	Universal gas constant	8.314	J/mol/K
ρ_{cat}	Catalyst density	2,200	kg/m^3
$c_{p,gas}$	Gas heat capacity	3,500	J/kg/K
$c_{p,cat}$	Catalyst heat capacity	1,100	J/kg/K
$m_{cat,R1}$	Catalyst mass bed 1	14,718	kg
$m_{cat,R2}$	Catalyst mass bed 2	21,186	kg
$m_{cat,R3}$	Catalyst mass bed 3	33,440	kg
$m_{cat,j}$	Catalyst mass in volume j	depending	kg
$\dot{m}_j$	Mass flow in volume j	depending	kg/s
ΔH_{rx}	Heat of reaction	-2.7×10^6	J/kg NH_3
U	Heat transfer coefficient	536	W/m^2/K
A	Heat exchanger area	283	m^2

A.3 Heat exchanger model

The heat exchanger is modelled using the number of transfer units (NTU) method. In this method, the NTU and the ratio of the enthalpy (C^*) of the cold (subscript c, feed) and hot stream (subscript h, outlet bed 3) are calculated and based on these values, the effectiveness (ε) can be calculated in Eq. (A.9).

$$C^* = \frac{\dot{m}_c c_{p,gas}}{\dot{m}_h c_{p,gas}} \tag{A.7}$$

$$NTU = \frac{UA}{\dot{m}_c c_p} \tag{A.8}$$

$$\varepsilon = \frac{1 - e^{-NTU(1-C^*)}}{1 - C^* e^{-NTU(1-C^*)}} \tag{A.9}$$

This effectiveness corresponds to the percentage of the maximum achievable energy transfer Q as shown in Eq (A.10).

$$\begin{aligned} Q &= \varepsilon Q_{max} \\ &= \varepsilon \dot{m}_c c_{p,gas} (T_{in,h} - T_{in,c}) \end{aligned} \tag{A.10}$$

Due to the assumption of a constant heat capacity, the outlet temperatures of the heat exchanger are then given as

$$T_{out,h} = T_{in,h} - \frac{Q}{\dot{m}_h c_{p,gas}} \tag{A.11}$$

$$T_{out,c} = T_{in,c} + \frac{Q}{\dot{m}_c c_{p,gas}} \tag{A.12}$$

It has to be noted, that these model equations do not add a further differential of algebraic equation to the system. This is caused by the fact, that the equations define a relationship between the temperature of the first CSTR in the first bed and the outlet temperature of the last bed. As the inlet temperature of a bed is not a state, no additional algebraic equations are defined.

A.4 Stream mixing and general requirement

The equations of mixing two streams 1 and 2 for the temperature and concentrations are given as

$$T_{mix} = \frac{\dot{m}_1}{\dot{m}_1 + \dot{m}_2} T_1 + \frac{\dot{m}_2}{\dot{m}_1 + \dot{m}_2} T_2 \tag{A.13}$$

$$w_{\mathrm{NH_3},mix} = \frac{\dot{m}_1}{\dot{m}_1 + \dot{m}_2} w_1 + \frac{\dot{m}_2}{\dot{m}_1 + \dot{m}_2} w_2 \tag{A.14}$$

Similar to the heat exchanger equations, they do not increase the number of algebraic equations as they are purely defining relationships between the outlet states of the previous bed and the inlet of the following bed. The inlet temperature of bed 1 is in the following denoted as T_0 as it has an important influence on the occurrence of limit-cycle behaviour.

Due to mass conservation, the following inequality constraint for the decision variables $\mathbf{u}$ has to be fulfilled as well.

$$h(\mathbf{u}(t)) = \sum_i u_i(t) - 1 \leq 0 \tag{A.15}$$

A.5 Optimization of the system

The differential states are defined as the temperatures $\mathbf{x} = \mathbf{T}$, the algebraic states as the weight fractions $\mathbf{z} = \mathbf{w}$. Both type of states have a total of $3n$ input variables per time step in which n defines the number of discrete volumes in each of the 3 reactor beds (*vide supra*). The manipulated variables are 3 per time step and correspond to the split ratio to the inlets of the reactor beds. To summarize, we can write

$$\mathbf{x} \in \mathbb{R}^{3n} \tag{A.16}$$

$$\mathbf{z} \in \mathbb{R}^{3n} \tag{A.17}$$

$$\mathbf{u} \in \mathbb{R}^{3} \tag{A.18}$$

The corresponding non-linear problem constraints are given in semi-explicit representation by

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{d}(t), \mathbf{u}(t)) \\ \mathbf{0} &= \mathbf{g}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{d}(t), \mathbf{u}(t)) \\ 0 &\geq h(\mathbf{u}(t)) \end{aligned} \tag{A.19}$$

in which $\mathbf{f}$ corresponds to the differential equations of the temperature defined in Eq. (A.5), $\mathbf{g}$ to the algebraic equations defined in Eq. (A.6), and $h(\mathbf{u})$ to the input inequality defined in Eq. (A.15). Furthermore, bounds on the variables are defined in Table A.1. This system represents an index 1 differential algebraic system which can be verified by taking the total differential of $\mathbf{g}(\mathbf{x}, \mathbf{z}, \mathbf{u})$ given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{g}(\mathbf{x}, \mathbf{z}, \mathbf{u}) = \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{f} + \frac{\partial \mathbf{g}}{\partial \mathbf{z}}\dot{\mathbf{z}} + \frac{\partial \mathbf{g}}{\partial \mathbf{u}}\dot{\mathbf{u}} = \mathbf{0} \tag{A.20}$$

The optimisation was performed using CasADi [4] with IPOPT [102]. The optimal control problem (OCP) was solved *via* the direct collocation method [10] with RADAU order 3 as collocation points. The used integrator for the simulation is IDAS, which is part of the SUNDIALS package [47], with a fixed integrator step length of t_{int}.

Appendix B

Model Description for the Synthesis Reactor Section of an Ammonia Plant

This chapter is describing the reaction section of the simple ammonia reaction section as case study for surrogate model definition. This case study is used in Chapters 11, 12, and 14.

B.1 Process Description

The reaction section of the ammonia synthesis gas loop is an example of an integrated process. The model consists of two reactor beds and is illustrated in Figure B.1. In this process, a feed consisting of hydrogen and nitrogen is reacting to ammonia. Addition-

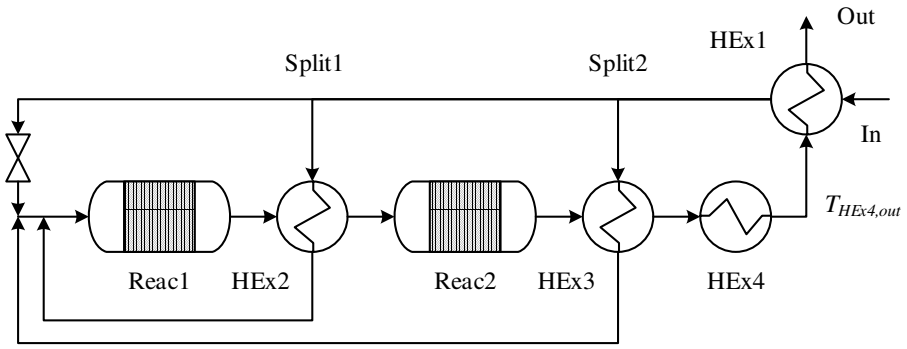


Figure B.1: Flowsheet of the reaction section of the ammonia synthesis loop.

ally, ammonia as well as the inert gases argon and methane are present in the feed stream. To exploit the produced heat of the exothermic reaction and improve reactor utilization through shifting of the thermodynamic equilibrium, several heat exchangers are introduced. A first heat integration is performed using heat exchanger 1 in which the feed is heated through the stream leaving the reaction section. Secondly, the feed is split into 3 streams going through a valve, an interstage heat exchanger and a heat exchanger post the second reactor bed. All streams are subsequently mixed with each other and fed to the first reactor bed. After heat exchanger 3, excess energy present through the exothermic reaction is used for creating high pressure steam in heat exchanger 4. In this model, we have two nested recycle loops (M-R1-HEx2-M and M-R1-HEx2-R2-HEx3-M) as well as a third recycle loop in contact with the two nested (HEx1-S-HEx3-HEx4-HEx1). Incorporating this model into a big flowsheet where an overall mass recycle loop is in contact with the third recycle loop leads to a complicated initialization of the model. In addition, small changes in the manipulated variables may lead to computational expensive flowsheet evaluations. Hence, incorporating the overall model into an optimization routine requires further simplifications of the model and frequently leads to crashes of the system.

B.2 Model Description and Assumption

The flowsheet was modelled in MATLAB and compromises a nonlinear system of equations with 282 states. The reactor beds are modelled as CSTR-cascades and the heat exchangers using the Number of Transfer Units Method. As heat exchanger 4 is defined *via* its outlet temperature, simple mass balances and an outlet temperature definition are sufficient. For the calculation of the energy transfer in heat exchanger 4,

The number of independent variables $n_u = 10$ is given by the variables of the feed stream (7 variables: p_{in}, T_{in}, and $\dot{n}_{i,in}$) plus the two split ratios through the valve (n_{Val}) and heat exchanger 3 (n_{HEx3}) as well as the outlet temperature ($T_{HEx4,out}$) of heat exchanger 4. The split ratio through heat exchanger 2 is defined *via* the aforementioned split ratios to maintain constant mass in the split.

References

[1] C. G. Alesandrini, S. Lynn, and J. M. Prausnitz. Calculation of vapor-liquid equilibria for the system NH_3-N_2-H_2-Ar-CH_4. *Industrial & Engineering Chemistry Process Design and Development*, 11(2):253–259, 1972.

[2] V. Alstad and S. Skogestad. Null space method for selecting optimal measurement combinations as controlled variables. *Industrial & engineering chemistry research*, 46(3):846–853, 2007.

[3] V. Alstad, S. Skogestad, and E. S. Hori. Optimal measurement combinations as controlled variables. *Journal of Process Control*, 19(1):138 – 148, 2009. ISSN 0959-1524.

[4] J. Andersson. *A General-Purpose Software Framework for Dynamic Optimization*. PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, October 2013.

[5] M. Appl. *Ammonia, 2. Production Processes*, page 139. Wiley-VCH Verlag GmbH & Co. KGaA, 2000. ISBN 9783527306732.

[6] M. Appl. *Ammonia, 2. Production Processes*, page 191. Wiley-VCH Verlag GmbH & Co. KGaA, 2000. ISBN 9783527306732.

[7] A. Araújo and S. Skogestad. Control structure design for the ammonia synthesis process. *Computers & Chemical Engineering*, 32(12):2920 – 2932, 2008. ISSN 0098-1354.

[8] K. B. Ariyur and M. Krstic. *Real-time optimization by extremum-seeking control*. John Wiley & Sons, 2003.

[9] A. Bhosekar and M. Ierapetritou. Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Computers & Chemical Engineering*, 108:250 – 267, 2018. ISSN 0098-1354.

[10] L. T. Biegler. *10. Simultaneous Methods for Dynamic Optimization*, pages 287–324.

[11] L. T. Biegler and R. R. Hughes. Infeasible path optimization with sequential modular simulators. *AIChE Journal*, 28(6):994–1002, 1982. ISSN 1547-5905.

[12] L. T. Biegler, I. E. Grossmann, and A. W. Westerberg. *Systematic Methods of Chemical Process Design*. Prentice Hall, 1997.

[13] L. T. Biegler, Y. dong Lang, and W. Lin. Multi-scale optimization for process systems engineering. *Computers & Chemical Engineering*, 60:17 – 30, 2014. ISSN 0098-1354.

[14] A.-L. Boulesteix and K. Strimmer. Partial least squares: a versatile tool for the analysis of high-dimensional genomic data. *Briefings in Bioinformatics*, 8(1): 32–44, 2007.

[15] J. A. Caballero and I. E. Grossmann. An algorithm for the use of surrogate models in modular flowsheet optimization. *AIChE Journal*, 54(10):2633–2650, 2008. ISSN 1547-5905.

[16] B. Chachuat, B. Srinivasan, and D. Bonvin. Adaptation strategies for real-time optimization. *Computers & Chemical Engineering*, 33(10):1557–1567, 2009.

[17] D. F. Chichka, J. L. Speyer, and C. Park. Peak-seeking control with application to formation flight. In *Decision and Control, 1999. Proceedings of the 38th IEEE Conference on*, volume 3, pages 2463–2470. IEEE, 1999.

[18] M. Chioua, B. Srinivasan, M. Guay, and M. Perrier. Performance improvement of extremum seeking control using recursive least square estimation with forgetting factor. *IFAC-PapersOnLine*, 49(7):424–429, 2016.

[19] M. M. Câmara, R. M. Soares, T. Feital, T. K. Anzai, F. C. Diehl, P. H. Thompson, and J. C. Pinto. Numerical aspects of data reconciliation in industrial applications. *Processes*, 5(4):56, 2017. ISSN 2227-9717.

[20] A. Cozad, N. V. Sahinidis, and D. C. Miller. Learning surrogate models for simulation-based optimization. *AIChE Journal*, 60(6):2211–2227, 2014. ISSN 1547-5905.

[21] M. L. Darby, M. Nikolaou, J. Jones, and D. Nicholson. RTO: An overview and assessment of current practice. *Journal of Process Control*, 21(6):874 – 884, 2011. ISSN 0959-1524.

[22] S. E. Davis, S. Cremaschi, and M. R. Eden. Efficient surrogate model development: Optimum model form based on input function characteristics. In A. Espuña, M. Graells, and L. Puigjaner, editors, *27th European Symposium on Computer Aided Process Engineering*, volume 40 of *Computer Aided Chemical Engineering*, pages 457 – 462. Elsevier, 2017.

[23] S. de Jong. SIMPLS: An alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 18(3):251 – 263, 1993. ISSN 0169-7439.

[24] J. Eason and S. Cremaschi. Adaptive sequential sampling for surrogate model generation with artificial neural networks. *Computers & Chemical Engineering*, 68:220 – 232, 2014. ISSN 0098-1354.

[25] J. P. Eason and L. T. Biegler. A trust region filter method for glass box/black box optimization. *AIChE Journal*, 62(9):3124–3136, 2016. ISSN 1547-5905.

[26] M. M. El-Halwagi. Introduction to process integration. In M. M. El-Halwagi, editor, *Process Integration*, volume 7 of *Process Systems Engineering*, pages 1 – 20. Academic Press, 2006.

[27] M. Ellis, H. Durand, and P. D. Christofides. A tutorial review of economic model predictive control methods. *Journal of Process Control*, 24(8):1156 – 1178, 2014. ISSN 0959-1524. Economic nonlinear model predictive control.

[28] S. Engell. Feedback control for optimal process operation. *Journal of Process Control*, 17(3):203 – 219, 2007. ISSN 0959-1524. Special Issue ADCHEM 2006 Symposium.

[29] T. Faulwasser, L. Grüne, and M. A. Müller. Economic nonlinear model predictive control. *Foundations and Trends® in Systems and Control*, 5(1):1–98, 2018. ISSN 2325-6818.

[30] R. Findeisen and F. Allgöwer. Computational delay in nonlinear model predictive control. *IFAC Proceedings Volumes*, 37(1):427 – 432, 2004. ISSN 1474-6670. 7th International Symposium on Advanced Control of Chemical Processes (ADCHEM 2003), Hong-Kong, 11-14 January 2004.

[31] W. Findeisen, F. Bailey, M. Brdys, K. Malinowski, P. Tatjewski, and A. Wozniak. *Control and Coordination in Hierarchical Systems*. International Series on Applied Systems Analysis. John Wiley & Sons, 1980.

[32] A. Forrester, A. Sobester, and A. Keane. *Engineering Design via Surrogate Modelling: A Practical Guide*. Wiley, 2008. ISBN 9780470770795.

[33] A. I. Forrester and A. J. Keane. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45(1):50 – 79, 2009. ISSN 0376-0421.

[34] A. S. Foss. Critique of chemical process control theory. *AIChE Journal*, 19(2): 209–214, 1973. ISSN 1547-5905.

[35] G. François, B. Srinivasan, and D. Bonvin. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *Journal of Process Control*, 15(6):701 – 712, 2005. ISSN 0959-1524.

[36] G. Froment, K. Bischoff, and J. De Wilde. *Chemical Reactor Analysis and Design, 3rd Edition*. John Wiley & Sons, Incorporated, 2010. ISBN 9781118136539.

[37] S. S. Garud, I. Karimi, and M. Kraft. Smart sampling algorithm for surrogate model development. *Computers & Chemical Engineering*, 96:103 – 114, 2017. ISSN 0098-1354.

[38] S. S. Garud, I. A. Karimi, and M. Kraft. Design of computer experiments: A review. *Computers & Chemical Engineering*, 106:71 – 95, 2017. ISSN 0098-1354. ESCAPE-26.

[39] S. S. Garud, I. A. Karimi, G. P. Brownbridge, and M. Kraft. Evaluating smart sampling for constructing multidimensional surrogate models. *Computers & Chemical Engineering*, 108:276 – 288, 2018. ISSN 0098-1354.

[40] B. Grimstad, B. Foss, R. Heddle, and M. Woodman. Global optimization of multiphase flow networks using spline surrogate models. *Computers & Chemical Engineering*, 84:237 – 254, 2016. ISSN 0098-1354.

[41] B. Grimstad et al. SPLINTER: a library for multivariate function approximation with splines. `http://github.com/bgrimstad/splinter`, 2015. Accessed: 2017-11-26.

[42] S. Gros, B. Srinivasan, and D. Bonvin. Optimizing control based on output feedback. *Computers & Chemical Engineering*, 33(1):191 – 198, 2009. ISSN 0098-1354.

[43] M. Guay and T. Zhang. Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainties. *Automatica*, 39(7):1283 – 1293, 2003. ISSN 0005-1098.

[44] F. Haber and R. Le Rossignol. Über das ammoniak-gleichgewicht. *Berichte der deutschen chemischen Gesellschaft*, 40(2):2144–2154, 1907. ISSN 1099-0682.

[45] T. Hager. *The Alchemy of Air: A Jewish Genius, a Doomed Tycoon, and the Scientific Discovery That Fed theWorld but Fueled the Rise of Hitler*. Crown/Archetype, 2008. ISBN 9780307449993.

[46] I. J. Halvorsen, S. Skogestad, J. C. Morud, and V. Alstad. Optimal selection of controlled variables. *Industrial & Engineering Chemistry Research*, 42(14): 3273–3284, 2003.

[47] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*, 31(3): 363–396, 2005.

[48] B. Hunnekens, M. Haring, N. van de Wouw, and H. Nijmeijer. A dither-free extremum-seeking control approach using 1st-order least-squares fits for gradient estimation. In *53rd IEEE Conference on Decision and Control*, pages 2679–2684. IEEE, 2014.

[49] International Fertilizer Association. Fertilizers, climate change and enhancing agricultural productivity sustainably. Technical report, Jul 2009.

[50] J. Jäschke and S. Skogestad. Nco tracking and self-optimizing control in the context of real-time optimization. *Journal of Process Control*, 21(10):1407–1416, 2011.

[51] J. Jäschke and S. Skogestad. Optimal controlled variables for polynomial systems. *Journal of Process Control*, 22(1):167 – 179, 2012. ISSN 0959-1524.

[52] J. Jäschke and S. Skogestad. Optimal operation of heat exchanger networks with stream split: Only temperature measurements are required. *Computers & Chemical Engineering*, 70(Supplement C):35 – 49, 2014. ISSN 0098-1354. Manfred Morari Special Issue.

[53] J. Jäschke, Y. Cao, and V. Kariwala. Self-optimizing control – A survey. *Annual Reviews in Control*, 43(Supplement C):199 – 223, 2017. ISSN 1367-5788.

[54] V. Kariwala. Optimal measurement combination for local self-optimizing control. *Industrial & Engineering Chemistry Research*, 46(11):3629–3634, 2007.

[55] V. Kariwala, Y. Cao, and S. Janardhanan. Local self-optimizing control with average loss minimization. *Industrial & Engineering Chemistry Research*, 47(4): 1150–1158, 2008.

[56] S. Karolius, H. A. Preisig, and H. Rusche. Multi-scale modelling software framework facilitating simulation of interconnected scales using surrogate-models. In

Z. Kravanja and M. Bogataj, editors, *26th European Symposium on Computer Aided Process Engineering*, volume 38 of *Computer Aided Chemical Engineering*, pages 463 – 468. Elsevier, 2016.

[57] B. D. Keating and A. Alleyne. Combining self-optimizing control and extremum seeking for online optimization with application to vapor compression cycles. In *American Control Conference (ACC), 2016*, pages 6085–6090. IEEE, 2016.

[58] R. E. Kopp and R. J. Orford. Linear regression applied to system identification for adaptive control systems. *AIAA Journal*, 1(10):2300–2306, Oct 1963. ISSN 0001-1452.

[59] D. G. Krige. A statistical approach to some mine valuations and allied problems at the Witwatersrand. Master's thesis, University of Witwatersrand, South Africa, 1951.

[60] D. Krishnamoorthy, A. Pavlov, and Q. Li. Robust extremum seeking control with application to gas lifted oil wells. *IFAC-PapersOnLine*, 49(13):205–210, 2016.

[61] D. Krishnamoorthy, E. Jahanshashi, and S. Skogestad. A feedback rto strategy using transient measurements. 2018, in preparation.

[62] M. Krstić and H.-H. Wang. Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4):595 – 601, 2000. ISSN 0005-1098.

[63] V. Kumar and N. Kaistha. Hill-climbing for plantwide control to economic optimum. *Industrial & Engineering Chemistry Research*, 53(42):16465–16475, 2014.

[64] L. Ljung. *System identification: theory for the user*. PTR Prentice Hall, Upper Saddle River, NJ, 2 edition, 1999.

[65] H. Manum and S. Skogestad. Self-optimizing control with active set changes. *Journal of Process Control*, 22(5):873 – 883, 2012. ISSN 0959-1524.

[66] S. Marinkov, B. de Jager, and M. Steinbuch. Extremum seeking control with adaptive disturbance feedforward. *IFAC Proceedings Volumes*, 47(3):383–388, 2014.

[67] S. Marinkov, B. de Jager, and M. Steinbuch. Extremum seeking control with data-based disturbance feedforward. In *2014 American Control Conference*, pages 3627–3632, June 2014. doi: 10.1109/ACC.2014.6858832.

[68] H. Martens. Reliable and relevant modelling of real world data: a personal account of the development of PLS regression. *Chemometrics and Intelligent Laboratory Systems*, 58(2):85 – 95, 2001. ISSN 0169-7439. PLS Methods.

[69] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979. ISSN 00401706.

[70] N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

[71] R. Misener and C. A. Floudas. ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. *Journal of Global Optimization*, 59 (2):503–526, Jul 2014.

[72] M. Morari, Y. Arkun, and G. Stephanopoulos. Studies in the synthesis of control structures for chemical processes: Part i: Formulation of the problem. process decomposition and the classification of the control tasks. analysis of the optimizing control structures. *AIChE Journal*, 26(2):220–232, 1980. ISSN 1547-5905.

[73] M. D. Morris. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174, 1991.

[74] J. Morud. *Studies on the Dynamics and Operation of integrated Plants*. PhD thesis, The Norwegian Institute of Technology, Department of Chemical Engineering, NTNU, Trondheim, Dec. 1995.

[75] J. C. Morud and S. Skogestad. Analysis of instability in an industrial ammonia reactor. *AIChE Journal*, 44(4):888–895, 1998. ISSN 1547-5905.

[76] L. Naess, A. Mjaavatten, and J.-O. Li. Using dynamic process simulation from conception to normal operation of process plants. *Computers & Chemical Engineering*, 17(5):585 – 600, 1993. ISSN 0098-1354.

[77] L. Narraway and J. Perkins. Selection of process control structure based on economics. *Computers & Chemical Engineering*, 18:S511 – S515, 1994. ISSN 0098-1354. European Symposium on Computer Aided Process Engineering—3.

[78] M. R. Naysmith and P. L. Douglas. Review of real time optimization in the chemical process industries. *Developments in Chemical Engineering and Mineral Processing*, 3(2):67–87, 1995. ISSN 1932-2143.

[79] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006. ISBN 9780387303031.

[80] A. Nuchitprasittichai and S. Cremaschi. An algorithm to determine sample sizes for optimization with artificial neural networks. *AIChE Journal*, 59(3):805–812, 2013. ISSN 1547-5905.

[81] L. M. Ochoa-Estopier, M. Jobson, and R. Smith. The use of reduced models for design and optimisation of heat-integrated crude oil distillation systems. *Energy*, 75:5 – 13, 2014. ISSN 0360-5442.

[82] R. Penrose. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51(3):406—-413, 1955.

[83] F. Provost, D. Jensen, and T. Oates. Efficient progressive sampling. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, pages 23–32, New York, NY, USA, 1999. ACM.

[84] N. Quirante and J. A. Caballero. Large scale optimization of a sour water stripping plant using surrogate models. *Computers & Chemical Engineering*, 92 (Supplement C):143 – 162, 2016. ISSN 0098-1354.

[85] H. H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960.

[86] M. Rovaglio, D. Manca, and F. Cortese. A reliable control for the ammonia loop facing limit-cycle and snowball effects. *AIChE Journal*, 50(6):1229–1241, 2004. ISSN 1547-5905.

[87] D. Santangelo, V. Ahón, and A. Costa. Optimization of methanol synthesis loops with quench reactors. *Chemical Engineering & Technology*, 31(12):1767–1774, 2008. ISSN 1521-4125.

[88] P. O. M. Scokaert and J. B. Rawlings. Feasibility issues in linear model predictive control. *AIChE Journal*, 45(8):1649–1659, 1999. ISSN 1547-5905.

[89] D. Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[90] S. Skogestad. Plantwide control: the search for the self-optimizing control structure. *Journal of Process Control*, 10(5):487 – 507, 2000. ISSN 0959-1524.

[91] S. Skogestad. Near-optimal operation by self-optimizing control: from process control to marathon running and business systems. *Computers & Chemical Engineering*, 29(1):127 – 137, 2004. ISSN 0098-1354. {PSE} 2003.

[92] S. Skogestad. *Chemical and Energy Process Engineering*. CRC Press, 2008. ISBN 9781420087567.

[93] S. Skogestad and C. Grimholt. *The SIMC Method for Smooth PID Controller Tuning*, pages 147–175. Springer London, London, 2012.

[94] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. Wiley & Sons, Chichester, West Sussex, UK, 2005. ISBN 9780470011676.

[95] I. Sobol. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4): 86 – 112, 1967. ISSN 0041-5553.

[96] B. Srinivasan, D. Bonvin, E. Visser, and S. Palanki. Dynamic optimization of batch processes: Ii. role of measurements in handling uncertainty. *Computers & chemical engineering*, 27(1):27–44, 2003.

[97] B. Srinivasan, L. Biegler, and D. Bonvin. Tracking the necessary conditions of optimality with changing set of active constraints using a barrier-penalty function. *Computers & Chemical Engineering*, 32(3):572 – 579, 2008. ISSN 0098-1354.

[98] J. Straus and S. Skogestad. Minimizing the complexity of surrogate models for optimization. In Z. Kravanja and M. Bogataj, editors, *26th European Symposium on Computer Aided Process Engineering*, volume 38 of *Computer Aided Chemical Engineering*, pages 289 – 294. Elsevier, 2016.

[99] Y. Tan, W. Moase, C. Manzie, D. Nešić, and I. Mareels. Extremum seeking from 1922 to 2010. In *Proceedings of the 29th Chinese control conference*, pages 14–26. IEEE, 2010.

[100] M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103:225–249, 2005.

[101] U.S. Geological Survey. Mineral commodity summaries 2018. Technical report, Jan 2018.

[102] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006. ISSN 1436-4646.

[103] H. Wendt. Neue konstruktive und prozeßtechnische konzepte für die wasserstoffgewinnung durch elektrolyse. *Chemie Ingenieur Technik*, 56(4):265–272, 1984. ISSN 1522-2640.

[104] S. Wold, H. Martens, and H. Wold. *The multivariate calibration problem in chemistry solved by the PLS method*, pages 286–293. Springer Berlin Heidelberg, Berlin, Heidelberg, 1983. ISBN 978-3-540-39447-1.

[105] S. Wold, M. Sjöström, and L. Eriksson. PLS-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58(2):109 – 130, 2001. ISSN 0169-7439.

[106] L. Würth, J. B. Rawlings, and W. Marquardt. Economic dynamic real-time optimization and nonlinear model-predictive control on infinite horizons. *IFAC Proceedings Volumes*, 42(11):219 – 224, 2009. ISSN 1474-6670. 7th IFAC Symposium on Advanced Control of Chemical Processes.

[107] R. Yelchuru and S. Skogestad. Convex formulations for optimal selection of controlled variables and measurements using mixed integer quadratic programming. *Journal of Process Control*, 22(6):995 – 1007, 2012. ISSN 0959-1524.