Hybrid Fuzzy-First Principles Modeling

Pascal van Lith

2002

Ph.D. thesis
University of Twente

Twente University Press

HYBRID FUZZY-FIRST PRINCIPLES MODELING

Twente University **Press**

Publisher: Twente University Press
P.O. Box 217, 7500 AE Enschede, the Netherlands, www.tup.utwente.nl

Cover design: Jo Molenaar, [deel 4] ontwerpers, Enschede
Print: Grafisch Centrum Twente, Enschede

ISBN 9036517060

HYBRID FUZZY-FIRST PRINCIPLES MODELING

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
prof.dr. F.A. van Vught,
volgens besluit van het College voor Promoties
in het openbaar te verdedigen
op vrijdag 25 januari 2002 te 13.15 uur

door

Pascal Ferenc van Lith
geboren op 6 juni 1974
te Arnhem

Dit proefschrift is goedgekeurd door de promotor
prof.dr.ir. B. Roffel

en de assistent-promotor
dr.ir. B.H.L. Betlem

Contents

1

Introduction

This thesis deals with the development and analysis of dynamic hybrid fuzzy-first principles models for chemical engineering applications. These hybrid models consist of a framework of dynamic mass and energy balances, supplemented with fuzzy models. The ability of fuzzy models to describe complex behavior in a simple, transparent and straightforward manner is used to model unknown or poorly understood physical phenomena, while the hybrid model structure is derived from first principles. Hybrid fuzzy-first principles models can be attractive if a complete first principles model is difficult to derive.

In this work, a structured modeling approach for hybrid fuzzy-first principles models is presented. Different modeling tools are discussed and evaluated. The approach is illustrated by developing hybrid models for several processes. In addition, the properties of these models are analyzed. This will provide modelers in process engineering with insight in the applicability and design of hybrid models.

1.1 Process modeling in chemical engineering

Similar to other fields of engineering, process models in chemical engineering are used to help solving problems which benefit from a mathematical description of the system. Applications can be found in the area of research and development, design and process control. For many applications, especially for control and optimization problems, dynamic models are necessary; the model needs to describe the behavior of the system with respect to time.

Dynamic process modeling in chemical engineering is often based on a combination of first principles and empirical relations. These models are interpretable, in the sense that, by analyzing the model, there is a physical understanding of the process behavior. Many process models, consisting of a framework of mass, component and energy balances describing the essential process accumulation, are available in a state-space representation. Within this framework, phenomena such as reaction rates, mass transfer or equilibria can be approximated by static empirical relations. Often, combinations of physical phenomena are described, when the distinction between these phenomena is difficult to make. Examples are the Monod equation, which is used to characterize biomass growth, the overall reaction rate equations to describe polymerization kinetics or mass transfer in distillation. Models which combine first principles with empirical descriptions are called *grey box models*. To make a distinction between these models and hybrid fuzzy-first principles models, this type of models will simply be denoted with *first principles models*.

If physical interpretation is less important and a complex system needs to be described by a simple input-output model, *black box modeling* can be applied. This is a data driven approach; the observed behavior is mapped by a mathematical representation that does not have a physical basis. Such approaches include transfer functions, which are used to approximate the response of an output to a change in an input and time series models (Box and Jenkins, 1970). Over the last decades, the use of soft computing approaches (such as artificial neural networks or ANN's and fuzzy logic) as a black box technique to model systems has gained substantial interest in different fields of engineering.

1.2 The use of soft computing

Soft computing is a collection of methodologies that aim to exploit the tolerance for imprecision and uncertainty to achieve tractability, robustness and low solution cost (Zadeh, 1994). The principle constituents of soft computing are fuzzy logic, neurocomputing and probabilistic reasoning, of which genetic algorithms are a popular form. These methodologies are also often referred to as artificial intelligence, computational intelligence or intelligent systems.

Soft computing applications can be found in many fields, from social studies to economics to medicine to engineering. There is an abundance of literature on soft computing. The overview given in Stephanopoulos and Han (1996) discusses over 500 papers on soft computing applications over a period of 10 years in process engineering alone. The review distinguishes seven different areas of application:

- Diagnosis of process operations
- Monitoring and analysis of process trends
- Intelligent control
- Heuristics and logic in planning and scheduling of process operations
- Modeling languages, simulation and reasoning
- Intelligence in scientific computing
- Knowledge based engineering design

In addition, Stephanopoulos and Han (1996) distinguish common trends in these areas that will characterize the nature of future developments:

- Specialization to narrowly defined classes of problems
- Integration of multiple knowledge representations
- Integration of processing technologies
- Rapid expansion of industrial applications

As will be shown in this work, hybrid modeling can be seen as part of the second and third trends. Hybrid fuzzy-first principles models combine different knowledge representations; fuzzy logic and first principles. In addition, identification and design techniques common for processing both representations are combined to construct hybrid models.

Fuzzy logic is the main soft computing methodology used in hybrid fuzzy-first principles modeling. Applications of fuzzy logic with respect to process engineering cover, amongst others, control systems, expert systems, decision support systems and classification systems

(Dubois and Prade, 1993). Of particular interest for this work are the modeling applications, which represent a relatively small portion of all fuzzy logic applications. Good introductions to fuzzy modeling can be found in Babuška (1996) and Yager and Filev (1994).

1.3 Hybrid modeling

In chemical engineering, dynamic process modeling often involves systems, of which physical information is limited or poorly understood. Examples of such processes are biochemical processes and polymerization processes. In such cases, it is difficult to develop a first principles model. *Hybrid models* can be a useful alternative in situations where a first principles model is required, but is difficult to construct.

By combining black box techniques with a physical model framework, hybrid models are obtained that combine first principles knowledge with the ability to deal with complex, poorly understood behavior. A partial model is derived from simple physical considerations (such as mass or energy balances), while a black box technique is used to augment the model. Hybrid models are especially suited to describe highly nonlinear behavior over a large operating domain. Examples are models of batch or fed-batch processes, cyclic processes or distributed parameter processes, such as plug flow reactors.

Combining black box techniques with physical equations has received some attention since the early 1990's. In addition to hybrid models, such models are referred to as grey box models, semi-mechanistic models or polytopic models. In this context, Thompson and Kramer (1994) distinguish a parallel approach, in which a model is augmented with black box techniques, and a serial approach, in which black box techniques are used to describe physical aspects of the process. As such, it has a serial connection with the physical equations of the model.

In Thompson and Kramer (1994), a hybrid modeling approach is presented that attempts to maximize the value of domain-specific knowledge. A hybrid model of a fed-batch bioreactor is developed, in which an artificial neural network augments the performance of a parametric model that describes the specific kinetic rates, such as biomass growth and substrate consumption. The combined output of the parametric model and the ANN is processed by an output model which calculates the system state. In their terminology, this is a combination of a parallel and serial hybrid modeling approach.

Bohlin (1994) presents the development of a grey box model for a full scale steel rinsing process. The approach is based on a so-called ”root model” that describes the known physical behavior and which is expanded iteratively. The result is a model that combines a physical model and an internal noise model. In Funkquist (1997), a similar grey box model is obtained by reducing a complex distributed first principles model to a state space model using orthogonal collocation and introducing stochastic disturbances.

In the serial approach, the black box part of the hybrid model calculates internal variables that the first principles part needs. The output of the black box part has a physical meaning or is a measure for a physical property. In the parallel approach, model equations which have limited validity are compensated for by the black box part, while in the serial approach, they are replaced by the black box part. This way, the interpretation of the validity of the model equations, both physical and black box, is much clearer. A popular approach is to combine artificial neural networks with first principles model structures (Gupta *et al.*, 1999; Otto and Hartmann, 1996; Piron *et al.*, 1997; Porru *et al.*, 2000; Psichogios and Ungar, 1992; Qi *et al.*, 1999; Van Can *et al.*, 1996; Van Can *et al.*, 1997). In the hybrid models presented in these papers, one or more internal variables of parameters are calculated by ANN's and processed by the physical part of the model. Applications include bioreactors, chemical reactors and separation processes.

Until now, little research has been presented in which fuzzy logic is used in a hybrid modeling context similar to artificial neural networks. Some applications have been reported (Babuška *et al.*, 1996; Johansen and Foss, 1997; Roubos *et al.*, 1999). For chemical engineering applications, the use of fuzzy logic is of particular interest. The ability to deal with different sources of information (such as process data as well as expert knowledge) and nonlinearities, in a straightforward manner, makes the approach of suitable for modeling complex chemical systems. In addition, fuzzy models are more transparent than neural networks, which make them useful in applications where transparency is required. It is therefore of interest to investigate the use of fuzzy logic in hybrid modeling.

1.4 Objective

This work deals with the development and analysis of *hybrid fuzzy-first principles models*. The objective is twofold. First, the *design* of hybrid fuzzy-first principles models is investigated. This involves the determination of a suitable hybrid model structure as well as model identification. This may be straightforward for simple systems, but can be difficult for more complex systems. A *structured modeling approach* will therefore be presented, which provides the possibility to divide the modeling problem into smaller subproblems. In addition, it provides the flexibility to use different modeling tools during model design. This may be beneficial for applications which require specific or customized techniques. The structured modeling approach will be applied to several processes. This provides insight in the use of the modeling approach and modeling tools.

Secondly, the *properties* of hybrid fuzzy-first principles models are investigated. A framework that characterizes the quality of hybrid models will be presented. Using this framework, properties such as performance, complexity and interpretability can be analyzed. This will form the basis for comparing hybrid models with other model structures. As such, this work provides modelers in process engineering with information to determine whether hybrid modeling is a suitable solution to their modeling problem.

1.5 Thesis outline

In this thesis, some understanding of fuzzy logic will be assumed. Basic concepts of fuzzy set theory are discussed in appendix A. For a more detailed discussion on fuzzy logic, the reader is referred to Yager and Filev (1994).

Chapter 2 discusses some general modeling paradigms. This will serve as an introduction into process modeling and related topics and provides a context for hybrid models.

Hybrid modeling will be discussed in chapter 3. A general hybrid model structure and the structured modeling approach will be presented. In addition, a framework for evaluating model quality will be derived, which provides a basis for a discussion on the applicability of hybrid models. The chapter also discusses the use of different sources of information: first principles, process data and expert knowledge.

Chapter 4 presents the design of hybrid models in more detail. The design phase involves the determination of the hybrid model structure, the identification of the fuzzy and physical parts and the integration of these parts. Different tools and identification algorithms will be discussed. The tools will be used to develop a hybrid model for a simple simulated fed-batch bioreactor, which will serve as an example throughout the chapter.

A hybrid model of a simulated continuous pulp digester will be developed in chapter 5. This illustrates the modeling approach for a more complex system. Using the model quality framework derived in chapter 3, the hybrid model will be compared with the two extremes of the hybrid fuzzy-first principles modeling ”spectrum”: first principles models and fuzzy models. This will provide modelers with more insight in the properties of hybrid models in relation to more conventional modeling approaches.

In chapter 6, two different hybrid models for a batch distillation column set-up will be designed and compared. This illustrates the application of the modeling approach in an experimental environment. It also illustrates two different approaches to the determination of a hybrid model structure which describes the essential dynamic characteristics of the process, based on prior knowledge.

The thesis will conclude with a discussion on the most important results and some suggestions for future research in chapter 7.

2

Process modeling

The notion of a *model* is widely used in almost every field of science. The term model is applied to entities varying from mathematical descriptions to scaled down replica's of the actual system. This chapter discusses some general aspects of models for and the modeling of chemical processes. It is not intended as an in depth discussion of various techniques, but more as an introduction into process modeling and related topics. The considerations presented here form a basis for chapter 3, where hybrid fuzzy-first principles modeling is discussed.

2.1 What is a model?

A general definition of a model in an engineering environment is given by Eykhoff (Eykhoff, 1974). He defines a model as a *representation of the essential aspects of an existing system (or a system to be constructed) which presents knowledge of that system in a usable form*. This means that a model is always a simplified representation of the real system. Such a representation can provide insight in the behaviour of the system, which does not necessarily mean that this insight is *physical*. For example, if an engineer is interested in developing a controller for a chemical reactor, he usually will want to know how the reactor behaves dynamically. Whether this knowledge is based on physical principles or not does not have to be relevant for his purposes.

A model is seldom a goal in itself. It is always a tool to help solving a problem, which benefits from a mathematical description of the system. Applications of models in engineering can be found in three general areas (Eykhoff, 1974; Luyben, 1990):

- **Research and development.** A model in research gives an interpretation of knowledge or measurements. An example is the determination of chemical kinetic mechanisms from laboratory or pilot-plant reaction data.
- **Design.** Models used in design can be used to determine the correct design parameters of a component or sub-system or can be used to study process stability, process safety, economical aspects, etc. This means that the available knowledge has to be expressed in such a way that it is compatible with these criteria.
- **Control.** The control actions are based on the knowledge that is available of the system. In addition, it is cheaper to conduct experiments concerning plant operation and control on models than on an operating unit (which does not mean that experimental setups are not needed at all). The control actions can for example be designed to keep the process on normal operating conditions, to handle the process in emergency situations (such as diagnostic systems) or to manage start-up and shut-down processes.

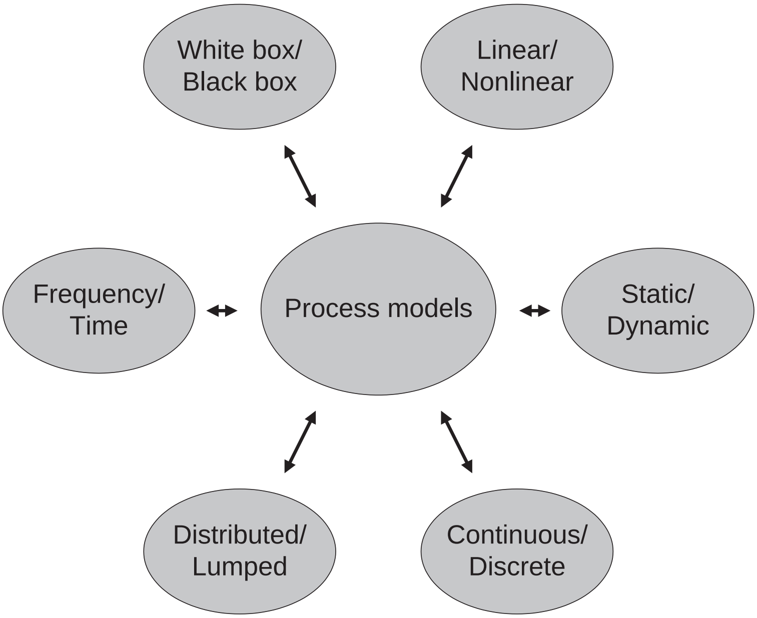


Figure 2.1: *Points of view for model structure comparison*

2.2 Types of models

The different applications of models and the performance criteria that result from these have lead to many different model structures. These can be compared from different points of view. Figure 2.1 illustrates this.

2.2.1 White box and black box models

Models that are entirely based on physical and chemical laws (thermodynamics, continuity equations) are called *white box, first principles* or *mechanistic* models. These models give a physical insight of the system and can even be built when the system is not yet constructed. Usually a set of (partial) differential equations supplemented with algebraic equations is used to give a mathematical description of the model. The effort needed to build these models is usually high, certainly for complex chemical systems.

Black box models do not use any structure that reflects the physical structure of the system: black box models (also called empirical models) give an input/output relation of the process. These models are useful if a physical understanding of the system is absent or not relevant for the purpose of the model. Mathematical descriptions used include autoregressive models (such as ARMA and ARMAX models), Artificial Neural Networks (ANN's, see for example

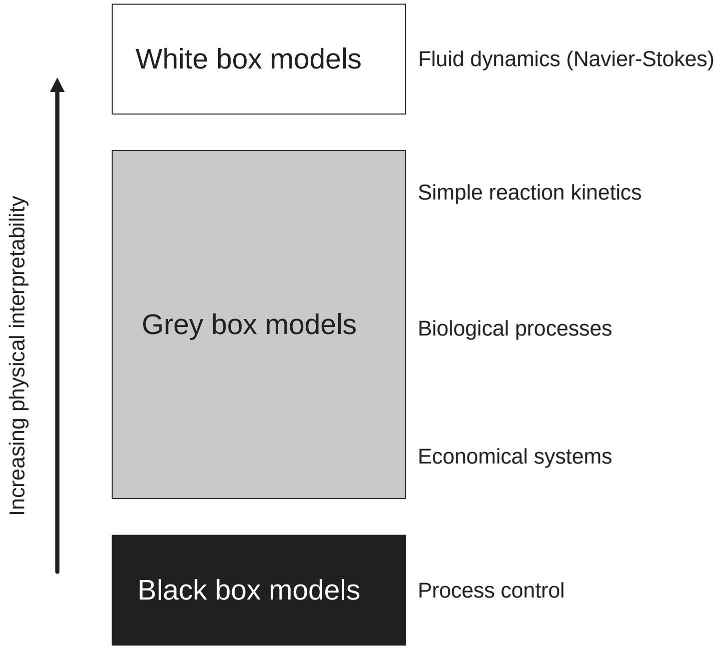


Figure 2.2: *Physical interpretability of grey box models and examples of applications*

Hertz *et al.* (1991)) and fuzzy logic as presented in this thesis.

Knowledge about the process may be incomplete, which can result in models that use both first principles and empirical modeling strategies. Models that combine both approaches are called *grey box* models. In practice, most models used will be grey to some extend, simply because knowledge about the system may be incomplete, due to lack of physical understanding or the lack of detailed process information.

The level of black box techniques used in grey box models denotes the level of physical interpretation that can be given to a model (see figure 2.2 (Sohlberg, 1998)). If physical interpretation is important then the model should be as white as possible. This means that the structure of the model should be based on first principles as much as possible. The level of interpretation also depends on the required level of detail of the model. For example, for process design it is sufficient to describe mass or heat transfer with a single coefficient. However, for more fundamental research studies at molecular level such descriptions are not adequate.Hybrid fuzzy-first principles models are a class of grey-box models.

2.2.2 Linear and nonlinear models

Models that have the property of superposition are called *linear models*. A model of the form $y = f(u)$ exhibits the property of superposition if:

$$\alpha f(u_1) + \beta f(u_2) = f(\alpha u_1 + \beta u_2) \tag{2.1}$$

where $\alpha, \beta \in \Re$ are constants. This also applies to differential equations. For example, equation 2.2 is linear and equation 2.3 is not.

$$\dot{x} = ax + u \tag{2.2}$$

$$\dot{x} = ax^2 + u \tag{2.3}$$

In many of the optimization methods and control algorithms, linear (or linearized) models are used. However, in chemical engineering, processes are hardly ever linear. To model these processes, linear models can be used, but to a limited extend. Linear models provide an accurate enough solution in a specific working range. An example is first order Taylor series expansion, where the first derivative describes the behaviour of the system around a working point.

Nonlinear models are hard to solve analytically, but several sophisticated methods are available to solve these models numerically. Commercial software packages such as gPROMS incorporate these methods, so that the modeler only has to be concerned with the formulation of the model and not the derivation of the solution.

2.2.3 Static and dynamic models

A model is static if the description of the independent variables does not change with respect to time. This is also denoted as a steady state description. Dynamic models are capable of describing the transition of variables with respect to time. Modeling the behavior of a batch reactor for example will result in a dynamical models, because a batch process is by definition unsteady state. If a model is based on continuity equations, the difference between a static and dynamical form often results in setting the accumulation term zero or not. Static models in chemical engineering are often used in design and optimization; dynamic models find their application in process control.

2.2.4 Distributed parameter and lumped parameter models

Variations in space of properties of the system require different modeling approaches. Spatial variations of variables or parameters (for example the concentration in a tubular reactor) can make the model description and solution complex. Models that take these variations into account are called *distributed parameter* models. Sometimes it can be convenient to ”lump” these variations by assuming homogenic properties. A lumped parameter model thus does not describe spatial variations, which may be an approximation of reality if these variations are present in the system. A lumped parameter model of the tubular reactor mentioned above can be made by dividing the reactor in homogenic ”cells”, for each of which a lumped parameter model can be constructed. Together these models from an approximate description of the process.

Care must be taken that a lumped parameter representation approximates the behaviour of the system sufficiently. It may be difficult to determine beforehand if a lumped parameter representation is valid. There are, however, some criteria which can help the modeler, such as the Péclet number Pe, which is defined as the ratio between the transport rate by convection and the transport rate by dispersion (Westerterp *et al.*, 1984).

2.2.5 Continuous and discrete models

In essence, continuous means that a variable can have any value at a given interval, whereas discrete variables can only have a given number of values. Although models can be discrete with respect to any variable (for example the number of trays in a distillation column, which is a discrete property of the system), the distinction between continuous and discrete models is usually made with respect to time. Discrete time models calculate the state of the system at given time intervals. When in this thesis a model is called discrete or continuous, it means that it is discrete or continuous *with respect to time*.

For digital computer simulation, a model has to be discrete at some point in the solution procedure. The numerical solution methods all require discrete models. However, many of the software packages available today are capable of discretizing the model automatically, so that the modeler can supply the continuous model, if desired.

Discretization of first order differential equations is done as shown in equation 2.4 (numerical differentiation).

$$\frac{dx}{dt} = \frac{x_{k+1} - x_k}{\Delta t} \tag{2.4}$$

where x denotes the variable to be discretized, t denotes the time and the subscript k denotes the time step. This approximation is only valid for a sufficiently small interval Δt. More on discretization and discretization errors can be found in Atkinson (1989).

Combinations of discrete and continuous models are also possible. These models describe systems which incorporate discrete time events, such as switching procedures in electronic circuitry. In literature these combinations of models are often called hybrid models. However, it must be noted that in this thesis the term *hybrid model* is only used to denote hybrid fuzzy-first principles models, unless stated otherwise.

2.2.6 Frequency domain and time domain models

The distinction between time domain models and frequency domain models is made for dynamical models only, because time dependence is by definition not incorporated in static models. For the analysis of dynamical systems in may be interesting to investigate the response of the system to a oscillating input with a certain frequency. In *frequency response analysis*, a system is subjected to a sustained sinusodial wave. The output of the system will eventually also become a sinusodial wave, which for systems that are unstable for that input frequency can be divergent. The features of this output signal (amplitude, phase shift) with respect to the frequency of the input signal can then provide information of the system behaviour as function of the frequency.

Frequency response analysis is an important tool in controller design. Bode diagrams and Nyquist plots are convenient ways to investigate the dynamic characteristics of a (controlled) system and can be used to determine the required control system characteristics.

2.3 Model building

Researchers often state that modeling is more an art than a science. Although the science behind the various techniques that have been developed can hardly be called an art, it is not necessarily disadvantageous that modeling is considered an art. The applications and requirements are far too different so that the development of a general model building approach would be extremely difficult and decisions concerning the modeling of a system can often best be made by an system expert. Some researchers argue that it is not desirable to develop a ”methodical modeling layer” and that research efforts in this area should be directed to practical and effective approaches to solve problems. While the decisions made by a modeler during the modeling process may be based more on his expertise than on a methodology, it may be worthwhile to discuss some aspects that can be taken into account.

The modeling process can be divided in four phases: problem definition, design, evaluation and application (figure 2.3). In the problem definition phase, the modeling problem and the goal of the model are properly formulated. This formulation is based on performance and structure requirements with respect to the application and on the modeling expertise of the modeler. This expertise can help beforehand to determine if these requirements are realistic and can be met using the modeling techniques that are available.

Based on this formulation, the key variables and the structure of the model are determined and the model parameters are identified in the design phase. For white box models, the structure of the model reflects the physical structure of the system, which means that additional design steps have to be taken: formulation of the basis of the model (physical and chemical laws) and making assumptions (finding a balance between microscopic, physically correct descriptions which require a lot of modeling effort and more general descriptions which yield less accurate results).

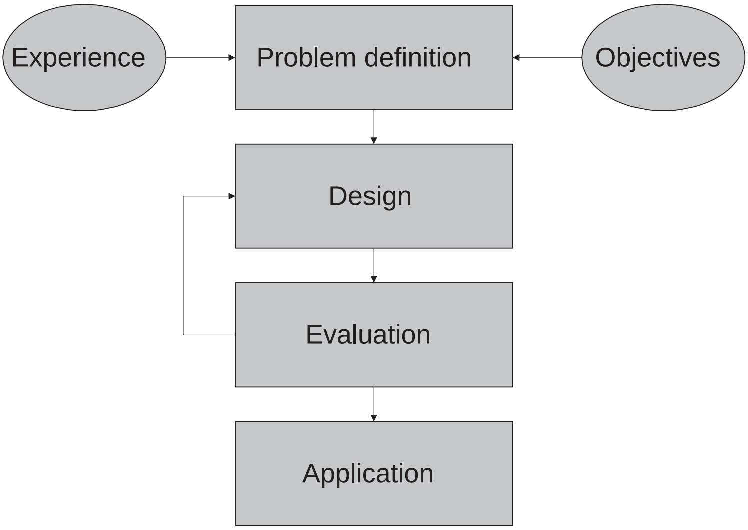


Figure 2.3: *General phases in model building*

In the evaluation phase, the model itself is verified with respect to its structure and the results of the model are validated with the real world situation. In addition to performance evaluation, the requirements with respect to the model structure (such as transparency of interpretability), as formulated in the problem definition phase, need to be evaluated. If the criteria are not met, the model needs to be improved, which makes modeling an iterative process. If the criteria are met, the model can be applied.

Chapter 3 will discuss model building in more detail with respect to hybrid fuzzy-first principles modeling.

2.4 Model structure considerations

Since models are used as a basis for further decisions, the knowledge should be presented in a usable form. The model should not be too complex, but must give a sufficiently accurate description of the system. Furthermore, requirements with respect to transparency and interpretability have to be taken into account.

The selection of the form of the model plays a more important role when white box modeling is applied than when black box modeling is used. While black box modeling techniques only require the specification of the input and output variables of the (sub)system and the correlation between these, white or grey box models in addition require the specification of the form of the mathematical relations that are used. This requires skill and experience.

Figure 2.4 shows some of the possible mathematical structures that are used in white or

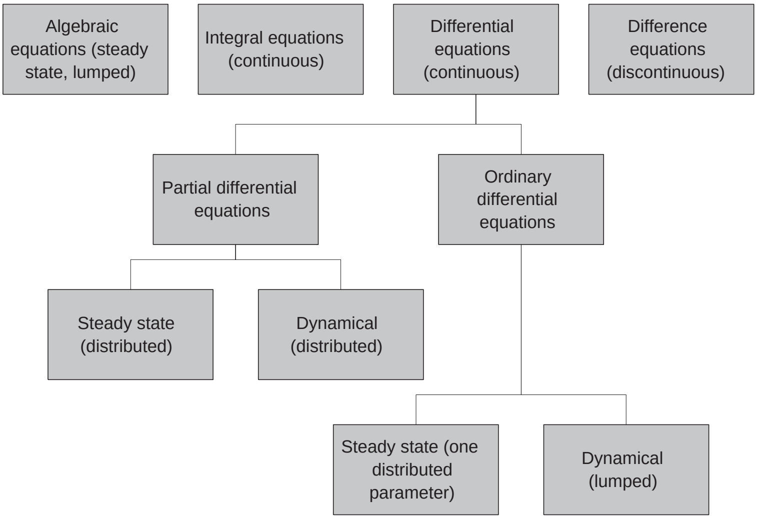

Algebraic equations (steady state, lumped)
Integral equations (continuous)
Differential equations (continuous)
Difference equations (discontinuous)
Partial differential equations
Ordinary differential equations
Steady state (distributed)
Dynamical (distributed)
Steady state (one distributed parameter)
Dynamical (lumped)

Figure 2.4: *Typical mathematical forms of models, with examples for differential equations*

grey box modeling (Edgar and Himmelblau, 1988). The eventual decision of what the best structure is can best be made *ad hoc*, based on the problem itself, the insight of the modeler and common practice in the field of application. Models for research purposes, for example, are preferably white box, or as white as possible. Models for online applications in control need to be solved quickly, so they cannot be too complex. If a model is used for plant debottlenecking, steady state models usually suffice. However, batch optimization requires dynamic models, etc.

2.5 Performance criteria

Model evaluation of deterministic models consists of two parts: numerical performance validation and model structure verification. These have to be evaluated with respect to the goals formulated in the problem definition phase. A clear distinction between verification and validation is made here.

2.5.1 Model verification

Verification is more an internal procedure: the mathematical correctness of the model structure is ascertained and the structure requirements are checked. Using common sense is of paramount importance, but there are some tools available which can help the verification process.

Parameter range
When using a physical base for modeling, usually there is some *a prioi* information about the magnitude of the estimated parameters. For example, some parameters can only be positive, such as viscosity. The parameters of the model can be checked with respect to this knowledge.

Sensitivity analysis
It may be interesting to investigate the influence of parameters on the outputs of the model. This can be done using sensitivity equations. The sensitivities are expressed by partial derivatives, such as:

$$\frac{\partial \hat{y}}{\partial \theta} \tag{2.5}$$

which denotes the sensitivity of model output $\hat{y}$ with respect to parameter θ. The calculation can be done numerically using Euler's method. The sensitivity can be approximated by:

$$S = \frac{\hat{y}(\theta + \Delta\theta) - \hat{y}(\theta)}{\Delta\theta} \tag{2.6}$$

where $\hat{y}(\theta)$ is output $\hat{y}$ as a result of the value of parameter θ and $\Delta\theta$ is a small change in the value of θ. A time plot of S can be useful to determine if θ can be assumed constant or not. This test can only be used for steady state models. If dynamic models are used, average, integral or final-time variants of the test can be used, although interpretation of the results is more difficult.

Structure tests
In modeling, the idea is to keep the model as simple as possible. If one model contains more parameters than another, it's performance can be better than the latter model, but it is also more complex. If desired, the extra complexity can be penalized by incorporating the number of parameters in an error criterion for the model. This way, a trade-off between a low modeling error and model complexity can be made. This kind of tests is especially useful in black box modeling. See Sohlberg (1998) for more information.

Transparency and interpretability
Structure requirements in the form of transparency and interpretability criteria are hard to formulate concretely. The modeler can form ideas about this beforehand, but the actual analysis can best be done *a posteriori*, because these criteria are subjective and can be subject to change. For example, if the physical interpretation that can be given to the model is lower than anticipated, but the extra modeling effort that is required to improve this is very high, the interpretability criterion could be weakened. So, while having the model available, the modeler can decide if these kinds of criteria are met or not. These issues are especially important in white and grey box modeling. Examples will be given in other chapters in this thesis.

Statistical tools
There are several statistical tests available to judge the structure of a model. Examples are the likelihood ratio test (were the ratio of the likelihood functions of two models provide information about the loss of performance with respect to model complexity), Lagrange Multiplier

tests (for nonlinear models) and Bayesian tests. These test provide possibilities to compare two models. Since the other tools presented here provide sufficient information for the purposes of this research, statistical tests are not discussed further. For a general overview of statistical structural tests, see Holst *et al.* (1992).

2.5.2 Model validation

Validation means that model output is compared with measurements from the actual system in order to determine if the model describes the real system accurately enough. This can be done using various error criteria.

Cross parameter validation
Cross parameter validation can be seen as a form of ”in duplo” parameter estimation. If possible, the experiment that yields the data for identification is divided in two sub-experiments, both of which are executed under the same circumstances. Based on this, it should be possible to reproduce the estimates. The estimates of the parameters are compared with each other. If one of the estimates resulting form the second experiment differs substantially from the corresponding estimate in the first experiment, the part of the model that uses this estimate should be investigated further.

Residual tests
The residuals are represented by the innovation of the process (or the model error). The innovation is defined as:

$$\boldsymbol{i} = \boldsymbol{y} - \hat{\boldsymbol{y}} \tag{2.7}$$

in which $\boldsymbol{i}$ is the innovation vector, $\boldsymbol{y}$ is the process measurement vector and $\hat{\boldsymbol{y}}$ is the estimated process output vector. The innovation is a suitable index to investigate the mismatch between the real process and the model. The residuals represent unmodeled parts and disturbances. For the ideal model the innovation should be white and normally distributed (independent of each other). The easiest way to analyze the innovation is to simply plot it, which can reveal trends, peaks or other deviations that are not desired.

The most common tests to check the behaviour of the innovation are the calculation of the auto-correlation and the cross-correlation with the input. The auto-correlation of a single innovation signal i calculated as:

$$r_{ii}(\tau) = \frac{1}{N-\tau} \sum_{k=1}^{N-\tau} (i(k) - \bar{i})(i(k+\tau) - \bar{i}) \tag{2.8}$$

where τ is the time lag, N is the number of innovation values and $r_{ii}(\tau)$ is the auto-correlation of the innovation at lag τ. Often, the normalized auto-correlation is used:

$$r'_{ii}(\tau) = \frac{r_{ii}(\tau)}{r(0)} \tag{2.9}$$

If the auto-correlation is plotted as a function of the lag τ and a confidence interval is set, the performance can be analyzed. The confidence interval is usually set at the 99% confidence interval for the asymptotic distribution, calculated as $[-2.58/\sqrt{N}, 2.58/\sqrt{N}]$. If the normalized autocorrelation at lag $\tau > 0$ falls outside this interval, the residuals are not independent.

The cross-correlation with respect to the input is similar to the auto-correlation test. The cross-correlation between the input u and the innovation i of a system is given by:

$$r_{ui}(\tau) = \frac{1}{N-\tau}\sum_{k=1}^{N-\tau}(u(k)-\bar{u})(i(k+\tau)-\bar{i}) \qquad (2.10)$$

The normalized form is given by:

$$r'_{ui}(\tau) = \frac{r_{ui}(\tau)}{\sqrt{r_{uu}(0)r_{yy}(0)}} \qquad (2.11)$$

where $r_{uu}(0)$ and $r_{yy}(0)$ are the auto-correlations of the input and the output at lag 0, respectively (calculated with equation 2.8). The cross correlation can be expressed for positive and negative lag. A peak for negative lag indicates feedback in the system. This means that feedback can be present in the process itself, or the experiments that were used to generate the data were carried out with a control signal which is based on the process output. The cross-correlation is compared with the 99% confidence interval, in the same way as the auto-correlation.

Another method is the normal distribution test. In this test, a histogram of the magnitudes of the residuals is plotted. In addition, a Gaussian curve based on the standard deviation of the innovation is plotted. This way, the distribution of the histogram can be compared with the normal distribution curve, which provides information about the residual distribution. Ideally, this distribution should also be normal.

Tracking index

A convenient performance index to check if the model describes the measurements accurately enough is the so called tracking error index. The index is also often used in optimization methods and for control optimization and essentially is an integrated square error. The tracking index for the continuous case is based in the innovation $\boldsymbol{i}$ as a function of time and is defined as (Ramirez, 1994):

$$e_t = \int_0^{t_f}(\boldsymbol{i}^T Q\boldsymbol{i})dt \qquad (2.12)$$

with $[0, t_f]$ the time interval for which the index is evaluated. Q is a weighting matrix. The discrete form of this equation is given by:

$$e_t = \sum_{k=0}^{N}(\boldsymbol{i}(k)^T Q\boldsymbol{i}(k))\Delta t \qquad (2.13)$$

Root mean squared error

One of the most commonly used error measures is the Root Mean Squared Error, or RMSE.

It is defined as (Sneddon, 1976):

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N} i^2} \tag{2.14}$$

where i is a single innovation signal.

Parameter identifiability
The identifiability criterion can be used to determine whether the measurements that were used provide sufficient information for parameter estimation. This evaluation can be done by using the covariance matrix $Cov(\hat{\boldsymbol{\theta}})$, which is the covariance matrix of the estimated parameter vector $\hat{\boldsymbol{\theta}}$. The lower bound for this covariance matrix (Cramer-Rao lower bound) is given by:

$$Cov(\hat{\boldsymbol{\theta}}) \geq M^{-1}_{\boldsymbol{\theta}=\boldsymbol{\theta_0}} \tag{2.15}$$

where M is Fisher's information matrix and $\boldsymbol{\theta}_0$ is the value of the estimated parameter vector. Fisher's matrix is determined by:

$$M = E(H(\boldsymbol{\theta}_0)) \tag{2.16}$$

with $H(\boldsymbol{\theta}_0)$ the Hessian of the parameter vector $\boldsymbol{\theta}_0$. The Hessian is defined in this context as the second order derivative of the likelihood function. The likelihood function $\mathcal{L}$ is defined as the functional relationship between the observed values of the variables of the system and the estimate of the parameter vector $\hat{\boldsymbol{\theta}}$, determined from these observations. This functional relation is governed by a probability function, which also incorporates the model equations. If the probability function is not known, an uniform distribution can be chosen as an approximation (Eykhoff, 1974). The Hessian then becomes:

$$H(\boldsymbol{\theta}) = \left(\frac{\partial}{\partial\boldsymbol{\theta}}\right)^T \left(\frac{\partial}{\partial\boldsymbol{\theta}}\right) \mathcal{L}(\boldsymbol{\theta}) \tag{2.17}$$

If the covariance of some parameters, found in the diagonal of the matrix, is very large, then those parameters are not very sensitive with respect to the given measured data. This means that the experiment is not sufficiently informative or that the model may be overparameterized. More about statistical validation measures can be found in Box *et al.* (1978) (experimental) or Kendall and Stuart (1979) (more fundamental).

2.6 Remarks

Within the spectrum of model structures, the hybrid models that are discussed in this work fall in the category of grey box models: part of the model structure will be based on first principles, part will be black box. Although they can be static, the focus will be on dynamic models, which makes them suitable for applications that require dynamic models (such as process control applications) as well as for applications that require static models.

The wide range of model structures that are used in chemical engineering are the result of the different requirements that applications demand. Even within classes of models, many variations and different substructures may exist. Based on the requirements, the modeler needs to select the model structure with the properties that, in his view, can solve the modeling problem.

As a result, the focus will be on the development and properties of hybrid fuzzy-first principles models instead of some application of a hybrid model. The general structure, design and analysis of hybrid models will be discussed and illustrated for different processes. This will provide modelers with a general basis to make the decision whether hybrid models can be used in their application.

3

Hybrid fuzzy-first principles models

The combination of fuzzy logic with first principles models as proposed in chapter 1 results in *hybrid models*. The term ”hybrid” is used here to denote the combination of two white box and black box modeling techniques. In the literature, such models are also called grey box, semi-mechanistic or polytopic models. Some research has been done in this area (most notably by (Psichogios and Ungar, 1992; Thompson and Kramer, 1994)), though little research has been presented in which fuzzy logic is used in a similar context.

This chapter will present a detailed definition of hybrid fuzzy-first principles models and will discuss different model structures. Subsequently, a framework for evaluating model quality will be derived. Using this framework, the applicability of hybrid models is discussed. In addition, a structured modeling approach, which will serve as the basis for the development of hybrid models, will be presented. The main sources of information for building these models and guidelines for applying them will be also be discussed.

3.1 What is a hybrid fuzzy-first principles model?

In hybrid modeling, a distinction can be made between a modular approach and a semiparametric approach. The latter approach falls apart into a serial and a parallel approach (Thompson and Kramer, 1994).

In modular design approaches, several blocks of fuzzy logic submodels are combined to constitute the process model. The structure of the overall model is determined using prior knowledge, while every block calculates one specific variable or parameter. Advantages of this approach are that it may improve interpretability and that it may reduce the number of model parameters. The hierarchical configuration of such models reduces for example the number of input variables which results in a less complex fuzzy block and thus a fuzzy block that contains less parameters. A major disadvantage is that good output behavior is not guaranteed because the combination of the blocks could generate an overall divergent behavior (Thompson and Kramer, 1994). The fuzzy submodels are only valid in the input-output domain for which data was present during identification. If the state of such a model is outside this domain, unpredictable divergent behavior can occur, which is much less likely to happen with physical equations.

With semiparametric modeling, a fuzzy logic submodel is placed in tandem with a physical model. The physical model structure is fixed and derived from first principles. In the serial approach, fuzzy logic submodels calculate model variables which the physical part of the model requires. The input of these fuzzy submodels is provided by the physical part of the model. In figure 3.1 an example for a serial semiparametric hybrid model with one fuzzy block is given.

In the parallel approach, the outputs of the fuzzy logic block and the physical model are combined to determine the total model output (figure 3.1). The model serves as a best estimate of the process. The fuzzy logic submodel is implemented such, that it is able to compensate for any discrepancy between the physical model output and measurements. A disadvantage of this approach is that desired behavior is not guaranteed, especially if the model is used

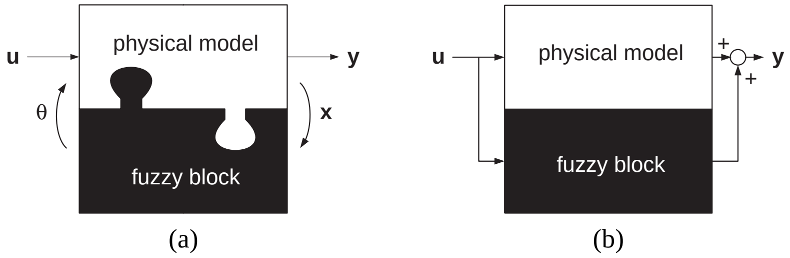


Figure 3.1: *Serial (a) and parallel (b) semiparametric design approaches. In both cases the hybrid model consists of a physical and fuzzy part, but in the parallel approach they are not connected*

under process conditions that were not included in the identification process.

If first principles models are preferred over black box models, it is proposed to leave the physical model structure intact as much as possible and only model those phenomena about which uncertainty exists (regarding model equations) with fuzzy submodels. The physical model structure is formed by dynamic mass and energy balances, while the fuzzy submodel(s) describe production rates, heat and mass transfer, equilibria, growth rates, etc. This way, hybrid fuzzy-first principles models are obtained which combine a high level of interpretability with the expectation of good extrapolating properties. Therefore, a serial semiparametric modeling approach is used. Thus in this work, hybrid models are defined as *a framework of dynamic mass and energy balances, formulated in state-space form and supplemented with algebraic and fuzzy equations*.

3.2 General properties and applicability

Because of their general form, hybrid fuzzy-first principles models can be applied in most research and engineering fields in chemical engineering. The structure of the model depends on the type of application and the process that is modeled. The focus here is on *dynamical hybrid models,* which can be used to describe the dynamical behavior of the process. Fields of application include process simulation, process control system design, process optimization and process behavior prediction.

Whether a hybrid model is suitable for a process depends on the application of the model and is the choice of the modeler. A hybrid model can be developed for any process, but it may not always be the best choice. This work only presents the construction and the properties of hybrid models, based on which the modeler can make his or her choice. It is possible, however, to give some guidelines for deciding whether hybrid models can be a solution to a modeling problem.

3.2.1 Model quality

Usually, the most important quality of a model is its performance. However, model performance does not provide sufficient information about the general properties of a model, certainly with respect to transparency and interpretability. A discussion about model properties should therefore not only include performance, but the different qualities of a model as a whole.

It is difficult to define the term ”quality”. This may be attributed to several reasons (Kan, 1994). First, quality is multidimensional concept. Second, there are different levels of abstraction; one can refer to it in the broadest sense or to its specific meaning. Third, the term quality is a part of our daily language and the popular views of the term may be very different from its use in professions in which it is approached from the engineering or management perspective.

Quality is viewed upon as intangible; it cannot be weighted or measured. Many people comment on quality that they ”know it when they see it.” In addition, luxury, class and taste are often associated with quality. These terms are rather vague. In engineering, more workable definitions have been formulated. (Crosby, 1979) defines quality as ”conformance to requirements”. This implies that requirements must be clearly stated. This definition does not take customers' requirements into account. A final product may conform to requirements, but it may not have been what the customers wanted. Therefore, the role of the customers should be explicitly incorporated in the definition of quality: conformance to customers' requirements (Kan, 1994).

In order to measure the quality of a model, it should be divided into several more tangible aspects. In other words, in order to measure quality, metrics should be defined. In software design, quality analysis is more common than in process modeling. (Boehm, 1973) formulates a hierarchy of software characteristics that results in a ”characteristics tree”. The *general utility* of a software product is determined by its *as-is utility*, its *maintainability* and its *portability*. These three characteristics are then divided into more specific characteristics such as device independence, completeness, accuracy, accessibility, etc. In analogy to Boehm's characteristics tree for software quality, a characteristics tree can be designed as a basis to map the different aspects of model quality. This is shown in figure 3.2.

To be applicable, model performance has to meet predefined requirements. This means that the model error has to be within acceptable boundaries. Error criteria can be defined to judge static performance as well as dynamic performance. In addition, model applicability can be determined by investigating interpolation and extrapolation properties. This provides information about the performance of the model outside its working area.

The second aspect of model quality is maintainability. Maintainability in this context can be evaluated by investigating model transparency. If a model is transparent, the model can easily be maintained by improving or replacing specific parts of the model. Model transparency is defined by its complexity and interpretability. The complexity and interpretability of the model equations as well as the model structure can be analyzed.

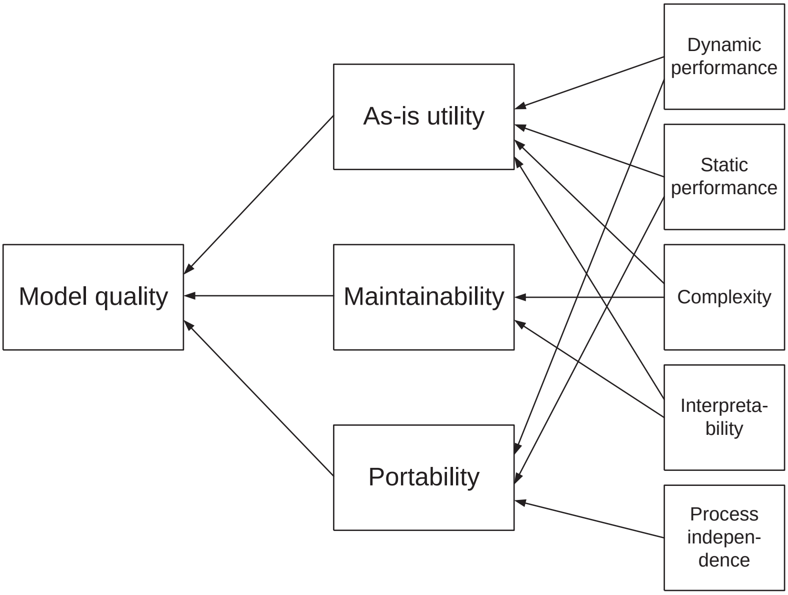


Figure 3.2: *Model quality characteristics tree*

Portability provides information about the extend to which a model is built for a specific instance of an installation. Portability may be important for applications in the field of research and development or design. High levels of process independence yield transferable models. Process independence can be investigated by analyzing the types of information that are used to build the model.

The three aspects are not completely independent. Good extrapolation properties may result in high levels of process independence but may yield models that are more complex than desired. Model interpretability is important for design purposes and affects model utility and maintainability. In addition, not all aspects are equally important for a given application. This has to be taken into account in when analyzing a hybrid model. Therefore, requirements with respect to each aspect of model quality should be determined as well as their relative importance.

3.2.2 Model applicability

It is infeasible to provide a complete list of properties of hybrid models since each model has a unique purpose. However, aspects for each of the general areas of application (research and development, design and control) can be formulated. This will enable a comparison between hybrid models and models at the extremes of the grey box modeling ”spectrum”: first principles models and black box models. Based on this comparison, the modeler can decide of hybrid modeling can provide a solution to the modeling problem.

	Model			*Application*		
Property	*First Prin.*	*Hybrid*	*Black box*	*R&D*	*Design*	*Control*
Static performance	++	+	+	●	●	
Dynamic performance	++	++	0	●		●
Complexity	0	+	++	●	●	●
Interpretability	++	+	−−	●		
Process independence	++	0	−−	●	●	

Table 3.1: *Comparison of general model quality aspects*

Research and development applications are the most demanding. The model usually has to provide detailed information about the process behavior. The model needs to be built in such a way that elements can be improved or changed without affecting the overall structure. For off line analysis, the model often needs to be process independent. Dynamic and static performance can be less important; the qualitative behavior is more important than the quantitative behavior.

Models built for design studies require accurate performance and need to be portable, certainly in the case a process has not been built yet. In addition, if optimization procedures are applied, extrapolation properties need to be good. Complexity and interpretability are less important; as long as the information that is required is provided.

The most important quality aspect of process models for control is dynamic performance. The model needs to provide accurate input-output information in order to be able to design a control structure. If the model is to be used in model-based control approaches, the model needs to be as simple as possible so that it can be solved quickly online. Portability is less important; controllers are often designed for a specific installation.

The most relevant quality requirements for each application area are shown in table 3.1. For every application, the as-is utility of the model will be important; this is the basic aspect by which to judge model quality. If the model cannot be applied, it is not useful. The importance of maintainability and portability depends more on the application.

In general, hybrid fuzzy-first principles models are useful when a physical model is required, but difficult to construct. In such cases, a *physically interpretable* model is needed for the application (interpretable in the sense that, by analyzing the model equations, the phenomena that are modeled and their mutual relations are clear). When there is a lack of understanding of the observed behavior of the process, which is often highly nonlinear, fuzzy logic can provide a means of describing this behavior in a accurate and transparent way. In addition, because the hybrid model's structure is based on first principles, a certain level of interpretability is guaranteed.

However, if a complete physical understanding of the process behavior is required, hybrid modeling should not be applied. Although hybrid models are transparent, the fuzzy equations do not have a physical meaning in the first principles sense. Fuzzy logic is a black box technique and not based on first principles. It may be possible to give a physical interpretation to the rules of the fuzzy model *a posteriori*, but this only gives qualitative insight, not insight in the actual physical processes that are taking place. First principles models can provide more detailed information, although this results in a higher level of complexity.

If a physical interpretation is not relevant for the application, black box models may be better suited than hybrid models. Black models are easier and faster to build than hybrid models. They are less complex than hybrid models. They are usually also solved faster than hybrid models (black box models usually calculate model outputs directly, while hybrid models in state-space form require numerical solvers to calculate model output), which may be important in online applications.

Hybrid fuzzy-first principles models are identified more quickly and more easily than first principles models, since they are less complex. In addition, there are modeling tools available which can determine the fuzzy equation structure and parameters fast and automatically. The identification effort is far less than is the case with complicated physical or empirical equations. If the development time for the model is limited and a complete first principles model is not required, hybrid modeling can be a good alternative.

As with first principles models, hybrid models need fairly detailed information about the behavior of the process. Process variables or parameters need to be available or it must be possible to estimate them (the system needs to be observable). If this is not possible, black box models can be a better alternative to hybrid models, because for black box models, usually input-output data of the phenomena under study is sufficient. However, to model the process dynamics, black box models need large amounts of dynamical data. In hybrid models, the dynamics are described by first principles. This means that less dynamical data is required and that hybrid models will perform better dynamically than black box models, certainly with respect to extrapolation.

The level of process independence is largely determined by the sources of information that are used to build the model. If process data is the main source, the model will be process dependent. This is the case for black box models. The use of first principles information increases the level of process independence. If, in hybrid modeling, the static properties are derived from process data or expert knowledge and these properties are process specific, the hybrid model will be more process dependent. Good static performance of the hybrid model will then be limited to the process or the operating conditions that were used during identification.

One of the main advantages of the use of fuzzy equations is that nonlinearities can be described in a simple way. These nonlinearities become apparent if the process has a large operating regime. A model for such a process is thus required to describe these nonlinearities adequately. In this case, hybrid models can be suitable. If the process is nonlinear, but these nonlinearities are not apparent during normal operating conditions, linear approximations may suffice. Examples of processes with large operating regimes are batch processes, distributed parameter processes or cyclic processes.

The differences between the types of models discussed, in relation to model quality, are given in table 3.1.

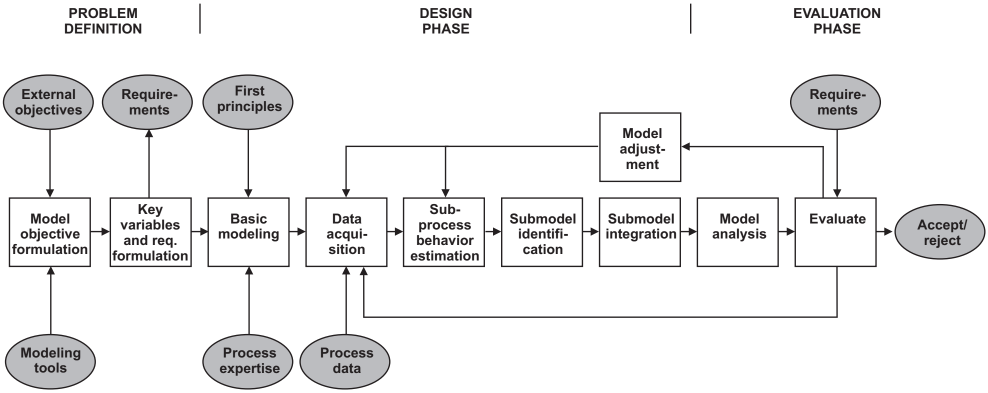


Figure 3.3: *Hybrid model building approach*

3.3 Modeling approach

Most literature about modeling focuses mainly on the parameter identification step. Relatively little is written on how to design a specific strategy for model development. This section will present such a strategy, based on the approaches presented in (Sohlberg, 1998) and (Edgar and Himmelblau, 1988). The two different approaches are integrated and adapted for hybrid fuzzy-first principles models.

The modeling approach is shown in figure 3.3. The approach consists of several sequential steps, performed independently of each other. Other research on hybrid modeling promotes a global approach (Psichogios and Ungar, 1992; Roubos *et al.*, 1999; Thompson and Kramer, 1994). A global approach is usually based on training the black box relations within the hybrid model using error feedback. The advantage of this approach is that it can reduce the number of steps that have to be taken during model development. The disadvantage of this approach is that one is easily inclined to only judge overall model fit, irregardless of the complexity and number of fuzzy relations. This is detrimental to model transparency. The advantage of independent steps is that the modeling problem is reduced to several smaller and simpler problems. The solutions of these problems are then combined to form the overall model.

Three main sources of information are generally available when constructing hybrid models. Physical understanding generally forms the basis of the model and is the result of fundamental research. The modeler has to acquire relevant first principles knowledge with respect to the modeling problem, that can be found in the general literature.

Process measurements are the most important source of information of a specific process. While first principles provide general information about the behavior of the process, process measurements are required to identify a suitable process model.

In addition to process measurements, human experience is an important source of information because it can be used to learn more about dependencies of relevant phenomena of the process and thus the structure of the model. A human can, based on his or her experience, denote whether certain effects are important or negligible. Based on this information, the modeler can decide whether these effects have to be accounted for in the model. In addition, human experience can be used to design fuzzy relations which quantify the human experience.

The modeling approach consists of three phases. In the first phase, the problem is defined, based on external objectives (the application of the model) and modeling tools and approaches that are available. In the second phase the model is build, which is evaluated in the third phase.

3.3.1 Problem definition

At the basis of model building lies the formulation of the objectives of the model. On the one hand, these objectives concern the specific process for which the model is being built. Objectives of models for product or process design will be different from objectives of models that are developed for process simulation or fault diagnosis. In addition, the domain in which the model must be valid is to be determined.

On the other hand, objectives can be formulated that apply to hybrid models in general. These are:

- Hybrid models need to consist of a framework of accumulation balances, supplemented with algebraic equations and fuzzy functions. The most important dynamics of the process need to be incorporated in the structure. Fuzzy functions only need to be applied if no suitable physical relation (or other acceptable empirical relation) can be found.
- Hybrid models need to be transparent. This means that the models need to be as undetailed as possible within the constraints of the objectives. The modeler should start with a global form and apply more detail if the objectives are not met.
- A more subordinate objective is that the fuzzy functions need to be interpretable. This means that the fuzzy functions need to be kept as transparent as possible (limited number of rules, for example). More complex relations are preferably represented by hierarchical fuzzy systems, which consist of a cascade of several simple fuzzy systems.

The modeling tools or techniques that are available provide some of the context in which the model is being built. These have to be taken into account when the model objective is formulated. For example, a model that is to be used for online model based control but that describes the behavior on a molecular level is impractical.

The objective results in a set of requirements. The key variables are distinguished and performance requirements that these key variables have to meet are set. In addition, other model quality requirements are formulated, for example the amount of information that the model has to provide or the level of transparency that is needed.

3.3.2 Hybrid model design

The aim of the design phase is to analyze the process and build the hybrid model, based on the requirements. This is done by dividing the modeling problem into smaller subproblems and dealing with them independently. The division is performed in two ”dimensions”. First, the process is partitioned into so-called ”subprocesses”, which represent different parts of the process, such as heat transfer, mass transformation or accumulation. Subsequently, a ”submodel” is developed for each subprocess. The development of the submodels consists of sequential steps: structure design, behavior estimation, identification and optimization.

The design phase consists of the following steps:

- **Basic modeling.** The process in analyzed and the model structure is designed using first principles and process expertise. In this step, a physical framework is designed that describes the key variables and the mathematical dependencies for nonlinear model parameters and additional variables are determined. In addition, the parameters that will be described with fuzzy logic are distinguished. The result can be interpreted as a number of subprocesses, for which submodels can be developed. The relation between these subprocesses is represented by a *data flow diagram* or DFD.
- **Data acquisition.** To be able to perform the identification of the various model parameters, sufficient process data needs to be available. This data can be gathered by doing experiments specifically designed for obtaining process behavior information if this information is not readily available. Parameter or state estimation techniques can also be used to obtain the correct information.
- **Subprocess behavior estimation.** Based on the model structure, the modeling problem is reduced to several subprocess behavior estimation problems, since the subprocess behavior often cannot be measured directly. See for example (Eykhoff, 1974; Luyben, 1990; Seinfeld and Lapidus, 1974; Sohlberg, 1998; Van Lith *et al.*, 2001).
- **Submodel identification.** Using the obtained data the model parameters and fuzzy blocks can be identified.
- **Submodel integration.** The submodels are integrated to form the hybrid model, which involves connecting the submodels and optimization of the hybrid model performance.
- **Model adjustment.** It may be necessary to adjust the model structure, based on the results of the evaluation phase. Usually, this means that a larger level of detail is needed. The choice of a more detailed description will depend on the experience and skill of the modeler.

The design phase will be discussed in more detail in chapter 4.

3.3.3 Hybrid model evaluation

Questions that need to be answered in this phase are: ”Has the correct model been built?” and ”Has the model been built correctly?” The quality of the model needs to be analyzed with respect to the requirements. Based on the quality analysis, the decision needs to be made if the model is really suitable or if it needs to be adjusted. The adjustment can take place in two ways: feedback through renewed identification (based on new identification data) or feedback through model adjustment. Based on the definition of model quality (section 3.2.1), some general pointers for model quality analysis can be given.

Model performance

Model performance can be divided in static and dynamic performance. Depending on the model application, a suitable error criterion needs to be defined, as well as performance requirements. Interpolation and extrapolation provide additional information about model validity.

Static performance involves calculating the model error under steady-state conditions. A distinction is made between model validation under conditions within the working area (*amplitude interpolation*) and validation under conditions outside the working area (*amplitude extrapolation*) (Van Can *et al.*, 1998). The working area can be predefined or imposed by process data that is available for identification; the model is most likely to be valid for the range of the identification data. An extension of amplitude extrapolation is dimension extrapolation, in which an input which is kept constant during identification is varied. A convenient way of visualizing interpolation and extrapolation properties is to plot the error criterion as a function of the variable that is changed during the experiments.

Dynamic performance can be evaluated by applying a varying input signal to the process and analyzing the model results. Again, the dynamic model performance can be validated under conditions within the working area (*frequency interpolation*) and outside the working area (*frequency extrapolation*). The working area is defined by the range of frequencies of change of the input variables for which the model is valid. Frequency extrapolation can also be investigated by changing variables that were kept constant during identification. The dynamic performance can be visualized by a Bode diagram.

Model transparency

Model complexity and interpretability indicate how transparent a model is. Complexity is determined by the amount of elements a (part of) a model consists of, while interpretability is determined by the meaning of those elements. Complexity and interpretability of the model structure and the model equations are taken into account.

The complexity of the model structure indicates how well the model is built. An important aid in analyzing model structure complexity are the data flow diagrams. These illustrate the connections between the different modeled phenomena (subprocesses) in a clear manner. Points of interest are the number of hierarchical layers in the model, the number of model equations and the connectivity of the equations. A transparent model shows maximum cohesion with a minimum number of connections between the equations.

Model equations are complex if they have a large input-output dimension, are highly nonlinear or contain many parameters and variables. In addition, the mathematical structure influences equation complexity; an second order polynomial is less complex than a high order partial differential equation. Equation complexity and model structure complexity are related; a simple model structure obtained by combining several relations can increase the complexity of model equations and vice versa.

Interpretability involves physical and non-physical evaluation. A model can be interpretable in the sense that it is understandable how the model output is derived from the inputs, but this may have no physical meaning. Again, interpretability of the model structure and the model equations are taken into account.

Interpretability of the model structure mainly concerns physical interpretability. The structure is physically interpretable if the elements (the equations) represent a physical process or phenomenon. This can be analyzed using the data flow diagrams. Interpretability of the model equations is based on the physical understanding of the mathematical structure of the equation (the influence of first order over second order effects, for example). In addition, equations can be interpretable while they have no physical basis; a fuzzy model may be interpretable in terms of qualitative classification but it is not based on physical principles.

Process independence

The level of process independence depends mainly on the sources of information that are used during model building. If the main source is process dependent, the model will also be process dependent. Three main sources are available: first principles, process data and human expertise.

First principles knowledge makes the model process independent. However, most models contain a certain level of empirical information which is based on observed process behavior, which makes the model more process dependent. The level of process dependence is therefore closely related to the amount of process data that is used during model building. If this is the main source, the behavior of the model will be based on the observed behavior of that specific instance of the process. It is then likely that the model will not perform as well for another instance of the process.

This is also the case for the use of human expertise. Categorizing knowledge is based on the behavior that is observed from the process and thus process dependent. Process descriptive or structural knowledge may also be process dependent, but usually to a lesser extend. A relation between two variables may be present in several instances of a process, but it may differ quantitatively.

In general, the best way to evaluate model quality is to use a top-down approach. Given the model application, first determine the relative importance of the as-is utility, maintainability and portability. Subsequently, set requirements with respect to static and dynamic performance, complexity, interpretability and process independence. Analyze only those elements that are relevant for the application. The requirements have to be incorporated in the model objectives and setting them is part of the problem definition phase.

3.4 The use of human expertise

Human expertise is used on different levels during hybrid modeling. In the first place, it is used to analyze the problem and the process and to design the model structure. In addition, human experience can be used during the design phase to provide quantitative information. This information can be captured using fuzzy logic. With the use of human expertise, three aspects need to be considered: the type of knowledge that is available, the eliciting of that knowledge and using the elicited knowledge for hybrid modeling.

3.4.1 Types of knowledge

Before knowledge processing is discussed it is insightful to investigate different types of and views to knowledge. Nonaka and Takeuchi (1995) accredits part the economical success of Japan about 20 years ago to the Japanese view on knowledge, which was unique at the time. According to their view, knowledge can be divided in two types:

- **Explicit knowledge.** Knowledge that easily can be expressed in words or numbers.
- **Implicit or tacit knowledge.** This type of knowledge can be segmented expertise and schemata/mental models/beliefs/perceptions so ingrained that it is taken for granted.

Explicit knowledge can be stored easily, implicit knowledge can not. Dealing with implicit knowledge includes learning something by mind and body. Heavy reliance is placed on figurative language and symbolism. Some investigation has been done about how implicit and explicit knowledge express themselves in process industry (Venkatasubramanian and Rich, 1988). A distinction between *deep knowledge* and *compiled knowledge* is made.

Deep knowledge is used to denote knowledge that is generic and process-independent involving concepts, constraints and behaviors of process units that are applicable to a variety of situations:

- Restrictions based on the laws of conservation of mass and energy.

- Confluence equations, which represent the influence of one state variable on another state variable.
- A library of fault models of different process units which attempt to explain the local cause of a low or high state variable value.
- Causal models of process units that would generate local effects given some cause.

Deep knowledge can be seen as *qualitative knowledge*. Deep knowledge gives information about relations between process units or variables and is not necessarily process specific. This knowledge can be very useful for designing model structures. This knowledge is the typical knowledge process engineers are likely to have.

Compiled knowledge is essentially a compilation of experiences typically restricted to a given process. Such knowledge can represent magnitudes of observed behavior and to a lesser extend relations between variables that are process specific. Compiled knowledge can be seen as *quantitative knowledge* and can be useful for determining model equations and parameters. Process operators are likely to have compiled knowledge about the process.

3.4.2 Knowledge structures

For knowledge elicitation, an overview about the form in which deep and compiled knowledge are available provides a good basis. Different approaches can be found, general as well as specifically for process knowledge.

A well known approach for developing knowledge-based systems is KADS[1]. To capture expert knowledge, KADS proposes to divide knowledge in four hierarchical layers (Tansley and Hayball, 1993): the domain layer (basic facts and concepts), the inference layer (the process of thinking), the task layer (inference sequences) and the strategy layer (selecting and planning of tasks). KADS aims to be a general approach, so it can also be suited for knowledge processing for hybrid modeling. However, since the type of knowledge that is available and its application are fairly clear, KADS may be too extensive.

Sestito and Dillon (1994) claims the following structures of knowledge are frequently employed by the problem solver:

- Heuristics or rules of thumb.
- Stereotypes, that are used to designate typical examples of some objects or situations.
- Solution hierarchies. These are frequently associated with the level of detail the problem solver wishes to deal with at one time.
- Procedures, that represent explicitly defined solution strategies and algorithms.

[1] KADS may have originally started life as an acronym, but uncertainty appears to have developed about what it is an acronym of, and it is generally used as a proper name

- Pattern matching to check if conditions are satisfied.
- Building a model and/or reasoning with that model.
- Reasoning with primary case material. This knowledge can be reduced to a set of heuristics because the knowledge is highly context dependent. This reasoning is often associated with legal domains, but it also appears in other areas such as patient care.

A more recent approach from the field of process technology and fault diagnosis in particular divides knowledge into three categories. First, there is categorizing knowledge, where (mostly numerical) data is divided into categories (”low”, for example) that provide a basis for further interpretation. Second, there is subdividing knowledge. Subdividing knowledge is knowledge about subdividing a process into autonomous units. This can be physical process units, but also specific phenomena. The final category is hypothesis testing knowledge. This is knowledge about how process variables are related to the behavior of the process.

These views all have some aspects in common. The different interpretations are caused by the different backgrounds and fields of application. About these similarities, the following can be said. Categorizing knowledge corresponds to Sestito’s pattern matching knowledge and has some overlap with stereotyping. These are specific examples of compiled knowledge. Furthermore, subdividing knowledge can be interpreted as the ability to distinguish hierarchies, such as with Sestito’s solution hierarchies. Reasoning with primary case material and hypothesis testing are both variants of the same compiled knowledge structure.

3.4.3 Knowledge contents

While information about the type and the structure of knowledge is important because it provides the context in which the knowledge must be elicited, the actual contents of the knowledge are obviously also important. The final model will be based on the contents of this knowledge. With the model structure in mind, care must be taken to elicit useful knowledge from the right person.

The contents of the knowledge a person has is highly dependent on the job of the person. Process engineers are likely to have knowledge that is deep and which is complemented with compiled knowledge about process behavior that is appropriate for the given process. The knowledge is mainly process descriptive. They have the ability to subdivide the process into logical units or distinguish structures. This represents deep knowledge is very useful during problem definitions, process analysis and model structure design.

Compiled knowledge that operators may have is quantitative and thus is useful for identification. The knowledge that will be available will mainly be concerning process control. The knowledge an operator has must be viewed in context with the proposed model structure and the task the operator has. For example, if an operator performs mainly supervisory control tasks, the knowledge must be viewed with regard to the process including its control structure. If an operator controls a variable manually, the operator acts as the controller and then is part of the control loop. The knowledge then also needs to be viewed with regard to the process including its control structure, but the knowledge in this case is much closer to the actual process. The knowledge and its application in hybrid model identification needs to be approached in a different way.

3.4.4 Knowledge elicitation

The actual elicitation of the knowledge concerns the acquisition of the knowledge from the human and recording this knowledge into an appropriate format. In knowledge elicitation, several phases can be distinguished (Jansen van der Sligte, 1999):

- Problem definition.
- Problem analysis.
- Knowledge acquisition and recording.
- Evaluation.

Problem definition and analysis

The goal of the first phase is to determine the context of the problem and to gain initial information about the process. This information is general in nature and provides information about the process structure and the modeling problem. This phase also investigates which human knowledge is available. A plant visit may also be beneficial to gain insight in the problem.

An appropriate knowledge acquisition technique needs to be used with respect to the knowledge. This is especially important for compiled knowledge. With hybrid modeling, a fuzzy model is the most appropriate structure to record compiled knowledge. This means that the choice of the knowledge acquisition technique is limited to those that are suited for recording the knowledge in fuzzy systems.

Knowledge acquisition and recording

Common knowledge acquisition techniques are based on interviews between a ”knowledge engineer” (who builds the knowledge based system) and the ”expert” and have been discussed extensively (see, for example, (Hart, 1992; Tansley and Hayball, 1993)). Different interviewing techniques are used; well known examples are the focused interview (where the interview is prepared in detail by the knowledge engineer), the structured interview (where a few topics are probed in depth by continuously asking for clarification and justification) and the tutorial interview (where the expert outlines the main themes and ideas of his knowledge domain).

Different variations aside, interviewing has been widely used as a basic knowledge acquisition mechanism (Sestito and Dillon, 1994) and it has been found by some researchers to be among the most effective. For acquiring deep knowledge, a focused interview usually will suffice. Based on the initial knowledge the knowledge engineer has obtained, an interview can be prepared to acquire information about variables, subprocesses, relations and strategies. The knowledge can simply be recorded by making notes or by designing a behavioral diagram.

More elaborate approaches have been developed for acquiring compiled knowledge. For recording this knowledge into fuzzy systems, the repertory grid and its variations are appropriate techniques. The expert's view of the problem and its domain is represented in a grid, where the first row is filled with elements that represent conclusions or solutions to the problem. The first column contains constructs (properties) that usually are bipolar (low/high, big/small). The rest of the grid is filled with a level of truth for a construct, given an element. The level of truth is given by a number between 1 and 5, where 1 and 5 represent the two poles of the construct. The grid provides a means to reveal patterns and derive decision rules.

Two variations on the repertory grid that have been designed to build fuzzy systems are Knowledge Acquisition for Fuzzy Expert Systems approach or KAFES (Hwang, 1995) and the Fuzzy Associative Memory or FAM (Marsh, 1994). In the KAFES grid, the constructs consist of variables and there corresponding bipolar values, described by linguistic values. The grid is filled with a number representing the linguistic value, given an element. The number denotes a linguistic value between the two poles. For example, if the bipolar value of a variable is ”low/high”, -2 denotes ”very low”, 1 denotes ”high” and 0 denotes ”normal”. In addition, a degree of certainty can be introduced. The grid forms the rule base; each column in the grid represents a fuzzy rule. A drawback of the approach is that all the premises for a conclusion are captured in one rule. In other words, each consequent part of the fuzzy model appears only once in the model, which may result in restrictions in flexibility that are not desirable.

For simple problems, the Fuzzy Associative Memory is a good alternative. All possible relations between input and output variables are showed in the grid, two inputs and one output at a time. The possible (linguistic) values of the two input variables are listed in the first column and the first row. The grid is filled with a linguistic value of the output variable, given the combination of the values of the two input variables. The grid is not very suitable for systems with more than two input variables. However, the class of systems with two input variables is very common and it is likely that an expert usually does not monitor more than two input variables at a time. When using more than two input variables, a FAM needs to be constructed for two inputs at a time, while keeping the other variables constant. Also, a hierarchical model could be build to describe the multiple input single output system.

The disadvantage of using the KAFES approach is that the expert needs to ”reason backwards”: given a conclusion the expert needs to distinguish premises that result in this conclusion. Most process operators or engineers are not used to reasoning this way, which may result in inaccuracy in the knowledge that is elicited. In addition, it is difficult to see from the

KAFES grid whether the obtained knowledge is complete. For problems where the several conclusions can be drawn from similar premises (such as medical diagnosis, where a disease is characterized by one set of different symptoms, whereas one symptom can be the result of many different diseases), KAFES may be a practical approach. For elicitation problems such as presented in this work, however, the FAM approach is better suited and more simple to apply.

Evaluation

When the knowledge based system is built, its performance can be judged by verification and validation. This is more important for systems based on compiled knowledge than systems based on deep knowledge. With verification, it is checked whether the system is build rightly; the proof of certain logical properties. Validation checks whether the right system is build; the determination of a homomorphism between a system and its representation.

Gonzalez and Dankel (1993) describes validation and verification of expert systems in more detail. His discussion is relevant for fuzzy systems also. In his view, verification concerns investigating the consistency of the rule base and the completeness of the rule base, while validation deals with the role of the expert system and determining the measure of performance for that system. In addition, more detailed information of the evaluation of fuzzy systems can be found in Marsh (1994) and Rausis (1998). A detailed discussion, however, goes beyond the scope of this work.

3.4.5 Using knowledge during identification

The use of compiled (quantitative) knowledge for identification concerns recording the knowledge into a fuzzy model. This model can be used in hybrid model design to determine subprocess behavior or can be used for hybrid model parameter identification. In doing so, under the assumption that a fuzzy model that contains the quantitative knowledge is available, a distinction is made between *model matching* and *model embedding*.

Model matching

In model matching, the fuzzy model that contains the expert knowledge (denoted as the ”expert model”) is transformed in such a way, that it matches the designed hybrid model structure (figure 3.4). If the hybrid model structure is designed in advance, it is likely that the expert model will not describe the input-output behavior that is required. There will be some form of model overlap, which has to be eliminated. Two different approaches for accomplishing this have been distinguished, based on the form of the expert model.

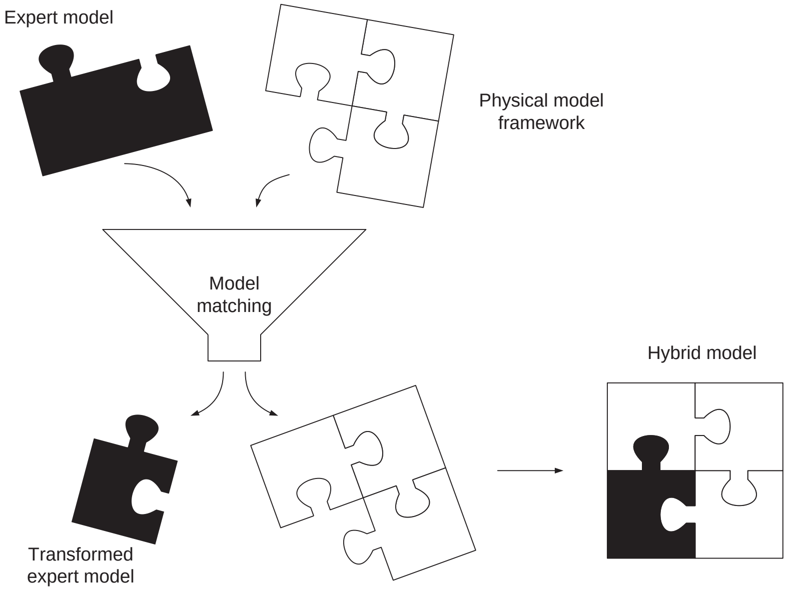


Figure 3.4: *Model matching. The expert model is transformed to match the physical framework*

In the case that the expert model describes (part of) the process behavior as opposed to control actions, *parallel model matching* is applied. The expert model is placed in parallel with the model and is used as a data generator that describes part of the process behavior. This data then can be used the same way as ordinary process measurements are used. To eliminate model overlap or to obtain the correct input-output mapping that is required for identification of the fuzzy submodel, the data needs to be transformed. This can be done using estimation techniques, which will be discussed in chapter 4. Example 3.1 illustrates a case where parallel model matching can be applied.

Example 3.1 Consider a system for which the following hybrid model structure has been designed:

$$\dot{x}_1 = f_1(x_1, x_4, u, \theta_1) \tag{3.1}$$

$$\dot{x}_2 = f_2(x_1, x_2, x_4, u, \theta_2) \tag{3.2}$$

$$\dot{x}_3 = f_3(x_1, x_3, x_4, u, \theta_3) \tag{3.3}$$

$$\dot{x}_4 = f_4(x_4, u) \tag{3.4}$$

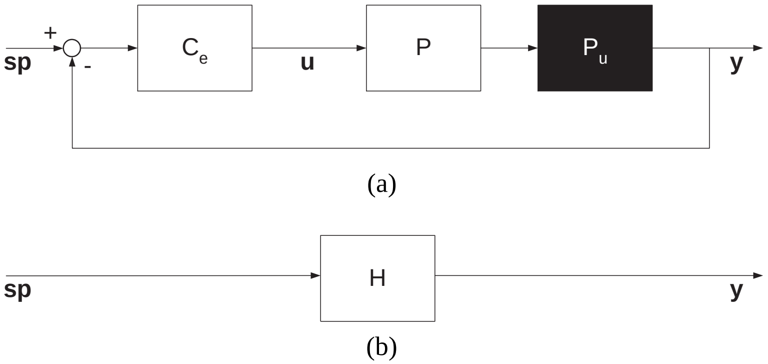
The parameters θ are described by the following algebraic equations:

$$\theta_1 = f_5(x_1, \theta_4) \tag{3.5}$$

$$\theta_2 = f_6(x_1, x_2) \tag{3.6}$$

$$\theta_3 = f_7(x_1, x_2) \tag{3.7}$$

$$\theta_4 = f_8(x_1, x_2) \tag{3.8}$$



(a)

(b)

Figure 3.5: *Serial model matching. The expert model C_e is placed in series with the hybrid model, consisting of a known part P and unknown part P_u (a)*

> Function f_5 is assumed to be unknown and will be described by a fuzzy equation. An expert model f_e is available, which describes the change in x_1, $\dot{x}_1$, as a function of x_1, x_2, x_4 and u:
>
> $$\dot{x}_1 = f_e(x_1, x_2, x_4, u) \tag{3.9}$$
>
> If this expert model is placed in parallel with the hybrid model structure, it can be seen that the model overlap is given by equations 3.8 and 3.1. The expert model can be transformed to provide a mapping of the form given by equation 3.5, by using parallel model matching. □

If the expert model is a controller model C_e, where the expert is part of the control loop, *serial model matching* can be applied. In this case, the expert model is in series with the process model. In serial model matching, the model structure is converted to transfer functions. The fuzzy parts of the hybrid model are concentrated to form an unknown block P_u that is placed in series with the known part. If the closed loop transfer function H is available or can be identified, this block is the only unknown block of the system and the problem can be solved. Figure 3.5 illustrates this.

If the closed loop transfer function H is defined as the transfer function between the setpoint and the process output, the unknown part can be calculated as:

$$P_u = C_e^{-1} P^{-1} \frac{H}{1-H} \tag{3.10}$$

where P is the known part of the model, C_e is the expert model and P_u is the unknown part of the model.

This requires inversion of the expert model as well as the known process model. Several techniques for inverting fuzzy models are available (both exact (Baranyi *et al.*, 1997; Baranyi *et al.*, 1998; da Costa Sousa, 1998; da Costa Sousa *et al.*, 1997) and inexact (da Costa Sousa *et al.*, 1997; Fischer and Isermann, 1996; Jordan and Rumelhart, 1992)), but inversion can be omitted if the closed loop transfer function is defined as the transfer function between the process input disturbances and the process output:

$$P_u = P^{-1} \frac{H}{1-C_e H} \tag{3.11}$$

Example 3.2 illustrates a case where serial model matching can be applied.

Example 3.2 Consider the system from example 3.1. Assume that instead of f_5, f_4 is unknown. The connection with the known part of the model is serial. Assume that an expert model C_e is available that describes the following behavior:

$$\dot{u} = C_e(e_{x_1}) \tag{3.12}$$

in which e_{x_1} is the error of x_1 with respect to its setpoint $x_{1,sp}$.

The unknown function f_4 can be determined using serial model matching. Interpret x_4 as a system input to equation f_1. The known process model P consists of a combination of f_1, f_5 and f_8. u and x_2 are interpreted as additional inputs. The controller C is given by the expert model f_e. The closed loop transfer function is defined as:

$$x_1 = H(x_{1,sp}) \tag{3.13}$$

while in addition

$$x_1 = P(x_4, x_2, u) \tag{3.14}$$

and

$$x_4 = P_u(u) \tag{3.15}$$

The unknown model part can be determined using equation 3.11, if P^{-1} can be determined with respect to x_4. □

Although the approach can successfully be applied to simple problems, the flexibility is limited. First of all, the approach is only useful if the unknown part is dynamic. If the unknown part is static, simpler approaches can be used. In addition, the connection between the known and unknown parts needs to be linear. If this is not the case, the system can be linearized, but this may result in a relatively complex model structure and limited model validity. Furthermore, the inversion of the known part can be cumbersome or impossible, certainly when dead time is present.

Model embedding

With model embedding, the expert model is the starting point for hybrid model development. The expert model is supplemented with the physical model framework, which needs to be transformed if model overlap is present. This way, the expert model is embedded in the physical model framework, instead of transforming it to match the framework structure. Model embedding is illustrated in figure 3.6.

If the expert model describes control actions, it needs to be inverted prior to embedding. In this case, it is required that the controller contains model information. This is the case for an internal model control structure, but may not always be true. Example 3.3 illustrates model embedding.

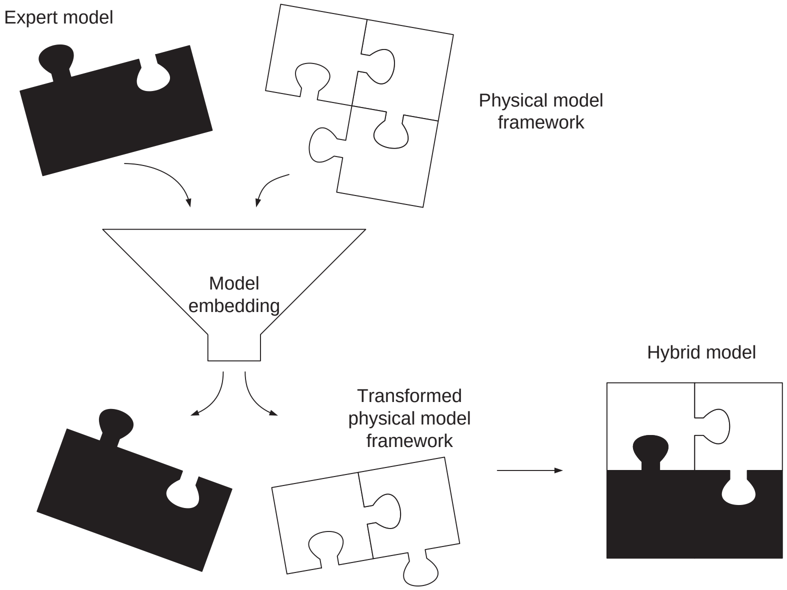


Figure 3.6: *Model embedding. The expert model is embedded by transforming the physical framework*

Example 3.3 Consider the system presented in example 3.1 and also assume that f_5 is unknown. In addition, assume that the same expert model is available:

$$\dot{x}_1 = f_e(x_1, x_2, x_4, u) \tag{3.16}$$

This model can be embedded in the physical framework by replacing equations 3.1, 3.5 and 3.8 by equation 3.16. □

3.4.6 Remarks

Although deep or qualitative knowledge can easily be applied in hybrid model design, the use of compiled or quantitative knowledge is complex. Serial model matching has limited applicability, while model embedding is not always desired if a specific hybrid model structure is designed in advance.

Since in modern plants most of the quantitative information is registered, it is recommended to use this registered information instead of eliciting it from humans. Humans base their experience on the measurements that are available. In addition, information may be lost during the elicitation process which may result in inaccuracy.

This makes the hybrid modeling approach mostly data-driven, but it should be emphasized that using deep knowledge to determine the model structure and dependencies is valuable and that it is worthwhile to investigate this knowledge when solving a hybrid modeling problem.

3.5 Mathematical considerations

For process simulation or online process behavior prediction, the model needs to be available in dynamical form. Usually, such systems are solved numerically. A mathematical advantage of using a dynamical physical framework with algebraic fuzzy equations (which results in a model in DAE form) is that there are no restrictions with respect to sampling time or solution method when solving the model. If autoregressive fuzzy submodels are used to describe process dynamics, these restrictions do apply, resulting in a less flexible model structure.

The type of fuzzy model used here is the Sugeno (TSK) type (Takagi and Sugeno, 1985). This type of model can be interpreted as a collection of local linear submodels. This type of relation is extremely suitable to describe highly nonlinear relations based on process data. It is shown that TSK fuzzy functions are universal function approximators (Ying, 1994). Furthermore, because of its local linear properties, simple algorithms can be used for parameter estimation, such as the least squares approach (Babuška, 1996). Many good algorithms for automatic identification of TSK models from data sets are available, including algorithms with structure optimization.

Although linguistic (or Mamdani) fuzzy models are easily interpretable for humans, they usually require more fuzzy rules than a TSK model to describe the same phenomenon, which makes them more complex. The advantage of using TSK models is reduced complexity, which makes the models more comprehensible. If the main source of data is human experience, Mamdani models could be easier to identify than TSK models. Since the main source of data for the identification are process measurements, TSK models are more suitable than Mamdani relations.

3.6 Concluding remarks

The structure of a hybrid fuzzy-first principles model consists of a framework of dynamic mass and energy balances, supplemented with algebraic and fuzzy equations, formulated in state space form. The advantage of such a structure is that it combines a high level of interpretability with the expectation of good extrapolation properties. Because fuzzy logic can deal with nonlinearity in a simple way, hybrid models are suited for nonlinear processes with a large operating regime, such as batch or distributed parameter processes.

Three main sources of information can be used to build hybrid models: first principles, process data and human expertise. First principles provide information about the physical framework, while the behavior observed from process data can be used to identify the fuzzy relations. Human expertise is mainly a source of structural information; process data is more suitable for quantitative information.

For building hybrid models, a structured modeling approach has been designed which consists of several independent steps, in which the modeling problem is reduced to several smaller and simpler problems which can be solved individually. This is an advantage over a global approach, in which the modeling problem is approached as a whole.

The modeling approach consists of three phases. In the first phase, the model objective and quality requirements are formulated. In the second phase, the hybrid model structure is designed and subprocesses are distinguished. Submodels for these subprocesses are subsequently identified and combined to form the hybrid model. The final phase determines the model quality and evaluates if the model meets the requirements. Model quality is determined by evaluating model performance, complexity, interpretability and the sources of information that were used.

The modeling approach is not a strict set of rules that has to be followed. It serves as a guideline for developing hybrid fuzzy-first principles models and its execution depends on the modeling problem at hand. The issues discussed in this chapter should, however, provide the modeler with a basis for hybrid model structure development.

The following chapter will present the design phase of the modeling approach in more detail. Chapter 5 will discuss general hybrid model properties.

4

Hybrid model design

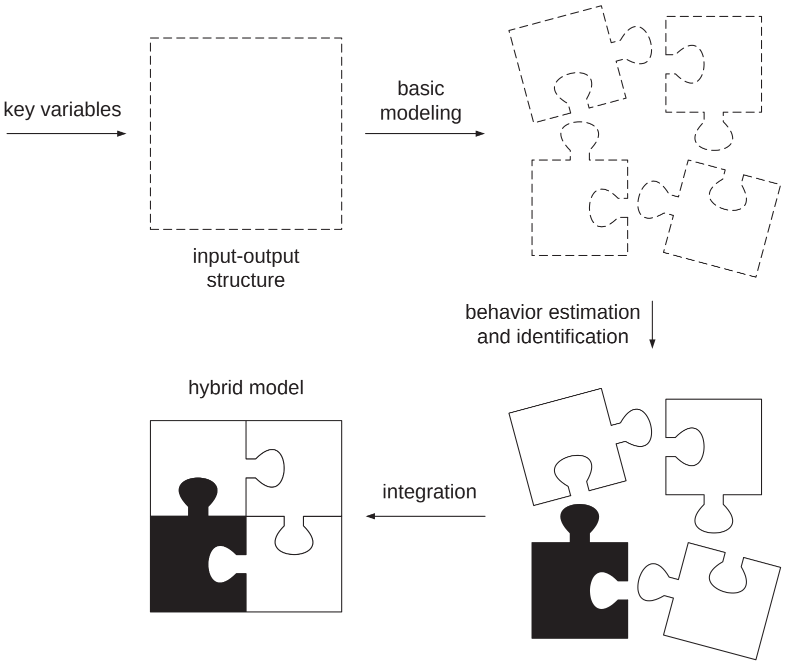


Figure 4.1: *Design phase. The problem is reduced to several smaller independent problems, which are solved sequentially*

The design phase of the hybrid modeling procedure is the ”work horse” in hybrid model development. This chapter will discuss the different steps in hybrid model design. The first step is basic modeling, in which the hybrid model structure is determined. In addition, the process is divided into several smaller ”subprocesses”, which are modeled individually. Basic modeling is followed by data acquisition, subprocess behavior estimation, submodel identification and submodel integration. The design phase is visualized in figure 4.1. For each step, different tools will be presented and their application will be discussed.

Throughout the chapter, the steps will be illustrated with a fed-batch bioreactor for penicillin production. For this process, a simulator is available which will be used to represent the actual process. The case is simple and does not represent an actual hybrid modeling problem. However, the process does posses the properties for which hybrid modeling is useful: a large operating regime and uncertainty about the phenomena that play a role, such as biomass growth. In addition, the fuzzy submodels that will be developed are double input single output systems. This allows simple visualization of the model and therefore provides a better understanding of the results. Because of the simplicity, it provides a good basis for illustrating and evaluating the various tools that are used during hybrid model design. The case will therefore be presented as an illustration.

4.1 Basic modeling

Basic modeling is the first step of the design phase: it determines the hybrid model structure and distinguishes the subprocesses. The following steps can be distinguished.

Step 1: Process description and information analysis
The function of the process is formulated and the available information is listed. It has practical and economical advantages if a model can be built on the readily available process information (that is, information that is available without performing specific experiments), be it experimental data or human expertise. This will not always be possible, but the available information should be analyzed before the model structure is designed. This step should also inform the modeler if, based on the readily available information, a model can be built in accordance with the model objectives or that additional experimentation is needed. The following steps should be executed within the context of this analysis.

Step 2: Process hypotheses
Before the system can be described with mathematical equations, the physical behavior needs to be described. Knowledge of basic principles from chemistry and physics can be used to denote qualitative relations. These hypotheses serve as the basis for the formulation of the mathematical equations.

Step 3: Process structure
The process is divided in subprocesses and the connections between these are determined. The subprocesses of the model denote the entities that will be described by a certain equation or a set of equations. Examples are chemical reaction, mass or heat transfer. These entities are combined using mass or energy balances (the model framework). The inputs and outputs of the model are also denoted. This step yields a behavioral or data flow diagram (Yourdon, 1989), in which the relations between the entities are shown.

In this step, the fuzzy and first principles part of the model are denoted. Based on model objectives, first principles, process expertise and the experience of the modeler, it can be decided if it is feasible to describe certain effects using physical equations. In addition, it can also be decided to ”lump” certain effects, too complex to describe physically, so that transparency is guaranteed. At this stage, the *structure of the hybrid model* is determined.

Step 4: Basic equations
The mass and energy balances can be drawn up with the help of the determined hypotheses about the behavior. Algebraic equations describing the driving forces and equilibria can also be determined. Model parameters are not determined at this stage. This is also the case for the fuzzy equations. These will be determined in the identification step.

The result of the basic modeling step is the model structure defined by the basic equations. The subprocesses that will be modeled with fuzzy logic are also determined. Appropriate process data can now be collected to facilitate model identification.

In the following example a hybrid model structure for a fed-batch bioreactor for penicillin fermentation will be developed. This reactor will serve as an example throughout the rest of the chapter and the model is based on the information provided in Thompson and Kramer (1994).

Example 4.1 Assume that the problem definition phase has been completed. The model objective is to describe the penicillin concentration during a batch run, based on process inputs and initial conditions. The model should be based on global physical principles. In order to accomplish this, the following key variables have been distinguished: the substrate concentration S, the biomass concentration X, the penicillin concentration P and the volume V.

Step 1: Process description and information analysis
At the beginning of a batch, a small culture of biomass X is present. The tank reactor is filled slowly with a feed F that contains a substrate S_F, which causes the biomass X to grow and produce penicillin P. The feed rate is constant. The tank is stirred and a typical batch run lasts 200 hours. Measurements of the biomass X, substrate S, product P and volume V as well as the inputs F and S_F are available every hour.

Step 2: Process hypotheses
Based on the available literature (such as (Bastin and Dochain, 1990; Dunn *et al.*, 1992)), the general mechanisms that play a role in the process can be identified. The key variables can be described directly by state equations.

The volume balance is straightforward; the accumulation is given by the feed rate F.

The substrate S is added by the feed stream and is consumed by the biomass. This consumption is used for biomass growth biomass maintenance and penicillin production. In addition, the dilution influences the substrate concentration.

The biomass accumulation is determined by biomass growth, biomass decay and a dilution factor that is a result of the changing volume. Since a detailed description of these processes is not required by the model objectives, the assumption is made that no dynamics are present in these phenomena and that they can be described by algebraic equations. This is common practice (Dunn *et al.*, 1992).

The accumulation of the penicillin concentration, finally, is determined by the production rate, dilution and product decay, which is assumed to be constant.

Although actual operation and behavior of these kind of processes are more complex than suggested by this model, its simple nature is useful for illustration purposes and procedure development. For example, the following phenomena also characterize the actual process but are not modeled:

- Growth rate decreases with biomass due to oxygen diffusion limitations.
- Cell decay becomes appreciable at high biomass concentration.
- Maintenance energy of the culture depends on biomass concentration.
- The penicillin product decays as the culture ages.

Step 3: Process structure
The process hypotheses can be organized in a data flow diagram which represents the hybrid model

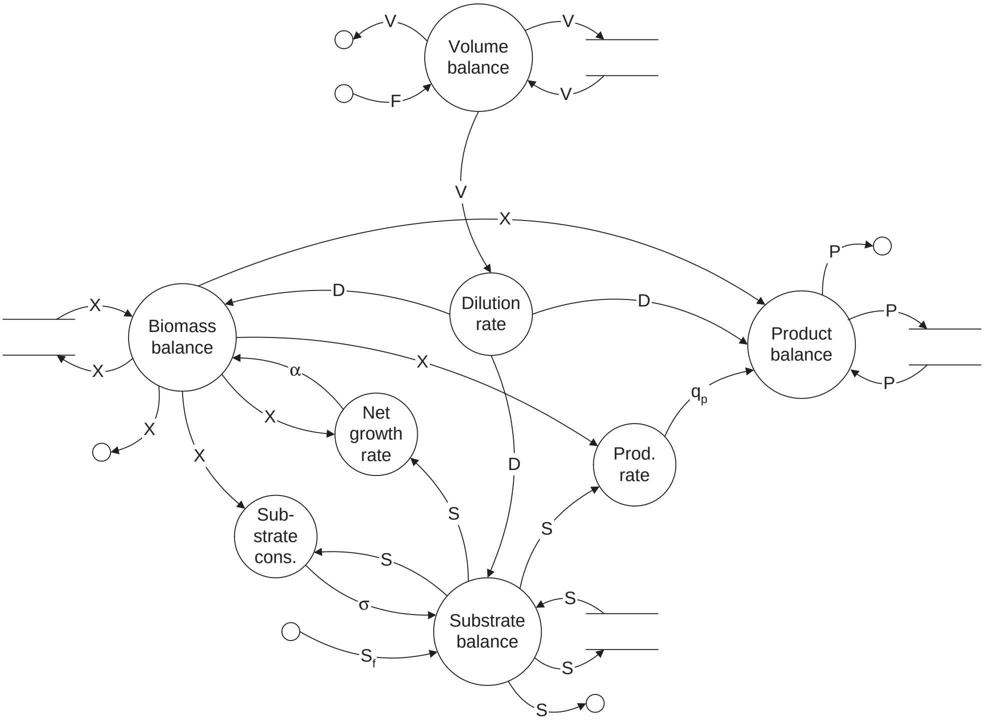


Figure 4.2: *Data flow diagram hybrid model bioreactor*

structure, as shown in figure 4.2. It is assumed that the equations that govern the biomass growth, the biomass decay and the production of penicillin are unknown. Biomass growth and decay are therefore lumped to form a net growth rate. The behavior of the net growth rate and product formation rate will be derived from the measurements. Using these measurements, fuzzy models will be built that describe the rates.

Step 4: Basic equations
The set of equations for the hybrid model can now be set up. The accumulation balances that form the physical framework are given by:

$$\frac{dX}{dt} = X(\alpha - D) \tag{4.1}$$

$$\frac{dS}{dt} = -\sigma X + (S_f - S)D \tag{4.2}$$

$$\frac{dP}{dt} = q_p X - P(D + K) \tag{4.3}$$

$$\frac{dV}{dt} = F \tag{4.4}$$

The supplemental algebraic equations are given by:

$$D = \frac{F}{V} \tag{4.5}$$

$$\alpha = f_{fuzzy}(S, X) \tag{4.6}$$

$$q_p = f_{fuzzy}(S, X) \tag{4.7}$$

$$\sigma = \frac{\mu}{Y_{x/s}} + \frac{q_p}{Y_{p/s}} + m_x \tag{4.8}$$

with

$$m_x = \frac{m_{xm} X}{X + 10} \tag{4.9}$$

and

$$\mu = \frac{\mu_m S}{K_X X + 10} \tag{4.10}$$

Here, $Y_{x/s}, Y_{p/s}, m_{xm}, K_X$ and K are constants. The relation for σ is based on information from Thompson and Kramer (1994). Now the structure is designed, the model parameters and the fuzzy equations can be identified. □

4.2 Data acquisition

In order to be able to identify the submodels that describe the subprocesses, appropriate process data needs to be available. This data can consist of historical plant data or it can be obtained by performing experiments that are designed especially for this purpose.

When designing experiments, care has to be taken that the data will best reveal the behavior of the subprocess. Experiments can be designed more effectively when knowledge that is available is used. In addition, exploratory experiments can be useful. Additional experiments are then designed on the basis of these initial results.

Experiments based on factorial design provide a useful way to measure behavior. In factorial design, the process behavior is measured at different levels of certain variables (or factors). Experiments are carried out for all possible combinations of the factors. To limit the number of experiments for systems with many factors, fractional factorial design can be used (Box *et al.*, 1978). In this approach, suitable fractions of full factorial designs are generated based on redundancy information.

Research on experiment design based on statistical methods is abundant and many text books about experiment design are available. See for example Box *et al.* (1978).

Example 4.2 Consider the bioreactor from example 4.1. For the identification of the two fuzzy relations (equations 4.6 and 4.7), sufficient measurements are needed in order to obtain the behavior of the relations. This means that the data needs to be distributed over a sufficiently large domain in the input space, so that all operating conditions are covered.

The input space is the same for both relations; the input space is formed by S and X. Figure 4.3 (a) shows the trajectory of a typical batch run in the input space. In order to obtain a good distribution, it is assumed that several experiments can be done by varying the initial conditions and system inputs. 16 different experiments were designed (see table 4.1) to accomplish this.

Batch #	$X(g/l)$	$S(g/l)$	$P(g/l)$	$V(l)$	$F(l/h)$	$S_f(g/l)$
ID1	5.0	0.5	0.0	20.0	0.110	525
ID2	5.0	0.5	0.0	20.0	0.132	525
ID3	5.0	0.5	0.0	20.0	0.154	525
ID4	5.0	0.5	0.0	20.0	0.176	525
ID5	5.0	0.5	0.0	20.0	0.198	525
ID6	5.0	0.5	0.0	20.0	0.220	525
ID7	7.5	0.5	0.0	20.0	0.110	525
ID8	10.0	0.5	0.0	20.0	0.110	525
ID9	12.5	0.5	0.0	20.0	0.110	525
ID10	15.0	0.5	0.0	20.0	0.110	525
ID11	17.5	0.5	0.0	20.0	0.110	525
ID12	20.0	0.5	0.0	20.0	0.110	525
ID13	22.5	0.5	0.0	20.0	0.110	525
ID14	25.0	0.5	0.0	20.0	0.110	525
ID15	27.5	0.5	0.0	20.0	0.110	525
ID16	30.0	0.5	0.0	20.0	0.110	525
VAL1	8.25	1.0	0.0	20.0	0.165	525

Table 4.1: *Initial conditions for batch runs*

No actual reactor setup was available to obtain the measurements. Instead, a simulator was used to generate the measurements. The simulator model is taken from Thompson and Kramer (1994). Noise was added to the simulation results. A detailed description of the model and the noise that was added is given in appendix B.

The distribution in the input space that was obtained is shown in figure 4.3. Since the outputs of the fuzzy relations, α and q_p, cannot be measured, they are estimated from the states of the model. Therefore, in addition to S and X, P and V are also recorded. □

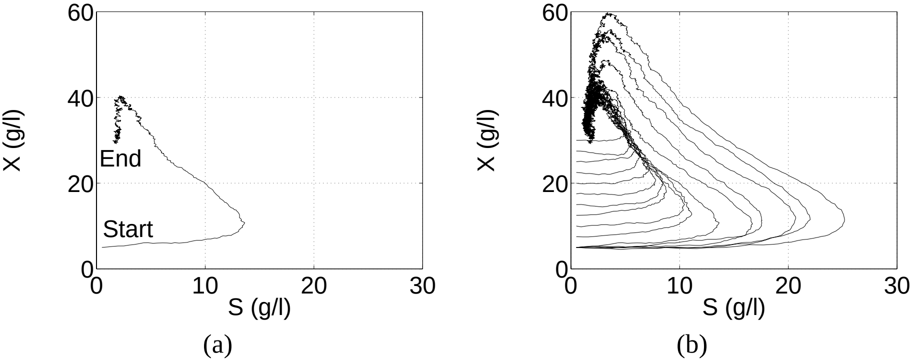


Figure 4.3: *Input space distribution of a typical batch run (a) and input space distribution of experiments*

4.3 Subprocess behavior estimation

Data preprocessing is an almost unavoidable step during process modeling. Obvious approaches such as filtering, scaling or data reduction can be applied, but more importantly, estimation techniques provide ways to obtain information about variables or parameters that cannot be measured directly. In hybrid modeling, this is the case for many subprocess parameters, such as reaction rates. Estimates of these parameters can then be used during the identification process. Over the years, much research has been done and numerous papers and text books have appeared on the subject (see for example (Eykhoff, 1974; Luyben, 1990; Ramirez, 1994; Seinfeld and Lapidus, 1974)). For modeling of chemical processes, the techniques should be able to estimate nonlinear and time varying variables or parameters. This section will discuss two approaches well suited for the problem: the extended Kalman filter and the PI-estimator.

4.3.1 Kalman filtering

The well-known Kalman filter (Kalman, 1960) was originally developed as a state estimator, but can also be used for parameter estimation. The discussion of the filter will be presented in discrete form (since measurements mostly will be available in discrete form). Consider the following linear system:

$$\mathbf{x}_{k+1} = \Phi \mathbf{x}_k + \Psi \mathbf{u}_k + \mathbf{w}_k \tag{4.11}$$

$$\mathbf{y}_k = H \mathbf{x}_k + \mathbf{v}_k \tag{4.12}$$

where $\mathbf{x}_k$ is the process state vector at time step k, $\mathbf{u}_k$ is the process input vector, Φ, Ψ are state transition and input transition matrices, respectively, $\mathbf{y}_k$ is the model output vector, H is the measurement matrix and $\mathbf{w}_k, \mathbf{v}_k$ are vectors assumed to be a sequence of white noise with zero crosscorrelation. The covariance matrices for these noise vectors are given by

$$E[\mathbf{w}_j \mathbf{w}_i^T] = \begin{cases} Q_{ij} & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{4.13}$$

$$E[\mathbf{v}_j \mathbf{v}_i^T] = \begin{cases} R_{ij} & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{4.14}$$

Q and R are diagonal matrices. Assume an estimate of the process state $\hat{\mathbf{x}}_k^-$ is available. The hat denotes estimate and the super minus indicates that it is an estimate prior to assimilating the measurement at time step k. Let

$$\mathbf{e}_k^- = \mathbf{x}_k - \hat{\mathbf{x}}_k^- \tag{4.15}$$

$$P_k^- = E[\mathbf{e}_k^- \mathbf{e}_k^{-T}] \tag{4.16}$$

When an estimate is made, an observer equation can be used to improve the estimate in the following way:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + G_k \mathbf{i}_k \tag{4.17}$$

$$\mathbf{i}_k = \mathbf{y}_k - H_k\hat{\mathbf{x}}^- \tag{4.18}$$

with $\mathbf{i}_k$ the innovation and G_k the *Kalman gain* at time step k. The Kalman gain is given by the following equation(Brown and Hwang, 1992):

$$G_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{4.19}$$

and the corresponding estimation covariance matrix for $\hat{\mathbf{x}}_k$ is given by:

$$P_k = (I - G_k H_k) P_k^- \tag{4.20}$$

The optimal estimate $\hat{\mathbf{x}}_k$ can thus be calculated. To determine the estimate at the next time step, the process state can be projected ahead using equation 4.11. Its error can be written as:

$$\begin{aligned} \bar{\mathbf{e}}_{k+1} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^- && (4.21)\\ &= (\Phi\mathbf{x}_k + \Psi\mathbf{u}_k + \mathbf{w}_k) - \Phi\hat{\mathbf{x}}_k - \boldsymbol{\Psi}\mathbf{u}_k && (4.22)\\ &= \Phi\mathbf{e}_k + \mathbf{w}_k && (4.23) \end{aligned}$$

which can be used to calculate the estimation error covariance matrix:

$$P_{k+1}^- = \Phi P_k \Phi_k^T + Q_k \tag{4.24}$$

The estimate and its covariance are now available, so the Kalman gain can be calculated for this estimate to obtain the optimal estimate. This process is repeated for each time step.

The filter equations were derived for linear systems. Most systems under investigation here however, are nonlinear. The filter can be adjusted to be able to deal with nonlinear systems by introducing a state transition matrix $\boldsymbol{\Phi}(\mathbf{x}_k, \mathbf{u}_k)$ that calculates the state transition for each time step. The same can be done for the input transition matrix. This results in a local linearization of the system. Equation 4.11 then becomes:

$$\mathbf{x}_{k+1} = \Phi(\mathbf{x}_k, \mathbf{u}_k)\mathbf{x}_k + \Psi(\mathbf{x}_k, \mathbf{u}_k)\mathbf{u}_k + \mathbf{w}_k \tag{4.25}$$

The standard Kalman filter configuration is shown in figure 4.7.

Parameter estimation

Parameters or parameter vectors of the process model can also be estimated using the Kalman filter. This can be done by introducing them as additional system states. This way, they can easily be incorporated in the general filter structure. Additional state equations of the following form are introduced:

$$\frac{d\boldsymbol{\theta}}{dt} = 0 \tag{4.26}$$

in which $\boldsymbol{\theta}$ denotes the parameter(vector). By setting the derivative zero, the parameters are assumed to be constant. However, the filter is also able to estimate time-varying parameters using this approach. The quality of the estimates of time-varying parameters depends on the tuning of the filter.

Observability and robustness

In order to determine the state of the process uniquely form a series of measurements over the interval $[t_0, t_f]$, the process needs to be observable. Consider the system described by equations 4.11 and 4.12 and do not consider the noise vectors (this is the noiseless process model). Because of its recursive structure, the state equation can also be written as:

$$\mathbf{x}_{k+1} = \Phi^k \mathbf{x}_0 + \Psi^k \mathbf{u}_0 \tag{4.27}$$

with $\mathbf{x}_0$ and $\mathbf{u}_0$ the state and input vector at $t = 0$. From the measurement equation, the process output vector can be represented by:

$$\mathbf{y}_k = H\Phi^k \mathbf{x}_0 \tag{4.28}$$

The linear system is called *observable* when a unique representation of $\mathbf{x}_0$ in terms of $\mathbf{y}_k$ is possible. When the system is observable, all the estimates can be determined uniquely with the measurement system. The observability is judged by evaluating the rank of the observability matrix:

$$O_{0,k_f} = \sum_{i=0}^{k_f} (\Phi^i)^T H^T H \Phi^i \tag{4.29}$$

with k_f the number of measurements. The system is observable if the observability matrix is non-singular. For non-linear systems, the observability can be calculated by replacing Φ with $\Phi(\mathbf{x}_k, \mathbf{u}_k)$. Instead of using Φ^k, the product of the various state transition matrices from time step 0 to time step k is used:

$$\prod_{i=0}^{k} \Phi(\mathbf{x}_i, \mathbf{u}_i) \tag{4.30}$$

The robust integrity of the Kalman filter can be verified by determining a stability region (see Ramirez (1994)). Consider the measurement perbutation slope n for the measurement configuration of the filter and assume that the measurement covariance matrix R is diagonal. The Kalman filter is stable for all n such that

$$n > \frac{1}{2}(1 - \lambda_0) \tag{4.31}$$

where λ_0 is the minimum eigenvalue of

$$(R^{1/2} G^T Q^{-1} G R^{1/2})^{-1} > \mathbf{0} \tag{4.32}$$

The criterion imposes an upper bound on the gains in the Kalman filter. Gains much larger than the process perbutations can cause the filter to respond too sensitive, resulting in instability. A value of equation 4.31 close to the maximum value of $1/2$ indicates a small region of robust stability. If a high degree of integrity and robustness is required, then λ_0 must be maximized by the choice of Q and R.

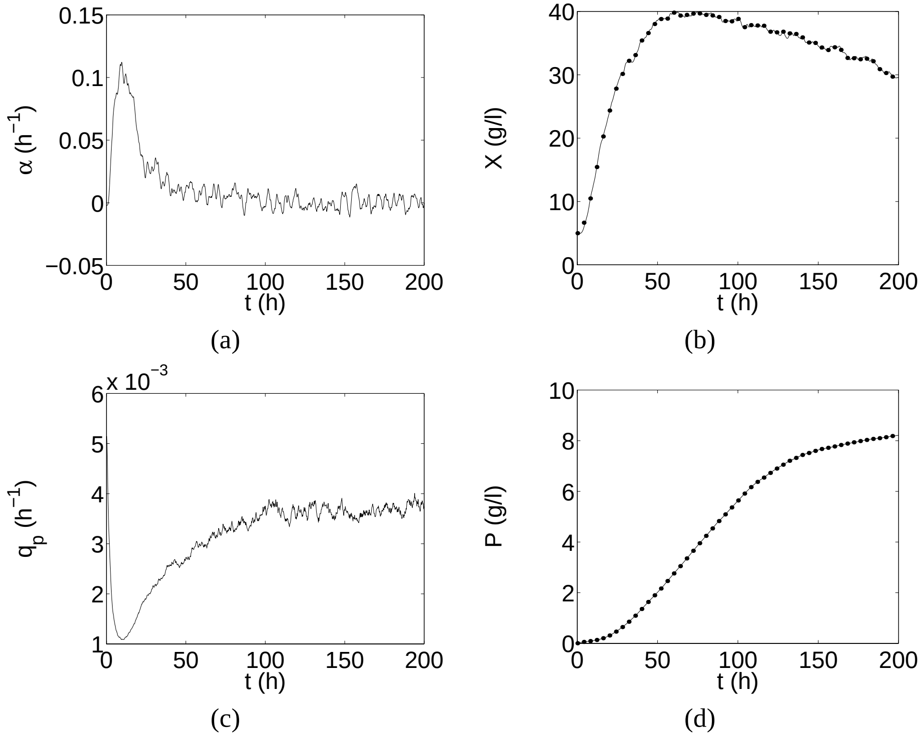


Figure 4.4: *Kalman estimates of α (a), X (b), q_p (c) and P (d). Lines indicate estimates, dots measurements.*

Filter tuning

Although matrices Q and R are defined as covariance matrices, they can be used to tune the Kalman filter and improve the performance. This is a trial and error process. The innovation (equation 4.18) is an important tool for tuning the filter. Since the noise vector $\mathbf{w}_k$ in equation 4.11 is assumed to be a sequence of white noise, it is a good indication that the filter is tuned well if the innovation is reduced to a sequence of white noise. This can be checked by calculating the autocorrelation ρ in the innovation.

Example 4.3 A simple Kalman filter can be designed to estimate α and q_p for the bioreactor. The measurements of X and P are sufficient for the system to be observable. The system can be represented as:

$$\begin{bmatrix} \dot{X} \\ \dot{P} \\ \dot{\alpha} \\ \dot{q}_p \end{bmatrix} = \begin{bmatrix} (\alpha - D) & 0 & 0 & 0 \\ q_p & -(D+K) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ P \\ \alpha \\ q_p \end{bmatrix} \tag{4.33}$$

in which

$$D = \frac{F}{V} \tag{4.34}$$

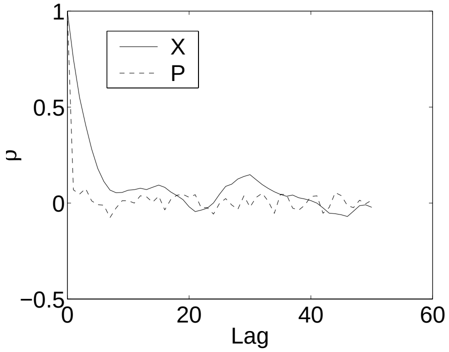


Figure 4.5: *Autocorrelation in Kalman filter innovation as a function of the lag*

The measurement matrix H for this system is given by:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.35}$$

and the innovation matrix is defined as:

$$\mathbf{i} = \begin{bmatrix} X \\ P \end{bmatrix} - \begin{bmatrix} \bar{\hat{X}} \\ \bar{\hat{P}} \end{bmatrix} \tag{4.36}$$

in which X and P are the measurements of the biomass concentration and the product concentration, while $\bar{\hat{X}}$ and $\bar{\hat{P}}$ are their estimates.

The filter was tuned by setting the diagonals of the measurement noise covariance matrix R and process noise covariance matrix Q as follows:

$$Q = \begin{bmatrix} 1e-4 & 0 & 0 & 0 \\ 0 & 1e-4 & 0 & 0 \\ 0 & 0 & 5e-6 & 0 \\ 0 & 0 & 0 & 7e-5 \end{bmatrix}, R = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.01 \end{bmatrix} \tag{4.37}$$

These settings were used for runs ID1 to ID16 and corresponded with an average stability border of 0.49, which indicates a small region of robust stability. The estimates, however, were found to be of acceptable quality. Estimation results for run ID1 are shown in figure 4.4 and the autocorrelation in the innovation is shown in figure 4.5. For clarity, only some of the measurements are shown. □

4.3.2 PI-estimation

A simple alternative to the Kalman filter for nonlinear parameter estimation is the PI estimator which is similar to the PI feedback controller (Van Lith *et al.*, 2001). It has many similarities with the Kalman filter, but is much easier to set up. Consider the nonlinear parameter of the system under study as a system input. The effect on the behavior of the model

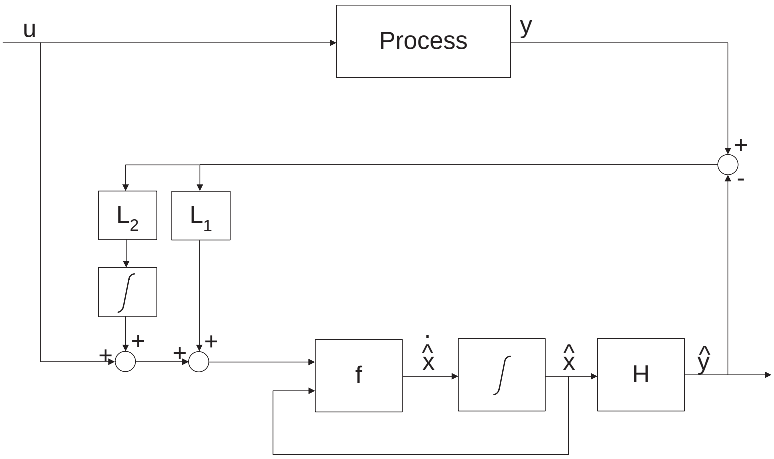


Figure 4.6: *PI-control scheme for parameter estimation*

of this new ”input” is dictated by the model equations (as with normal system inputs), while the nonlinear time-varying behavior of the parameter itself needs to be determined by some other means.

Since the parameter is viewed as an input, this input can be used to *control* the behavior of the model. This behavior needs to be the same as the actual physical behavior of the system described by measurements. The actual behavior can be seen as a trajectory that the model needs to follow. The correct nonlinear estimation of this new ”input” will result in correct model behavior.

This task can be accomplished by designing a simple feedback controller which controls the behavior of the model by manipulating the nonlinear parameter. The controller output serves as an estimate for the behavior of the nonlinear parameter. This is only true if the complete initial state is known. The controller takes the difference between the controlled model output and the desired trajectory (i.e. the innovation) as input. The controller is chosen to be a familiar PI-controller for fast response and elimination of offset:

$$\boldsymbol{\theta} = L_1\mathbf{i} + L_2 \int \mathbf{i}dt \tag{4.38}$$

where $\boldsymbol{\theta}$ is the parameter vector and $\mathbf{i}$ is the innovation. L_1 and L_2 are diagonal matrices of appropriate dimensions (with respect to the parameter vector $\boldsymbol{\theta}$).

The ”control scheme” is shown in figure 4.6. In this figure, the ”control action” is concatenated to the input vector of the system to obtain the model input vector $\mathbf{u}^*$:

$$\mathbf{u}^* = \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\theta} \end{bmatrix} \tag{4.39}$$

Following more conventional PI-controller notation, the tuning parameters, the elements on the diagonals of L_1 and L_2, can be written in terms of the controller gain K and integral time constant τ_i as follows:

$$l_{1,ii} = K \tag{4.40}$$

$$l_{2,ii} = \frac{K}{\tau_i} \tag{4.41}$$

Comparison with Kalman filtering

The controller interpretation of parameter estimation shows some analogies with conventional parameter estimation using state estimators. If a state estimator, such as a Kalman filter (or Luenberger observer) is used to make parameter estimations, the parameter is introduced as an additional state variable (see equation 4.26). The filter configuration is shown in figure 4.7.

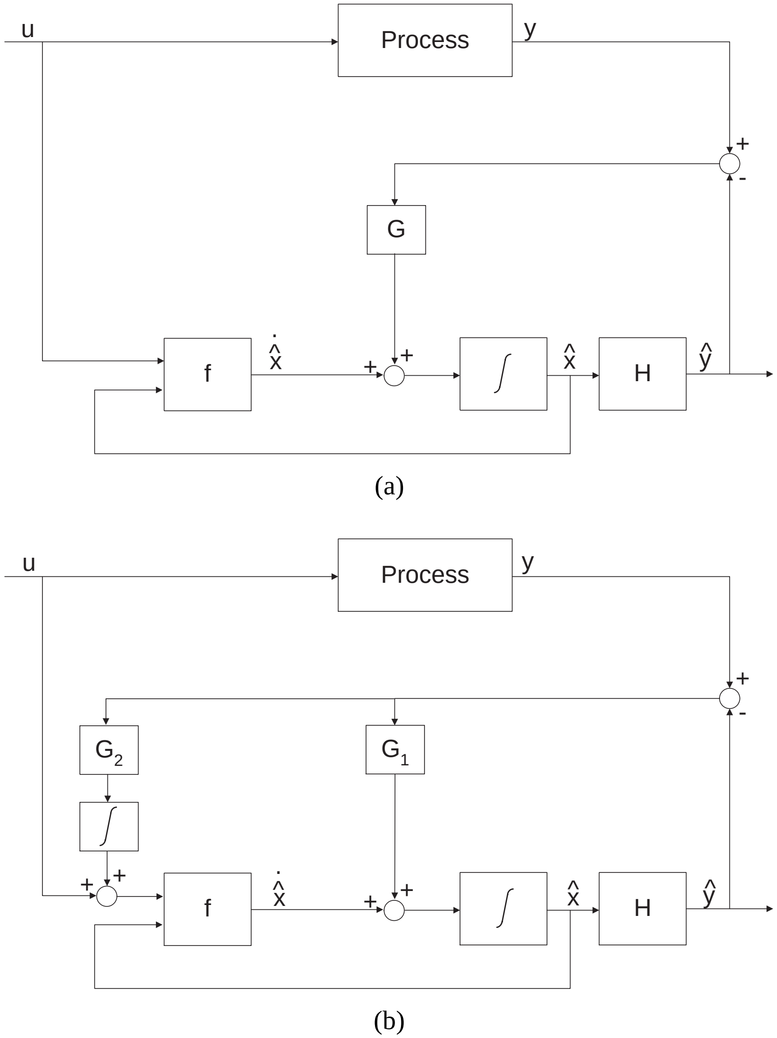
In figure 4.7 (a), equation 4.26 is a part of the function block f. Rearranging the equations yields the configuration shown in figure 4.7 (b). The extra integral block is the state equation for the parameter (equation 4.26), which is taken outside the function block f. This results in G being replaced with a different gain matrix G_1 and the introduction of an additional gain matrix G_2. Thus figure 4.7 (a) and (b) are different representations of the same Kalman filter.

The configurations in figure 4.6 and figure 4.7 (b) are quite similar with respect to integral action. But there are also some differences. Consider a single parameter θ. First of all, in the controller configuration there is an additional proportional term that adjusts the estimates for θ. In the Kalman filter, this term is not present. Secondly, the matrix G_1 of the Kalman filter in figure 4.7 (b) adjusts the estimates of the complete state $\hat{\mathbf{x}}$. These adjustments are made in addition to the adjustments that are only made to parameter θ. So the estimates of θ, propagated through the model f, result in estimates of $\hat{\mathbf{x}}$, which are subsequently adjusted by the innovation $\mathbf{i}$ and G_1. Thus this mechanism can correct estimation errors in $\hat{\mathbf{x}}$ caused by the estimates of θ. This mechanism is not present in the controller configuration. However, this can be seen as an advantage, because in the controller configuration, $\hat{\mathbf{x}}$ is only calculated accurately if the estimates of θ are accurate.

Obviously, the calculation of the gain matrices G, L_1 and L_2 differs. The different gains L_1 and L_2 are fixed and are set using PI tuning procedures. In practice, tuning of the Kalman filter is not done by adjusting G, which is calculated by the Kalman equations, but by adjusting the noise covariance matrices Q and R, from which G is calculated.

Controller configuration

When the controller configuration is used in multiple parameter estimation problems, appropriate ”control loops” need to be chosen. In other words, for each parameter which state is used to obtain the estimates has to be decided. Analysis has to be done to determine the



(a)

(b)

Figure 4.7: *Standard (a) and alternate (b) Kalman filter configuration*

interactions between the various parameters and states, so that sensible control loops can be chosen. Using the *relative gain array* (see Bristol (1966)), pairs of inputs θ_j (the parameters that have to be estimated) and outputs x_i (the states that are ”controlled” by the parameter estimates) can be selected in order to minimize the amount of interaction among the resulting loops.

If the static gain matrix G_{static} of the transfer functions of the system is available, the relative gain array Λ can also be calculated as follows (Roffel and Chin, 1987):

$$\lambda_{ij} = g_{ij}(G_{static}^{-1})_{ij}^{T} \tag{4.42}$$

in which g_{ij} denotes the ij-th element of the static gain matrix G_{static} and λ_{ij} is the ij-th element of the relative gain array Λ.

The relative gain array provides a measure of the interaction based on steady-state considerations. Therefore, the rule given above for the selection of loops does not guarantee that the dynamic interaction between the loops will also be minimal. The relative gain array can be replaced by its dynamic counterpart to account for this (Roffel and Chin, 1987). The interaction can then be calculated for different frequencies.

Example 4.4 The parameters α and q_p of the bioreactor can be estimated by two PI-controllers. The model f (see figure 4.6) is formed by the mass balance for the biomass concentration (equation 4.1) and the product concentration (equation 4.3); both states are measured and the parameters α and q_p directly influence the behavior of these states.

Form the equations, it is obvious to control the biomass concentration X with α and the product concentration P with q_p. This is also confirmed by the relative gain array Λ. Define the manipulated variable vector as

$$\begin{bmatrix} \alpha \\ q_p \end{bmatrix} \tag{4.43}$$

and the controlled variable vector as

$$\begin{bmatrix} X \\ P \end{bmatrix} \tag{4.44}$$

To determine the static open loop gain matrix G_{static}, the nonlinear dynamic system can be linearized using a first order Taylor expansion and transformed to deviation variables, assuming stationary operation. Although the system is non-stationary, the only purpose of the relative gain array and thus the open loop static gain matrix is to determine which input should control which output, so this assumption can be made. The linearization resulted in the following static gain matrix:

$$G_{static} = \begin{bmatrix} \frac{-X_0}{\alpha_0 - F_0/V_0} & 0 \\ \frac{q_{p0}}{F_0/V_0 + K}\frac{-X_0}{\alpha_0 - F_0/V_0} & \frac{X_0}{F_0/V_0 + K} \end{bmatrix} \tag{4.45}$$

in which the subscript ”$_0$” denotes the working point where the linearization was made. Element ij of G_{static} is the open loop static gain of output i with respect to input j. Using equation 4.42, the relative gain array becomes:

$$\Lambda = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.46}$$

The relative gain array is independent of the point of linearization and shows that X should be matched with α and P should be matched with q_p.

Estimator type	*Settings*
PI-Estimator α	$K = 0.06, \tau_i = 2$
PI-Estimator q_p	$K = 0.15, \tau_i = 10$

Table 4.2: *PI-Estimator settings for bioreactor problem*

The innovation vector is the same as for the Kalman filter. The two PI-estimators were tuned manually. The model structure allows the estimators to be tuned separately. First, the controller for X is tuned. Since q_p has no influence on X, the controller for P has no influence on X, which means that this can be done. If this controller is tuned well, the error in the estimates of P is only caused by the error in q_p. The controller for P can be tuned subsequently.

Estimator settings are given in table 4.2. These settings were used for experiments ID1 to ID16. The results of experiment ID1 are shown in figure 4.8.

The estimates are good; the trajectories of X and P are followed closely. The Kalman filter adjusts the estimates of α as well as the estimates of X to minimize the innovation $\mathbf{i}$, whereas the PI-estimator only uses the estimates of α. As a result, the noise level in the estimates of α is somewhat higher than in the estimates made by the Kalman filter.

The PI-estimator only estimates the biomass X well if α is estimated well, making the estimates of α plausible. A similar statement can only be made for the Kalman filter in example 4.3 if the tuning parameters (the elements of the process noise covariance matrix Q) corresponding to the

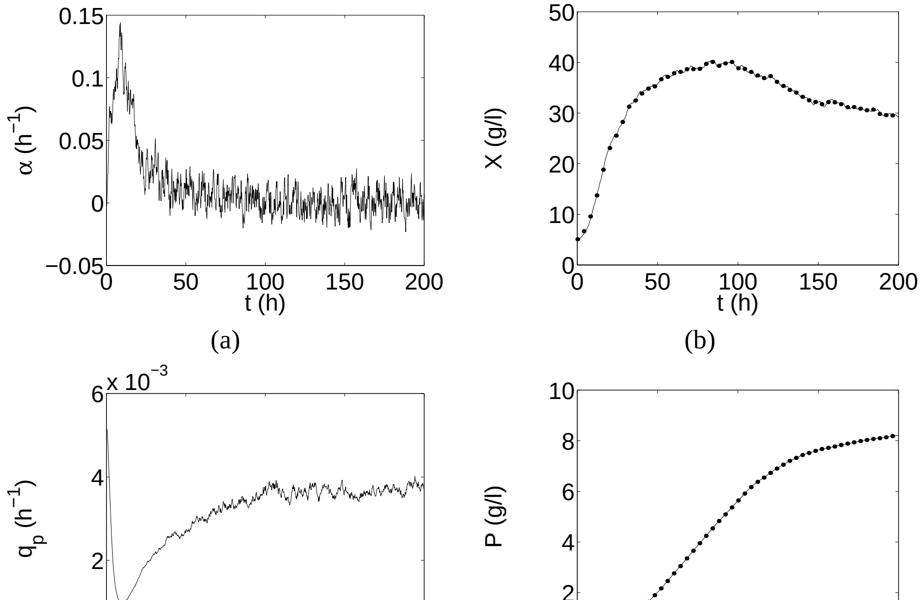


Figure 4.8: *PI-estimates of α (a), X (b), q_p (c) and P (d). Dots measurements, line estimates.*

measured states are set at low values, although in this case, some direct correction on the measured states remains. □

4.3.3 Remarks

With regard to the computational structure, the PI-estimator shows many resemblances with Kalman filtering. Therefore, it can be a good alternative to Kalman filtering. The results presented in the example were comparable with Kalman filter estimates. Although the Kalman filter has versatile application possibilities in combined state and parameter estimation, the PI-estimator has the advantage that it is simple, easy to use and easy to tune for simple single or multiple parameter estimation problems and therefore may be preferred over Kalman filtering.

4.4 Submodel identification

Submodel identification involves the design of the equations of the hybrid model. These include the accumulation balances, algebraic equations and the fuzzy equations. The identification of the balances and the algebraic equations done in the same fashion as with first principles modeling. This section will focus on the identification of the fuzzy equations.

In hybrid modeling, TSK type fuzzy models are used. The design of these models involves the determination of the number of local linear models (the number of rules), the parameters of these models or ”hyperplanes” and the operating range for which they are valid (the premise part). The parameters of the fuzzy equations are derived from the input-output data.

Research efforts in the field of identification of TSK fuzzy models has been enormous, as is the number of algorithms. They vary from manual design, tree search methods (Nelles, 1997) to an abundance of combinations of soft computing algorithms. It is unfeasible to present a thorough evaluation of the different techniques. However, three different approaches representing three different classes of identification algorithms are presented in order to be able to give some guidance with regard to building hybrid models. They are fuzzy clustering, genetic algorithms and neuro-fuzzy methods.

The calculation of the consequent part parameters using the three approaches is similar. The approaches differ in the way the rule structure and antecedent membership functions are determined. Fuzzy clustering is the most flexible. The presented approach determines the number of rules and the premise part parameters with minimal a priori knowledge. Genetic algorithms require more initial information. This approach searches for premise part parameters in a predefined search space. In addition, the number of rules needs to be specified. Finally, the neurofuzzy approach which is presented requires a complete initial fuzzy model, which is optimized to provide a desired input-output mapping.

For basic fuzzy set theory concepts, the reader is referred to appendix A.

4.4.1 Fuzzy clustering

The basic idea behind clustering is to divide a set of objects into self-similar groups (or clusters). This similarity is often defined as a distance norm. Clustering methods are usually based on assumptions about the geometry of the clusters, which include spheres, lines, hyperplanes, ellipsoids, etc. Various clustering methods can be used to develop fuzzy models. The goal of this section, however, is to present a useful approach for use with hybrid modeling. More detailed information about fuzzy clustering can be found in Jain and Dubes (1988), Pal *et al.* (1997) or Babuška (1996).

Since TSK models are used, the clustering algorithm must search for linear subspaces. Each cluster can then be represented by a rule of the fuzzy model. Cluster prototypes can be defined as linear subspaces (lines, planes, hyperplanes), while the similarity measure can be defined as the distance to such a prototype. Algorithms that use this approach include fuzzy c-varieties (Bezdek, 1981), fuzzy c-elliptotypes (Bezdek *et al.*, 1981) and fuzzy regression models (Hathaway and Bezdek, 1993).

Gustafson-Kessel clustering

Gustafson-Kessel clustering (GK-clustering) has found to be an effective tool for building TSK models (Babuška and Verbruggen, 1994; De Bruin and Roffel, 1996; Zhao *et al.*, 1994). GK-clustering uses the covariance matrix of a cluster during the calculation of the distance measure. The advantage of GK-clustering is that the clusters can have different shapes (while they all have the same volume). This gives the GK-algorithm more flexibility in describing complex systems. In addition, it is insensitive to scaling of the data or initialization (Babuška, 1996).

The clustering algorithm can be formulated as follows. Let $X = \{x_j | j = 1, \ldots, N\}$ be a set of feature vectors in $\Re^n$, where N is the number of measurements and n is the dimension of the input-output space. Let $P = (P_1, P_2, \ldots, P_K)$ be a K-tuple of cluster prototypes, characterized by a cluster center ν and a covariance matrix F. A partition of X into K fuzzy clusters will be performed by minimizing the objective function

$$J(P,U;X) = \sum_{i=1}^{K} \sum_{j=1}^{N} (u_{ij})^m d^2(x_j, P_i) \tag{4.47}$$

where $U = [\mu_{ij}]_{K \times N}, \mu_{ij} \in [0,1]$ is the fuzzy partition matrix, which denotes to what extend a feature vector ”belongs” to each cluster. This matrix satisfies the following conditions:

$$0 < \sum_{j=1}^{N} \mu_{ij} < N, \forall i \tag{4.48}$$

$$\sum_{i=1}^{K} \mu_{ij} = 1, \forall j \tag{4.49}$$

The minimization of the objective function is performed iteratively. Repeat for $l = 1, 2, \ldots,$ given the initial fuzzy partition matrix $U^{(0)}$ (which is initialized randomly), the number of clusters K and the termination tolerance ε:

- Compute the cluster centers:

$$\nu_i^{(l)} = \frac{\sum_{j=1}^{N} \mu_{ij}^m x_j}{\sum_{j=1}^{N} \mu_{ij}^m},\ 1 \leq i \leq K \tag{4.50}$$

- Compute the new covariance matrices:

$$F_i^{(l)} = \frac{\sum_{j=1}^{N} \mu_{ij}^m (x_j - \nu_i)(x_j - \nu_i)^T}{\sum_{j=1}^{N} \mu_{ij}^m},\ 1 \leq i \leq K \tag{4.51}$$

- Compute the distances of the features to the cluster centers $d^2(x_j, P_i)$:

$$d^2(x_j, P_i) = (x_j - \nu_i)^T |F_i|^{1/n} F_i^{-1} (x_j - \nu_i),\ 1 \leq i \leq K, 1 \leq j \leq N \tag{4.52}$$

- Compute the membership new grades:

$$\mu_{ij}^{(l)} = \frac{d^2(x_j, P_i)^{-1/m-1}}{\sum_{k=1}^{K} d^2(x_j, P_l)^{-1/m-1}},\ 1 \leq i \leq K, 1 \leq j \leq N \tag{4.53}$$

  in which m is the ”fuzzy exponent”, which influences the resulting partition. As m approaches 1 from above, the partition becomes hard. As m approaches ∞, the partition becomes maximally ”fuzzy”. For GK-clustering, m is usually set to 2.

  If $d^2(x_j, P_i) = 0$ for some i, then set $\mu_{ij} = 1$ and set $\mu_{ij} = 0, \forall i \neq k$.

until $\|U^{(l)} - U^{(l-1)}\| < \varepsilon$.

Structure optimization

The number of clusters K needs to be determined a priori. However, it is usually not possible to determine the optimal number of clusters beforehand. The number of clusters can be limited by merging ”compatible” clusters that show a certain degree of conformity. A suitable technique that can accomplish this is the modified compatible cluster merging (MCCM) algorithm (Kaymak and Babuska, 1995). This approach merges clusters based on cluster distances and covariance matrix eigenvectors.

Let the centers of two clusters be ν_i and ν_j. Let the eigenvectors of the two corresponding covariance matrices be $\{\phi_{i,1}, \ldots, \phi_{i,n}\}$ and $\{\phi_{j,1}, \ldots, \phi_{j,n}\}$ and arranged in descending order. The two criteria used for cluster merging are defined as:

$$c_{ij}^1 = |\phi_{i,n} \cdot \phi_{j,n}| \geq k_1, k_1 \text{ close to } 1 \tag{4.54}$$

$$c_{ij}^2 = \|\nu_i - \nu_j\| \leq k_2, k_2 \text{ close to } 0 \tag{4.55}$$

The first criterion states that if the clusters, viewed upon as hyperplanes, are almost parallel, then they should be merged. The second criterion states that if the two clusters are sufficiently close, then they should be merged. Evaluating these criteria for all clusters yields two matrices, $C^1[c_{ij}^1]$ and $C^2[c_{ij}^2]$, whose elements indicate the degree of similarity between clusters i and j.

Based on these matrices, a decision must be made whether to merge two clusters. To make this decision, C^1 and C^2 are transformed to $\tilde{C}^1$ and $\tilde{C}^2$ using two exponential membership functions. These membership functions indicate the degree of compatibility between two clusters and are defined as follows:

$$\tilde{c}_{ij}^1 = \exp \frac{-7(c_{ij}^1 - 1)^2}{(1-a)^2} \tag{4.56}$$

$$\tilde{c}_{ij}^2 = \exp \frac{-7(c_{ij}^2 - 1)^2}{(1-b)^2} \tag{4.57}$$

in which

$$a = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} c_{ij}^1 \tag{4.58}$$

$$b = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} c_{ij}^2 \tag{4.59}$$

This makes the decision algorithm problem dependent. The criteria for closeness and parallellity may partially compensate for each other. Two clusters that are very close but that are slightly non-parallel may need to be merged. This is also true for parallel clusters that lie somewhat apart. To account for this, $\tilde{C}^1$ and $\tilde{C}^2$ are combined to a matrix C^0 in the following way:

$$c_{ij}^0 = \sqrt{\tilde{c}_{ij}^1 \tilde{c}_{ij}^2} \tag{4.60}$$

To decide which clusters are to be merged, the matrix C^0 is thresholded. One or several groups of clusters can be merged. The merging is done transitively, which means that if the values of C^0 indicate that cluster i and j should be merged and that cluster j and k should be merged, all three clusters i, j and k are merged together. This is accomplished by relational clustering (Dunn, 1974; Yang, 1993):

```
$C^0 := C$, $i := 0$
repeat
  $i = i + 1$
  $C_i = C^0 \circ C_{i-1}$
until $C_i = C_{i-1}$
$C := C_i$
```
(4.61)

in which $C^0 \circ C_{i-1} = [c'_{ij}]$ with $c'_{ij} = \vee_{k=1}^{K} (c_{ik} \wedge c_{kj})$. In addition, $i = 1, \ldots, K$ and $j = 1, \ldots, K$.

The matrix C is now thresholded with a value γ. If $c^0_{ij} > \gamma$ then cluster i and j should be merged. Since c_{ij} always lies between 1 and 0, a threshold γ of 1 means that clusters are never merged, while a threshold of 0 means that clusters are always merged. A value of γ between 0.55 and 0.75 is recommended (Kaymak and Babuska, 1995).

Undesired results may be obtained by cluster merging if incompatible clusters are located between clusters that are compatible for merging. A heuristic step can solve this problem (Babuška, 1996). This step prevents merging of clusters if there is an incompatible cluster close to the mutual centers of the compatible clusters in the set M:

$$\min \max_{\nu_i \in M, \nu_k \notin M} d_{ik} > \max_{\nu_i, \nu_j \in M} d_{ij} \tag{4.62}$$

in which d_{ij} is the distance between cluster centers ν_i and ν_j projected on the input space:

$$d_{ij} = \|\text{proj}(\nu_i) - \text{proj}(\nu_j)\| \tag{4.63}$$

If this criterion is not met, the set of clusters M is not merged.

A fuzzy partition matrix U' can be calculated based on the partition matrix that was available before merging, by adding up the rows of U that correspond to the clusters that are merged. After merging, the data is reclustered with the new number of clusters. The new fuzzy partition matrix is used for initialization of the GK-clustering algorithm.

The clustering and cluster merging algorithm are performed consecutively until no more clusters can be merged, i.e. when $U = U'$.

Model derivation

The result of the clustering algorithm is a description of the fuzzy system. The number of rules is given by the number of clusters. The premise part is given in terms of the fuzzy partition matrix U, which represents the clusters by multi dimensional, point-wise defined membership functions in the input space. In essence, these membership functions represent the Degree Of Fire (DOF) of the rules. The cluster centers ν and their covariance matrices F represent the consequent part. A model description that is independent of the identification data must be derived in order to be able to use the model.

Premise part parameters

The antecedent membership functions can be derived in several ways. The degree of membership of a feature for a cluster can be directly calculated in the product space of the input variables by using the distance measure from the clustering algorithm. An alternative is using a multi-input version of a parametric membership function that is fitted to the point-wise definition, given by the fuzzy partition matrix. The shape of the membership function given by the fuzzy partition matrix is similar to sigmoidal membership functions, while the shape of the clusters is ellipsoidal. The multi-input membership function can then be defined as a

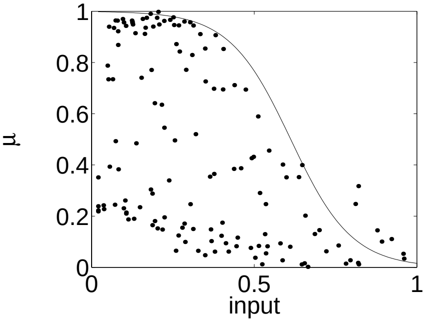


Figure 4.9: *Projection of fuzzy partition matrix and fit of parametric membership function*

sigmoidal membership function with an ellipsoidal basis. For a system with two inputs x_1 and x_2, the multi input membership function can be given by:

$$\mu = \frac{1}{1 + \exp -a(r-b)} \text{ with } r = \sqrt{\frac{(x_1 - x_1^*)^2}{c^2} + \frac{(x_2 - x_2^*)^2}{d^2}} \qquad (4.64)$$

with a, b, c, d function parameters and x_1^*, x_2^* the cluster center.

A disadvantage of multi dimensional membership functions is that their description is less transparent than the descriptions by single input membership functions. A more straightforward way is to project the multi dimensional point-wise membership functions onto the input variables. A parametric single input membership function can then be fitted through this projection. This is illustrated in figure 4.9.

The projection can be done directly onto the input variables, which results in an orthogonal projection. Projection is carried out for all of the input variables. After fitting, the clusters can be reconstructed by applying the AND intersection operator in the product space of the input variables, so that the Degree Of Fire is calculated.

The reconstruction is not exact. If clusters are located in a non-orthogonal way with respect to the input variables, the reconstruction may result in *decomposition errors*, illustrated in figure 4.10. The error can be reduced by ”rotating” the input space to obtain an orthogonal orientation, which involves transformation of the input variables. This can be done by using eigenvector information form the cluster covariance matrices (Babuška, 1996) or by using cluster boundary information (De Bruin and Roffel, 1996). However, this error can also be reduced by least-squares calculation of the consequent parameters, which will be explained later on. Some information may be lost by projecting non-orthogonally oriented clusters, but it provides a better possibility to interpret the model because the input variables are not transformed.

Consequent part parameters

The calculation of the consequent parameters involves determining the hyperplanes that describe the identification data for each cluster. The calculation is divided in two categories. In the first category, the parameters for each hyperplane are calculated directly from the cluster

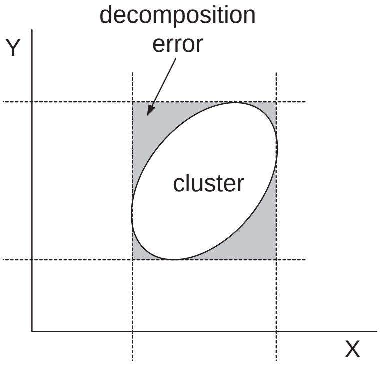


Figure 4.10: *Decomposition error caused by orthogonal projection*

center and eigenvector information from the covariance matrix. The normal of the desired hyperplane is the eigenvector that corresponds to the smallest eigenvalue of the covariance matrix. The parameters can be calculated from this eigenvector (Babuška, 1996) or from the remaining eigenvectors, which span the hyperplane (De Bruin and Roffel, 1996).

A disadvantage of direct calculation of the consequent parameters is that model errors can occur due to the decomposition error. This model error can be partially compensated for by estimating the consequent parameters using a least squares approach. The estimation problem can be approached as several independent weighted least squares problems (one for each cluster) or as a global least squares problem following from the weighted mean defuzzification equation (Babuška, 1996). The global least squares approach provides an optimal fuzzy model in terms of a minimal model error. However, due to the global approach, the estimates of the local models are biased. In this work, the weighted least squared approach is preferred because it provides local interpretation and analysis of the fuzzy model.

The weighted least squares approach assumes that each cluster represents a local linear model of the system. The membership values μ_{ik} of the fuzzy partition serve directly as the weights expressing the relevance of the data feature to the local model i. Let X denote a matrix containing the N input features in each row, let $\boldsymbol{y}$ denote a vector containing the corresponding output values and let W_i denote a diagonal matrix having the membership values μ_{ik} as its kth diagonal element:

$$X = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix}, \; \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \; W_i = \begin{bmatrix} \mu_{i1} & 0 & \cdots & 0 \\ 0 & \mu_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_{iN} \end{bmatrix} \tag{4.65}$$

The consequent parameters of rule i, $\boldsymbol{a}_i$ and b_i, are concatenated into a single parameter vector $\boldsymbol{\theta}_i$:

$$\boldsymbol{\theta}_i = \left[\boldsymbol{a}_i^T; b_i\right]^T \tag{4.66}$$

Appending a unitary column to X gives the extended input variable matrix X_e:

$$X_e = [X; \mathbf{1}] \tag{4.67}$$

If the columns of X_e are linearly independent and $\mu_{ik} > 0$ for $1 \le k \le N$, then

$$\boldsymbol{\theta}_i = \left[X_e^T W_i X_e\right]^{-1} X_e^T W_i \boldsymbol{y} \tag{4.68}$$

is the least squares solution of $\boldsymbol{y} = X_e \boldsymbol{\theta} + \epsilon$ where the kth data feature is weighted by μ_{ik}. A more efficient computer implementation is obtained if each row of X_e and $\boldsymbol{y}$ is first multiplied with $\sqrt{\mu_{ik}}$:

$$\tilde{X}_i = \begin{bmatrix} \sqrt{\mu_{i1}}\boldsymbol{x}_1^T \\ \sqrt{\mu_{i2}}\boldsymbol{x}_2^T \\ \vdots \\ \sqrt{\mu_{iN}}\boldsymbol{x}_N^T \end{bmatrix}, \ \tilde{\boldsymbol{y}}_i = \begin{bmatrix} \sqrt{\mu_{i1}}y_1 \\ \sqrt{\mu_{i2}}y_2 \\ \vdots \\ \sqrt{\mu_{iN}}y_N \end{bmatrix} \tag{4.69}$$

The parameters are then given by:

$$\boldsymbol{\theta}_i = \left[\tilde{X}_i^T \tilde{X}_i\right]^{-1} \tilde{X}_i^T \tilde{\boldsymbol{y}}_i \tag{4.70}$$

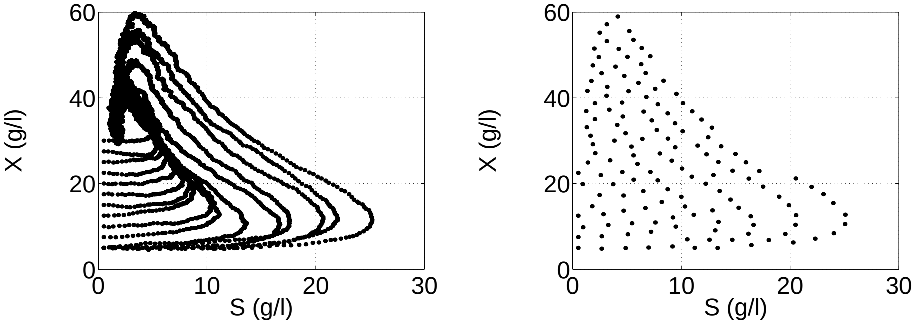
An α-*cut* can be incorporated in the calculation in order to prevent biasing by data features with low membership values for a certain cluster. The low membership of these features is already taken into account by the weighting. However, if a cluster is small than the number of features with low membership value may be high, which can bias the calculation. The α-cut α_c can be performed as follows:

$$\tilde{X}_i^\alpha = \{\boldsymbol{x}_{ik} | \boldsymbol{x}_{ik} \in \tilde{X}_i, \mu_{ik} > \alpha_c, \alpha_c \in [0,1]\} \tag{4.71}$$

$$\tilde{\boldsymbol{y}}_i^\alpha = \{y_{ik} | y_{ik} \in \tilde{\boldsymbol{y}}_i, \mu_{ik} > \alpha_c, \alpha_c \in [0,1]\} \tag{4.72}$$

Example 4.5 As an example, fuzzy clustering will be used to build a model of the net growth rate α as a function of the substrate S and biomass X for the hybrid model of example 4.1. Assume that an input-output data set has been created using experiments ID1 to ID16 (example 4.2) and that the PI-estimator designed in example 4.4 is available. This means that input-output data is available in the form of measurements of S and X and estimates of α.

The data set can be presented directly to the clustering algorithm. However, due to the nature of the batch runs, the data is not evenly distributed in the input space. During the latter hours of a batch, a sort of ”pseudo steady state” is obtained in which X and S do not vary as much as during the first hours. This causes data features to lie closer together in the area where X is high and S is low than data features in the area where X is low and S is high. Since the clustering approach uses least squares calculation, results may be biased because of this. This problem is solved by reduction of the data set: all features, for which the distance in the input space to an other feature is smaller than a pre-determined threshold, are removed. This threshold is set manually by finding a balance between the number of data features in the reduced data set and the average distance between these features. The threshold was set to 2 and the distribution of the resulting data set is shown in figure 4.11.



(a) (b)

Figure 4.11: *Input space distribution before (a) and after (b) reduction*

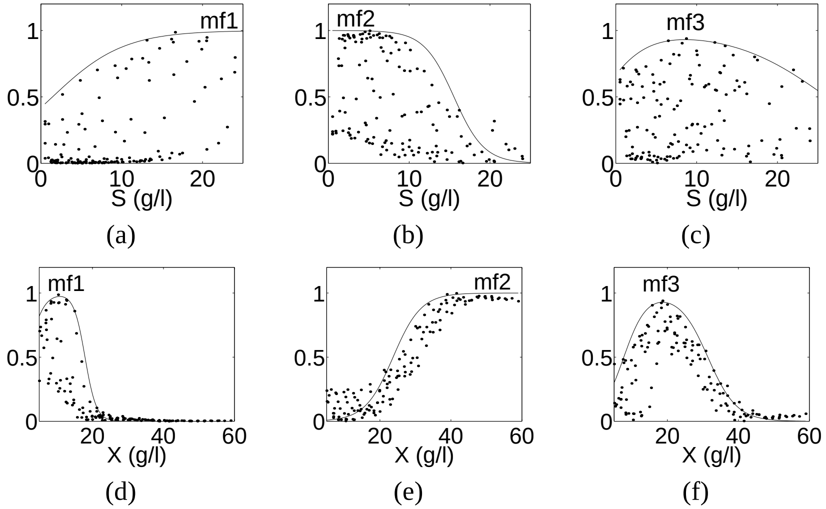
The reduced data set was clustered and the threshold γ was set to 0.4. The initial number of clusters was 10. This number was reduced to three by the merging algorithm. The next step is to project the fuzzy partition matrix onto the S and X axis, so that parametric membership functions can be determined. This is shown in figure 4.12.

The fuzzy model can now be completed by calculating the consequent part parameters using the least squares approach. For the α-cut, α_c was set to 0.5. This yields the following local linear models:

$$y_1 = 0.0114S + 0.0048X - 0.0859 \tag{4.73}$$

$$y_2 = 0.0061S - 0.0001X - 0.0054 \tag{4.74}$$

$$y_3 = 0.0092S - 0.0028X - 0.0327 \tag{4.75}$$



(a) (b) (c)

(d) (e) (f)

Figure 4.12: *Fuzzy partition matrix projections and fitted parametric membership functions*

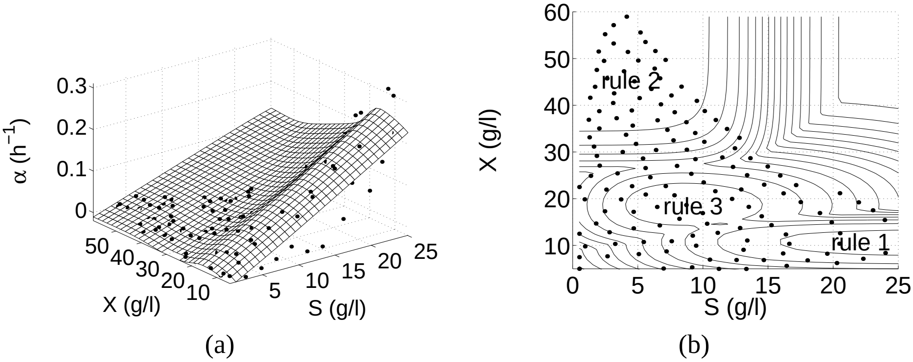


Figure 4.13: *Fuzzy relation for α and rule locations. Dots indicates identification data*

The fuzzy model can now be simply formulated as:

$$\begin{array}{l} \text{IF } S = \text{mf}_1 \text{ AND } X = \text{mf}_1 \text{ THEN } \alpha = y_1 \\ \text{IF } S = \text{mf}_2 \text{ AND } X = \text{mf}_2 \text{ THEN } \alpha = y_2 \\ \text{IF } S = \text{mf}_3 \text{ AND } X = \text{mf}_3 \text{ THEN } \alpha = y_3 \end{array} \tag{4.76}$$

Figure 4.13 shows the function hyperplane and the locations of the rules. The Root Mean Squared Error (RMSE) with respect to the identification data is 0.0248. This error is mainly the result of a bad fit in the region where X is low; the RMSE with respect to data features in the region where X is high is 0.0091. A more complex fuzzy relation can provide a better description. However, the observed behavior of the net growth rate is not so complex that a more complex fuzzy relation is justified from a transparency point of view. In addition, the PI-estimator lags behind the actual value of the growth rate due to the feedback nature. Estimates of α are thus lower than the actual value. Therefore, the final evaluation of the fuzzy model should be done after integration in the hybrid model structure.

Although fuzzy logic is a black box technique, *a posteriori* analysis of the model shows that the rules represent three phases during a batch: an initial phase, in which the substrate concentration is still high and the biomass culture is growing rapidly (rule 1), a final phase which occurs at the end of a batch, in which not much substrate is present and the net growth rate is not very high (rule 2) and an intermediary phase (rule 3). □

4.4.2 Genetic algorithms

Genetic Algorithms (GAs) are well known for their optimization capabilities. Following basic Darwinistic propagation, the method is based on a ”survival of the fittest” principle, in which only the solution candidates with the best desirable properties (e.g. smallest error) from a ”population” will survive. The candidates that will survive are selected by evaluating their fitness value through the fitness function (similar to the objective function in more traditional optimization algorithms).

This section describes the basic concept of genetic algorithms and the techniques used in this work, for a more detailed discussion see (Goldberg, 1989).

When genetic algorithms are used to identify a fuzzy system, the identification is viewed upon as an optimization problem. Many applications of developing fuzzy systems with GAs have been reported (Back and Kursawe, 1995). Using GAs to set up a fuzzy system involves coding the problem into ”chromosomes” or ”strings” and setting up a fitness function (goal function). Since TSK models are used, the consequent part of the fuzzy model can be calculated using a least squares approach, if the premise part is available (see section 4.4.1). Therefore a hybrid identification approach is used: only the premise part of the fuzzy model is coded into chromosomes and optimized by the GA. In each iteration, the consequent parts of all candidates are calculated using the least squares approach, after which the fitness function is calculated. Since the local models in the consequent part of each rule are least squares optimal, no rule structure optimization is necessary. Optimization of the number of rules involves a more elaborate approach and significantly increases the search space.

In order to be able to code the problem, an initial model structure needs to be formulated. Based on the number of rules and membership functions in this structure, a binary string can be designed.

Premise part coding

Usually, the problem is coded into a binary string or ”chromosome” which is presented to the GA for optimization. Real-valued coding is less common, but also can be used. The coding of the premise part of the fuzzy system involves representing the parameters of the membership functions in a binary format. The representation chosen here is straightforward and taken from KrishnaKumar *et al.* (1995).

The first step is to map a single parameter to a binary string. To accomplish this, allowed maximum and minimum values of the parameter are chosen. In addition, the string length κ should be chosen. The parameter is discretized by mapping to the string from the minimum value to the maximum value with a linear mapping in between. For example, for a string length of 5, 00000 corresponds to the minimum value and 11111 corresponds to the maximum value. The string length is determined by selecting the resolution r, the number of discrete values in the mapping.

The discrete mappings of the different parameters of a membership function are concatenated to form a string that represents the membership function. Double sigmoid membership functions, which contain 4 parameters, will be used. Different membership function strings are then concatenated to form the string with length κ that represents the premise part, as shown in figure 4.14.

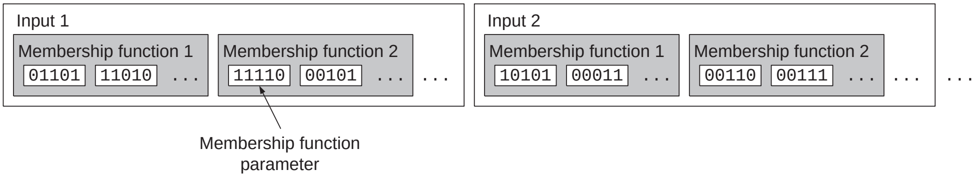


Figure 4.14: *GA coding of the premise part*

Optimization procedure

The identification involves three main steps, which are performed iteratively:

- Optimization of the consequent part using weighted least squares
- Evaluation of the ”fitness” of the population of possible solution candidates for the premise part
- ”Evolution” of the population, based on the population ”fitness”

The procedure is initialized by creating a ”population” of possible solution candidates, in this case a set of different premise parts for the system, coded as binary strings. This set is initialized randomly in the search space, spanned by the selected maximum and minimum values of the coded parameters. The size M of the population depends on the complexity of the problem and the number of parameters that need to be optimized. One way to assure that a suitable solution can be found is to create a relatively large population, but this can result in a large computational effort. Some researchers have found that a good rule of thumb is to select a population size of 25% of the string length or larger.

The first step in the procedure is to optimize the consequent part for each member in the population. The weighted least squares calculation is explained in section 4.4.1. The next step is to calculate the ”fitness” of each member in the population. Since a set of data needs to be described by the model, the goal is to minimize the model error with respect to the data. The ”fitness” is expressed in terms of this model error: the smaller the error, the fitter the member of the population. A squared error, common in optimization procedures, is used:

$$J = \frac{1}{2}\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2 = \frac{1}{2}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{4.77}$$

in which $\hat{\boldsymbol{y}}$ denotes the model output vector and $\boldsymbol{y}$ denotes the desired output vector.

”Evolution” of the population takes place by reproduction. Based on their fitness value, members of the population are selected for reproduction. Members that are selected to reproduce are copied and replace members with a low fitness value. The number of members M of a population remains unchanged. This means that good strings get more copies in the next ”generation”, which emphasizes the basic survival of the fittest concept.

Different strategies exist for the selection of population members. The best known procedure is the roulette wheel. Each member of the population has a roulette wheel slot sized in proportion to its fitness. The roulette wheel is spun and the string at which slot the roulette wheel stops is selected for reproduction. This means that population members with high fitness are more likely to be selected than strings with low fitness. Spinning is repeated until the next generation is full.

However, performance difficulties using the roulette wheel have been reported and improvements of this basic selection procedure have been presented (Goldberg, 1989). Tournament selection is an example of a procedure that does not have these restrictions. With tournament selection, a pair of individuals is selected at random from the population. The fittest of the two becomes a member of the next generation. This process is repeated until the next generation is full.

Two processes, crossover and mutation, take place during reproduction. The result of these processes is that population members with new properties can be created. A simple crossover process is performed in three steps. First, the newly selected strings are paired together at random. Second, an integer position n along every pair of strings is selected uniformly at random. Finally, based on a probability of crossover p_c, the pairs of strings undergo crossover at the integer position n along the string. This results in new pairs of strings that are created by swapping all the characters between and including characters 1 and n. Crossover can be at a single point, two points or at k points. The probability of crossover p_c is typically around 0.75.

Mutation is simply an occasional random alteration of a string character, which in this case involves changing a 1 to a zero and vice versa. Mutation is based on the probability of mutation p_m, which typically is around 0.001. The mutation operator helps in avoiding the possibility of mistaking a local minimum for a global minimum because it introduces a random search element in the vicinity of the population.

The three iteration steps - consequent part optimization, fitness evaluation and evolution - are repeated until a pre-determined number of generations G have been evaluated or when there is no improvement.

Example 4.6 To illustrate the GA approach, a fuzzy model will be identified for the net growth rate α as a function of the substrate concentration S and the biomass concentration X, as a part of the hybrid model for the bioreactor described in example 4.1. This will be done using the same reduced data set as was used in example 4.5.

The GA does not incorporate the optimization of the number of rules nor does it derive the model structure from the data. Therefore, several different identification runs were performed and analyzed in order to determine an acceptable number of rules. In each case, the rules were independent.

Double sigmoid membership functions were used in each experiment. In order to guarantee a certain level of transparency, the parameter description of this membership function was slightly changed (see equation A.4):

$$\mu = \frac{1}{1 + \exp -a(x-b)} \frac{1}{1 + \exp -c(x-d-\delta)} \tag{4.78}$$

# rules	r	κ	G	M	p_c	p_m	$RMSE$
2	64	96	250	101	0.77	0.0077	0.0197
3	64	144	250	101	0.77	0.0077	0.0132
4	64	192	250	101	0.77	0.0077	0.0129

Table 4.3: *GA setting and results*

with

$$\delta = (\frac{1}{c} - \frac{1}{a}) \ln (1/0.99 - 1) + b \tag{4.79}$$

If $d \geq 0$, then this membership function always reaches approximately 1. Parameters b and d determine the position of the membership function, while a and c determine the slope of the S-curves. For the coding of the premise part, maximum and minimum values for b and d were based on the input domain. Maximum and minimum values for a and c were chosen arbitrarily to guarantee a certain level of ”fuzziness”; high values lead to crisp sets, while low values lead to sets that are too fuzzy.

GA settings and results for three different runs are shown in table 4.3. All models describe the identification data well and model errors are in the same order of magnitude. Not surprisingly, the model with 4 rules has the smallest error.

The GA searches for a solution in a large search space. It determines starting points for the search itself, which makes it not very sensitive to initial parameter values. Not the initial values of the parameters affect the result, but the specified search space does. However, the GA does not take the identification data as a starting-point for model derivation, such as fuzzy clustering does. A priori information about the model structure has to be supplied. In addition, rules can be located in areas of the domain where no data is present. This means that, depending on the model structure, undesired extrapolation behavior can occur. In this example, this is the case for the fuzzy models with 3 and 4 rules (see figures 4.16 and 4.17).

Because of the undesired rule locations and extrapolation behavior of the more complex models, the 2-rule model is selected, despite its larger error. The model is shown in figure 4.15.

Figure 4.18 shows the parametric membership functions of the 2-rule model. For the α-cut in the consequent part parameter calculation, α was set to 0.5, which yields the following local linear models:

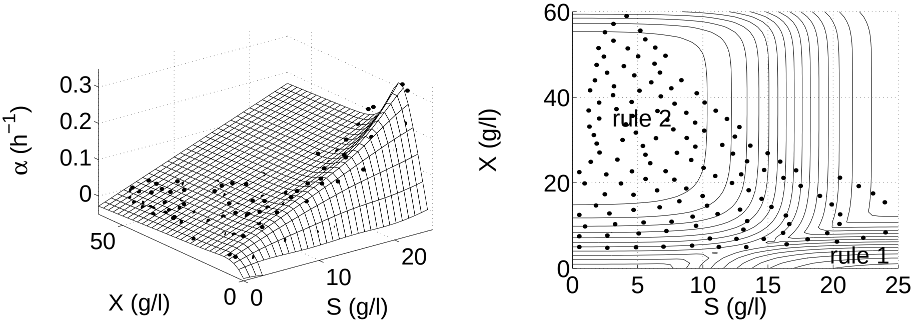
$$y_1 = 0.0060S + 0.0355X - 0.2264 \tag{4.80}$$

$$y_2 = 0.0079S - 0.0003X - 0.0070 \tag{4.81}$$

The fuzzy model can now be formulated as:

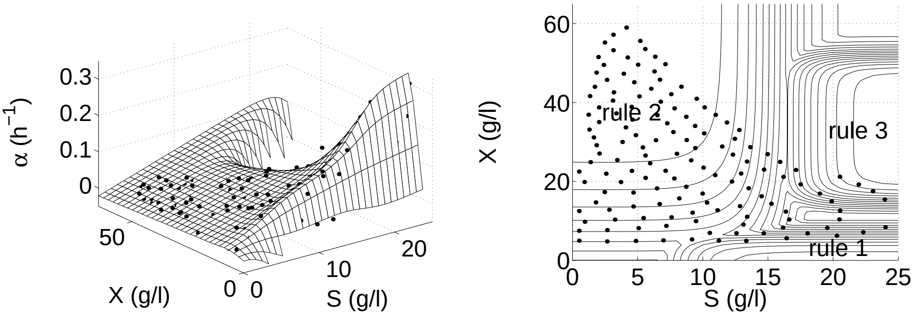
$$\begin{array}{l} \text{IF } S = \text{mf}_1 \text{ AND } X = \text{mf}_1 \text{ THEN } \alpha = y_1 \\ \text{IF } S = \text{mf}_2 \text{ AND } X = \text{mf}_2 \text{ THEN } \alpha = y_2 \end{array} \tag{4.82}$$

□



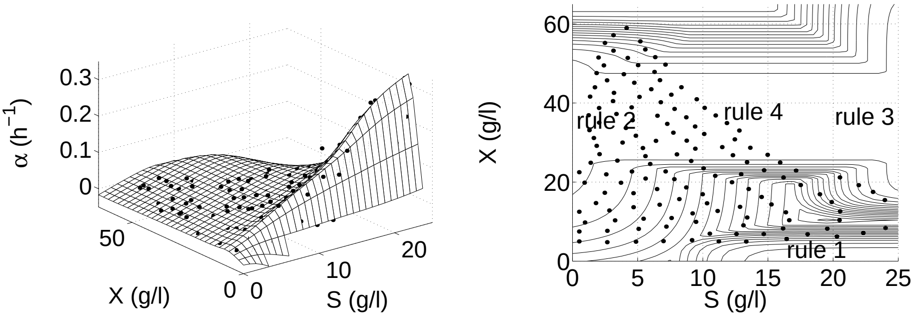
(a) (b)

Figure 4.15: *Fuzzy relation for* α *and rule locations for 2-rule model. Dots indicates identification data*



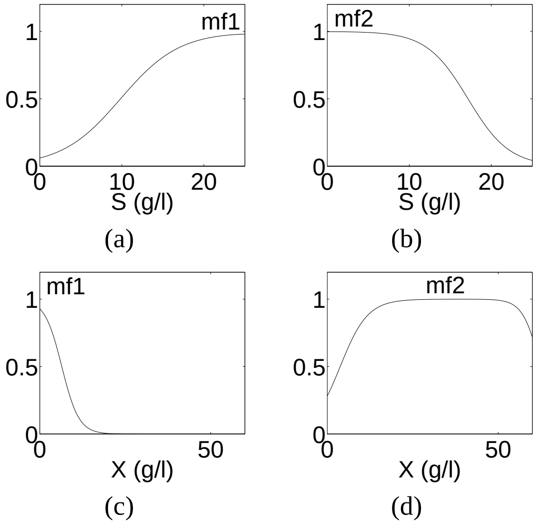
(a) (b)

Figure 4.16: *Fuzzy relation for* α *and rule locations for 3-rule model. Dots indicates identification data*



(a) (b)

Figure 4.17: *Fuzzy relation for* α *and rule locations for 4-rule model. Dots indicates identification data*



(a) (b) (c) (d)

Figure 4.18: *Parametric membership functions for the 2-rule fuzzy model*

4.4.3 Neurofuzzy systems

Neurofuzzy systems can be viewed upon as a combination of fuzzy systems and artificial neural networks (ANNs). The fuzzy inference system is implemented in the framework of these adaptive networks. This provides the possibility to use backpropagation learning rules, commonly used to train these nets. Several approaches have been developed in the past. Some of these have been applied within the context of hybrid modeling, such as NEFPROX (Nauck and Kruse, 1997) and ASMOD (Kavli, 1993), covering both linguistic and TSK type fuzzy models. Here, Jang's well-known Adaptive Network based Fuzzy Inference System (ANFIS) approach is discussed in more detail Jang (1993).

Jang interprets a TSK fuzzy system as an adaptive network, on which adaptive learning rules can be applied to optimize the system parameters. The interpretation consists of 5 layers. Consider a simple 2 input, 1 output TSK model consisting of two rules:

$$\begin{array}{l}\text{IF } x_1 = \mathrm{mf}_{11} \text{ AND } x_2 = \mathrm{mf}_{21} \text{ THEN } y = y_1 \\ \text{IF } x_1 = \mathrm{mf}_{12} \text{ AND } x_2 = \mathrm{mf}_{22} \text{ THEN } y = y_2\end{array} \tag{4.83}$$

in which

$$y_1 = p_1 x_1 + q_1 x_2 + r_1 \tag{4.84}$$

$$y_2 = p_2 x_1 + q_2 x_2 + r_2 \tag{4.85}$$

The ANFIS interpretation of this fuzzy model is shown in figure 4.19 and can be described as follows:

- Layer 1
  The nodes in layer 1 contain the premise part membership functions of the model. For

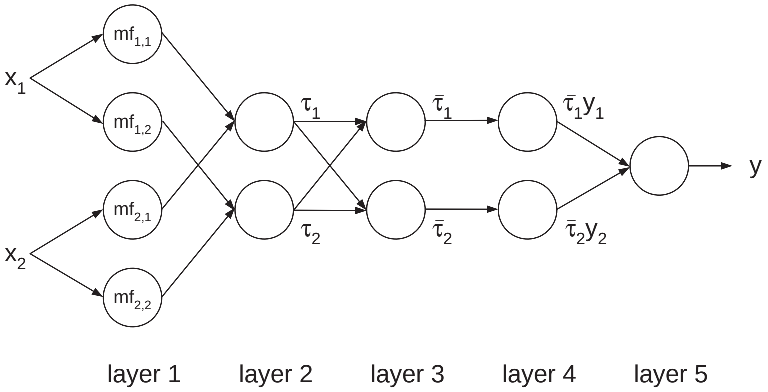


Figure 4.19: *ANFIS architecture for a 2 input, 1 output TSK model*

example, the node function for membership function j of input i is given by:

$$o_{ij} = \mathrm{mf}_{ij}(x_i) \tag{4.86}$$

- Layer 2
  The nodes in layer 2 determine the Degree of Fire τ (DOF) of the rules of the model. The DOF of rule i is calculated by the AND intersection operator:

$$\tau_i = o_{1i} \wedge o_{2i} \tag{4.87}$$

  In this case, the product is used as the AND operator.

- Layer 3
  Layer 3 calculates the weighted DOF of each rule. The weighted DOF for rule i is calculated by:

$$\bar{\tau_i} = \frac{\tau_i}{\sum_j^2 \tau_j} \tag{4.88}$$

- Layer 4
  Using the local linear models, the weighted output of each rule is calculated. The output of rule i is calculated as:

$$\bar{\tau_i} y_i = \bar{\tau_i}(p_i x_1 + q_i x_2 + r_i) \tag{4.89}$$

- Layer 5
  The output of the fuzzy model is calculated by layer 5:

$$y = \sum_i^2 \tau_i y_i \tag{4.90}$$

Parameter optimization is carried out using a so-called forward pass and a backward pass. During every iteration, the consequence part parameters of the fuzzy model are determined using the weighted least squares approach (see section 4.4.1). An alternate approach to the weighted least squares calculation is an iterative calculation of the solution to the least squares problem (Jang, 1993). In this approach, no matrix inversion is used which results in a computationally more efficient calculation. In this work, the weighted least squares approach is used. Calculation of the consequent parameters is called the forward pass.

In the backward pass, the error rates propagate backwards and the premise part parameters in layer 1 are updated by a standard gradient descent learning rule. The parameters in layer 1 are the only parameters that are updated by the learning rule. This means that the error in the output needs to be propagated to this layer.

Denote the output of a node in the ith position of the kth layer by o_i^k and let O_i^k be its function. Let $X = \{x_j | j = 1, \dots, N\}$ be a set of feature vectors in $\Re^n$, where N is the number of measurements and n is the dimension of the input-output space. In this example, $n = 3$. The error measure of the ANFIS network is defined as the sum of the squared errors:

$$E_i = (x_{i,3} - y_i)^2 \tag{4.91}$$

$$E = \sum_{i=1}^{N} E_i \tag{4.92}$$

In order to develop a learning procedure that implements gradient descent in E over the parameter space, the error rate $\partial E_i / \partial o$ for the ith feature vector and for each node output o needs to be calculated. The error rate for the outputnode can be calculated from equation 4.91:

$$\frac{\partial E_i}{\partial o_{1,i}^5} = -2(x_{i,3} - o_{1i}^5) \tag{4.93}$$

For a node j in layer k, the error rate can be derived by the chain rule:

$$\frac{\partial E_i}{\partial o_{j,i}^k} = \sum_{m=1}^{\#(k+1)} \frac{\partial E_i}{\partial o_{m,i}^{k+1}} \frac{\partial o_{m,i}^{k+1}}{\partial o_{j,i}^k} \tag{4.94}$$

in which $\#(k)$ denotes the number of nodes in layer k. This means that the error rate of an internal node can be expressed as a linear combination of the error rates of the nodes in the next layer.

Now if a is a membership function parameter in a node in layer 1, the error rate can be described by:

$$\frac{\partial E_i}{\partial a} = \sum_{o* \in S} \frac{\partial E_i}{\partial o*} \frac{\partial o*}{\partial a} \tag{4.95}$$

where S is the set of nodes whose outputs depend on a. Then the derivative of the overall error measure E with respect to a is:

$$\frac{\partial E}{\partial a} = \sum_{i=1}^{N} \frac{\partial E_i}{\partial a} \tag{4.96}$$

The update equation for the parameter a is:

$$\Delta a = -\eta \frac{\partial E}{\partial a} \tag{4.97}$$

in which η is a learning rate which is expressed by:

$$\eta = \frac{K}{\sqrt{\sum_a \left(\frac{\partial E}{\partial a}\right)^2}} \tag{4.98}$$

where K is the step size, the length of each gradient transition in the parameter space. The value of K can be changed to vary the speed of convergence. If K is small, the gradient method will closely approximate the gradient path, but convergence will be slow. If K is large, convergence will initially be fast, but the algorithm will oscillate about the optimum. Based on this, K can be updated according to the following heuristic rules (Jang, 1993):

- If the error measure undergoes four consecutive reductions, set $K = r_{Ki}K$, in which r_{Ki} is the increase ratio.
- If the error measure undergoes two consecutive combinations of one increase and one reduction, set $K = r_{Kd}K$, , in which r_{Kd} is the decrease ratio.

Jang selects to increase or decrease the step size by 10%. This number is chosen arbitrarily. In addition, the initial value of K is usually not critical as long as it is not too large.

The forward and backward pass are performed iteratively until some criterion is satisfied, for example, until the change in parameters is very small or until a predefined number of iterations e has been completed.

The procedure starts with an initial fuzzy model. This model is trained by optimizing its membership function parameters, while the rule structure is left intact. This means that the rule structure needs to be defined beforehand. Neurofuzzy approaches that include rule learning are available (for example Kavli (1993)), but are not presented here.

Example 4.7 The ANFIS approach is illustrated by building a fuzzy model for the net growth rate α. The same data set is used as in examples 4.6 and 4.5.

As with the genetic algorithms example, no rules optimization is included. In the ANFIS case, this involves the optimization of the network structure. Therefore, several identification experiments were executed in order to determine a suitable number of rules. All rules were independent and double sigmoid membership functions were used in each experiment.

The structure of the initial models influences the end result. Therefore, the initial models were built properly, i.e., the membership functions were designed in such a way that a fuzzy partition in the input space was formed.

Settings and results for three different runs are shown in table 4.4. As with the GA, all models perform well and the most complex model has the smallest error.

# rules	e	r_{Ki}	r_{Kd}	K_{init}	$RMSE$
2	1000	1.1	0.9	0.05	0.0208
3	1000	1.1	0.9	0.05	0.0197
4	1000	1.1	0.9	0.05	0.0179

Table 4.4: *ANFIS setting and results*

For all three runs, it is observed that the premise part is not altered much after identification. The membership functions are only slightly changed. This was observed for all three experiments. As can be expected from a gradient descent approach, ANFIS is very sensitive to initialization.

No mechanism is provided to maintain a fuzzy partition in the input space. Although this does not have to impair model results, it may result in models that are difficult to interpret. For the 3-rule model, rules 1 and 2 completely overlap after identification (see figure 4.21). This indicates that the chosen model structure may be too complex.

Similar to the presented GA approach, ANFIS does not take the data as a starting point to determine the model structure but uses an a priori defined model. This can result in undesired extrapolation behavior if the model structure is too complex, especially in areas of the input-output domain where there is little data. In this case, the 3-rule and 4-rule model show this behavior. The anomalous extrapolation is caused by the rules in the area where there is little data.

The rules in the 2-rule model have sufficient data in their activation domain to calculate the consequent parameters successfully. The output surface is similar to the 2-rule model identified with the GA. The premise part membership functions are shown in figure 4.23 and the local linear models are:

$$y_1 = 0.0111S + 0.0132X - 0.1383 \tag{4.99}$$

$$y_2 = 0.0054S + 0.0030X - 0.1964 \tag{4.100}$$

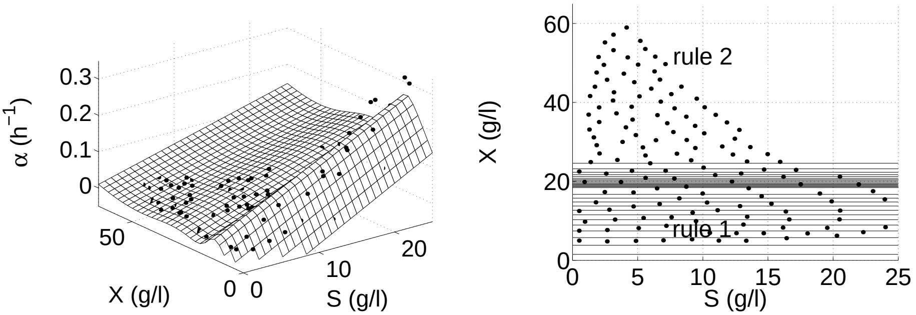
The fuzzy model can be formulated as

$$\begin{array}{l} \text{IF } S = \text{mf}_1 \text{ AND } X = \text{mf}_1 \text{ THEN } \alpha = y_1 \\ \text{IF } S = \text{mf}_2 \text{ AND } X = \text{mf}_2 \text{ THEN } \alpha = y_2 \end{array} \tag{4.101}$$

□

4.4.4 Remarks

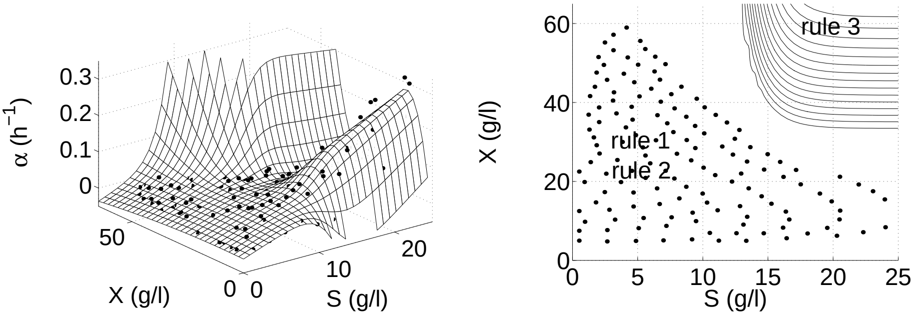
Fuzzy clustering requires less *a priori* structure information than the GA and ANFIS. The latter two methods need a pre-determined rule base structure and membership functions to initialize parameter identification, whereas the clustering approach determines the number of rules automatically. The results are therefore less sensitive to initialization. ANFIS uses an initial model as a starting point for further optimization. Such an initial model may be difficult to set up if no prior information is available, unless an expert could give some qualitative information. Furthermore, experimental results have shown what can be anticipated:



(a)

(b)

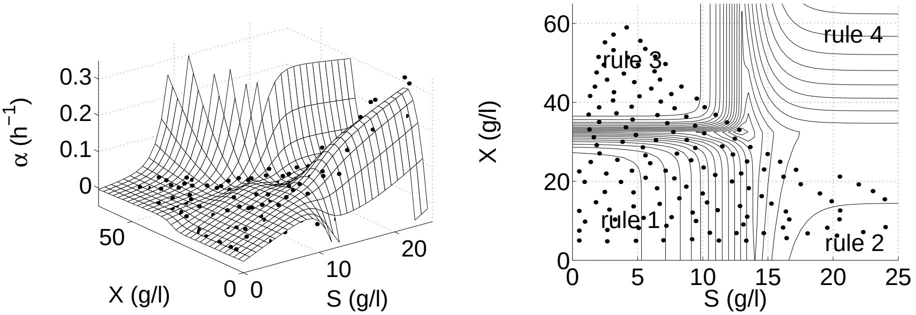
Figure 4.20: *Fuzzy relation for α and rule locations for 2-rule model. Dots indicates identification data*



(a)

(b)

Figure 4.21: *Fuzzy relation for α and rule locations for 3-rule model. Dots indicates identification data*



(a)

(b)

Figure 4.22: *Fuzzy relation for α and rule locations for 4-rule model. Dots indicates identification data*

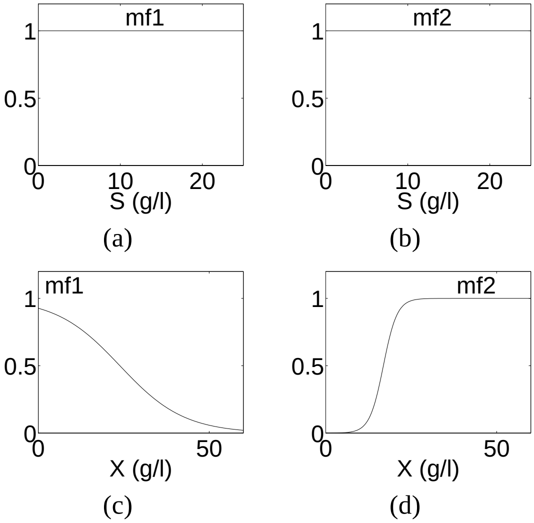


Figure 4.23: *Parametric membership functions for the 2-rule fuzzy model*

the identified model is closely related to the initial model with respect to premise part parameters. This makes ANFIS sensitive to choices made before identification. With the GA, information about the structure of the model also has to be provided (in terms of the rule base and corresponding membership functions) in advance. The GA searches for a solution in a much larger search space than the backpropagation algorithm and determines starting points for the search itself, which makes it not very sensitive to initial parameter values. Not the initial values of the parameters affect the result, but the specified search space does.

The input-output data for the growth rate is not distributed over the complete input space of the system. Normal operation of the reactor causes S and X to be limited to a certain part of the input space. A fuzzy model with independent rules will be able to cope with this data much better, because the rules of these models will be able to describe working areas within the part where data is present, without influencing other rules. For example, if the best partitioning of the input space is non-orthogonal, fuzzy models with dependent rules will be limited in providing a simple rule base that handles this.

The advantage of fuzzy clustering is that it focuses on the data and derives a fuzzy model with independent rules. To obtain the same result with the GA or with ANFIS, prior knowledge has to be provided about the structure and initial location of the rules. This may be cumbersome for high dimensional systems. If this prior knowledge is not provided, rules may be present that have no meaning and that can complicate optimization. For example, this was observed with the 4-rule model that was built with ANFIS. Although the overall result is good for the part where data is present, rule 4 in the area with high S and X is not desirable (figure 4.22 (b)).

As with all black box techniques, care has to be taken in extrapolating these fuzzy models. The 3- and 4-rule models that were built with ANFIS show an example of extrapolation properties that seriously will impair hybrid model results. Post-processing of the identification results can improve this by assuming linear behavior when extrapolating, which often is the best assumption that can be made. Since TSK models are a collection of local linear models, evaluating rules located at the edge of the input space and adjusting membership functions when necessary will ensure this.

Based on these results, the Gustafson-Kessel clustering approach with structure optimization will be used as the main identification tool for fuzzy systems throughout the rest of this work.

4.5 Submodel integration

Since the general structure of the hybrid model is a framework of accumulation balances accompanied with algebraic fuzzy relations, integration of the physical and fuzzy parts is straightforward. The fuzzy submodels are part of the set of equations that make up the hybrid model.

However, with respect to the fuzzy submodels, two sources of error may result in unacceptable hybrid model performance. First of all, estimates are made in order to obtain input-output data. Estimation errors will manifest themselves through the fuzzy model in the hybrid model. Secondly, the fuzzy models are fit to input-output data. Errors resulting from fuzzy model identification can also cause hybrid model errors. Since the hybrid models are dynamical and usually are simulated as a ”free run” (numerically and in an autoregressive manner), small errors are integrated which eventually can result in large offset.

If hybrid model performance is unacceptable, it can be improved by manipulating the fuzzy parts of the hybrid model. This means optimizing fuzzy model parameters with respect to the hybrid model output.

This section will discuss the integration of the fuzzy submodels of a hybrid model by optimization. The main idea is to improve hybrid model performance without discarding the properties of the fuzzy submodels.

4.5.1 Parameters

The number of parameters in the fuzzy submodels that has to be optimized is quite large. One rule of the fuzzy model for the net growth rate α in the bioreactor case, for example, contains about 10 parameters, depending on the type of membership function that is used. Due to the ”curse of dimensionality” the number of parameters increases exponentially for systems with higher dimensions.

During optimization, it is proposed to account for the meaning of the parameters of the fuzzy model. With TSK models, the premise part parameters determine the operating range for which the local linear models are valid. In order to maintain some of the information about the operating range that the fuzzy identification algorithm has determined, an optimization algorithm has to be selected that can deal with this. This means constraints for the *premise part parameters* should be introduced. These constraints can put limitations on the level of fuzziness of the sets and their location in the input domain. The constraints can be determined from the fuzzy model and heuristic knowledge. Usually the constraints can be formulated as an allowed percentage of variation.

It is clear that the *consequent parameters* have the largest impact on the model performance. Since the impact on model transparency is relatively low, the optimization of these parameters should be less constrained. In particular, if local interpretation of the model is important, the optimization should be more constrained than if this is not the case. Since the optimization is performed globally, and not with respect to the local linear model output, local interpretation may be impaired.

To reduce the number of parameters, the optimization can also be performed on *rule weights* or *fuzzy model weights*. The optimization of rule weights corresponds with the adjustment of the parameters of a single local linear model by the same amount, while optimization of the model weights corresponds with the adjustment of all consequent parameters. Optimization of rule weights or model gain can be useful for large parameter systems, if similar error behavior over the entire input domain is observed, or if model errors can be traced to a specific rule in the fuzzy model. In addition, local interpretation is preserved better than with unconstrained consequent parameter optimization.

4.5.2 Optimization

In principle, any optimization algorithm that the modeler finds appropriate, can be applied for hybrid model optimization. Care has to be taken, however, that the optimization algorithm is able to deal with the number of parameters that will be optimized.

Since the goal is to improve the hybrid model performance by using the results from the submodel identification step as a starting-point, direct search (simplex) or gradient-based approaches are suitable to perform the task. Approaches such as genetic algorithms (for example applied in Roubos *et al.* (1999)) discard the existing model parameters and are only appropriate if the search space is limited, thus if heavy constraints are imposed that are based on the parameters of the fuzzy submodel before optimization.

For large parameter optimization problems, the approach described in Branch *et al.* (1999) was found to be suitable. This approach transforms large parameter problems into a two dimensional quadratic approximation for a certain ”trust region” by using a preconditioned conjugate gradient approach. This quadratic problem is subsequently solved. Box constraints are incorporated by ”reflecting” the search path when it encounters a bound. The algorithm is available commercially as part of the MATLAB Optimization Toolbox. Here, only a brief overview is given.

Trust region method

Let $f(\boldsymbol{x})$ be the objective function to be minimized with respect to the vector $\boldsymbol{x}$ with n parameters. The basic idea is to approximate $f(\boldsymbol{x})$ with a simpler function q that describes the original function reasonably well in the neighborhood of $\boldsymbol{x}$. This neighborhood is the so-called ”trust region” N. A trial step $\boldsymbol{d}$ is calculated by minimizing q over N. The optimization problem can be reformulated as:

$$\min_{\boldsymbol{d}}\{q(\boldsymbol{d})|\boldsymbol{d} \in N\} \tag{4.102}$$

The current point is updated to $\boldsymbol{x}+\boldsymbol{d}$ if $f(\boldsymbol{x}+\boldsymbol{d}) < f(\boldsymbol{x})$. If this is not the case, the current point remains unchanged and N is shrunk, after which a new trial step is calculated. To be able to solve the problem, an appropriate approximation function q, a method for choosing the trust region N and a solution method for the trust region subproblem (equation 4.102) need to be available.

The approximation q can be defined by the first two terms of the Taylor expansion of f at $\boldsymbol{x}$. The trust region N is usually spherical or ellipsoidal. The problem can now be stated as:

$$\min_{\boldsymbol{d}}\{\frac{1}{2}\boldsymbol{d}^T H\boldsymbol{d} + \boldsymbol{d}^T\boldsymbol{g}\|D\boldsymbol{d}\|_2 \leq \Delta\} \tag{4.103}$$

with $\boldsymbol{g}$ the gradient and H the Hessian of the objective function, D a positive diagonal scaling matrix and Δ a positive scalar.

Good algorithms are available for solving equation 4.103 (Byrd *et al.*, 1988) but they are time consuming for large numbers of parameters. Therefore, a different approach is applied. If the trust region problem is restricted to a two-dimensional subspace S, the solution of 4.103 requires little computational effort.

Subspace calculation

The two-dimensional subspace $S =< s_1, s_2 >$ is spanned by two vectors: the direction of the gradient $s_1 = D^{-2}\boldsymbol{g}$ (the scaled gradient vector) and either a direction of negative curvature (the steepest descent direction):

$$s_2^T H s_2 < 0 \tag{4.104}$$

or an approximate scaled newton step:

$$\hat{M}s_2 = -\hat{\boldsymbol{g}} \tag{4.105}$$

where

$$\hat{\boldsymbol{g}} = D^{-1}\boldsymbol{g} = diag(|\boldsymbol{v}|^{1/2})\boldsymbol{g} \tag{4.106}$$

$$\hat{M} = D^{-1}HD^{-1} + diag(\boldsymbol{g})J_a \tag{4.107}$$

J_a is the Jacobian of $|\boldsymbol{v}|$ and $\boldsymbol{v}$ is a vector that implements box constraints. The solution to equation 4.105 can only be found if $\hat{M}$ is positive definite. If this is not the case, the direction of negative curvature 4.104 is selected for s_2.

Box constraints

If constraints are imposed on the parameter vector $\boldsymbol{x}$, then the optimization problem can be formulated as:

$$\min_{\boldsymbol{x}}\{f(\boldsymbol{x})|\boldsymbol{l} < \boldsymbol{x} < \boldsymbol{u}\} \tag{4.108}$$

where $\boldsymbol{l}$ is a vector of lower bounds and $\boldsymbol{u}$ is a vector of upper bounds. Some (or all) components of $\boldsymbol{l}$ may be equal to $-\infty$, some components of $\boldsymbol{u}$ may be equal to ∞. The constraints are implemented by the vector $\boldsymbol{v}$:

$$\begin{array}{llllll} \text{IF} & g_i < 0 & \text{AND} & u_i < \infty & \text{THEN} & v_i = x_i - u_i \\ \text{IF} & g_i \geq 0 & \text{AND} & u_i > -\infty & \text{THEN} & v_i = x_i - l_i \\ \text{IF} & g_i < 0 & \text{AND} & u_i = \infty & \text{THEN} & v_i = -1 \\ \text{IF} & g_i \geq 0 & \text{AND} & u_i = -\infty & \text{THEN} & v_i = 1 \end{array} \tag{4.109}$$

If, during a step, a constraint is hit, the step is reflected off the constraint. This increases the step size (Branch *et al.*, 1999). For a (single) reflection step, given a step $\boldsymbol{d}$ that intersects a constraint, consider the first bound constraint crossed by $\boldsymbol{d}$ and assume it is the i-th bound constraint. Then the reflection step $\boldsymbol{d}^R = \boldsymbol{d}$ except in the i-th component, where $d_i^R = -d_i$.

Algorithm

The optimization algorithm can be summarized as follows:

1. Subspace calculation: Formulate the two-dimensional trust region subproblem S
2. Trust region method: Solve the two-dimensional subproblem S. Calculate the trial step and use reflections if necessary
3. If $f(\boldsymbol{x}+\boldsymbol{d}) < f(\boldsymbol{x})$ then set $\boldsymbol{x} = \boldsymbol{x}+\boldsymbol{d}$ and start again with step 1 or else go to step 4
4. Adjust Δ to shrink the size of the trust region and determine a new trial step $\boldsymbol{d}$ in step 2

4.5.3 Objective function

All relevant hybrid model outputs should be incorporated in the objective function. For dynamical models, the goal function should be able to deal with measurement trends instead of steady state errors. In addition, scaling of the different errors is appropriate. A straightforward and common least squares criterion that incorporates this can be formulated as follows:

$$J = \frac{1}{2}\sum_{j=1}^{m}\sum_{i=1}^{n} w_i(y_{ij} - \hat{y}_{ij})^2 \tag{4.110}$$

in which $\boldsymbol{y}_i$ is the vector with (normalized) measurements of output i and $\hat{\boldsymbol{y}}_i$ is the vector with (normalized) model output values of output i. $\boldsymbol{w}$ is a weighting vector. The choice of the weights depends on the problem.

In Psichogios and Ungar (1992), the parameters of a neural net (that is part of the hybrid model) are optimized with respect to the hybrid model output. The objective function in this work incorporates sensitivity functions (partial derivatives of the output to the parameters that are optimized) in the objective function. This way, the error is propagated back to the output of the neural net. However, if gradient descent based optimization approaches are used, sensitivity information of the model output with respect to the parameters will be calculated by the algorithm. This omits the need to incorporate sensitivity equations explicitly in the objective function.

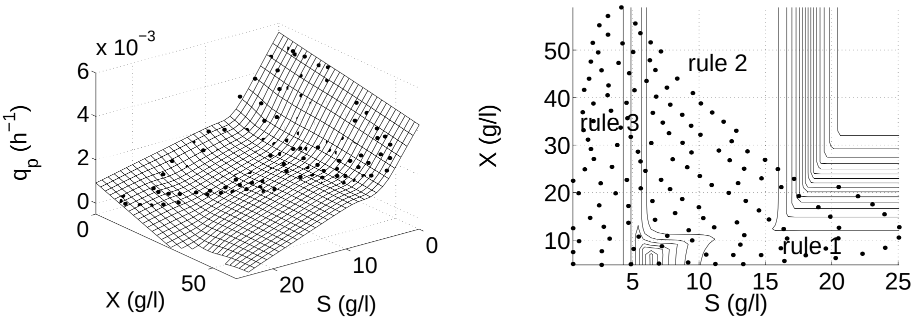
4.5.4 Remarks

The optimization of the hybrid model as presented in this section can be extended to incorporate other model parameters. However, the larger the number of parameters that is optimized, the more likely it is that the different parameters will compensate for each others errors. For example, if several fuzzy models are optimized simultaneously, then they can compensate for each others contribution to the hybrid model output, which results in a loss of meaning of the fuzzy model output. This is even more the case if other model parameters are included.

To gain insight in the influence of the different fuzzy models or model parameters on the hybrid model output, sensitivity analysis can be worthwhile before performing the integration step. This will provide information about the main sources of the error in the model output. With respect to this, it is also recommended to investigate the contributions of the specific rules in the fuzzy models. The analysis forms the basis for selecting the set of parameters that is optimized and for determining objective function weights.

It is clear that the complexity of the problem is reduced if the integration is performed on smaller, independent parts of the hybrid model. For example, if the fuzzy model for a reaction rate is optimized, the mass balance which contains this reaction rate should only be incorporated. Whether this is possible depends on the measurements that are available, the model structure and the sensitivity analysis.

With the optimization of large sets of parameters, it is difficult to determine whether the global optimum is found. By using gradient based algorithms, it is more likely that a local optimum is found. However, it is usually possible to formulate criteria for acceptable performance of a model. The goal of the optimization should therefore not be the determination of the optimal



(a)

(b)

Figure 4.24: *Fuzzy relation for q_p and rule locations. Dots indicates identification data*

set of parameters, but finding a solution that meets these criteria.

Example 4.8 Consider the hybrid model for the fed-batch bioreactor (example 4.1) and assume that the fuzzy model for the net growth rate α that was identified with fuzzy clustering in example 4.5 is available. Furthermore, assume that a fuzzy model for the production rate q_p as a function of the substrate concentration S and the biomass concentration X is available and that this model was identified with fuzzy clustering, based on estimates by a PI-estimator. The estimates of q_p were made by using measurements of P and the mass balance for the product concentration (equation 4.3). The settings for the PI-estimator are shown in table 4.2, while the settings for the clustering were 5 for the initial number of clusters and 0.55 for the cluster merging threshold γ. The fuzzy model is shown in figure 4.24.

In this example, the large scale algorithm will be illustrated. The initial performance of the hybrid model for run VAL1, before optimization, is shown in figure 4.25. It can be seen that for this run, the concentration of P starts to decrease after about 160 hours, which is not possible. This is caused by anomalous behavior of q_p. The production rate decreases after 160 hours, which results in a decrease in production of P. Because the amount of P in the reactor is diluted, the net result is a decrease in the concentration of P. The increase in S is caused by a decrease in the substrate consumption rate, which is a result of the decrease in q_p.

The goal is to improve hybrid model performance with respect to the errors in the biomass X, the substrate S and the product P. This will be done by using the measurements from the identification batch runs ID1 to ID16 (see example 4.2) as a reference. The objective function is formulated according to equation 4.110:

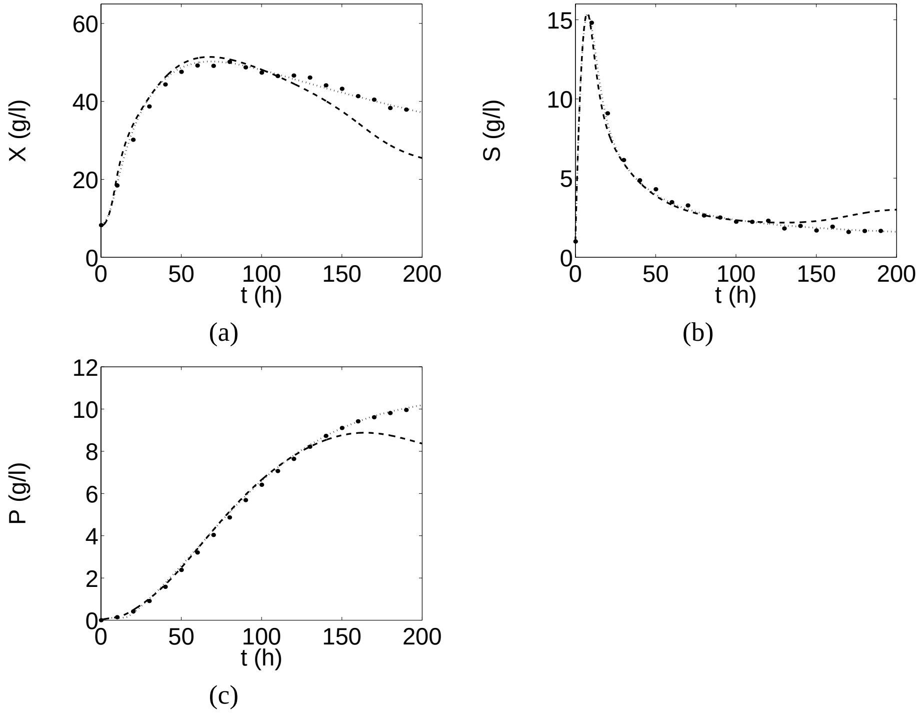
$$J = \frac{1}{2}\sum(e_S^2 + e_X^2 + e_P^2) \tag{4.111}$$

in which e_S, e_X and e_P are normalized error signals, defined as:

$$e_{S,k} = |\frac{S_{ij} - \hat{S}_{ij}}{\bar{S}_j}| \text{ with } k = i \cdot j \tag{4.112}$$

$$e_{X,k} = |\frac{X_{ij} - \hat{X}_{ij}}{\bar{X}_j}| \text{ with } k = i \cdot j \tag{4.113}$$

$$e_{P,k} = |\frac{P_{ij} - \hat{P}_{ij}}{\bar{P}_j}| \text{ with } k = i \cdot j \tag{4.114}$$



(a)

(b)

(c)

Figure 4.25: *Run VAL1 results before and after optimization. Dots indicate measurements, dashed line hybrid model before optimization, dotted line hybrid model after optimization for X (a), S (b) and P (c)*

with index i indicating the time step and index j indicating the batch run. $\hat{S}$ indicates model estimates for S and $\bar{S}$ indicates the average value of S for a batch run, with similar definitions for X and P.

Although it is possible to optimize the fuzzy models for α and q_p sequentially (first, α with respect to X and then q_p with respect to P), it is interesting to see how the large scale algorithm deals with optimization of a large set of parameters in a hybrid model. Therefore, all the parameters of the two fuzzy models will be optimized simultaneously. The result is that a set of 66 parameters will be optimized; 48 premise part parameters and 18 consequent part parameters. The premise part parameters are constrained; the bounds are set at the initial values +/- 10 %. No constraints were placed on the consequent part parameters.

The results of the optimization are shown in figure 4.25 (for clarity, only some of the measurements are shown). The anomalous behavior has been removed and model offset has been reduced to acceptable levels. The results are shown in table 4.5. The errors are the normalized errors for run VAL1 at the end of the batch.

Figure 4.26 shows the local linear models of the fuzzy relations before and after the optimization. In both models, the rule locations have not been altered much, due to the constraints. The consequent parameters of the fuzzy model for α have only been changed slightly in order to accomplish good behavior of X. The consequent parameters in the model for q_p have been changed substantially. The local interpretation of the linear submodels has been impaired.

Error	*Before optimization*	*After optimization*
e_X	0.29	0.03
e_S	0.76	0.06
e_P	0.17	0.02

Table 4.5: *Optimization results*

These results can be explained in two ways. The rules in the fuzzy model overlap, as a result of which the overall model output is different than described by the linear models. This is shown in figure 4.27. It is this overall output that determines the performance of the hybrid model. In addition, the sensitivity of P for changes in q_p is relatively small for rules 1 and 2. The sensitivity can be calculated using the sensitivity equations (Caracotsios and Stewart, 1985). Results are shown in figure 4.28. This means that changes in the parameters of these rules do not have a large impact on hybrid model performance. The sensitivity is relatively large for rule 3. The parameters of this rule have only been adjusted slightly to improve the hybrid model (see figure 4.26).

Whether behavior as illustrated by the model of q_p should be accepted depends on the objectives of the modeler. The overall hybrid model performance is good. If, however, according to the modeler's judgement, the fuzzy relationship in a certain working area is unrealistic, it could be rejected. It should be noted that fuzzy logic is still a black box technique and that care should be taken in associating a physical meaning with the results. Furthermore, it could be argued that less importance should be attached to areas with low sensitivity in relation to areas with larger sensitivity. An advantage is that fuzzy logic provides a means to learn more about the relative

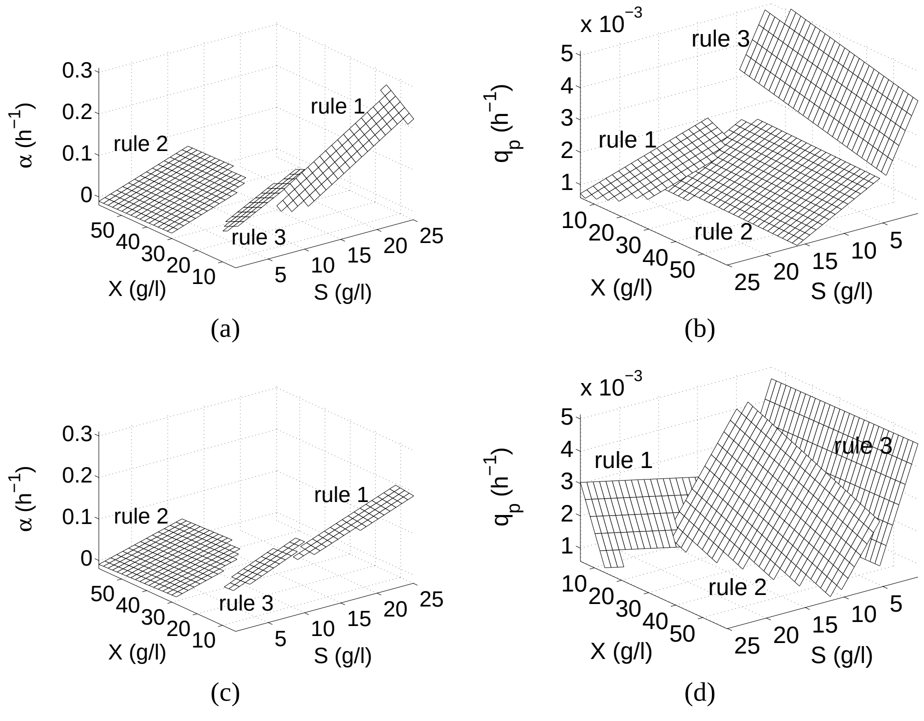


Figure 4.26: *Local linear models for α and q_p before ((a) and (b)) and after ((c) and (d)) optimization*

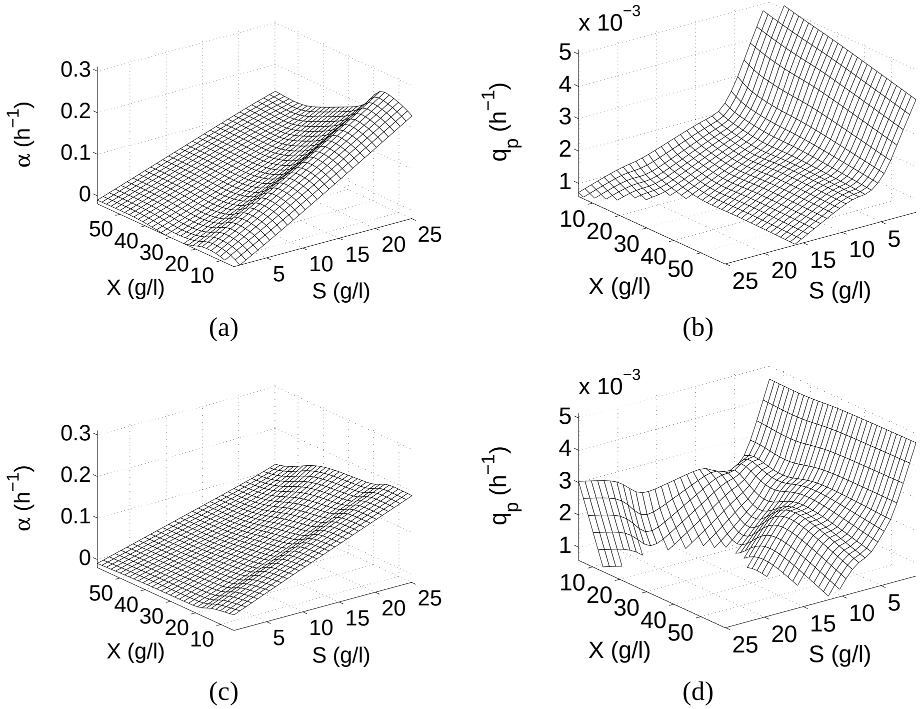


Figure 4.27: *Fuzzy models for α and q_p before ((a) and (b)) and after ((c) and (d)) optimization*

importance of working areas. In addition, it allows the modeler to tune performance independently in these areas.

The sensitivity equations could be used to reduce the size of the optimization problem in advance by leaving out optimization of parameters with limited sensitivity. It should be noted, however, that the analysis above was done after the results were obtained and that optimization results may be slightly different if the complete model is not included in the optimization. It may, for example, be difficult to decide when the sensitivity is low enough to exclude a rule from optimization beforehand. □

4.6 Model adjustment

The model that is built using basic modeling is based on the hypothesis and the available process measurements. Sometimes, this information may not prove to be sufficient to build the model. In addition, some of the hypothesis may not be correct (mainly due to simplification). This means that the model needs to be adjusted.

In essence, model adjustment is basic modeling, based on the model analysis results. The results from this step need to state the nature of the problems as clearly as possible. With this information, a new hybrid model structure is designed, following the basic modeling

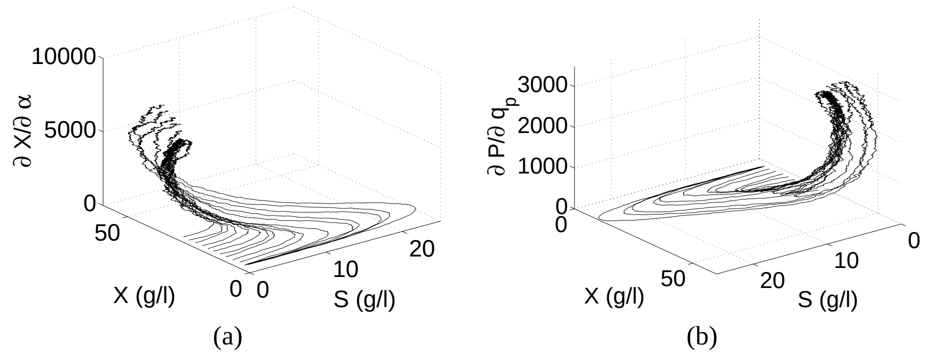


Figure 4.28: *Sensitivities for X with respect to α (a) and P with respect to q_p (b)*

procedure.

Model adjustment can be done in two ways: at the same level of detail or at a higher level of detail. It is advisable to first try developing another model at the same level of detail before going to a higher level of detail. If this not possible, then a higher level of detail needs to be selected. The increase in detail needs to be done in small steps. Based on the results of the model analysis step, the modeler needs to decide which structure adjustments are necessary. In principle, this means the addition of model equations. An example is the introduction of temperature dependency in heat capacity coefficients. In some cases, the adjustment of the model structure requires the addition of balance equations. If this is the case, the deliberation needs to be made whether the increase in modeling effort and complexity is justified with respect to model performance.

4.7 Concluding remarks

The design phase of the hybrid modeling procedure can be summarized as follows:

- Determine the hybrid model structure and distinguish subprocess modeling problems, for which fuzzy submodels should be developed
- Based on this structure, acquire relevant process data
- Estimate unmeasurable behavior of the subprocesses
- Identify the fuzzy submodels using fuzzy clustering
- Integrate the fuzzy submodels by optimizing their parameters with respect to the hybrid model output

The steps are performed sequentially and independently of each other. The advantage is that the modeling problem is split into several simpler problems. Combining the solutions of these problems may lead to a non-optimal hybrid model; the result is that an integration step is needed.

For the subprocess behavior estimation, the PI-estimator is a simple and useful alternative to more elaborate estimation approaches. It's simple nature makes it possible to set it up quickly. Care should be taken, when there is large coupling between two or more parameters that are estimated simultaneously. For many applications, however, the approach can be applied without many problems (Van Lith *et al.*, 2001).

Fuzzy clustering provides a good way to identify the fuzzy submodels. It requires little a priori model structure information. In addition, because of the structure optimization algorithm, it is insensitive to initialization. It also derives a fuzzy model with independent rules directly from the data, which results in models that are not likely to show anomalous extrapolation behavior.

During submodel integration, the parameters of the fuzzy submodels are optimized with respect to the hybrid model output. To maintain part of the results from the identification step, these fuzzy models should be used as a starting point, which makes gradient based optimization algorithms suitable tools.

Although algorithms for optimizing large sets of parameters are available, it is obvious that the number of parameters should be kept as low as possible. This makes sensitivity analysis worthwhile. In addition, the optimization problem can be reduced by optimizing rule weights or model weights.

Model adjustment essentially provides a feedback procedure to adjust the model structure if a hybrid model performs unacceptably. The model adjustment step incorporates the same elements as the basic modeling step. Based on the initial hybrid model, a new model structure can be developed, after which the data acquisition, subprocess behavior estimation, submodel identification and integration steps are performed.

5

Hybrid model properties

To illustrate the concepts of the structured hybrid modeling approach, this chapter will deal with the development of a more elaborate hybrid model than discussed in the previous chapters. In addition, this model will provide the basis for a discussion on hybrid model properties, by comparing it with the two ”extremes” of the hybrid modeling spectrum: a first principles model and a fuzzy model. This provides more information about the applicability of hybrid model in general.

A continuous pulp digester was selected as the test case. This is a unit operation used to produce paper pulp from wood chips. It is a nonlinear distributed parameter process, of which the process conditions vary considerably. In addition, the description of the internal behavior often is empirical, for which many different approaches exist. This makes the process suitable for development of a hybrid model.

For the comparison, a first principles model is available in the form of the Extended Purdue Model, an industrially accepted benchmark model for the process. This is a rigorous model that describes the internal and overall dynamic behavior of the digester. The first principles model will provide the basis for hybrid model development, which means that the hybrid model is not based on actual process data. However, by using this approach, it will be possible to investigate which essential characteristics of the process should be described, while good performance and interpretability is maintained. The rigorous model will be reduced to a simpler model structure which describes the essential behavior of the digester, given the requirements that were defined for the hybrid model. As a result, new model equations need to be derived, by which the hybrid modeling approach can be illustrated. The fuzzy model will also be based on the first principles model. Input-output data, generated by the first principles model, is used to construct the fuzzy model.

The comparison of the three models will be based on the five model quality aspects: static performance, dynamic performance, complexity, interpretability and process independence. The evaluation will be related to the three general areas of model applications: research and development, design and control.

First, this chapter will discuss the pulping process and the first principles model. Based on this information, the hybrid model objective and model requirements will be discussed, followed by the design and evaluation. Then the fuzzy model will be designed and evaluated, after which the three models will be compared.

5.1 Process description

The production of pulp for paper products has been a widely studied subject, especially in the Scandinavian countries and Canada. In particular, the continuous digester, the unit operation for manufacturing pulp, has received much attention. It is a challenging unit operation for control studies due to unmeasured disturbances, long time delays and nonlinear behavior (Wisnewski *et al.*, 1997). In addition, different interpretations about the physical processes that take place have been the basis for model development, resulting in a variety of models. This makes the unit operation an interesting candidate for developing a hybrid model.

For more information about wood and pulp processing, the reader is referred to Jahn and Strauss (1992).

5.1.1 Wood and pulp processing

Wood contains a mixture of polymers that can be split in three groups:

- Lignin (20 - 30%)
- Cellulose
- Hemicellulose; anhydrides of Xylose, Arabinose, Glucose, Mannose, Galactose, Xylan and Galactoglucomman

The strength of the wood structure comes from the cellulose fibers. Cellulose is a very strong, chemically highly resistant, fiberous polymer. The other elements are easily degradable by treatment with acids or bases.

Lignin serves to hold the cellulose fibers and other elements and binding them together into the anatomical structure we know as ”wood”. Lignin is susceptible to degradation and dissolution by treatment with strong bases under elevated temperatures. Lignin can also disintegrate when treated with acid sulphite solutions or oxidizing agents. Thus, lignin can be removed from the wood, leaving separated cellulose fibers in the form of a pulp. This pulp can then be processed to produce paper. The main quality parameter for the pulp is the so-called Kappa number $\kappa\#$, which is a measure for the amount of lignine in the wood.

In general, the paper production process incorporates:

- Wood preparation
- Pulping (mechanical or chemical)
- Pulp screening and cleaning
- Pulp bleaching
- Paper production

Mechanical pulping is used for low quality paper. With mechanical pulping, a log is pressed against a rotating disc or cylinder, which yields what is called groundwood pulp. In this work, the focus is on chemical pulping.

Wood preparation

Bark contains many short fibers besides colored, non-fiberous materials. This yields pulp that is not suitable of making paper. If the pulp contains bark, the resulting paper is not very strong and discolors. Therefore, the bark needs to be removed as much as possible. This is accomplished by letting the logs scrub each other in a rotating cylinder. Other methods exist to debark the logs, for example removing the bark with knives or by means of water jets.

After the logs have been debarked, the wood can be used directly in the following production steps. When the wood is used as raw material for chemical pulping, it is necessary to chop the logs into wood chips. These woodchips usually have dimensions of about 0.5 x 2 cm. The logs are chopped into thin slices using a rotating disc containing radial placed knives. The slices fall apart to the desired woodchips due to the enormous forces the rotating discs exert on them.

Chemical pulping

Lignin constitutes approximately 20 to 30 % of the total weight of wood. In manufacturing purified pulps of high whiteness, it is necessary to remove as much lignin as possible with minimum loss or degradation of the carbohydrate cell wall. Usually this is done by pulping to liberate the fibers and then bleaching the fibers to the desired whiteness. Unbleached pulps are usually dark brown in colour and are used in that form for grocery bags and wrapping paper.

Lignin is not a pure component, but can be considered as a class of substances. Lignin is a complex, three-dimensional polymer containing mostly phenylpropane units joined together by various ether and carbon-carbon linkages. Aim of the chemical pulping processes is to deliberate the lignin by dissolving or degradation. There are two methods to accomplish this: reactions with acids or bases. Both methods are carried out in an aqueous environment under elevated pressure and temperature.

The sulphite process uses a cooking liquor of sulphurous acid and a salt of this acid. While calcium was the most widely used base at a time, it has been replaced with sodium, magnesium, and ammonia. The sulphate process uses a mixture of sodiumhydroxide and sodiumsulphide as the active chemical. The term sulphate process is used because sodiumsulphate is used as make-up chemical. The word *Kraft* is also used to refer to this process for it is the German word meaning ”strength”. This method produces the strongest pulp. In the past, ammonium base NH_4OH was used and had the ”advantage” that this liquor could be vaporized and burned without any residue. Only air pollution occurred. The SO_2 in the gas effluents can be lead through a gas-washer and react with fresh ammoniumhydroxide.

The sulphite process is carried out in batch reactors using longer residence times than the Kraft process with temperatures of 140-150 $°C$. This process is most appropriate for wood species such as spruce and pine trees where a relatively light colored and strong pulp is

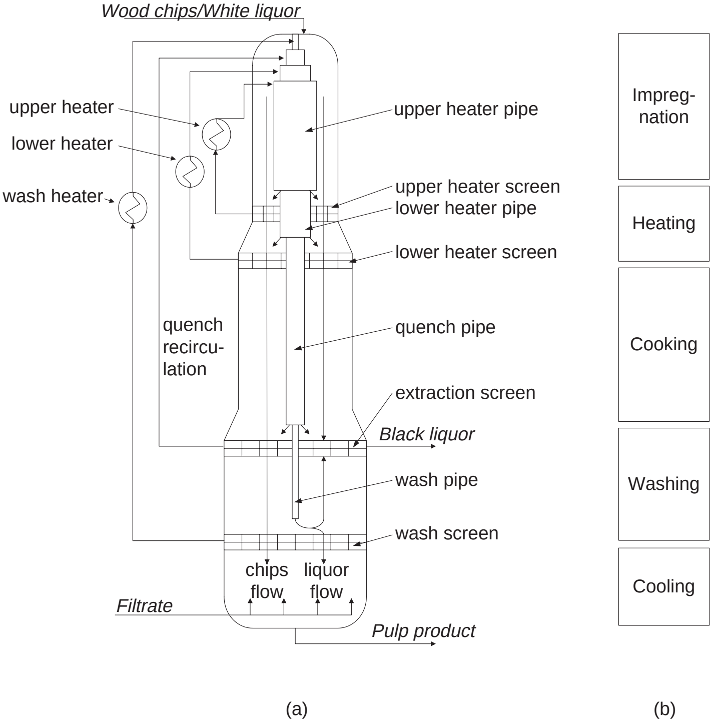


Figure 5.1: *Digester unit operation (a) and functional zones (b)*

produced. The sulphite process can therefore be found in those regions where the wood species mentioned above exist.

5.1.2 The continuous pulp digester

The continuous pulp digester is a unit operation designed to convert woodchips into a cellulose fiber pulp. The pulping of the woodchips is accomplished by cooking the woodchips in a hot solution of sodium hydroxide and sodium sulphide, referred to as ”white liquor”. The digester is essentially a tubular reactor, where woodchips travel from the inlet at the top to the outlet at the bottom of the digester. The surrounding liquor flow is either in co-current or in counter-current with respect to the woodchips, dependent on the functional zone where the woodchips are flowing. A schematic overview of the digester and its functional zones is given in figure 5.1.

In the impregnation zone the woodchips are brought into contact with a co-current flow of white liquor. The white liquor components (primary sodium hydroxide and sodium sulphide) diffuse into the pores of the porous woodchips. As the temperatures in this zone are relatively low (around 385 K), practically no delignification takes place. The length of the impregnation zone is 5.5 m.

Reactor height	32 m
Reactor diameter	5 m
Throughput	3.7 m^3/min
Pulp yield	600 t/d
Residence time	2 h
Pressure cooking zone	7 bar
Reaction temperature	430 K

Table 5.1: *Typical digester dimensions and operating conditions*

In order to reach the required reaction temperatures, in the various zones part of the white liquor flow is drawn off the digester. After being heated in external heat exchangers, the white liquor flow is fed back into the heating zone. The white liquor is supplied through a system of concentric pipes. The digester contains two heating zones for gradually heating the white liquor to the desired reaction temperatures. This way, the upper heater may be considered as a pre-heater. The actual temperature control is achieved by adjusting the lower heater outlet temperature. The upper heater outlet temperature is approximately 410 K, the lower heater outlet temperature is about 430 K. The total length of the heating zones extends to 4.7 m.

The actual delignification takes place in the cooking zone, which extends to 12 m. The majority of the delignification takes place in this section. The Kappa number of the pulp is drastically reduced. The reaction products diffuse into the white liquor at the same time the reactants diffuse from the white liquor into the pores of the woodchips. In the cooking zone, typical temperatures are about 430 K and pressures are about 7 bar (Jahn and Strauss, 1992).

A concentric pipe ending 1.5 m above the end of the cooking zone injects a relatively cold, non-reactive liquor flow. The temperature of the woodchips is reduced and the delignification reactions are quenched. The quench liquor is a process flow coming from the digester, referred to as ”black liquor” which is essentially a contaminated mixture of white liquor and wash liquor. Black liquor contains few reactive components.

In the washing zone, a counter current flow washes the degradation products from the pulp. In the counter current region a large average driving force is maintained between the woodchips and the washing liquor. A process flow referred to as ”filtrate” is used as washing liquor. The wash liquor feed configuration presented in figure 5.1 is a simplified one. In an actual digester, a part of the wash liquor is fed into the digester through the cold blow line, which prevents caking of the pulp in the bottom of the digester. The temperature of the wash liquor has a large influence on the pulp quality. As diffusion is favored by high temperatures, more degradation products are washed from the pulp at higher temperatures. However, too high temperatures cause continuation of the delignification process, damaging the cellulose fibers structure yielding an inferior paper quality.

In the cooling zone, part of the injected filtrate flow goes down and thus travels co-current with respect to the woodchips. This flow cools the woodchips. At the bottom of this zone, the outlet device is located.

Typical digester dimensions and process conditions are given in table 5.1.

5.1.3 Digester control

The primary control objective for the digester is to produce pulp at a specified Kappa number at maximal yield. A Kappa number that is too high increases the cost necessary to bleach the pulp, a Kappa number that is too low means damaging the cellulose fibers and using more chemicals and energy than necessary. Process optimization involves using a minimum input of energy and chemicals to achieve the target Kappa number. Adequate process control minimizes the variation in the measured Kappa number and creates room for optimization because it enables operating more closely to the target Kappa number.

Controlling the digester is a difficult task because of the long time delays involved, non-linear behaviour and unfrequent availability of process measurements. The Kappa number can only be determined by an off-line analysis, taking up to one hour. If a disturbance affecting the Kappa number occurs, it will take at least one process dead time increased by the time required for analysis before the disturbance is detected. Furthermore, because many inputs affect the Kappa number, multivariable control is highly appropriate for the digester.

Traditional digester control involves stabilizing the most important variables of the process in such a way that reaction conditions remain constant. Variations in observed or estimated Kappa number are compensated by adjusting the lower heater outlet temperature.

5.2 Extended Purdue Model

The Kraft pulping process for both batch and continuous digesters has been modeled to various levels of complexity. Wisnewski *et al.* (1997) present a short overview of digester modeling based on physical principles. In addition, modeling approaches include black box techniques such as Wiener models (Godasi and Palazoglu, 2001), Partial Least Squares models (Alexandridis *et al.*, 2001) and neural networks (Dufour *et al.*, 2001).

A well known first principles model for a continuous pulp digester is the Purdue Model (Smith and Williams, 1974) which is a detailed and industrially accepted digester model (Wisnewski *et al.*, 1997). The model has been extended by providing improved definitions of mass concentrations and volume fractions and a more detailed description of the mass and energy transport. This Extended Purdue Model (EPM) is described in detail in Wisnewski *et al.* (1997) and will serve as the reference model for hybrid and fuzzy model development.

5.2.1 Model structure

In Wisnewski *et al.* (1997), the continuous pulp digester is modeled as a series of 50 Continuously Stirred Tank Reactors (CSTR's) leading to a lumped parameter system. Consequently, true plug flow behaviour is approached. Modeling the digester as a Plug Flow Reactor (PFR) instead would yield partial differential equations (PDE's). However, modern

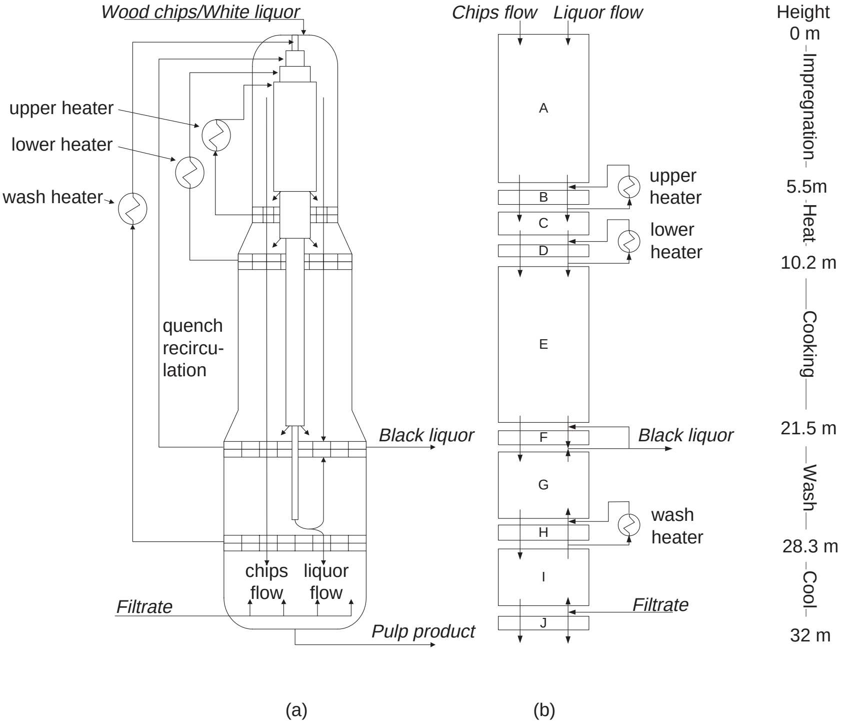


Figure 5.2: *Digester unit operation (a) and model flowsheet (b)*

simulation packages such as gPROMS are able to solve PDE's directly. Therefore, the CSTR description is transformed to a distributed parameter system (Jansen, 2000). This yields the flexibility to solve the model as accurately as needed. By specifying more discretization intervals, more accurate results will be obtained. In a CSTR series approach, this can only be achieved by changing the model structure. The digester model ”flowsheet” is based on the CSTR model structure and incorporates the heater and recycle loops. The model flowsheet is shown in figure 5.2.

The digester model is implemented by treating certain disturbances constant and implementing them as parameters. In an actual digester, these parameters are not constant and change over time. Some of the disturbances are non-measurable and difficult to anticipate. In the context diagram (figure 5.3), the most important digester unit operation inputs, outputs and disturbances are shown.

In the model, the following process input disturbances are treated as constant with respect to time:

- Compaction behaviour. The model uses a linear compaction profile throughout the digester. In an actual digester, the compaction profile will vary with time and place. Especially at the extraction screen, the woodchip hold-up fluctuates. Furthermore, the assumed compaction profile will represent a mean situation that never occurs in actual

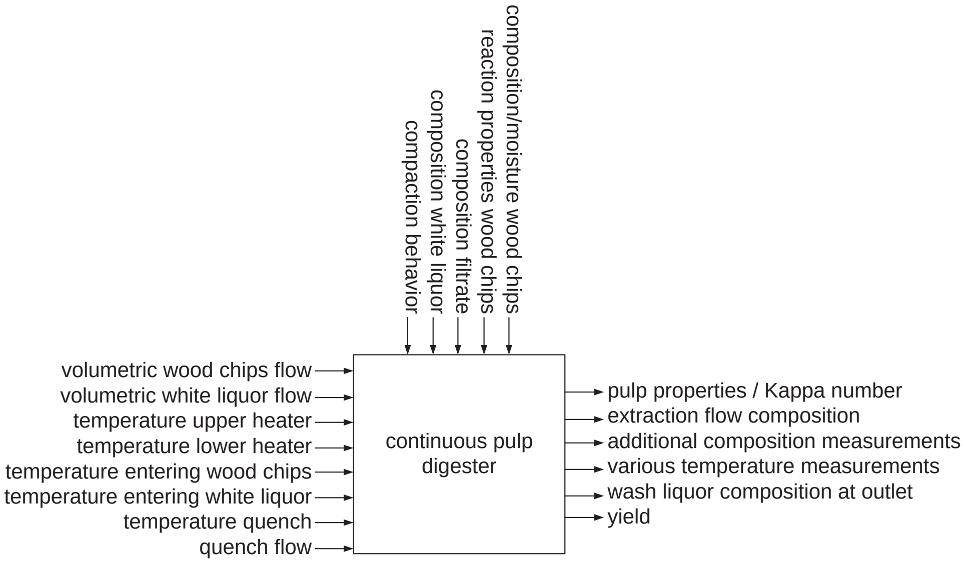

composition/moisture wood chips
reaction properties wood chips
composition filtrate
composition white liquor
compaction behavior
volumetric wood chips flow
volumetric white liquor flow
temperature upper heater
temperature lower heater
temperature entering wood chips
temperature entering white liquor
temperature quench
quench flow
continuous pulp digester
pulp properties / Kappa number
extraction flow composition
additional composition measurements
various temperature measurements
wash liquor composition at outlet
yield

Figure 5.3: *Digester context diagram*

digester operation.

- White liquor is supplied from a preceding unit and will posses varying alkali contents with respect to time.
- The filtrate composition (weak black liquor) will also vary, because it is made up in preceding process apparatus.
- Kinetic parameters. Different wood species have different kinetic parameters and these will more or less vary among woodchips. The modeled values are to be treated as mean values.
- Composition of woodchips. There are many wood species and composition will vary along individual woodchips of the same species, due to variable moisture content or age.

5.2.2 Modeled states

The most important output variable of a digester is the Kappa number. The primary objective of digester control is to maintain the Kappa number within certain limits. The Kappa number measures the amount of residual lignin within the woodchips. In addition, the overall digester yield is often calculated. The yield is a measure of the amount of wood substance recovered in comparison with the amount of wood substance fed to the digester.

In order to calculate these properties using a physical model, the appropriate wood and liquor components need to be modeled, as well as the temperature in the digester. In the EPM,

Phase	*Index*	*State*	*Modeled effects*	*Additional algebraic equations*
Chip	$s,1$	High reactive lignin	Convection, reaction	Reaction rate
	$s,2$	Low reactive lignin	,,	Reaction rate
	$s,3$	Cellulose	,,	Reaction rate
	$s,4$	Galactoglucomannan	,,	Reaction rate
	$s,5$	Araboxylan	,,	Reaction rate
		Temperature	Convection, reaction heat, conduction, diffusion, bulkflow	Diffusion coefficient
Entrapped	$e,1$	Active alkali	Convection, reaction, diffusion, bulkflow	-
	$e,2$	Passive alkali	,,	-
	$e,3$	Active hydrosulfide	,,	-
	$e,4$	Passive hydrosulfide	,,	-
	$e,5$	Dissolved lignin	,,	-
	$e,6$	Dissolved carbohydrates	,,	-
Free liquor	$f,1$	Active alkali	Convection, diffusion, bulkflow	-
	$f,2$	Passive alkali	,,	-
	$f,3$	Active hydrosulfide	,,	-
	$f,4$	Passive hydrosulfide	,,	-
	$f,5$	Dissolved lignin	,,	-
	$f,6$	Dissolved carbohydrates	,,	-
		Temperature	Convection, reaction heat, conduction, diffusion, bulkflow	-

Table 5.2: *EPM overview*

three phases are distinguished: the solid phase, which comprises the wood substance, the free liquor phase, and the entrapped liquor phase, which comprises the liquor that has entered the porous wood chips. This phase serves as a transport medium for mass and energy from the internal wood surface of the woodchips to the surrounding bulk, the free liquor phase. The woodchips, having a slightly larger velocity, travel down with the free liquor. The modeled components are shown in table 5.2. The index denotes how the components will be referenced in the equations.

The Kappa number $\kappa\#$ and the yield γ can be calculated as follows:

$$\kappa\# = \frac{\rho_{s,1} + \rho_{s,2}}{0.00153 \sum_{i=1}^{5} \rho_{s,i}} \tag{5.1}$$

$$\gamma = \frac{\sum_{i=1}^{5} \rho_{s,i,exiting}}{\sum_{i=1}^{5} \rho_{s,i,entering}} \tag{5.2}$$

in which ρ denotes the mass concentration of a substance and the subscripts refer to the species given in table 5.2.

The EPM incorporates temperature dependent mass diffusion, temperature dependent reaction rates, composition dependent heat capacities and heat exchange between entrapped and free liquor phase due to conduction. In addition, the following assumptions have been made:

- Adiabatic behavior; no heat exchange with the surrounding environment
- No radial gradients in temperature or composition
- Entrance, wall and cross flow effects are neglected

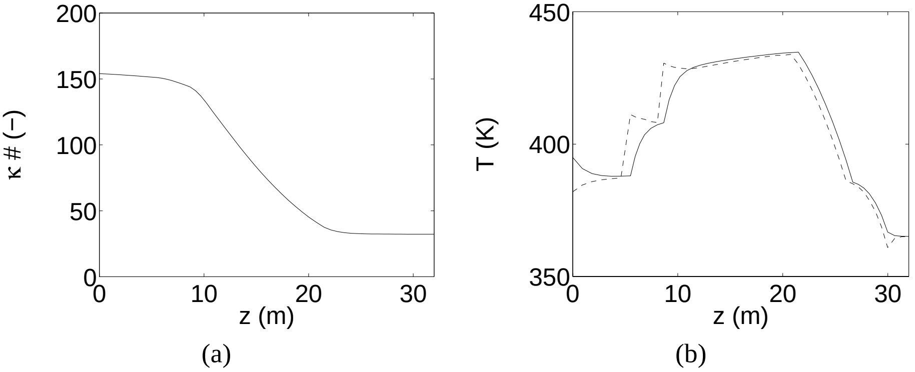


Figure 5.4: *Steady state Kappa profile (a) and temperature profiles (b). Solid line T_c, dashed T_f*

- Compaction profile is known and constant with time

The EPM equations and parameter values are given in appendix C. The data flow diagram (DFD) is also presented there.

5.3 Extended Purdue Model analysis

Since the EPM will serve as the reference in this study, the complexity, interpretability and process independence of the model will be analyzed (see figure 3.2). This analysis can be used later for comparison with the hybrid model and the fuzzy model. In addition, the EPM will be analyzed with respect to its purpose in this work; to provide a basis for developing a hybrid model that is used to investigate hybrid model quality properties. First, the general behavior of the EPM will be discussed.

5.3.1 Model behavior

Static and dynamic behavior

The steady state Kappa number profile $\kappa\#(z)$ of the reactor for typical operating conditions is given in figure 5.4. The temperature profile of the solid and entrapped phase $T_c(z)$ and of the free liquor phase $T_f(z)$ are shown as well. The model simulation was carried out with 50 discretization steps for the reactor height z.

The Kappa profile indicates that little reaction occurs in the impregnation zone ($0 < z < 5.5$). This is due to relatively low temperatures. After the heating zone, the temperature is much higher and significant reaction takes place, which results in a decrease of $\kappa\#$. In the

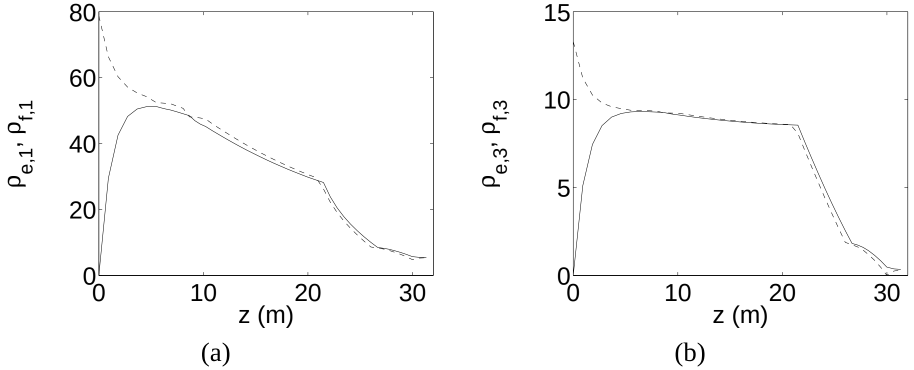


Figure 5.5: *Steady state active alkali (a) and active hydrosulfide profiles (b). Solid line entrapped phase, dashed free liquor phase*

wash zone ($21.5 < z < 28.3\ m$), the reaction is quenched by the cold counter current wash flow. The degradation products are washed from the pulp, not affecting the Kappa number $\kappa\#$.

The wood chips that enter the digester have a higher temperature than the white liquor (figure 5.4). At the end of the impregnation zone, the temperatures are the same. At the upper heater outlet, hot free liquor enters the digester and as a result, there is a discontinuity in the free liquor temperature. The same is true for the temperature at the lower heater outlet. In the cooking zone, reaction temperature is reached. The exothermic reactions cause a further increase in temperature; the temperature of the wood chips is higher than that of the free liquor. The counter-current flow in the washing section results in a constant temperature difference between the chips and the liquor.

Figure 5.5 shows the active alkali $\rho_{e,1}, \rho_{f,1}$ and active hydrosulfide $\rho_{e,3}, \rho_{f,3}$ concentration profiles, the main components of the white liquor. In the impregnation zone, concentration equalization due to diffusion takes place. The pores of the woodchips are initially filled with water. As the woodchips travel co-currently with the surrounding free liquor along the impregnation zone, concentrations of entrapped and free phase components approach each other. At the heater outlets, small discontinuities occur due to mixing with free liquor having a slightly different composition. Along the cooking zone, the active alkali concentration decreases due to exhaustion. Diffusion from the free phase into the entrapped phase within the woodchips takes place. In the washing zone, the effective alkali concentrations are reduced drastically, due to the counter current wash flow.

In the impregnation zone, the dissolved lignin diffuses from the free liquor phase ($\rho_{f,5}$) into the entrapped phase ($\rho_{e,5}$). Because the entering free liquor flow is mixed with black liquor, there are already dissolved solids present in the entering free liquor flow (figure 5.6). In the beginning of the cooking zone, the temperature is high enough to cause significant delignification. As a result, the concentration of dissolved lignin in the entrapped phase becomes higher than the surrounding free liquor phase. Diffusion is the mechanism responsible for the flow of dissolved solids to the free liquor. Due to the delignification process, the free liquor absorbs more and more dissolved lignin, causing the concentrations to increase. In the

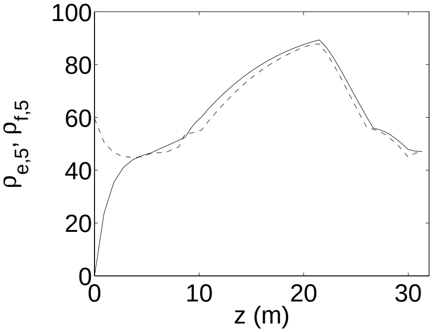


Figure 5.6: *Steady state dissolved lignin profile. Solid line entrapped phase, dashed free liquor phase*

washing zone, the degradation products are washed from the pulp by bringing the entrapped liquor into contact with a relatively clean counter-current washing liquor.

To illustrate the dynamic behavior of the EPM, several experiments were carried out, based on the experiments in Wisnewski *et al.* (1997). The experiments involve step changes in the lower heater heat supply Q_{lh} (with constant upper heater heat supply Q_{uh}) and the white liquor flow ϕ_f. In addition, Bode diagrams for the Kappa number and the yield as a function of Q_{lh} and ϕ_f are constructed. Frequency analysis results are shown in table 5.3. The table also shows the response amplitudes $\Delta\kappa\#$ and $\Delta\gamma$ for a positive and a negative step change, respectively.

The step changes are performed at $t = 0\ min$. The response to a 10 % increase or decrease in lower heater heat supply (figure 5.7) indicates a negative gain (the higher temperature results in a higher reaction rate and thus in a lower kappa number). The gain of the response to a 7 % change step in the free liquor flow rate ϕ_f at constant Q_{lh} and Q_{uh} is positive (reaction temperature is lower since the same heat is supplied to more liquor), as shown in figure 5.8.

Figures C.4 to C.7 in appendix C show Bode diagrams for the yield and the Kappa number

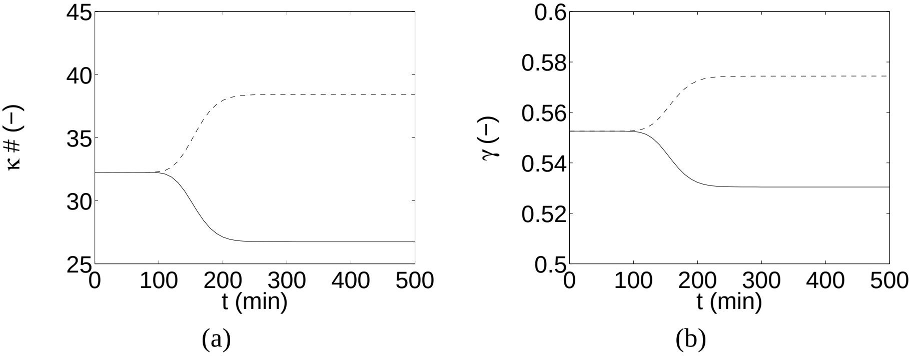
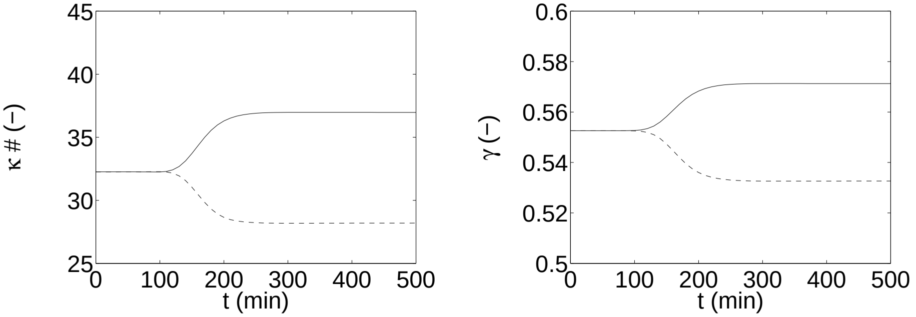


Figure 5.7: *Kappa (a) and yield (b) response to steps in lower heater heat supply. Solid line positive step, dashed negative*



(a)

(b)

Figure 5.8: *Kappa (a) and yield (b) response to steps in the liquor flow rate. Solid line positive step, dashed negative*

of the EPM. The response of the system was not simulated for very high frequencies due to numerical inaccuracies. Time constants τ and system order O are obtained from the Bode diagrams and are listed in table 5.3. The figures show second order behavior for changes in Q_{lh}, while third order behavior is observed for changes in ϕ_l. The higher order for changes in the flow rate (while Q_{lh} is constant) is caused by the change in concentration. This is an additional effect that is not observed for changes in Q_{lh}, since during these changes, the flow rate is kept constant. The fraction in the orders is caused by numerical dispersion. For example, a block pulse is affected by numerical dispersion, which results in detection of a slightly higher order. In addition, there may be some inaccuracies in the Bode diagram. The dead times were determined using step changes in the inputs.

Overall mass balance

The performance of the EPM can be verified with respect to its overall mass balance. The mass balance is calculated for standard operating conditions, while ignoring the mass flow of the solvent (water). This way, the mass balance describes only the ingoing and outgoing mass flows for the components that are solved and the solid phase mass flow. The actual throughput will be larger. Results are shown in table 5.4.

Minor differences due to numerical inaccuracies may be tolerated. The error in the mass balance amounts to 1.0 %. This is very reasonable and indicates that there are no errors in model consistency with respect to the mass balances.

Input	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$ (min)	$\tau_{[input],\gamma}$ (min)	$\theta_{[input]}$ (min)	$\Delta\kappa\#$ $(-)$	$\Delta\gamma$ $(-)$
Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020

Table 5.3: *Frequency response analysis results for EPM*

Stream	*Mass flow in* (kg/min)	*Mass flow out* (kg/min)
Solid	751.9	416.1
Entrapped	0.057	118.7
Free	496.5	51.7
Extraction	-	802.4
Filtrate	154.7	-
Total	1403	1389

Table 5.4: *Overall mass balance EPM*

5.3.2 Complexity

The complexity of the EPM is analyzed using the model flowsheet and the DFD, given in appendix C. The model flowsheet consists of 10 sections of different length. These sections are a more detailed representation of the five functional sections in the digester (impregnation, heating, cooking, washing and cooling) and are needed to be able to implement the various heater recycle loops and screens. The flowsheet is straightforward and not complex.

The DFD however, is somewhat more complex. It represents the model that is used for each section in the flowsheet. The diagram is constructed in accordance with general guidelines for designing DFD's (Yourdon, 1989). The connections between the effects that are modeled (see also table 5.2) are clearly shown.

The DFD is constructed for the general classes of components (solid phase, entrapped phase or free liquor phase components). Including each component separately would result in an unnecessary complex diagram; there are 17 components in total. The disadvantage is that the connection between the components through the reaction stochiometry and heat capacity is not represented.

The most complex equations in the EPM are the mass and energy balances. The other equations are nonlinear algebraic equations that are rather straightforward. The complexity of the model equations is relatively low because equation substitutions are kept to a minimum.

5.3.3 Interpretability

The model flowsheet represents the physical structure of the digester and its connection with other unit operations (the heaters). Due to the low level of complexity, interpretation is straightforward. The strength of the model is that it provides information at a detailed level. The different physical phenomena and phases that are modeled by section models of the EPM are clearly represented in the DFD and can be interpreted.

The equations that can be interpreted best are the mass and energy balances. Each term in these equations represents a physical phenomenon such as mass transfer or heat generated by the reaction. The algebraic equations that describe these phenomena cannot be easily interpreted from a first principles point of view. Many include lumped parameters (such as the heat transfer coefficient U) or are fitted to empirical data (such as the relation for the diffusion coefficient D_{ce}. This is the result of the level of detail of the model; more detailed behavior (for example on the level of a wood chip particle) is not included. The only interpretation that can be given to these equations is the type of mathematical dependence (linear, quadratic, etc.).

5.3.4 Process independence

Process independence is determined by the sources of information that are used during the construction. For the EPM, first principles are the most important source of information. No or little experimental data is available to construct a model that can describe the concentration profiles in the way the EPM does. The behavior is mainly determined by the level of first principles information that is used. This is augmented by expertise that is available in the industry, which is based on lab experiments or observed behavior. The relation for the diffusion coefficient and the introduction of several fit parameters are examples of this.

The detailed description of the behavior and the first principles basis result in a certain level of process independence. The model can be used for different installations of this type of reactor, probably after fine-tuning of some of the coefficients. In addition, the EPM is an extension of an industrially accepted digester model, the Purdue Model (Wisnewski *et al.*, 1997), which illustrates successful application on a wide variety of installations.

With respect to process design, the EPM has limited application possibilities. It only provides global information about design parameters. The empirical relations may have limited validity, which makes the model more suited for process operation studies than for process design.

5.3.5 Remarks

The quality of the EPM with respect to a traditional application (controller design, process optimization) is not discussed here. Some aspects of application in the three general areas (R&D, design and control) will be discussed later. However, whether the model quality meets the requirements with respect to the application in this work can be analyzed.

The EPM will serve as the basis for hybrid and fuzzy model development. In addition, it will be compared with the hybrid and the fuzzy model with respect to model quality. The EPM is suitable for these purposes, as was shown in this section. It provides a dynamical description of a nonlinear process with a large operating regime. The description is detailed and can be reduced to a more simple structure, which only describes the essential behavioral characteristics. This means that new model equations need to be derived, as will be presented in the following section. In addition, the EPM can be used as a simulator to provide process data for the identification of hybrid and fuzzy model parameters.

5.4 Hybrid model problem definition

5.4.1 Objective

The objective of the hybrid model of the pulp digester is to facilitate the investigation of different hybrid model properties in comparison to models at the extremes of the spectrum: first principles models (in this case the EPM) and fuzzy models. Based on this analysis, conclusions can be drawn about the ”position” of hybrid models in this spectrum, which provides more information about the applicability of hybrid models in general. The analysis can be performed by evaluating model quality with respect to model performance, complexity, interpretability and process independence.

In addition, it is interesting to determine how well such a hybrid model can deal with simplified dynamics, without impairing performance and physical interpretation. In other words, the hybrid model needs to describe the *essential physical characteristics* of the pulp digester with respect to the application of the model. The key variables and model requirements then provide the basis for determining the most important characteristics that should be included in the model.

5.4.2 Key variables and requirements

The major quality requirement of the hybrid model is that it can be compared to the EPM. Together with the objective, this breaks down into the following requirements:

1. **The hybrid model of the pulp digester should describe the Kappa number $\kappa\#$ and the yield γ**.
The two most important variables are the Kappa number $\kappa\#$ and the yield γ, especially at the reactor outlet. These are the *key variables* of the hybrid model. This means that the model needs to describe the concentrations of the reactants in such a way that $\kappa\#$ and γ can be calculated.
2. **Dynamic and static behavior should match the behavior of the EPM**.
This means that the hybrid model should be dynamic and that the states should be modeled in such a way, that the same type of behavioral interpretation can be given to the hybrid model as can be given to the EPM.
3. **The hybrid model structure needs to be based on physical principles**.
Since the model will be hybrid, part of the structure will be based on first principles. In addition, the input-output structure of the fuzzy submodels needs to be physically interpretable. In general, this does not have to be the case. For this model however, input-output structure interpretability is required to maintain a certain level of hybrid model interpretation, in order to make comparisons to the EPM possible.
4. **The hybrid model structure needs to be less complex than the EPM**.
One of the goals is to describe the digester behavior with a model that is based on the

essential physical phenomena that play a role. This results in a model that is less complex than the EPM, but still provides a high level of interpretability. With respect to this, it is interesting to investigate how model performance depends on the simplifications.

From equations 5.1 and 5.2, it can be concluded that the concentrations of the wood components need to be modeled in order to calculate $\kappa\#$ and γ. Not all concentrations need to be modeled separately, though. With respect to the EPM, components $(s, 3)$ to $(s, 5)$ can be lumped, which still provides sufficient information to calculate the two key variables $\kappa\#$ and γ. The next step is to determine in detail which components need to be modeled and what the hybrid model structure should be to meet the requirements.

5.5 Hybrid model design

The design of the hybrid model for the pulp digester can be interpreted as a ”bottom up” procedure. Taking the key variables and the requirements as a starting point, the relevant states and physical processes are determined, followed by the model structure with respect to the unit operation. After this, model identification can take place.

This section presents the various steps of the design phase: basic modeling, data acquisition, subprocess behavior estimation, submodel identification and submodel integration.

5.5.1 Basic modeling

The apparent choice is to derive the hybrid model structure from the EPM by a form of *model reduction*. By formulating interpretability requirements, the model reduction is performed from a physical point of view instead of a mathematical point of view. This way, requirement 4 is met. Based on requirements 1, 2 and 3, the behavior of the digester can be analyzed and process hypotheses describing the essential behavior can be formulated.

Step 1: Process description and information analysis
The process is described in section 5.1. The EPM provides detailed information about the behavior, which can be used during model identification. In particular, the state profiles over the reactor are useful; it is quantitative physical information. From these, all additional variables and parameters can be derived. In addition to the model structure and expertise that is available from the field of digester modeling, these profiles are an important source of information.

Step 2: Process hypotheses
In the digester, wood reacts with liquor to remove lignin from the wood. It is a two-phase system; there is a wood phase and a liquor phase. The flow regime is plug flow. To start the reaction, the liquor is heated. At a certain point, the reaction is quenched by washing the

wood with cold liquor.

In this system, the following phenomena take place: convection, reaction, mass transfer and heat transfer. These phenomena can be unified in a framework of dynamical mass and energy balances. In the EPM, the phenomena are described by static algebraic equations, which determines the level of detail that is available. To meet requirement 2, the phenomena should by described in a similar fashion in the hybrid model.

Step 3: Process structure

The derivation of the process structure involves the model reduction step. The reduction will be performed on two levels: on the level of the model flowsheet and on the level of the section model. In both cases, the goal of the reduction is to determine the most important model characteristics, with a minimum loss of transparency (requirement 2 and 3).

The model flowsheet of the EPM consists of 10 sections, which are located in 5 ”functional zones” (see figure 5.1). The functional zones are relevant for the interpretation of the behavior and should be maintained in the hybrid model. The number of sections, however, is less important for meeting the interpretability requirements; interpretation of the Kappa number profile, for example, does not require detailed information about the quench recirculation. In particular, the double heater recycle loop can be omitted without compromising interpretability. The function of the heaters is to initiate the reaction, which for the most part takes place in the cooking zone. This is indication by the drop in $\kappa\#$ in figure 5.4. The same functional interpretation can be given to a single heater that provides the same amount of energy as the two heaters combined. In addition, the heater recycles can be omitted. This may influence the dynamic behavior of the model and therefore should be investigated in the evaluation phase.

As a result, the hybrid model can describe each functional zone with one section. There are thus four sections (figure 5.9):

- Section I: Impregnation
- Section II: Cooking
- Section III: Washing
- Section IV: Cooling

The section model of the EPM describes three phases which contain 17 components in total. Mathematically, there is no need to describe the solid and entrapped phase separately. The EPM does not describe mass or heat transfer between the two phases. The description of the reactions that take place do not require specific information about the phases, except for the concentrations of the reactants. The concentration of the solid species is based on the volume of the chips (which is equal to the sum of the volumes of the solid and entrapped phase), while the concentration of the entrapped species is based on the volume of the entrapped phase. In addition, the concentrations in the entrapped phase are required to describe mass and heat transfer between the entrapped and free liquor phase.

The section model can be simplified by lumping the entrapped and solid phase to a ”reaction phase”, which essentially describes the physical phenomena in the wood chips. The poros-

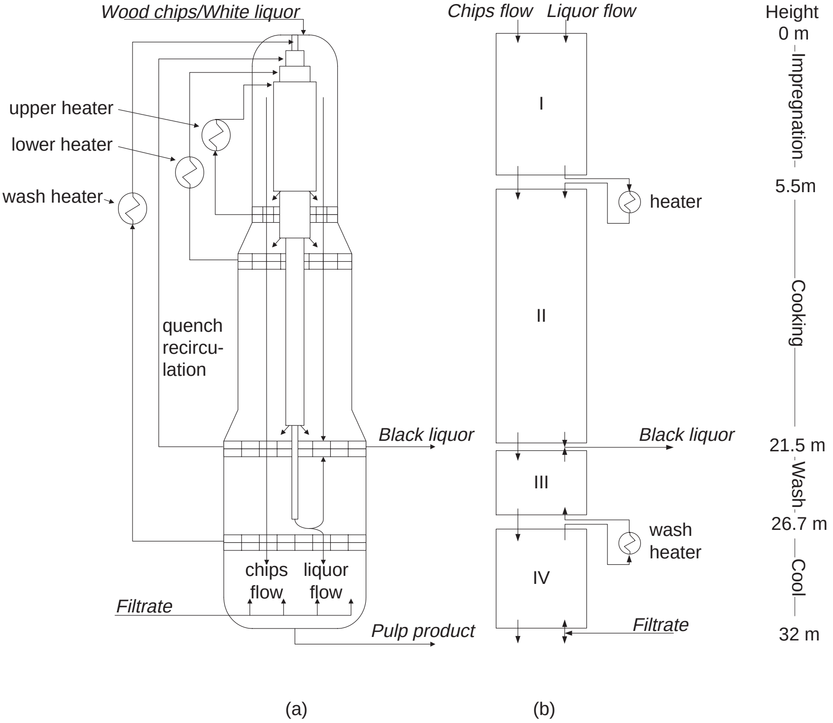


Figure 5.9: *Digester unit operation (a) and hybrid model flowsheet (b)*

ity ε (see equation C.16) can be used to compensate for the change in concentration of the entrapped components, which as a result of the lumping is based on the chip volume instead of the entrapped liquor volume. As a result, the hybrid model will describe two phases, a *reaction phase* which corresponds to the solid and entrapped phases of the EPM, and a *liquor phase*, which corresponds with the free liquor phase in the EPM. The assumption will be made that the reaction phase is ideally mixed and that bulkflow can be neglected.

The hybrid model needs to describe the Kappa number $\kappa\#$ and the yield γ. This means that the concentrations of the wood species $(s, 1)$ to $(s, 5)$ are required. However, species $(s, 3)$ to $(s, 5)$ are not required explicitly. The section model can be simplified by lumping them to a species ”carbohydrates”.

A similar approach can be followed for the species in the liquor phase. In the EPM, the use of different species in the free liquor phase is only required to be able to describe the reaction rate; the reaction rate equation C.6 requires species $(e, 1)$ and $(e, 3)$. Thus $(f, 1)$ and $(f, 3)$ are required. A transparent simplification can be accomplished by viewing upon the reaction as a reaction between ”lignin” (species $(s, 1)$ and $(s, 2)$), ”carbohydrates” (species $(s, 3)$ to $(s, 5)$ lumped) and ”liquor” (species $(e, 1)$ and $(e, 3)$ lumped). This reduces the number of differential equations in the model, while still meeting interpretability requirements.

The result of the lumping of the phases and species is that the kinetic equations of the EPM

Phase	*Index*	*State*	*Modeled effects*	*Additional fuzzy equations*
Reaction	$r, 1$	High reactive lignin	Convection, reaction, diffusion	Reaction rate
	$r, 2$	Low reactive lignin	,,	Reaction rate
	$r, 3$	Carbohydrates (cellulose, galactoglucomannan, araboxylan)	,,	Reaction rate
	$r, 4$	Active liquor (active alkali, active hydrosulfide)	Convection, reaction, diffusion	-
	$r, 5$	Passive liquor (passive alkali passive hydrosulfide)	,,	-
	$r, 6$	Dissolved lignin	,,	-
	$r, 7$	Dissolved carbohydrates	,,	-
		Temperature	Convection, reaction heat, conduction, diffusion	Diffusion coefficient
Liquor	$l, 1$	Active liquor (active alkali, active hydrosulfide)	Convection, reaction, diffusion	-
	$l, 2$	Passive liquor (passive alkali passive hydrosulfide)	,,	-
	$l, 3$	Dissolved lignin	,,	-
	$l, 4$	Dissolved carbohydrates	,,	-
		Temperature	Convection, reaction heat, conduction, diffusion	-

Table 5.5: *Hybrid model overview*

are no longer valid. These relations are nonlinear and cannot be easily transformed to accommodate the new species (see equations C.6, C.7 and C.8). To describe the reaction rates in the hybrid model, fuzzy equations will be used. These will be derived from observed behavior. In addition, the diffusion coefficient D_{ce} will be described by a fuzzy equation. In the EPM, an empirical equation was used. Due to the lumping of the entrapped and free liquor phase species, diffusion behavior is affected (see equation C.3).

The reaction rates of the other species in the EPM are calculated through reaction stochiometry (equations C.9 and C.35). Since this operation and the lumping are both linear with respect to the components, the new reaction stochiometry can readily be calculated from the stochiometry in the EPM.

To build a consistent hybrid model with closed mass and energy balances, the reaction products need to be described as well. This results in the overall section model as is presented in table 5.5.

Step 4: Basic equations

Similar to the EPM, each section of the hybrid model will be modeled using the same section model. The Kappa number $\kappa\#$ and the yield γ can be calculated as follows:

$$\kappa\# = \frac{\rho_{r,1} + \rho_{r,2}}{0.00153 \sum_{i=1}^{3} \rho_{r,i}} \tag{5.3}$$

$$\gamma = \frac{\sum_{i=1}^{3} \rho_{r,i,exiting}}{\sum_{i=1}^{3} \rho_{r,i,entering}} \tag{5.4}$$

in which ρ denotes the concentration in kg/m^3 of a species and the subscripts refer to the species given in table 5.5.

The physical framework of the hybrid model will consist of 13 states:

- Mass balances for species $(r, 1)$ to $(r, 7)$
- Mass balances for species $(l, 1)$ to $(l, 4)$
- Energy balance for the reaction phase, resulting in the reaction phase temperature T_r
- Energy balance for the liquor phase, resulting in the liquor phase temperature T_l

To meet requirement 3, the states are described as a function of time and location in the reactor. The structure of the state equations is similar to the structure of the state equations in the EPM.

The reaction rates $R_{r,i}$ of components $(r, 1)$ to $(r, 3)$ will be described by fuzzy equations, while the other reaction rates are derived from them through stochiometry. The reaction rate depends on the species concentration, the liquor concentration, and the reaction phase temperature. This results in the following structure:

$$R_{r,i} = f_{fuzzy}(\rho_{r,i}, \rho_{r,4}, T_r) \text{ for } i = 1 \ldots 3 \tag{5.5}$$

The diffusion coefficient D_{cr} is also described by a fuzzy equation. The simplifications affect the behavior of the diffusion. The lumping of species $(e, 1)$ and $(e, 3)$ and $(f, 1)$ and $(f, 3)$, respectively, is nonlinear with respect to the diffusion term. This will be discussed later. Similar to the EPM, it will depend on the reaction phase temperature:

$$D_{cr} = f_{fuzzy}(T_r) \tag{5.6}$$

Other section model equations involve simple linear algebraic equations that calculate the total mass of a phase, the overall heat capacity of a phase, etc.

The heaters in the hybrid model are described by the same equations as in the EPM. The complete set of model equations and the hybrid model DFD are presented in appendix D.

5.5.2 Data acquisition

Except for the fuzzy models, all parameters of the hybrid model can be obtained from the EPM. The EPM is used as a benchmark model and has been extensively verified with industrial data. To be able to identify the fuzzy relations, sufficient data about the behavior is needed, which can be generated by using the EPM as a simulator. Since the hybrid model describes the internal behavior of the digester, the data needs to be available in the form of ”state profiles”, that represent the states as a function of the location in the reactor.

This data can be obtained by performing several simulation ”experiments” under different operating conditions. The ”measurements” of the state profiles can then be used as identification data. Since the fuzzy relations are static, no dynamic data is required. This means that a limited number of experiments is sufficient; only steady state measurements of the process

	ϕ_f					
T_{lh}	1.76	2.0	2.34	2.6	2.94	3.2
420	ID1		ID4		ID7	
430		VAL1		VAL3		VAL5
438	ID2		ID5		ID8	
445		VAL2		VAL4		VAL6
450	ID3		ID6		ID9	
460		VAL7		VAL8		

Table 5.6: *Identification and validation run settings*

operated under different conditions need to be obtained. Measurements for identification as well as for validation need to be available.

The different steady state profiles can be obtained by manipulating process inputs. During simulation of the model, the assumption is made that the wood flow rate and composition do not change. This reduces the number of experiments that have to be set up while the hybrid model requirements are still met. The main process inputs that can be used to control the Kappa number $\kappa\#$ (lower heater temperature and free liquor flow rate) will be used to acquire the data. It is also possible to manipulate the filtrate flow rate, but this only affects the process behavior in the last part of the reactor, which results in a marginal contribution.

To compare results after identification, the hybrid model has to be simulated under the same conditions as the EPM. Since in the hybrid model the upper heater and lower heater are lumped, the simulation conditions of the hybrid model are based on the heat supply Q_h instead of heater temperature T_h. The total heat supply can be determined by calculating the heat supply of the lower and upper heater in the EPM, using equation C.29 and the lower and upper heater temperatures.

The hybrid model is being built based on model reduction of the EPM. This means that the EPM generates the required identification data. To acquire sufficient information, the lower heater temperature or energy supply and the liquor flow rate can be varied within the validity range of the EPM. However, to facilitate investigation of extrapolation behavior with physically meaningful experiments, the range of variation during identification should be limited. Therefore, the lower heater temperature and free liquor flow rate are varied between approximately +/- 10 % of their normal operating values (appendix C). This results in 9 different ”identification runs” ID1 to ID9. In addition, 8 ”validation runs” VAL1 to VAL8 are designed. An overview is given in table 5.6.

5.5.3 Subprocess behavior estimation

To identify the fuzzy models for $R_{r,1}$, $R_{r,2}$, $R_{r,3}$ and D_{cr}, input-output data is required. The inputs can be obtained from the EPM, the outputs cannot. In the hybrid model, species $(e, 1)$ and $(e, 3)$ are lumped to form species $(r, 4)$. Species $(e, 1)$ and $(e, 3)$ are both inputs to the EPM kinetic equations. However, since the kinetics are nonlinear, they are no longer valid for species $(r, 4)$. This means that the outputs of the fuzzy models, $R_{r,1}$, $R_{r,2}$ and $R_{r,3}$, cannot be readily obtained from the EPM. The outputs are therefore derived from observed behavior by estimation. Since the lumping affects the behavior of the diffusion term in the model, the diffusion coefficient is also estimated.

The reaction rates and diffusion coefficient can be estimated with the help of the mass balances of species $(r, 1)$ to $(r, 3)$ and $(l, 1)$ of the hybrid model (equations D.1 and D.3) and the steady state profiles from the identification experiments. Because these profiles describe a steady state, the state equations reduce to ODE's. This facilitates the use of PI-estimators (section 4.3.2). The system is mathematically equivalent to a batch process; as in a batch process the conditions vary with time, the conditions in the digester vary with the location in the reactor. The profiles can be interpreted as a dynamic response with respect to the location in the reactor.

Four PI-estimators were designed. Since detailed information on the behavior of the EPM is available (all state variables can be "measured"), the model parts f of the PI-estimators (as shown in figure 4.6) are kept simple. Consider the estimation of reaction rate $R_{r,1}$. The estimates can be derived from the steady state mass balance of species $\rho_{r,1}$:

$$0 = -\frac{\phi_r}{S(1-\eta)}\frac{\partial \rho_{r,1}}{\partial z} + R_{r,1} \tag{5.7}$$

which can be rewritten to

$$\frac{\partial \rho_{r,1}}{\partial z} = \frac{S(1-\eta)}{\phi_r} R_{r,1} \tag{5.8}$$

Incorporating this equation in a PI-estimator structure, an "estimated profile" for reaction rate $R_{r,1}$ can be obtained. The model equations for the PI-estimators of $R_{r,2}$ and $R_{r,3}$ are similar. For the estimation of D_{cr}, the PI-estimator model equation becomes:

$$\frac{\partial \rho_{l,i}}{\partial z} = \frac{S\eta}{\phi_l} D_{cl}(\frac{1}{\varepsilon}\rho_{r,4} - \rho_{l,1}) \tag{5.9}$$

Substituting

$$D_{cl} = D_{cr}\frac{1-\eta}{\eta} \tag{5.10}$$

results in

$$\frac{\partial \rho_{l,i}}{\partial z} = \frac{S(1-\eta)}{\phi_l} D_{cr}(\frac{1}{\varepsilon}\rho_{r,4} - \rho_{l,1}) \tag{5.11}$$

In this equation, $\rho_{r,4}$ is treated as an input.

The PI-estimators were tuned manually by comparing the concentration profiles with the reference values from the EPM. During setup, it was not possible to find appropriate tuning parameters that yielded acceptable estimate profiles for the complete digester. A single PI-estimator cannot cope sufficiently with the change in EPM behavior in the different sections. Therefore, PI-estimators for each of the four sections of the hybrid model were designed.

While designing the PI-estimators for D_{cr} in the four sections, it was observed that the estimators in sections I, III and IV provided little useful information. In section I, both temperature T_r and estimated D_{cr} do not change much. In sections III and IV, no acceptable tuning

Estimator	*Controlled* variable	*Other* variables (input)	K I	II	III	IV	τ_I I	II	III	IV
$R_{r,1}$	$\rho_{r,1}$	ϕ_r	0.2	0.25	0.25	0.35	2	2	0.4	0.8
$R_{r,2}$	$\rho_{r,2}$	ϕ_r	0.15	0.2	0.2	0.4	1	0.5	0.3	10
$R_{r,3}$	$\rho_{r,3}$	ϕ_r	0.2	0.2	0.2	0.3	5	1	0.3	0.8
D_{cr}	$\rho_{l,1}$	$\phi_l, \rho_{r,4}, \varepsilon$	-	-0.045	-	-	-	5	-	-

Table 5.7: *PI-estimator structures and settings*

parameters could be determined. To obtain small errors, the estimators had to be tuned aggressively. Due to the limited number of measurements that were available for the estimators of the smaller sections III and IV, overshoot and oscillation was observed. Therefore, only a PI-estimator for D_{cr} in section II was used.

The PI-estimator settings were used for all identification runs (ID1 to ID9). Table 5.7 gives an overview of the PI-estimator structures and tuning parameters K and τ_I.

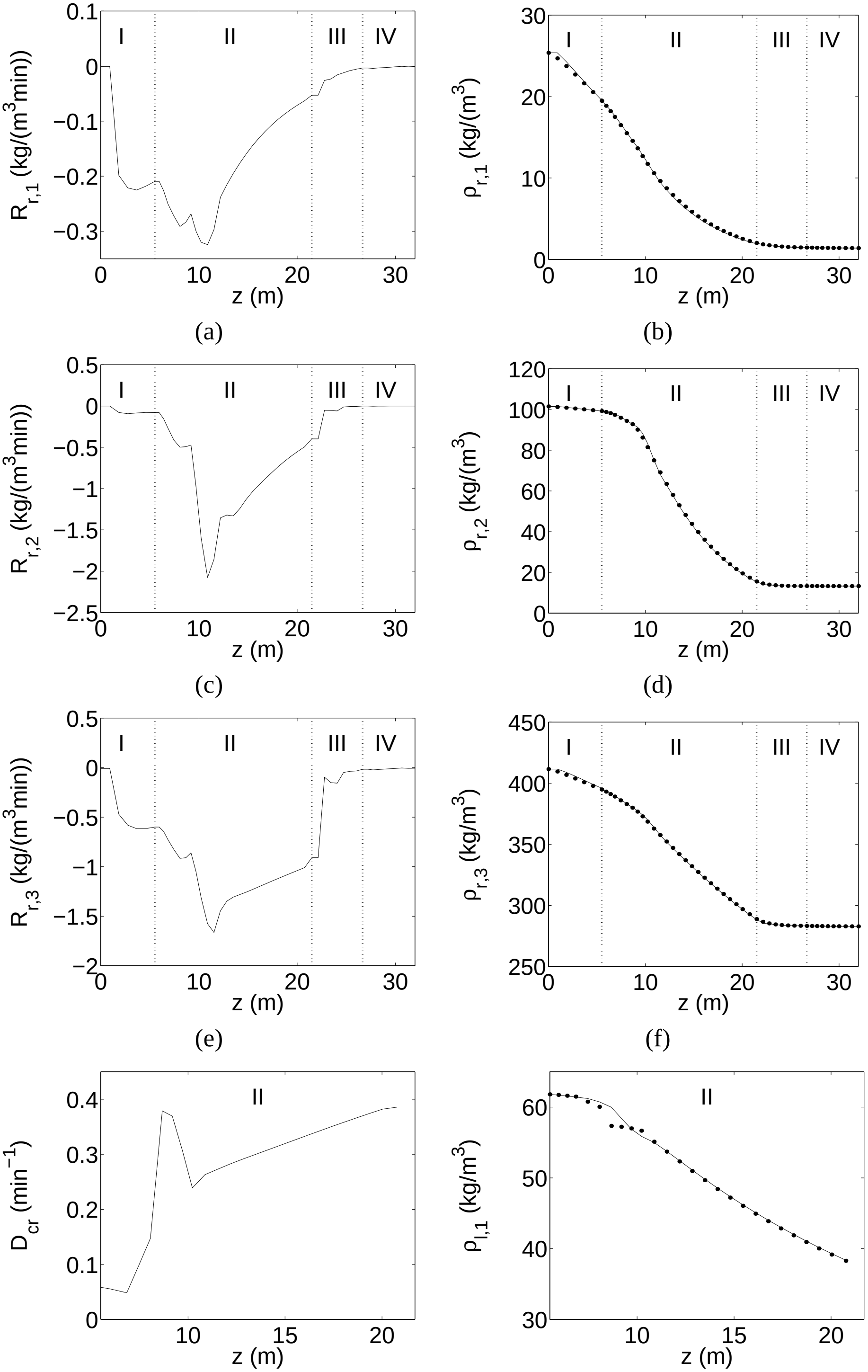
The estimates of the reaction rates and the corresponding species are shown in figure 5.10. Only the results of run ID5 (the standard operating conditions) are shown; the results of the other runs are similar. The estimates are of good quality. The errors in the estimates $\rho_{r,1}$ to $\rho_{r,3}$ and $\rho_{l,1}$ are small. The estimates of the reaction rates are negative because they describe consumption.

In the estimates of D_{cr}, a spike occurs around $z = 8\ m$. This is caused by a sudden increase in the estimation error of $\rho_{l,1}$ (figure 5.10 (h)). The ”measured” profile shows a sudden decrease in $\rho_{l1,}$, which is the result of the lower heater. The lower heater outlet is located at $z = 8.67\ m$. The lower heater inlet is located at $z = 10.2\ m$, where the concentration in $\rho_{l,1}$ is lower. The recycle flow is mixed with the liquor phase at the lower heater outlet, resulting in a decrease in concentration. The estimator cannot compensate for this ”bump”.

Consider the estimates of the reaction rates $R_{r,1}$ to $R_{r,3}$. For each of the reaction rates, PI-estimator reinitialization can be observed. At the start of each section, the estimation error is set to zero and the initial estimate of the reaction rate is set to the last estimate of the previous section (for section I, initial estimates of the reaction rates are set to zero, based on the assumption that no reaction takes place at $z = 0\ m$). Because the PI-estimator lags 1 step (due to the feedback structure), the estimates of the reaction rates remain constant for the first time step (figures 5.10 (a), (c) and (e)).

The estimates of the reaction rates can be interpreted physically. In section I at $z = 0\ m$, the reaction rate is low. In the beginning of section I, the reactants are brought together and the reaction starts slowly. By the end of section I, the rates have converged to a pseudo steady state.

In section II, the reaction phase is heated up. The influence of the two heaters can clearly be observed. The upper heater (at $z = 5.5\ m$) causes an increase of the reaction, followed by another increase caused by the lower heater (at $z = 8.7\ m$). After that, the reaction rate decreases because the reactants are being depleted. In sections III and IV, the reaction is quenched and the reaction rate decreases to zero.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

Figure 5.10: *PI-estimates run ID5. Solid line estimates, dots measurements*

Fuzzy model	*Inputs*	γ_{dr}	*# data features*	k_0	γ_{cm}	*# rules*	$RMSE$
$R_{r,1}$	$\rho_{r,1}, \rho_{r,4}, T_r$	2	179	10	0.7	4	0.024
$R_{r,2}$	$\rho_{r,2}, \rho_{r,4}, T_r$	1	333	10	0.7	6	0.14
$R_{r,3}$	$\rho_{r,3}, \rho_{r,4}, T_r$	5	198	10	0.7	6	0.12
D_{cr}	T_r	1	36	5	0.2	2	0.026

Table 5.8: *Fuzzy model identification settings and results*

The influence of the lower heater results in some overshoot in reaction rates $R_{r,2}$ and $R_{r,3}$. From a physical point of view, more smooth behavior is expected. However, the overshoot could not be reduced any further without compromising the estimates of the concentrations. Therefore, the estimation results as presented in figure 5.10 were accepted.

The use of the nine identification runs resulted in a total of 450 estimates for the reaction rates and 243 estimates for the diffusion coefficient D_{cr}. The required input-output data for the identification of the fuzzy models is now available.

5.5.4 Submodel identification

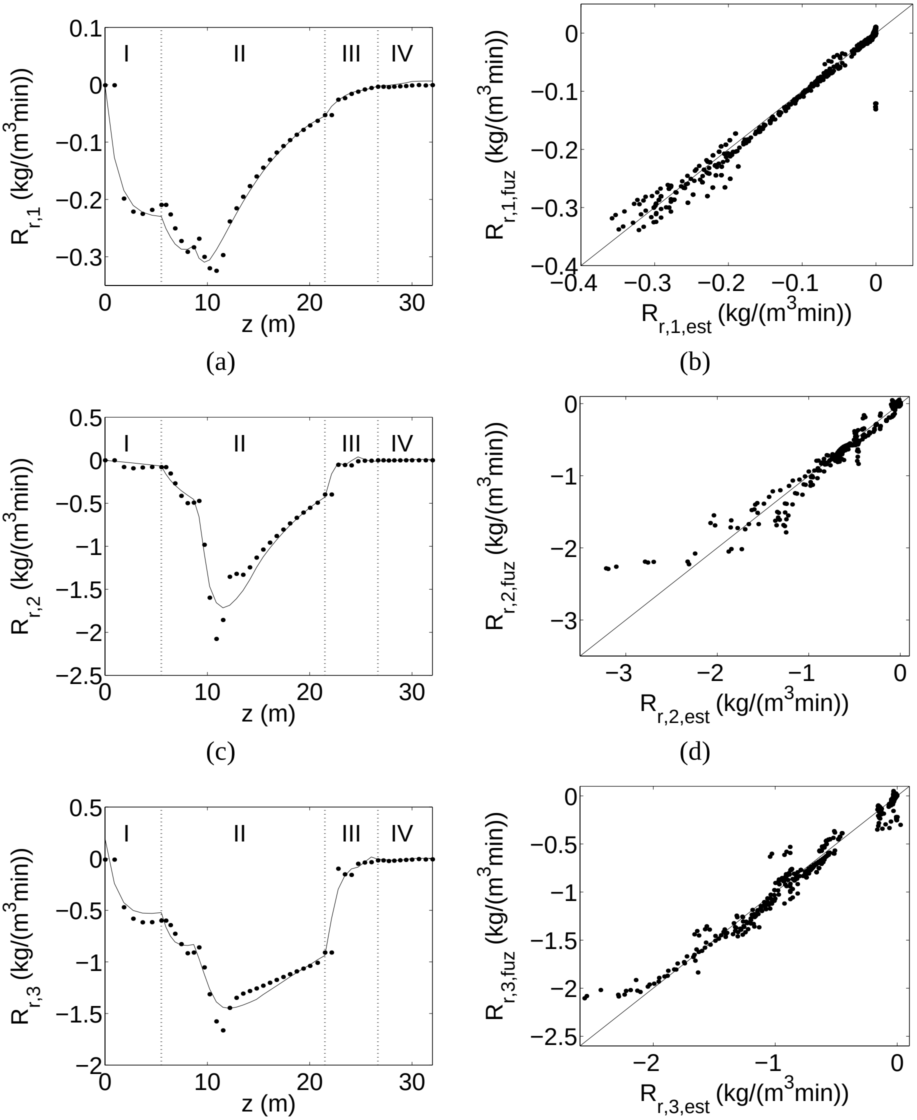
The four fuzzy models will be identified with GK-clustering in combination with cluster merging. This way, a simple fuzzy model is derived from the input-output data without the need to impose an initial model structure (section 4.4.1). The identification step consists of preparation of the data sets, clustering in combination with cluster merging, premise part construction by projection and consequent part construction using weighted least squares.

The first step is to reduce the data sets in such a way that the data features are distributed evenly across the input space. Usually, the data reduction threshold γ_{dr} (see example 4.5) is related to the maximum distance of two neighboring features. However, in this case, the data sets for the four fuzzy models consist of 9 trajectories, in which the data features have a small mutual distance, but between which the distance is relatively large. Care has to be taken that the threshold is not set at a value which results in loss of information. The thresholds were set manually and are shown in table 5.8.

For clustering, the following parameters have to be set: the initial number of clusters k_0 and the merging threshold γ_{cm}. For each model, several values of γ_{cm} were investigated. The values that were selected yielded relatively simple models with acceptable performance. Settings as well as clustering results are also shown in table 5.8.

The model for the reaction rates are shown in appendix D. Figure 5.11 (a), (c) and (e) show the performance of the reaction rates for run ID5. These results were generated by supplying the ”measured” profiles of the input variables $\rho_{r,i}$, $\rho_{r,4}$ and T_r. In addition, figure 5.11 (b),(d) and (f) show parity plots with respect of all identification data, which indicate the quality of the fit.

All fuzzy models perform well. The overall behavior is similar to that of the estimates. The parity plots indicate that modeling errors are larger when the reaction rate is larger (more



(a) (b)

(c) (d)

(e) (f)

Figure 5.11: *Fuzzy model results reaction rates (ID5) and parity plots (ID1-ID9). In (a), (c) and (e), dots indicate estimates, solid line fuzzy model.*

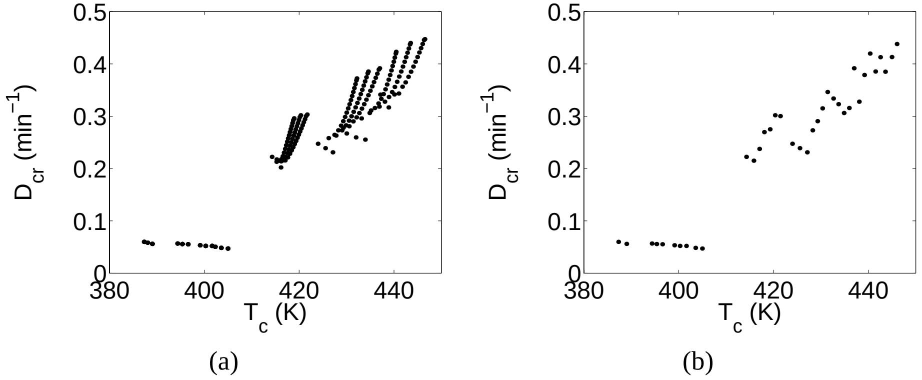


Figure 5.12: *Fuzzy model results diffusion (a) and reduced data set (b)*

negative). When the fuzzy models are incorporated in the hybrid model, these errors are integrated. Whether this effect plays a major role in the performance of the hybrid model, will become apparent during submodel integration.

Figure 5.12 shows the input-output identification data of runs ID1 to ID9 for the fuzzy model for D_{cr}. In figure 5.12 (a), the ”spikes” in the estimates are removed manually, resulting in limited information between $T_r = 405\ K$ and $T_r = 415\ K$. Figure 5.12 (b) shows the reduced data set. The figure shows that the diffusion coefficient D_{cr} is not a strict function of the reaction phase temperature T_r but that there is some other influence. The results are caused by the nonlinear effect that the lumping of species $(f, 1)$ and $(f, 3)$ has. This can be explained by solving the mass balances for $(f, 1)$ and $(f, 3)$ analytically (with neglection of the bulk flow and assuming steady state) and lumping the analytical solutions to form an equation that describes species $(l, 1)$. If this result is compared with the analytical solution of the mass balance that is imposed on $(l, 1)$ (equation 5.11), it can be seen that the structures are different. The PI-estimator derives D_{cr} from the behavior that is described by the lumped analytical solution by using an ”incorrect” imposed model structure, which results in the estimates for D_{cr} as shown in figure 5.12.

The problem can be solved by lumping the diffusion term into a relation that describes diffusion as a function of ρ_{l1}, ρ_{r4} and T_r. However, this results in a loss of transparency. Despite the mediocre quality of the estimates, the proposed structure to describe diffusion is maintained. Consequently, diffusion can still be interpreted as described by a driving force $(\rho_{l,i} - (1/\varepsilon)\rho_{r,i+3})$ and a diffusion coefficient, which is temperature dependent. The reduced data set is used to build a fuzzy model for D_{cr}. Results are shown in figure 5.13. The influence of errors in the fuzzy model on hybrid model results will be investigated during submodel integration.

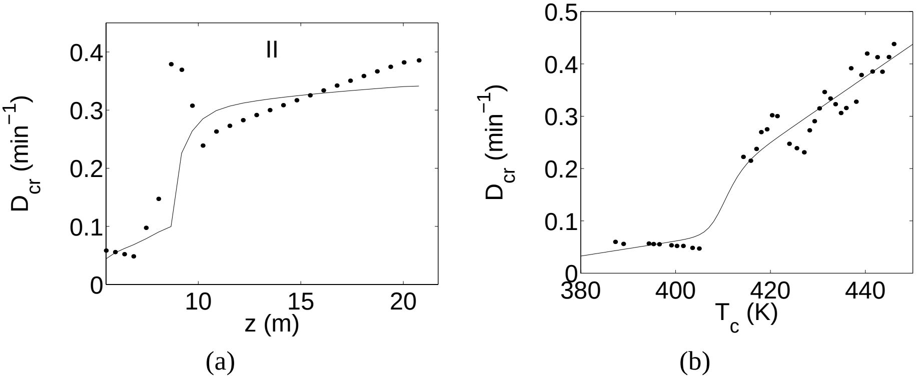


Figure 5.13: *Fuzzy model results for D_{cr}, run ID5 (a) and D_{cr} as a function of T_c*

5.5.5 Submodel integration

The fuzzy submodels for reaction rates $R_{r,1}$, $R_{r,2}$, $R_{r,3}$ and D_{cr} were combined with the physical framework to form the hybrid model. Figure 5.14 shows the Kappa number profile for ID5, table 5.9 shows key variable errors at $z = 32\ m$, the reactor outlet.

The hybrid model shows the same qualitative behavior for the Kappa number as the EPM. However, there is a discrepancy between the two profiles. Table 5.9 shows the model errors at the reactor outlet; there are significant errors in the hybrid model ($\varepsilon_{\kappa,hm}$ and $\varepsilon_{\gamma,hm}$ denote absolute errors, $\bar{\varepsilon}_{\kappa,hm}$ and $\bar{\varepsilon}_{\gamma,hm}$ denote relative errors).

The hybrid model errors are caused by:

- Model reduction. The simplifications result in slightly different behavior.
- PI-estimates. Errors in the estimates of the reaction rates and the diffusion coefficient manifest themselves through the fuzzy models in the hybrid model results.

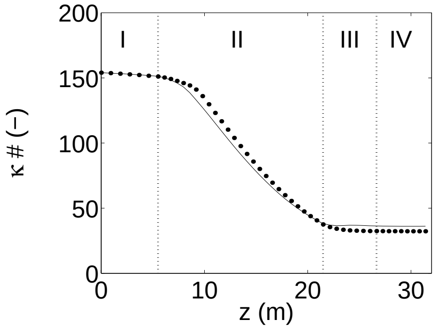


Figure 5.14: *Initial $\kappa\#$ profile hybrid model run ID5*

Error	*ID1*	*ID2*	*ID3*	*ID4*	*ID5*	*ID6*	*ID7*	*ID8*	*ID9*
$\varepsilon_{\kappa,hm}$	-2.60	-3.03	3.88	-5.22	-3.85	2.68	-5.78	-3.23	5.57
$\bar{\varepsilon}_{\kappa,hm}$	0.03	0.07	0.15	0.07	0.12	0.17	0.08	0.11	0.43
$\varepsilon_{\gamma,hm}$	-0.02	-0.03	-0.03	-0.02	-0.03	-0.03	-0.01	-0.02	-0.02
$\bar{\varepsilon}_{\gamma,hm}$	0.03	0.05	0.06	0.03	0.05	0.07	0.02	0.03	0.04

Table 5.9: *Hybrid model errors before submodel integration*

- Identification errors. The error of the fuzzy models with respect to the estimates also is present in the hybrid model results.
- Error integration. The error increases towards the end of the reactor.

The fuzzy models will have to be adjusted in order to improve hybrid model performance. This involves optimization of the fuzzy model parameters. To gain insight in the influence of the fuzzy models on the errors, the sensitivity of $R_{r,1}$, $R_{r,2}$, $R_{r,3}$ and D_{cr} with respect to the Kappa number and the yield is calculated.

Sensitivity analysis

The sensitivity of the kappa number and the yield with respect to changes in the reaction rates and diffusion coefficient was determined by introducing fuzzy model weights w_m in the hybrid model. The model weights can be interpreted as multipliers of the reaction rates and the diffusion coefficient. The model is evaluated for different values of these multipliers. Determining the change in the Kappa number and the yield with respect to changes in the multipliers give the sensitivity:

$$S^{\kappa}_{R_{r,i}} = \left| \frac{\Delta\kappa\#/\kappa\#}{\Delta w_m^{R_{r,i}}/w_m^{R_{r,i}}} \right| \quad \text{for } i = 1,\ldots,3 \tag{5.12}$$

$$S^{\kappa}_{D_{cr}} = \left| \frac{\Delta\kappa\#/\kappa\#}{\Delta w_m^{D_{cr}}/w_m^{D_{cr}}} \right| \tag{5.13}$$

$$S^{\gamma}_{R_{r,i}} = \left| \frac{\Delta\gamma\#/\gamma\#}{\Delta w_m^{R_{r,i}}/w_m^{R_{r,i}}} \right| \quad \text{for } i = 1,\ldots,3 \tag{5.14}$$

$$S^{\gamma}_{R_{r,i}} = \left| \frac{\Delta\gamma\#/\gamma\#}{\Delta w_m^{D_{cr}}/w_m^{D_{cr}}} \right| \tag{5.15}$$

The hybrid model was simulated for the conditions of run ID5; the assumption is made that sensitivity does not change much within the identification domain. Each multiplier was changed from 1.00 to 1.05. Results are shown in table 5.10.

The sensitivity of the yield γ is the highest for changes in $R_{r,3}$. The sensitivity for the other parameters is lower than for $R_{r,3}$, but approximately in the same order of magnitude. The sensitivity of the Kappa number $\kappa\#$ is the highest for changes in $R_{r,2}$, while the sensitivity for changes in the other parameters is relatively low.

	$S_{R_{r,1}}$	$S_{R_{r,2}}$	$S_{R_{r,3}}$	$S_{D_{cr}}$
$\kappa\#$	0.19	1.50	0.26	0.03
γ	0.01	0.07	0.14	0.05

Table 5.10: *Sensitivities hybrid model*

Parameter optimization

The most straightforward approach is to improve γ by optimizing $R_{r,3}$ and $\kappa\#$ by optimizing $R_{r,2}$. This means that two optimizations are performed sequentially. If this yields unacceptable results, other reaction rates or the diffusion coefficient can be incorporated in the optimization procedure.

The reaction rates $R_{r,2}$ and $R_{r,3}$ influence both $\kappa\#$ and γ. The sensitivity $S^{\kappa}_{R_{r,2}}$ is much higher than $S^{\gamma}_{R_{r,2}}$, while $S^{\kappa}_{R_{r,3}}$ and $S^{\gamma}_{R_{r,3}}$ are in the same order of magnitude. Because $S^{\kappa}_{R_{r,2}} > S^{\gamma}_{R_{r,2}}$, the optimization of $R_{r,2}$ will not affect γ as much. Therefore, $R_{r,2}$ should be optimized after $R_{r,3}$ is optimized.

The two fuzzy models contain 192 parameters. To keep the optimization simple, the number of parameters that is optimized should be kept as low as possible. This can be accomplished by optimizing fuzzy model weights or rule weights instead of the individual model parameters. To provide sufficient flexibility to deal with nonlinearity, the rule weights $w_j^{R_{r,ij}}$ of the two models are optimized. This results in the optimization of 6 parameters in each of the two fuzzy models.

The goal function for the optimization of $R_{r,3}$ is a standard quadratic error criterion. Instead of incorporating the error in γ, the error in $\rho_{r,3}$ is used; if the states of the model are described well, the model outputs are described well. In addition, interpretation of the behavior of the states is improved. The goal function is given by:

$$J = \frac{1}{2}\sum_{j=1}^{9}\sum_{i=1}^{n}(\rho_{r,3,epm,i,IDj} - \rho_{r,3,hm,i,IDj})^2 \tag{5.16}$$

in which $\rho_{r,3,epm,i,IDj}$ denotes the ith value in the EPM concentration profile of $\rho_{r,3}$ for experiment IDj and $\rho_{r,3,hm,i,IDj}$ denotes the ith value in the hybrid model concentration profile of $\rho_{r,3}$ for experiment IDj. n is the number of features per experiment. The goal function for the optimization of $R_{r,2}$ is similar to equation 5.16:

$$J = \frac{1}{2}\sum_{j=1}^{9}\sum_{i=1}^{n}(\rho_{r,2,epm,i,IDj} - \rho_{r,2,hm,i,IDj})^2 \tag{5.17}$$

Bounds were imposed on the rule weights:

$$0 \leq w_j^{R_{r,i}} \leq 2 \tag{5.18}$$

in which i denotes the species and j denotes the rule number.

Rule #:	1	2	3	4	5	6
$R_{r,2}$	0.73	0.93	0.47	1.19	1.14	1.70
$R_{r,3}$	0.99	1.38	1.04	1.95	1.25	1.12

Table 5.11: *Optimized rule weights*

A standard Nelder-Mead simplex algorithm is used to perform the optimization. A direct search method omits the need to calculate gradient information, which is cumbersome for the hybrid model. Although the algorithm may require many iterations and can get stuck in local optima, it can be used as long as performance requirements are met.

Results

The optimized weights are shown in table 5.11. Figure 5.15 shows the results before and after the optimization step, table 5.12 shows the errors after optimization.

Figure 5.15 shows that the hybrid model concentration profiles follow the measurements more closely after optimization. The error in the yield γ is negligible (table 5.12). The overall error in the Kappa number $\kappa\#$ at the reactor outlet has been decreased, although for some runs, the error is larger (ID2 and ID5).

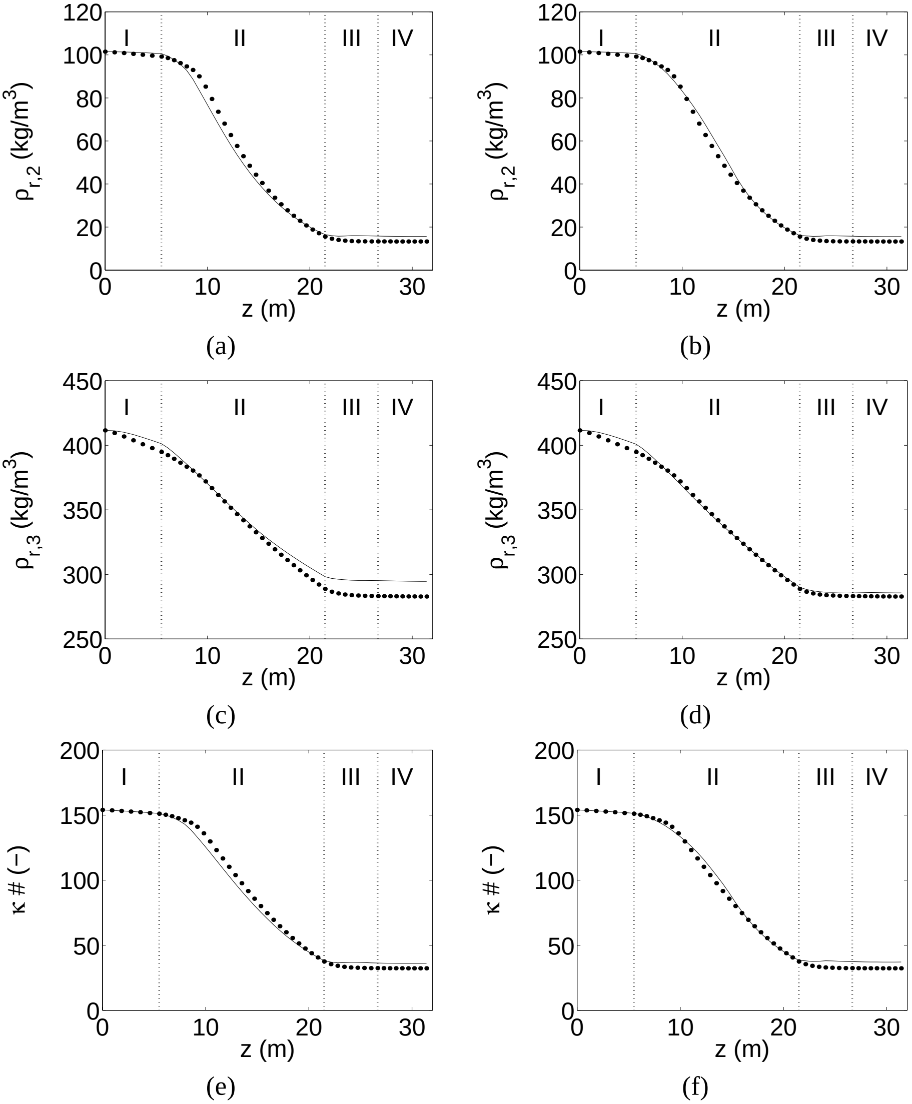
The optimization results show that the hybrid model behavior matches the static behavior of the EPM for runs ID1 to ID9 closely. Table 5.13 shows that the hybrid model is consistent with respect to the mass balances within 3 %, the result of rounding errors. The liquor flow at the reactor outlet is assumed to be zero; it is assumed that the amount of liquor phase that leaves the reactor at the bottom is negligible.

5.6 Hybrid model analysis

The next step is to analyze the hybrid model in detail to determine whether the quality requirements are met. The hybrid model of the digester will be analyzed with respect to static performance, dynamic performance, complexity, interpretability and process independence.

Error	*ID1*	*ID2*	*ID3*	*ID4*	*ID5*	*ID6*	*ID7*	*ID8*	*ID9*
$\varepsilon_{\kappa,hm}$	2.47	-3.86	-0.28	-0.82	-4.87	-1.09	-1.00	-2.61	2.23
$\bar{\varepsilon}_{\kappa,hm}$	0.03	0.10	0.01	0.01	0.15	0.07	0.01	0.09	0.17
$\varepsilon_{\gamma,hm}$	0.00	-0.01	-0.01	0.00	-0.01	-0.01	0.01	0.00	0.00
$\bar{\varepsilon}_{\gamma,hm}$	0.01	0.03	0.02	0.01	0.02	0.02	0.01	0.01	0.00

Table 5.12: *Hybrid model errors after submodel integration*



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5.15: *Simulation results run ID5 before (a,c,e) and after (b,d,f) optimization. Solid line hybrid model, dots measurements*

Stream	*Mass flow in* (kg/min)	*Mass flow out* (kg/min)
Reaction	748.5	564.0
Liquor	494.4	-
Extraction	-	879.4
Filtrate	154.6	-
Total	1398	1443

Table 5.13: *Overall mass balance hybrid model*

5.6.1 Static performance

The static performance of the hybrid model will be analyzed by evaluating the results of the hybrid model for runs VAL1 to VAL8 (table 5.6) and by investigating the static extrapolation behavior.

Table 5.14 shows the errors at the reactor outlet for the validation runs. In figure 5.16, parity plots for $\kappa\#$ and γ are given. The overall static performance is good. The errors are slightly larger than the errors for the identification runs, but still within acceptable limits. Similar to the identification runs, the error in $\kappa\#$ is larger than the error in γ. The relative error in $\kappa\#$ for VAL7 is large, while the absolute error is in the same order of magnitude as the errors in VAL1-VAL4. The relative error is large since the $\kappa\#$ at the reactor outlet is relatively low for VAL7 (figure 5.17).

The qualitative behavior is in accordance with the measurements. Figure 5.17 illustrates this for VAL1 (interpolation) and VAL7 (extrapolation). Details, however, are different. For example, in VAL7, $\kappa\#$ of the hybrid model starts to decrease at a lower z than the $\kappa\#$ of the EPM, but in section II, the reaction rate in the hybrid model is lower than in the EPM, resulting in a higher $\kappa\#$.

Because the error in γ is similar to the error during the identification experiments (tables 5.12 and 5.14), it can be concluded that the fuzzy model for $R_{r,3}$ performs well for the validation runs. Since the fuzzy model for $R_{r,3}$ performs well, the larger error in $\kappa\#$ is caused by the error in $R_{r,2}$, for which the Kappa number has the highest sensitivity. The influence of the other fuzzy models on the Kappa number is much smaller. As a result, the errors caused by these fuzzy models are much smaller than the error caused by the fuzzy model for $R_{r,2}$.

To investigate the interpolation and extrapolation properties in more detail, the hybrid model performance as a function of Q_h (which is equivalent to the energy supply of the lower and upper heater, that can be derived from T_{uh} and T_{lh} in the EPM) and ϕ_l (equivalent to the free liquor flow ϕ_f in the EPM) was calculated.

Error	*VAL1*	*VAL2*	*VAL3*	*VAL4*	*VAL5*	*VAL6*	*VAL7*	*VAL8*
$\varepsilon_{\kappa,hm}$	7.55	-4.37	-5.81	-2.00	-6.04	0.81	-6.62	-1.16
$\bar{\varepsilon}_{\kappa,hm}$	0.14	0.17	0.12	0.10	0.12	0.04	0.55	0.18
$\varepsilon_{\gamma,hm}$	0.03	-0.02	0.00	-0.01	0.00	0.00	0.01	0.01
$\bar{\varepsilon}_{\gamma,hm}$	0.05	0.04	0.00	0.02	0.01	0.01	0.01	0.02

Table 5.14: *Hybrid model errors validation runs*

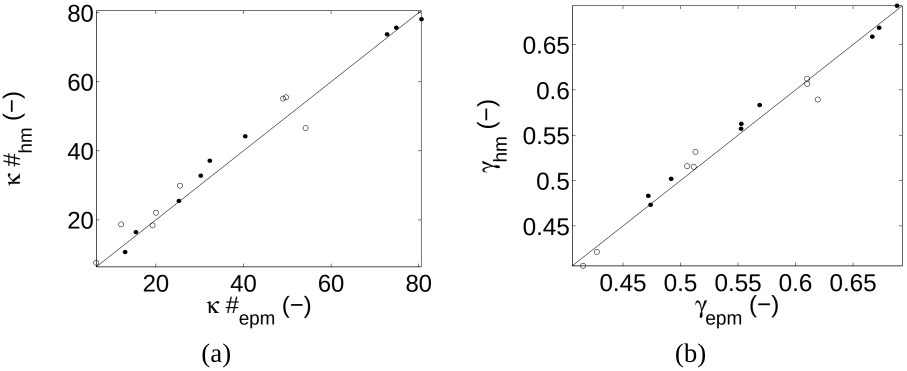


Figure 5.16: *Parity plots for $\kappa\#$ (a) and γ (b). Solid dots ID runs, open dots VAL runs*

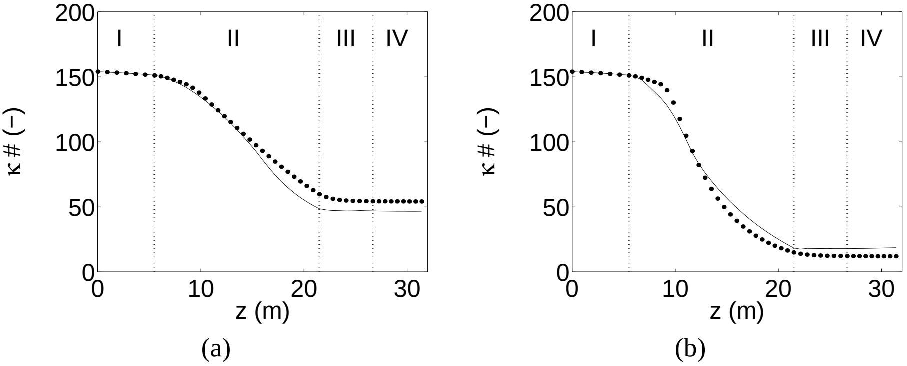


Figure 5.17: *Simulation results $\kappa\#$ run VAL1 (a) and VAL7 (b). Solid line hybrid model, dots measurements*

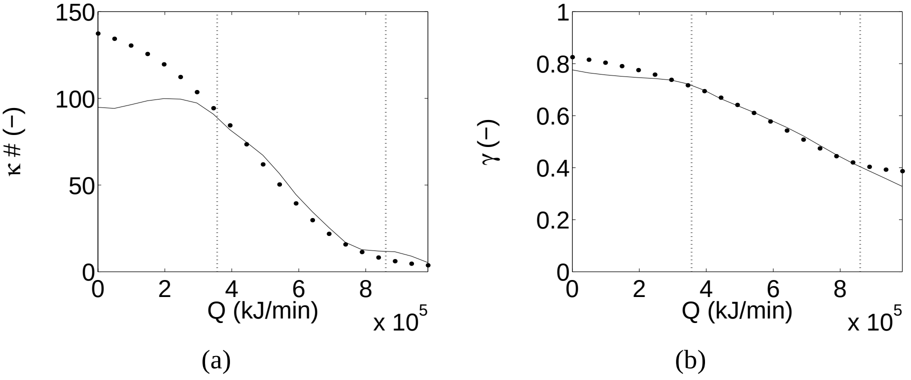


Figure 5.18: *Static extrapolation results for $\kappa\#$ (a) and γ (b). Solid line hybrid model, dots EPM. Vertical lines indicate identification domain*

Figure5.18 (a) shows the Kappa number at the reactor outlet as a function of the heat input Q, which can be interpreted as a general production curve. As temperature rises, the production increases exponentially. At higher temperatures, exhaustion is observed, resulting in depletion of the reactants.

The liquor flow ϕ_l was kept constant at $2.34\ m^3/min$. Q was varied from 0 (heaters switched off) to 25 % of the identification range beyond the maximum identification value. Within the identification range ($350000 \leq Q \leq 860000$), the hybrid model error is relatively constant and small. For $Q > 860000\ kJ/min$, the Kappa number of the EPM converges to zero due to exhaustion of components $\rho_{r,1}$ and $\rho_{r,2}$. Since the mass balances of the hybrid model are based on first principles, the hybrid model Kappa number will also converge to zero.

At low Q, the production as calculated by the hybrid model becomes constant. Under these conditions, only rule 2 is active in the fuzzy model for $R_{r,2}$. Although the reaction conditions differ slightly due to the changing Q, the overall reaction rate does not change much. The result is that $\kappa\#$ remains constant.

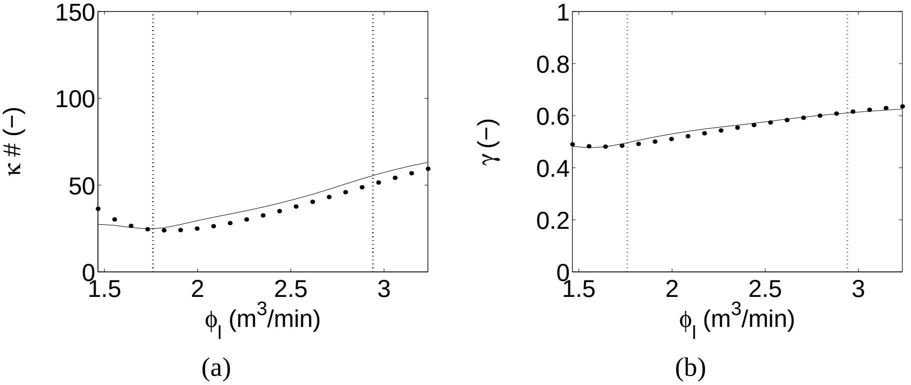


Figure 5.19: *Static extrapolation results for $\kappa\#$ (a) and γ (b). Solid line hybrid model, dots EPM. Vertical lines indicate identification domain*

Figure5.18 (b) shows the yield as a function of Q. The results are similar to figure5.18 (a). However, at high Q, the yield of the EPM converges to a constant value, while the yield of the hybrid model continues to decrease. In the EPM, the conversion is limited by ”inert concentrations”; a fraction of species $s, 3$ and $s, 4$ does not react, which means that the yield has a minimum value that is greater than zero. This limitation is not accounted for in the hybrid model, as a result of which the yield converges to zero.

To investigate extrapolation performance resulting from variations in the liquor flow rate, Q_h was kept constant at 627000 kJ/min (the value for ID5). The flow rate ϕ_l was varied from 1.46 m^3/min to 3.23 m^3/min. The results for the Kappa number and the yield are shown in figure 5.19.

Although the concentration of the liquor in the reaction phase ($\rho_{r,4}$) rises as a result from the increase in ϕ_l, the reaction phase temperature decreases, since the heat supply of the heater remains constant. The net result is that conversion decreases with higher flowrates, as illustrated in figure 5.19. At low flow rates, $\rho_{r,4}$ becomes the limiting factor, which also results in a decrease in conversion.

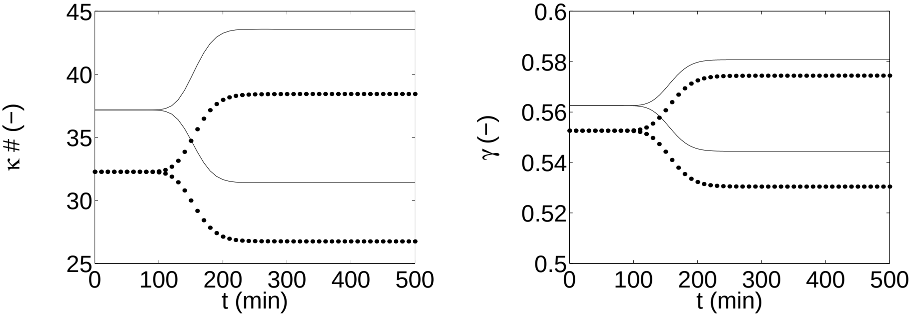
Changes in the liquor flow rate do not affect the production as much as changes in the heat supply of the heater. As a result, similar performance is observed throughout the extrapolation range. For $\kappa\#$, the error remains constant at approximately the same value as the error for ID5. The error is mainly caused by the error in $\rho_{r,2}$, which in turn is the result of the performance of the fuzzy model for $R_{r,2}$. The performance of the yield is good over the entire extrapolation domain, indicating good performance of the fuzzy model for $R_{r,3}$.

5.6.2 Dynamic performance

Since the dynamic structure of the hybrid model is based on first principles, the model is expected to perform well dynamically. This will be analyzed using step responses of step changes in the heat supply of the heater Q_h and the liquor flow rate ϕ_l. In addition, frequency extrapolation will be investigated.

Figure 5.20 shows the results for a step change in the lower heater heat supply at $t = 0$ min. The change in Q_{lh} of the EPM and Q_h of the hybrid model was identical. The step was performed for the operating conditions of run ID5. The steady state offset between the EPM and the hybrid model can clearly be seen.

Table 5.15 shows the amplitudes $\Delta\kappa\#$ and $\Delta\gamma$ for the step change. Values for both a positive and a negative step change are given. Figure 5.20 (a) shows the response of the Kappa number at the reactor outlet for a change in Q_h. The change in Kappa number is approximately the same for both models. This means that under these conditions, the steady state offset remains constant. The dynamic behavior of both models is identical; both show a second order response and similar dead times. Figure 5.20 (b) shows the response of the yield. The influence of the step on the yield is very small, but performance of the hybrid model is similar to the performance of the EPM.



(a)

(b)

Figure 5.20: *Hybrid model step response results for $\kappa\#$ (a) and γ (b) for positive and negative step change in Q. Dots EPM, lines hybrid model*

Figure 5.21 shows the response of the Kappa number and the yield for a step change in the free liquor flow ϕ_l. The gain of both models with respect to changes in ϕ_l is different. This can be explained as follows. A step change in ϕ_l results in a change of the liquor concentration $(r, 4)$. It was found there is not much difference between the concentrations of species $(r, 4)$ in the EPM and the hybrid model. This indicates that diffusion is described in a similar fashion by both models and that the difference in gain is the result of different sensitivities of the reaction rates to changes in species $(r, 4)$.

In addition, the figure shows that two effects play a role. Consider the response of the hybrid model. If the flow rate is increased, then the concentration of the liquor reactants in the reaction phase will increase over the entire reactor as a result from the increase in the driving force of the diffusion. This results in a decrease of the Kappa number. However, since the heat supply of the heaters is constant, the temperature will drop, resulting in an overall increase in the Kappa number. These effects have the same influence on the yield, but since the yield does not change much, the influence of the increase in concentration can be neglected.

The first effect is not observed in the EPM. Since the volume of the free liquor phase in the EPM is not constant due to compaction, the effect of a step change in the free liquor flow rate is different. The second effect has a similar influence as it has on the hybrid model. The overall Kappa number is increased as a result form the decrease in reaction temperature.

The model orders O, time constants τ and dead times θ indicate that dynamic performance of the hybrid model is good (see table 5.15). The dead times were derived from the step responses, while the orders and time constants were determined using the Bode diagrams, given in appendix D. The influence of the simplifications is observed in the hybrid model

Model	*Input*	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$ $(min$	$\tau_{[input],\gamma}$ $(min$	$\theta_{[input]}$ (min)	$\Delta\kappa\#$ $(-)$	$\Delta\gamma$ $(-)$
EPM	Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
	ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020
Hybrid	Q_{lh}	1.8	1.9	40	45	84	-5.77, 6.40	-0.018, 0.018
model	ϕ_f	2.4	2.7	34	34	109	4.23, -3.69	0.014, -0.013

Table 5.15: *Frequency response analysis results for hybrid model*

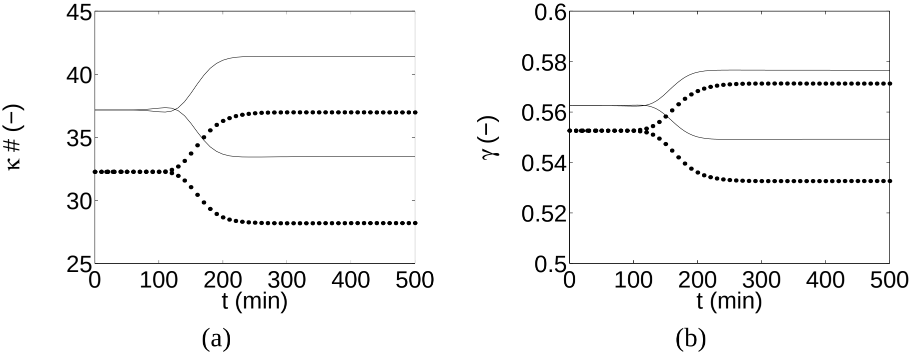


Figure 5.21: *Hybrid model step response results for $\kappa\#$ (a) and γ (b) for positive and negative step change in ϕ_l. Dots EPM, lines hybrid model*

response to a change in the flow rate, but the overall response is in accordance with the EPM. The time constants and dead times of both models are approximately the same. There are some differences with respect to the orders of the models. The order of the hybrid model is slightly lower than the order of the EPM, since fewer PDE's are present. In addition, numerical inaccuracies during the construction of the Bode plots influence the results; the frequency range limits accurate determination of the order. However, the overall performance indicates that the dynamic behavior of both models is similar.

The analysis of the dynamic behavior of the hybrid model can also be viewed in the context of frequency extrapolation. Frequency extrapolation occurs when the model is simulated at different frequencies than the frequencies that were used during identification. In the case of the hybrid model, the identification was performed using static data. Within this context, all dynamic experiments presented in this section can be interpreted as frequency extrapolation experiments. From this, it can be concluded that the hybrid model possesses good frequency extrapolation properties.

5.6.3 Complexity

The flowsheet of the hybrid model contains four sections that describe the most important functional zones of the digester. The upper and lower heater are lumped and the heater recycle flows are omitted. The quench recirculation is also omitted; in the EPM, quench recirculation is set to zero (Wisnewski *et al.*, 1997). The flowsheet is more straightforward than the flowsheet of the EPM, while it is still possible to distinguish the different zones in the digester.

The DFD of the section model (appendix D) clearly shows the reduced complexity of the relations between the modeled phenomena. In comparison to the EPM, the entrapped phase mass balance is not present, as is the bulk flow. Although the DFD is simpler, it does not show the relations between the different components. The relation is only illustrated by the model equations, which makes interpretation more cumbersome.

The complexity of the model equations is similar to the EPM. Although less states are modeled, the structure of the mass and energy balances is identical; in both models, the balances consist of partial differential equations, describing the states as a function of time and place. The connection between the balances in both models is also similar. The main difference is that the hybrid model contains fuzzy submodels.

The fuzzy submodel for the diffusion coefficient is a very simple two rule SISO model. The three fuzzy submodels for the reaction rates are more complex. The MISO models each are a collection of local linear models. The activation of these models is based on the values of the input variables. Although the structure of these submodels is clear, it is difficult to determine which of the local models is most important, or how the individual local models determine the behavior of the reaction rates.

The complexity of the fuzzy models is mainly determined by the number of rules and the rule base. The clustering algorithm in combination with structure optimization provides a way to minimize fuzzy model complexity, which yielded acceptable results. Other approaches for evaluating fuzzy model complexity exist and more information can for example be found in Setnes (1999).

5.6.4 Interpretability

The static behavior of the model can be explained physically by using the model flowsheet and section model DFD. For example, the interpretation of the temperature profile was based on the model flowsheet in order to distinguish the different functional zones. On a more detailed level, the development of the concentration profiles can be explained by using the section model DFD. A decrease in temperature results in a decrease in reaction, as a result of which the Kappa number rises.

The interpretation that is given to the simulation results or the model structure is based on the process hypotheses that were used to build the model. This result in a level of interpretation which is limited to the information that these hypotheses provide. In the case of the hybrid models, the level of interpretation is limited to relations in terms of accumulation, convection, reaction or diffusion. Physical interpretation on a more detailed level is more difficult and even impossible for the fuzzy submodels, which give an input-output mapping.

The fuzzy models do, however, provide information about characteristic operating regimes that can be interpreted; the reaction rate is high for high concentrations and high temperature. In addition, the distribution of the membership functions gives information about the nonlinearity of the model output with respect to an input. For both $R_{r,2}$ and $R_{r,3}$, a characteristic transition can be observed at around $T = 425\ K$. Such a transition can also be observed for $R_{r,1}$, but is less clear. The nonlinearity with respect to $\rho_{r,4}$ is comparable for all three reaction rate models. The membership functions of $\rho_{r,4}$ in the models for $R_{r,2}$ and $R_{r,3}$ are similar. The model for $R_{r,1}$ shows the same trends; $\rho_{r,4} = mf4$ and $\rho_{r,4} = mf4$ in the model for $R_{r,1}$ corresponds with $\rho_{r,4} = mf3$ and $\rho_{r,4} = mf5$ in the model for $R_{r,2}$, respectively. The nonlinearity with respect to the wood species differs for the three models, which indicates different kinetic characteristics.

To explain the differences in the behavior of the fuzzy models in more detail, the consequent parameters should be included in the analysis. However, the analysis of the observed behavior (section 5.5.3) and the premise part membership functions provides some a posteriori information about the reaction kinetics. This is partly possible due to the transparent structure of fuzzy models.

The data used to construct the fuzzy relations for the reaction rates and diffusion coefficient is derived form the framework that was proposed a priori. The data was not obtained from measurements or lab analysis. The fuzzy relations are specifically designed for the physical framework that is presented. Therefore, the reaction rates that are calculated by the fuzzy models should be interpreted as a *measure* for the reaction rate, appropriate for the chosen model structure. This is also true for the diffusion coefficient.

The level of physical interpretation of the hybrid model is similar to the EPM. In the EPM, the level of interpretation is also limited by the physical framework. Interpretation on a more detailed level involves analysis of empirical relations. The EPM, however, has a more detailed model structure, which means that it provides more information about the unit operation and the relation between specific wood and liquor components.

5.6.5 Process independence

The level process independence is based on the sources of information that are used. In the case of the hybrid model, the main source of information was the EPM. This means that in general, the hybrid model can be used for the same applications as the EPM, as long as the model can provide the required information. Some comments, however, need to be made.

The EPM can be viewed upon as the ”process” for which the hybrid model was built. The hybrid model is completely based on this ”process”, which makes it process dependent. It has the same possibilities and limitations as the EPM has. In addition, with respect to static performance, it is only valid for operating conditions that were used during identification. If the model is to be used for other conditions, for example other wood species, it can be adjusted by re-identifying the fuzzy equations. New identification data is then required. The rest of the model can remain unchanged.

The application possibilities of the hybrid model on different continuous pulp digester installations are the same as the possibilities the EPM has. The model may need adjustment of the fuzzy relations or other model parameters to provide acceptable performance. However, due to the first principles information, the main physical phenomena that play a role are accounted for, which means that the adjustments will be minor.

Similar to the EPM, the hybrid model has limited process independence with respect to process design. It provides little information about design parameters, except for reactor height and width. In addition, the need for parameter adjustment in order to use the model for different digester installations makes it more suited for studying process operation than for process design.

5.6.6 Remarks

The hybrid model describes the key variables of the pulp digester acceptably (requirements 1 and 2). The maximum static error is around 5 units for $\kappa\#$ and around 0.01 for γ. The dynamic characteristics of the hybrid model and its reference are similar, although the detected order of the hybrid model was somewhat lower than the order of the EPM.

The description of the variables is based on a simplified interpretation of the process, which results in a model that is transparent and a model structure which can be interpreted physically (requirements 3 and 4). Although it provides less detailed information than the EPM, it is still possible to explain the digester behavior in terms of convection, reaction and diffusion.

The differences in model structure between the EPM and the hybrid model manifest themselves in the states that are modeled and hence the observed order (requirement 5). The lumping of some of the species is based on physical assumptions and results in differences in the equation structure (such as for the reaction rates) or in the model structure (the connections between the modeled effects).

Since the fuzzy models describe static relationships, no dynamical data was needed during identification. The dynamic characteristics are determined by the first principles framework. This makes the modeling approach useful if limited amounts of dynamic data are available and first principles provide sufficient information to design the model framework.

5.7 Fuzzy model problem definition and design

To determine hybrid model properties in relation to first principles models and fuzzy models, a third, fuzzy model for the digester is developed. This model is completely black box and represents one extreme of the model ”spectrum” (section 5.4.1). As with the hybrid model, the EPM will serve as the reference. In order to be able to compare the fuzzy model with the hybrid model, the input-output structure has to be similar. The fuzzy model should describe the static as well as the dynamic behavior of the digester.

5.7.1 Model structure

To meet the model requirements, the fuzzy model will describe the key variables: the Kappa number $\kappa\#$ and the yield γ at the reactor outlet. The most simple model structure is a black box approach, in which $\kappa\#$ and γ are described as a function of the inputs. This results in a model that does not provide information about concentration profiles, such as the EPM and the hybrid model do. However, the purpose of the fuzzy model is to provide a black box description of the digester behavior. This means that a description of the key variables based on concentration information is not required and that the fuzzy model can describe $\kappa\#$ and γ directly instead.

Similar to the hybrid model, the fuzzy model will describe the kappa number and the yield for a fixed wood chip flow rate and one type of wood. The input variables that are required are the heat input Q_h and the free liquor flow ϕ_l. The model will consist of two Multiple Input Single Output (MISO) models; one for each of the output variables.

The dynamics are incorporated by using an ARX structure. A second order structure will be used (see section 5.3.1). In addition, the dead times with respect to the input variables needs to be incorporated. This yields the following model structure:

$$\kappa\#_k = f_{fuzzy}(\kappa\#_{k-1}, \kappa\#_{k-2}, Q_{h,k-\theta_Q}, \phi_{l,k-\theta_\phi}) \tag{5.19}$$

$$\gamma_k = f_{fuzzy}(\gamma_{k-1}, \gamma_{k-2}, Q_{h,k-\theta_Q}, \phi_{l,k-\theta_\phi}) \tag{5.20}$$

in which k denotes the time step and θ_Q and θ_l denote the dead times with respect to Q_h and ϕ_l.

The fuzzy model will be of the Sugeno type. The MISO models can then be interpreted as a collection of local linear ARX models.

5.7.2 Data acquisition

The identification data needs to represent the dynamic behavior of the pulp digester. To accomplish this, identification and validation data sets have been designed based on ramped-RMRI (Random Magnitude Random Interval) input signals. Such a signal consists of a series of ramped steps of random magnitude with, spaced at random intervals.

To design the input signals, a magnitude domain, an interval domain and a ramp domain need to be specified. The input signals are allowed to vary between the standard operating conditions +/- 10 %, similar to the identification data of the hybrid model.

To generate the input signal for Q_h, the EPM was subjected to ramped-RMRI signals for both the upper heater heat supply Q_{uh} and the lower heater heat supply Q_{lh}. The signal that is used for Q_h during identification of the fuzzy model consists of the sum of these two signals.

The length of the interval between two steps, τ_{is}, is chosen at random using the following rule of thumb:

$$\frac{1}{4}\tau_{ss} < \tau_{is} < \frac{5}{4}\tau_{ss} \tag{5.21}$$

with

$$\tau_{ss} = \theta + 5\tau \tag{5.22}$$

Model	$\theta_{Q,[output]}\ (min)$	$\theta_{\phi,[output]}\ (min)$	k_0	γ_{cm}	*# rules*
$\kappa\#$	87	113	6	0.7	3
γ	87	113	6	0.4	5

Table 5.16: *Settings and structure results fuzzy model*

where τ is the first order time constant and θ is the dead time with respect to the input signal. The ramp domain was arbitrarily set: the minimum duration was 2 min, the maximum duration was 200 min. To acquire sufficient information, 25 different steps were performed, resulting in a signal duration of approximately 8000 min.

The EPM was simulated using the ramped-RMRI input signals. The resulting input-output data sets for identification and validation are shown in appendix A.

5.7.3 Model identification

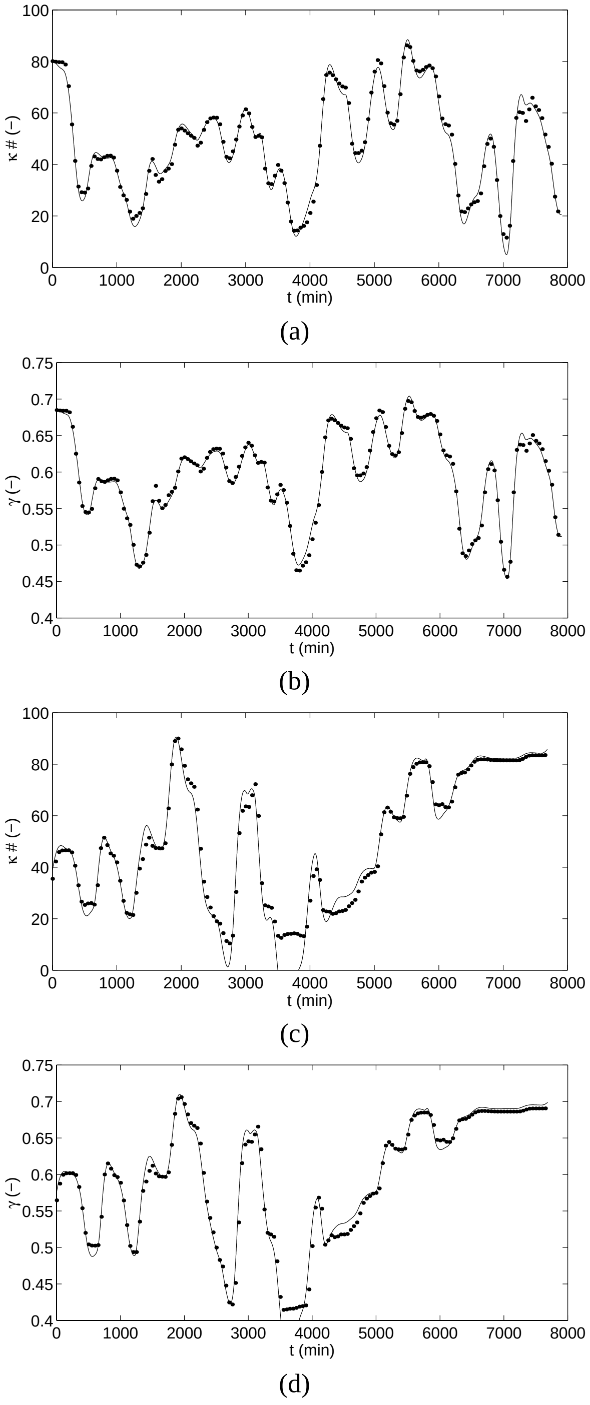
Both MISO models were identified using GK-clustering in combination with structure optimization (section 4.4.1). Since the large values of Q_h can give computational problems while evaluating the membership functions, the signal was scaled by a factor of $1e5$. Settings and fuzzy model structure results are shown in table 5.16. The fuzzy models are given in appendix A.

Figure 5.22 shows the simulation results of a free run, in which the fuzzy model is simulated in an autoregressive manner. For clarity, only some of the measurements are shown. The performance of the fuzzy model is good for the kappa number as well as for the yield. Some mismatch is observed during fast changes (for example around $t = 7500\ min$. Overall, the fuzzy model matches the identification data.

For all rules in the fuzzy MISO models for the Kappa number, the consequent parameters corresponding with $\kappa\#_{k-1}$ have approximately the value of 2. All consequent parameters corresponding with $\kappa\#_{k-2}$ have the value of -1 (equation E.5). This indicates second order behavior with approximately critical damping (Stephanopoulos, 1984).

5.7.4 Model validation

To validate the fuzzy model, a validation input-output data set was designed, similar to the identification data set (see appendix A). Simulation results are shown in figure 5.22 (c) and (d). Similar to the results for the identification data set, model mismatch occurs for fast changes. In addition, at around $t = 3700\ min$, large errors are observed for $\kappa\#$ and γ. Here, a combination of input signals is encountered that was not present in the identification data set, resulting in a large model error. This indicates poor extrapolation properties. Otherwise, performance is similar to the performance for the identification data set.



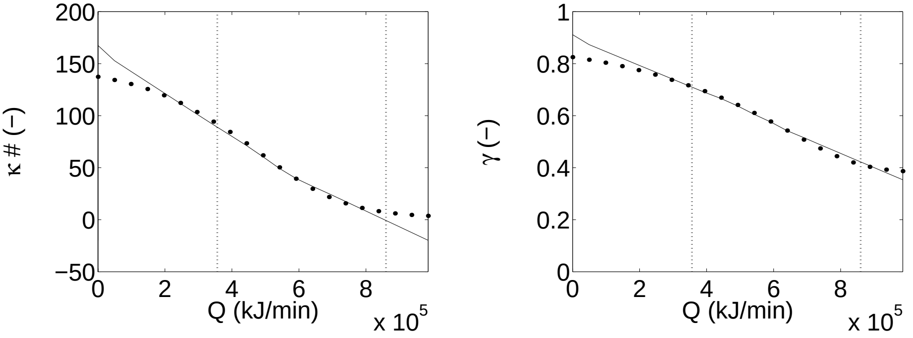
(a)

(b)

(c)

(d)

Figure 5.22: *Fuzzy model simulation results identification data set ($\kappa\#$ (a) and γ (b)) and validation data set ($\kappa\#$ (c) and γ (d)). Dots EPM, lines fuzzy model*



(a) (b)

Figure 5.23: *Static extrapolation results fuzzy model for $\kappa\#$ (a) and γ (b). Lines indicate identification domain*

5.8 Fuzzy model analysis

The fuzzy model will be analyzed with respect to the same aspects as the hybrid model: static and dynamic performance, complexity, interpretability and process independence. This makes comparison between the results possible. Of the quality aspects, the performance is the most important.

5.8.1 Static performance

Figure 5.23 shows the performance of the fuzzy model as a function of Q_h. The liquor flow ϕ_l was kept constant at the standard operating value (ID5). For both the Kappa number and the yield, the fuzzy model performs well within the identification domain. Outside this domain, the extrapolation behavior is approximately linear. This is mainly caused by the linear nature of the identification data with respect to Q_h in the identification domain. In addition, if the model is extrapolated far enough, only one of the fuzzy rules will become active. This means that an output variable is described by one linear ARX model. For $\kappa\#$, this results in negative values for high Q_h. This shows the black box nature of the model; no information is incorporated stating that the Kappa number cannot become negative.

Evaluating the static performance of the fuzzy model as a function of ϕ_l (figure 5.24) gives similar results. In the identification domain, the fuzzy model performs well. The fuzzy model shows linear extrapolation behavior for changes in the liquor flow rate, mainly caused by the linear nature of the data in the identification domain.

The results show that the static extrapolation performance of the fuzzy model for the yield is comparable to the performance of the hybrid model. For the Kappa number, the hybrid model performs slightly better, especially for high values of Q_h.

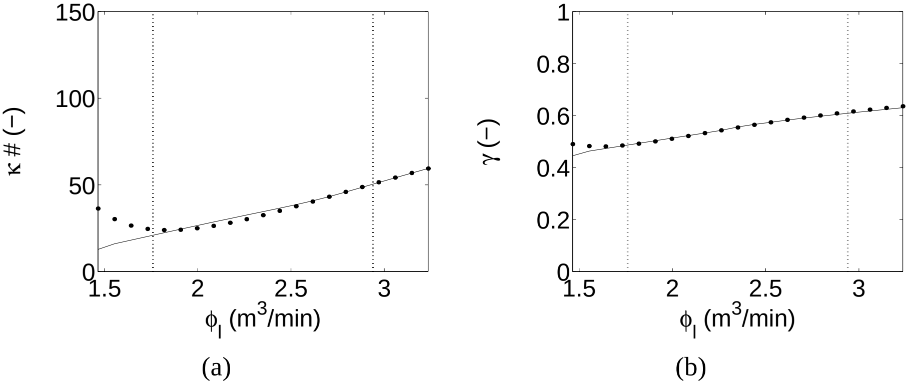


Figure 5.24: *Static extrapolation results fuzzy model for $\kappa\#$ (a) and γ (b). Lines indicate identification domain*

5.8.2 Dynamic performance

The behavior of the fuzzy model for a step change in Q_h is different from the behavior of the EPM (figure 5.25). The step was performed at $t = 0\ min$, while the initial conditions were taken from experiment ID5. The steady state offset can be observed, while the model gain is approximately the same as the gain of the EPM. The fuzzy model shows overshoot; the response is underdamped.

The results for a step change in the liquor flow ϕ_l are given in figure 5.26. The step was performed at $t = 0\ min$ and the initial conditions were taken from ID5. For these conditions, the steady state performance of the fuzzy MISO model for the yield is good (as was already observed in figure 5.24). Table 5.17 shows the amplitudes $\Delta\kappa\#$ and $\Delta\gamma$. Values for a positive and a negative step change, respectively, are given. The gains of the fuzzy model are smaller than the gains of the EPM. The gains for step changes in ϕ_l are approximately equal for positive an negative steps. This indicates that probably one of the rules is active under those

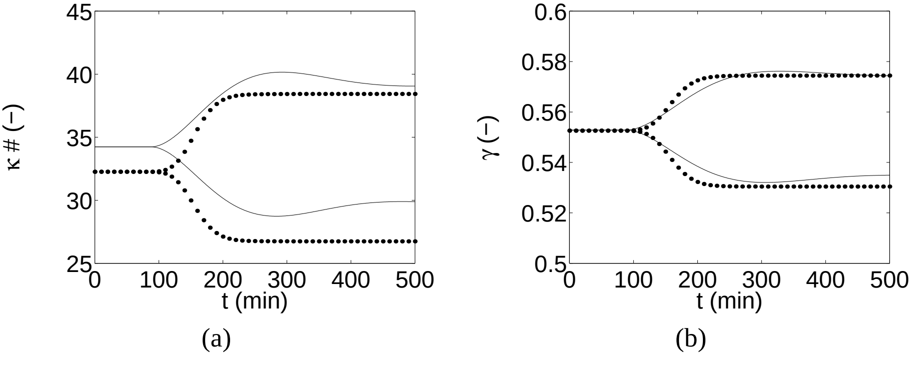


Figure 5.25: *Fuzzy model step response results for $\kappa\#$ (a) and γ (b) for positive and negative step change in Q_h. Dots EPM, lines fuzzy model*

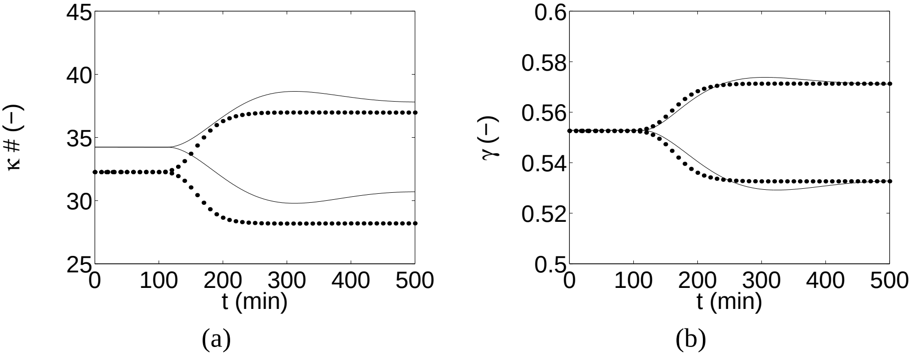


Figure 5.26: *Fuzzy model step response results for $\kappa\#$ (a) and γ (b) for positive and negative step change in ϕ_l. Dots EPM, lines fuzzy model*

conditions. As a result, the behavior is given by one of the local linear second order ARX models.

Frequency response analysis shows that there are considerable differences between the system orders O and time constants τ of EPM and the fuzzy model (table 5.17). The system orders and time constants were determined using the Bode diagrams. During model design, different model orders were investigated and the second order structure provided the best results. The simple model structure does not allow the orders that were observed from the EPM to be described accurately. The fuzzy model imposes second order behavior for step changes in both Q_h and ϕ_l. However, in the EPM, the order of the behavior for a steps change in Q_{lh} is different than for ϕ_f.

The fuzzy model describes the identification and validation data set sufficiently accurate. However, frequency response analysis reveals that there are differences between the dynamic characteristics of the EPM and the fuzzy model. These differences will be more apparent for high frequencies. To improve the dynamic characteristics of the fuzzy model, more information needs to be incorporated during model identification.

5.8.3 Complexity, interpretability and process independence

The structure of the fuzzy model is very simple. It consists of two MISO fuzzy ARX models, each with four inputs. The models represent a collection of linear ARX models, that

Model	*Input*	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$ (min)	$\tau_{[input],\gamma}$ (min)	$\theta_{[input]}$ (min)	$\Delta\kappa\#$ $(-)$	$\Delta\gamma$ $(-)$
EPM	Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
	ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020
Fuzzy	Q_h	2.1	2.0	52	58	87	-4.54, 4.98	-0.018, 0.022
model	ϕ_l	2.0	2.0	50	50	113	3.70, -3.68	0.019, -0.020

Table 5.17: *Frequency response analysis results fuzzy model*

	Model			*Application*		
Quality aspect	*EPM*	*Hybrid*	*Fuzzy*	*R&D*	*Design*	*Control*
Static performance		+	+	●	●	
Dynamic performance		++	0	●		●
Complexity	0	+	++	●	●	●
Interpretability	++	++	−	●		
Process independence	++	0	−−	●	●	

Table 5.18: *Comparison of model quality aspects, ranging from poor quality (−−) to good quality (++)*

describe the Kappa number and the yield as a function of the liquor flow and the heat supply of the heater.

Obviously, the simple structure does not provide much information about the physical processes that take place. Interpretation of the fuzzy model can only be done in terms of input-output behavior. With the EPM and the hybrid model, the model structure provided much information about the relation of physical phenomena. Since the fuzzy model maps the process inputs directly to the process outputs, such a structure is not present.

Since the information that was used to build the fuzzy model only consisted of process data, the model is very process dependent. As is common for these black box models, transportability is very limited. The performance analysis has already shown that there are differences in the dynamic characteristics of the fuzzy model and the ”process” it was built for. As a result, the fuzzy model is only suitable for application under the process conditions that are represented by the identification data. It is easy to improve performance by providing more information during model identification, for example by providing additional ”measurements”. This, however, yields a new fuzzy model.

5.9 Model evaluation

In order to evaluate the properties of the hybrid model of the pulp digester, the modeling results are summarized in table 5.18. This table shows how well each of the models performs with respect to one of the five quality aspects. In addition, the relevance of a quality aspect with respect to a general field of application is given (see also table 3.1).

The static performance of the hybrid and fuzzy model is comparable. Both describe the key variables well within the identification domain; the fuzzy model performs slightly better for the Kappa number $\kappa\#$ than the hybrid model. The fuzzy model is directly fit to $\kappa\#$ and has, because of the high number of parameters in comparison to the model equations, the flexibility to describe the key variables well for steady state conditions. In addition, it is an input output mapping, which means that error integration is not present, as is the case for the hybrid model.

Dynamically, the hybrid model performs better than the fuzzy model. Time constants and system orders match the EPM better than the fuzzy model. This is the result of the inclusion of first principles information for the dynamics. In the hybrid model, the key variables are based on the concentrations of several species, which in turn are described by physical phenomena. This way, the observed behavior is matched better. The MIMO reference system shows different orders for changes in the inputs (a change in heat supply Q_h results in a second order response, a change in the liquor flow rate results in a third order response), which is accounted for in the hybrid model. The simple structure of the fuzzy model is not able to describe this behavior, although the performance with respect to the identification and validation data is acceptable.

Obviously, the hybrid model is less complex than the EPM and more complex than the fuzzy model. The result is that the hybrid model is less interpretable than the EPM, but provides more information about the behavior than the fuzzy model does. Although the fuzzy model is transparent, it provides no physical information about the internal behavior. The hybrid model can be interpreted physically; the observed behavior can be explained with the model structure, in which the relation between the modeled physical phenomena is represented.

The level of interpretability of both the EPM and the hybrid model is comparable. In both models, the structure provides the most explanation of the behavior. The model equations provide information about the relative importance of the modeled effects. In both models, some of these equations are based on empirical data. The reaction rates in the EPM are based on empirical data, while the reaction rates in the hybrid model are derived from observed behavior. The level of interpretability is limited by the information that the specific equations provide.

Since part of the identification of the hybrid and fuzzy models was data driven, the models are more process dependent than the EPM. The fuzzy model is the most process dependent, since the observed behavior was the only information that was provided during identification. The hybrid model contains more information from other sources, such as first principles, which results in better transportability. The use of other sources in addition to process data also means that the data does not have to provide full information about the behavior. If the model is used for other installations, the fuzzy submodels may need to be adjusted, but the structure can remain unchanged.

The level of process dependence, although limited, makes the hybrid model less useful for process or equipment design studies on a detailed level. This is also true for research and development applications. A hybrid model can be very useful in studies that have the goal to gain more insight in the behavior of processes, but should not be used for applications that require models that are process independent. Regarding control applications, the hybrid model can be useful for specific control design or optimization problems. For use of the model in online control applications, the computational effort should be carefully evaluated. The level of complexity determines the amount of information that a model provides, but also the computational effort that is required to solve it, which may be limited in online applications.

The evaluation indicates that hybrid models are suited for applications that require models that are transparent, provide sufficient explanation of the process behavior and focus on a specific installation. Whether this is in the field of R&D, design or control, entirely depends on the application.

5.10 Concluding remarks

The hybrid model of the continuous pulp digester meets the requirements that were set. The behavior of unknown parts was derived from the observed behavior without making a priori assumptions about the internal structure of the submodels. In addition, since the fuzzy relations are static, there is no need for dynamic data during the identification of these unknown parts.

The hybrid model contains more information than the complete fuzzy model. As a result, it possesses better extrapolation properties. Although the error increases during extrapolation, the hybrid model performs better than the fuzzy model. This was observed during the evaluation of the static and dynamic performance.

The hybrid model was designed by incorporating the essential characteristics of the process, given the model requirements. The number of states has been reduced from 19 to 13, while the number of model sections has been reduced from 10 to 4. This yields a reduction in model complexity of roughly 30 %. Still, the behavior of the hybrid model approaches the behavior of the EPM closely. This is the result of the incorporation of the most important states that describe the physical background of the key variables. It also provides an explanation for the behavior of the model. If these states are not taken into account, it is not possible to combine similar performance with a similar level of interpretation.

It is difficult to determine exactly for which applications hybrid models can be beneficial and for which not. The enormous amount of literature about process modeling illustrates that each modeler presents a different approach for each application. This chapter has tried to provide more insight in general aspects of hybrid modeling and hybrid model properties. This has been done by presenting a detailed discussion on the design of a hybrid model and by comparing this model to two other models. It thus provides a basis for modelers to determine whether hybrid modeling is a suitable approach for their modeling problem.

The next chapter will deal with the development of a hybrid model for an experimental process unit setup. Although the hybrid model will be simpler than the model for the pulp digester, it will investigate the performance of the modeling approach in an actual process environment where limited amounts of data are available.

6

Application

The aim of this chapter is to build a hybrid model, based on experimental data. A suitable process, of which measurements were available, is a batch distillation column. In addition, dynamic modeling of this column has been applied to characterize column operation (Betlem *et al.*, 1998). It is therefore interesting to see if a hybrid model can be applied for such a purpose.

Two different hybrid models will be developed for the column. The models differ with respect to the dynamics that are incorporated. The goal of this exercise is to illustrate the development of the minimal dynamic framework to obtain a useful dynamic description. The best model will be used to determine the optimal production rate for a batch.

First, the application of the hybrid model will be discussed, from which the key variables and model requirements will be derived. Then, the two different hybrid models will be designed and compared. Finally, the best model will be applied in a simulation study to determine the optimal production rate.

6.1 Problem definition

An important parameter during batch operation is the duration of a batch or the *batch time*. Often, the objective is to maximize the production within a certain amount of time, for a given quality. A model that calculates the production as a function of the batch time provides insight in this problem. Several approaches have been presented for the column under study, ranging from rigorous to simplified reduced models. A simplified model has been used to evaluate different basic control policies for a single distillation cut (which means distillation of the feed stock without slop recycle) (Betlem *et al.*, 1998):

- Constant reflux control
- Constant quality control
- Dynamic optimization according to Pontryagin's maximum principle

The model that was used, does not take the start-up of the column into account. Start-up of a batch requires about 20 to 30 min, while a typical batch run for a single cut takes about 3 h. This means that start-up comprises a significant part of the batch run. A model that can describe the start-up of the column would therefore be beneficial for optimization studies.

Measurements of several batch runs with constant quality control were available. These involve the separation of ethanol and 1-propanol for different feed stocks and equal product qualities. It is assumed that for this separation, constant quality control is equivalent to dynamic optimal control (Betlem *et al.*, 1998). The data includes a description of the start-up of the column. Based on this data, *a hybrid model will be developed that describes the production for one quality and for the constant quality control strategy* during start-up and normal operation.

The measurements limit the level of detail of the hybrid model. No internal data is available, only input-output measurements. This means that the hybrid model will be similar to the reduced models in Betlem (1997) that describe overall dynamic behavior.

6.2 Hybrid model design

The design phase will follow the approach presented in chapter 4. First, the model structure will be determined, after which the data is pre-processed and the parameters and fuzzy models are identified. Subsequently, the fuzzy submodels will be integrated into the first principles framework.

6.2.1 Basic modeling

Process description and information analysis

The experimental setup involves a batch distillation column with 21 bubble cap trays with an internal diameter of 76 mm and a Murphee vapor tray efficiency of about 53 %. The only manipulated variable is the reflux fraction R^*. The heat supply E is constant during a batch run. The maximum exhaustion time is about 4 h and the dominant time constant is about 25 min (Betlem, 2000). A simplified diagram of the column setup is given in figure 6.1.

Process data for five batch runs, concerning the separation of ethanol from 1-propanol, is available. Since the data only provides information about the behavior under constant quality control, *the hybrid model can only be applied for simulations of batch runs using constant quality control as the control strategy*. The following measurements were made:

- Product quality x^0_{n+1}
- Reflux fraction R^*
- Vapor flow after condensation L_V

Table 6.1 shows the feed stock properties, in which $M_{col,0}$ is the amount of feed stock and $x_{col,0}$ is the composition of the feed stock, given by the mole fraction of ethanol. The desired product quality is given by $x^0_{n+1,sp}$. To be able to validate the model, three runs are designated as identification runs (ID), the other two are designated as validation runs (VAL).

Figure 6.2 illustrates the measurements of a batch run. A typical batch run can be described as follows. When the column is heated, the vapor flow slowly builds up in the column before reaching the condenser. This means that the during the first part of the start-up procedure, no vapor flow is measured. The measurements start when the vapor flow reaches the condenser.

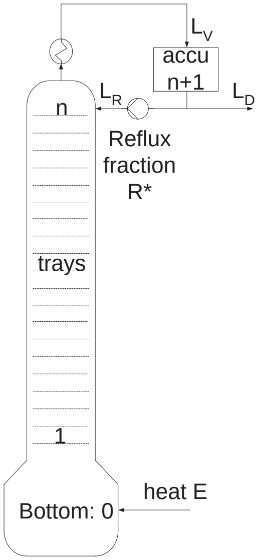


Figure 6.1: *Experimental batch distillation setup*

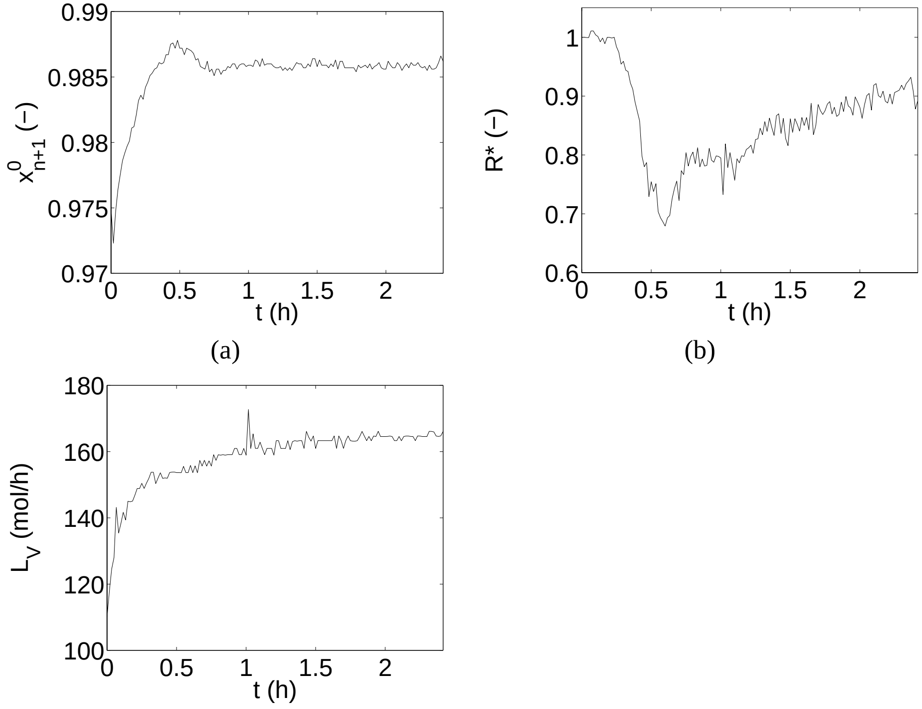
At that moment, the temperature in the column is still rising and settles after about 0.5 h. This results in the vapor flow as shown in figure 6.2 (c).

At the same time, the quality x^0_{n+1} is still rising. The quality is controlled with a PI controller in combination with a Partial Least Squares (PLS) estimator. This estimator has limited validity and the control loop only works properly if the deviation of the quality from the setpoint is limited. Therefore, the quality controller is switched on if the deviation is less than 0.002 from setpoint. Initially, the reflux fraction is set to 1. It takes about 15 min before the controller is switched on, as clearly can be seen in figure 6.2 (b).

When the control loop is active, the quality quickly reaches the desired value. There is some ”overshoot” present at around $t = 0.5\ h$ (figure 6.2 (a)). During warming up of the column, the equilibria on the trays are slightly more in favor of the volatile component than when the column is warmed up. This results in a slightly higher product quality. As the batch run advances, the feed stock becomes exhausted. To maintain the desired product quality, the reflux fraction is increased (figure 6.2 (b)).

Run no.	$M_{col,0}(mol)$	$x_{col,0}(-)$	$x^0_{n+1,sp}$
ID1	201	0.403	0.986
ID2	194	0.499	0.986
ID3	200	0.603	0.986
VAL1	193	0.410	0.986
VAL2	187	0.490	0.986

Table 6.1: *Initial conditions for batch distillation column*



(a)

(b)

(c)

Figure 6.2: *Available measurements, run ID1*

Process structure and basic equations

To describe the production dynamically, the overall dynamics of the column should be included in the model. The short term dynamics are less important; they describe the tray to tray behavior inside the column. In addition, the measurements do not provide sufficient information to include short term dynamics.

To describe the overall dynamics of the column, a simplified model comprised of an overall separation approximation, combined with first order dynamics for the exhaustion, has been derived (Betlem, 1997):

$$\frac{dM_{col}}{dt} = -L_V(1 - R^*) \tag{6.1}$$

$$M_{col}\frac{dx_{col}}{dt} = -L_V(1 - R^*)(x^0_{n+1} - x_{col}) \tag{6.2}$$

$$x^0_{n+1} = f(R^*, x_{col}) \tag{6.3}$$

in which $M_{col}(mol)$ is the mass of the column, $L_V\ (mol/h)$ is the vapor flow rate, $-L_V(1 - R^*) = L_D(mol/h)$ is the distillate flow rate, x_{col} is the average molar ethanol fraction in the column, x^0_{n+1} is the molar ethanol fraction of the product and R^* is the reflux fraction. M_{col} and x_{col} are introduced to avoid the need for tray-to-tray equations to describe the column.

The production M_p can be calculated as follows:

$$M_p = M_{col,0} - M_{col} \tag{6.4}$$

This model assumes constant vapor flow. During start-up, this flow changes due to the warming up of the column. This behavior can be approximated by a first order relation:

$$\frac{dL_V}{dt} = \frac{1}{\tau_V}(L_{V,ss} - L_V) \tag{6.5}$$

in which τ_V is the first order time constant that characterizes column heating and $L_{V,ss}$ is the steady state vapor flow for the given heat supply (this is an approximation, since the vapor flow changes during the batch run due to exhaustion). The actual behavior is more complex, as explained before. After the vapor flow reaches the condensor the behavior of the flow can be approximated by the first order equation. For this approximation, an estimate for the initial condition $L_{V,0}$ should be used such that the equation matches the observed behavior. The start-up behavior of the model before the vapor flow reaches the condensor is not incorporated and not represented by the measurements. As a result, the model describes only the final stage of the start-up procedure.

Equation 6.3 describes the separation as a function of the reflux fraction and the column quality. This relation is difficult to derive. Relations based on a so-called separation factor have been used, but they only approximate the separation for steady state conditions as they are valid for continuous operation (Shinskey, 1984).

Based on the process data, a fuzzy relation that describes the separation can be derived. Since it is derived form the data, it can match the observed behavior without imposing a functional relation a priori, as is the case with empirical separation factor relations. In addition, the influence of the start-up on the product quality can be taken into account by incorporating the vapor flow in the input space of the fuzzy model. This way, a hybrid model is obtained that includes first order exhaustion dynamics, first order startup dynamics and a static fuzzy relation that describes the product quality. The structure of the fuzzy relation is therefore given by:

$$x^0_{n+1} = f_{fuzzy}(R^*, x_{col}, L_V) \tag{6.6}$$

This model does not take top concentration dynamics, characterized by the dominant time constant of the column (Betlem, 2000), into account. It is interesting to investigate to what extend incorporating column dynamics influences the result. This illustrates the effect that omitting characteristic dynamic information has on model performance. To incorporate the dominant column dynamics, a first order state equation for the product quality is used:

$$\frac{dx^0_{n+1}}{dt} = \frac{1}{\tau_x}(x^* - x^0_{n+1}) \tag{6.7}$$

in which τ_x is the dominant time constant and x^* is given by:

$$x^* = f_{fuzzy}(R^*, x_{col}, L_V) \tag{6.8}$$

Equation 6.8 can be interpreted as the forcing function for the product quality.

Run	$\tau_V\ (h)$	$L_{V,0}\ (mol/h)$	$L_{V,ss}\ (mol/h)$
ID1	0.2	120	165
ID2	0.2	120	165
ID3	0.2	120	168
Average	0.2	120	166

Table 6.2: *Estimation results parameters start-up dynamics*

The result is two different hybrid model structures: a model that does not incorporate top concentration dynamics, denoted as the *simplified overall static model,* given by equations 6.1, 6.2, 6.5 and 6.6, and a model that incorporates top concentration dynamics, denoted as the *simplified overall dynamic model,* given by equations 6.1, 6.2, 6.5, 6.7 and 6.8. Appendix F shows the DFD's of both structures. Both models will be build and compared.

6.2.2 Simplified overall static model identification

Identification of the simplified overall static model involves the determination of τ_V, $L_{V,0}$, $L_{V,ss}$ and identification of the fuzzy relationship for the product quality (equation 6.6).

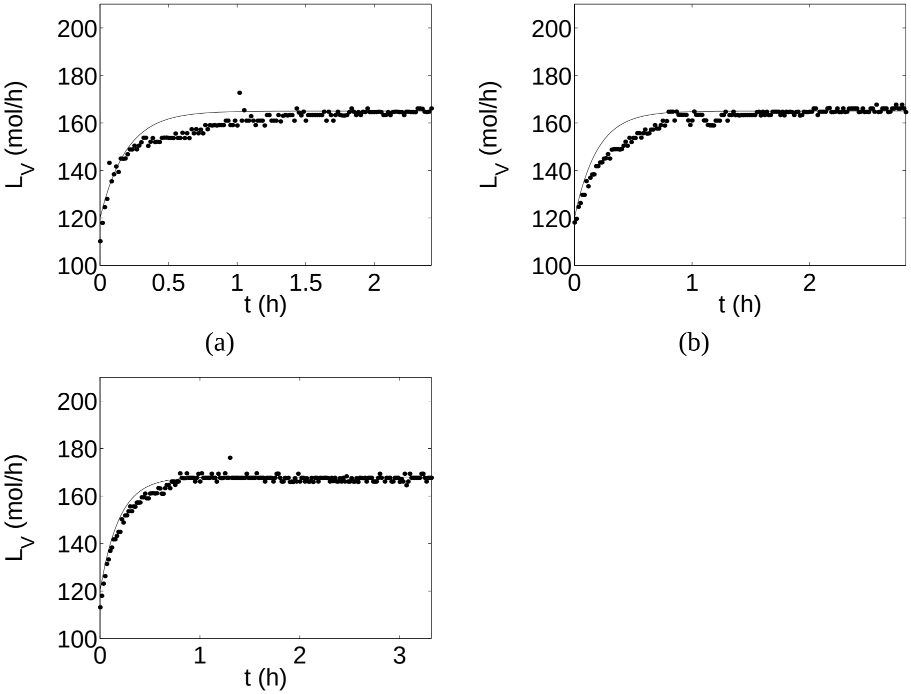
Start-up dynamics

The time constant τ_v and the steady state value $L_{V,ss}$ were determined by fitting equation 6.5 to the measurements of L_V. In addition, the initial condition $L_{V,0}$ was determined to provide a better approximation. Results are shown in figure 6.3. Table 6.2 shows the values of τ_V, $L_{V,0}$ and $L_{V,ss}$ for the three identification runs. The final values that will be used in the model are the averages of the values for the three runs.

Product quality

For the identification of the fuzzy submodel, R^*, x_{col}, L_V and x^0_{n+1} need to be available. The reflux fraction R^*, the vapor flow L_V and the product quality x^0_{n+1} are measured. The column quality x_{col} can be obtained by solving equations 6.1 and 6.2. This is done for each of the three identification runs. The combined runs result in an input-output data set of 512 features.

The fuzzy model was identified using GK-clustering in combination with structure optimization (see section 4.4.1). No data reduction was applied; it was found that data reduction did not improve clustering results. Clustering settings and results are shown in table 6.3. The table also gives the Root Mean Squared Error with respect to the identification data. The fuzzy model is given in appendix F.



(a)

(b)

(c)

Figure 6.3: *Model results for L_V, run ID1 (a), ID2 (b) and ID3 (c)*

Model validation

To analyze the performance of the model, the model will be solved for the initial conditions of the identification and validation runs. The simulation will use the measurements of the manipulated variable R^* as the model input. The results for the product quality x^0_{n+1} for the five batch runs are given in figure 6.4. The hybrid model describes the identification data well, although the overshoot during start-up (at about $t = 0.5\ h$) is not described accurately. The overshoot is a relatively small part of the behavior during a batch run, as a result of which it is not taken properly into account by the clustering algorithm.

Model	k_0	γ_{cm}	# rules	$RMSE$
x^0_{n+1}	10	0.7	3	6.5e-4

Table 6.3: *Clustering settings and fuzzy model results for x^0_{n+1}*

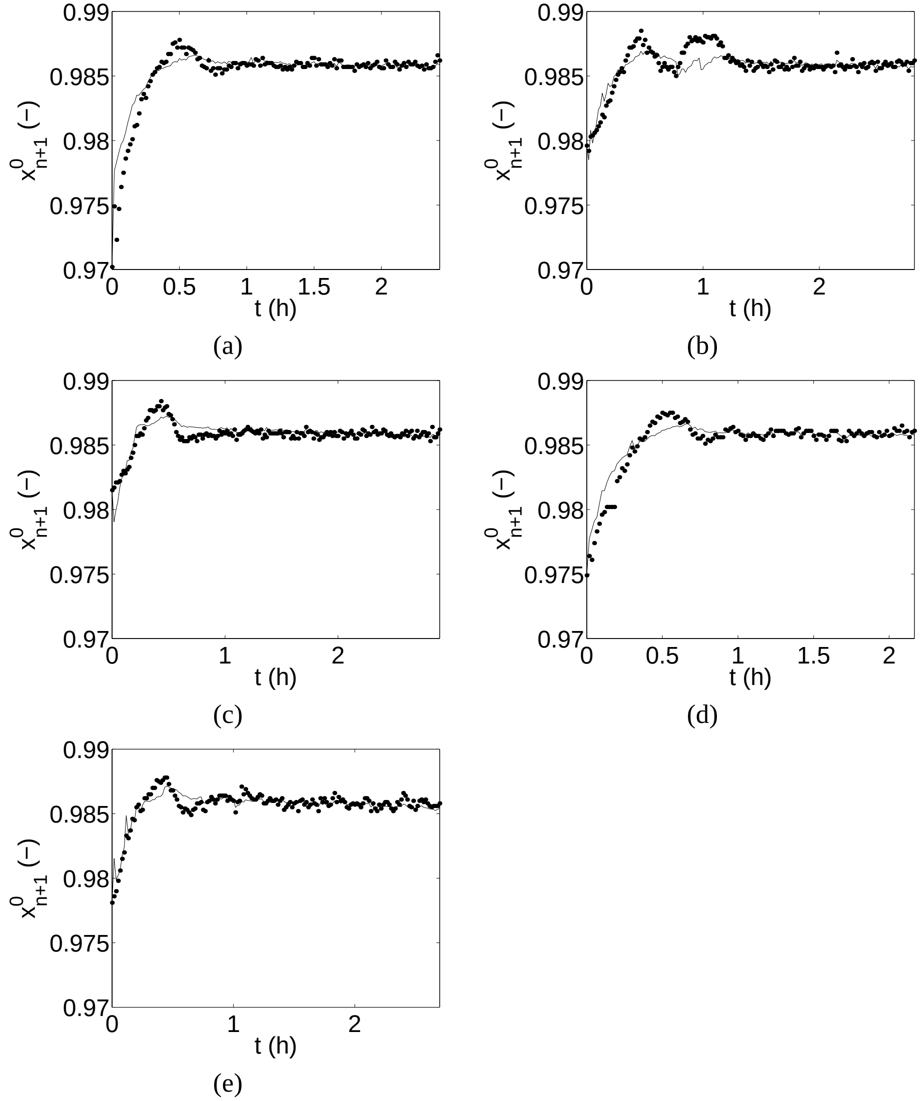


Figure 6.4: *Model validation results for x^0_{n+1} of the simplified overall static model. Dots indicate measurements, line indicates hybrid model results for ID1 (a), ID2 (b), ID3 (c), VAL1 (d) and VAL2 (e)*

Batch	K	τ_I
ID1	5	0.04
ID2	5	0.07
ID3	4	0.1

Table 6.4: *PI-estimator settings for* x^*

6.2.3 Simplified overall dynamic model identification

Identification of the simplified overall dynamic model involves the determination of τ_V, $L_{V,0}$, $L_{V,ss}$ and identification of the fuzzy relationship for the forcing function of the product quality (equation 6.8). The parameters τ_V, $L_{V,0}$, $L_{V,ss}$ are the same as for the static model.

Product quality forcing function

To identify the forcing function, R^*, x_{col}, L_V and x^* need to be available. As before, x_{col} can be obtained by solving equations 6.1 and 6.2. x^* will be estimated using a PI estimator (section 4.3.2). The measurements of x^0_{n+1} will serve as the reference and equation 6.7 is the model part of the estimator (figure 4.6). The estimator was tuned manually and good results were achieved for the settings given in table 6.4, as shown in figure 6.5.

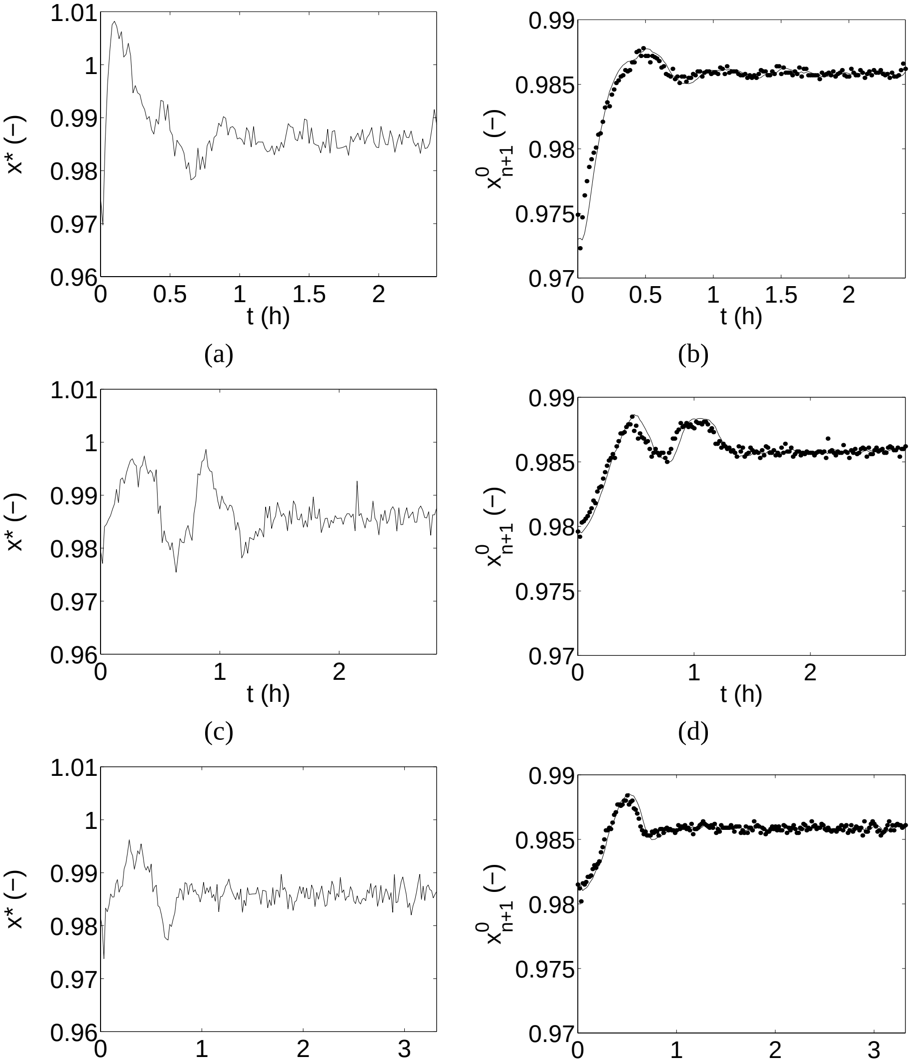
The fuzzy relationship for x^* was identified with fuzzy clustering. Settings are shown in table 6.5 and the fuzzy model can be found in appendix F. The error in the fuzzy model is larger than the error in the fuzzy model for x^0_{n+1}. This is caused by the amount of noise that is present in the estimates of x^*. It is expected that this does not influence the simulation results much, which should be investigated during model validation.

Model validation

The validation simulation results are acceptable (figure 6.6). The initial condition $x^0_{n+1,0}$ was estimated at 0.98, which is the average value for x^0_{n+1} at $t = 0\ h$ of the different batch runs. Similar to the static model, the fuzzy relation is not able to describe the overshoot in x^* well, as can be seen in appendix F. This error is integrated and the result is that x^0_{n+1} lags behind, as is the case for ID1 and VAL2. The overshoot in x^0_{n+1} is described slightly better than with the static model. In addition, the simulation results for x^0_{n+1} contain less noise. This is the result of the dynamic equation for the product quality, which has a filtering effect.

Model	k_0	γ_{cm}	# rules	$RMSE$
x^*	10	0.6	3	3.7e-3

Table 6.5: *Clustering settings and fuzzy model results for* x^*



(a) (b)

(c) (d)

(e) (f)

Figure 6.5: *Estimation results of x^* for batch run ID1 (a), ID2 (c) and ID3 (e), and corresponding estimates of x^0_{n+1} ((b), (d) and (f), respectively).*

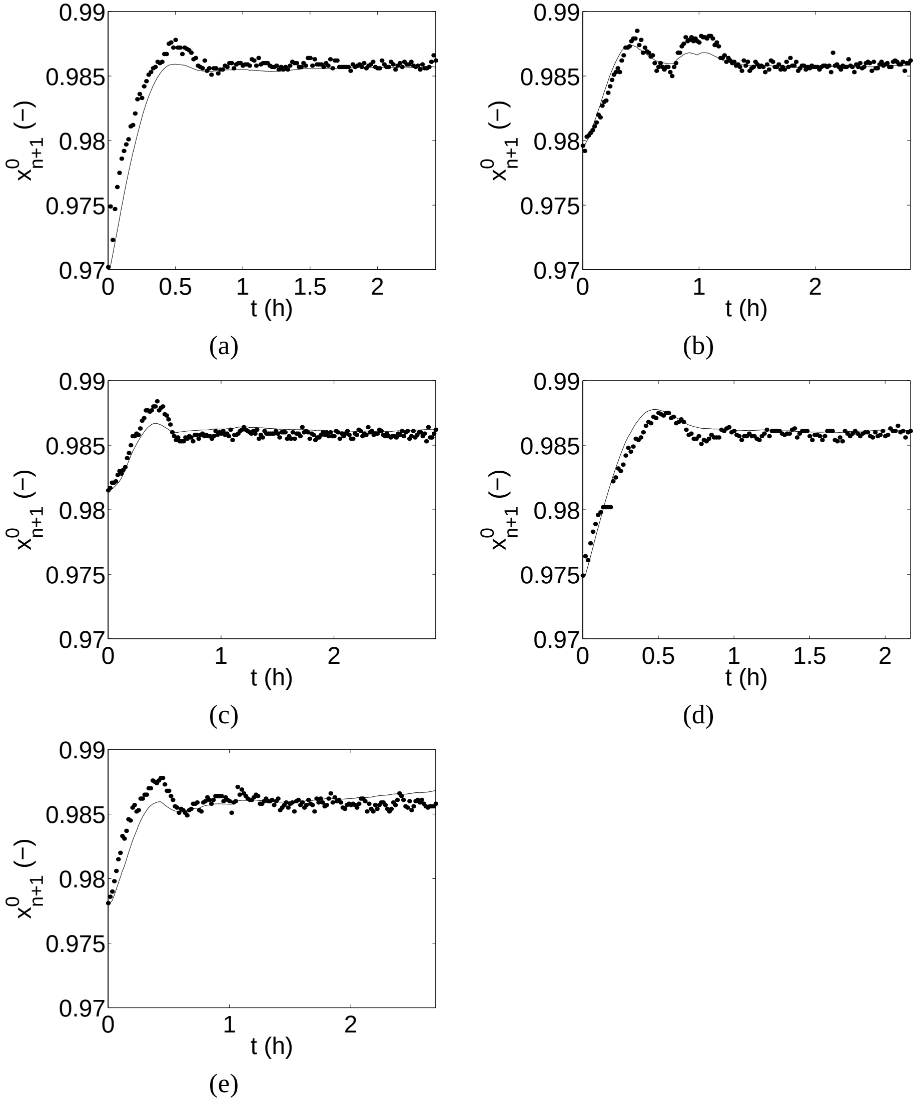


Figure 6.6: *Model validation results for x^0_{n+1} of the simplified overall dynamic model. Dots indicate measurements, line indicates hybrid model results for ID1 (a), ID2 (b), ID3 (c), VAL1 (d) and VAL2 (e)*

6.3 Hybrid model evaluation

Both the static and dynamic models describe the overall behavior of the distillation column and have a simple structure, which makes detailed evaluation of model interpretability and complexity of less interest. The focus of the evaluation phase will therefore be on the application of the models; the models will be used to determine the production curve which describes M_p in relation to the batch time t_b.

6.3.1 Production curve

Evaluating the production as a function of the batch time results in a *production curve* (see also Rippin (1983)). The curve represents the amount of product produced as a function of the duration of a batch. To construct such a curve, it is required that each batch produces product with the same quality. Since the hybrid models calculate the production under constant quality control, the product quality is constant over the course of a batch. A production curve can then simply be constructed by calculating the production during the course of a single batch run.

The production curves will be determined by simulating batch runs using the constant quality control strategy. To accomplish this, a simple PI controller is constructed that controls the quality x^0_{n+1} by manipulating R^*, similar to the experimental setup. This results in closed loop simulation in which the models are used as stand alone simulators. Controller switching is also included; the controller is switched on if the deviation from the setpoint is less than 0.002. Until then, $R^* = 1$.

Production curves were constructed for the conditions of the three identification batch runs. This enables comparison to the experimental results. Both the hybrid models were used. Controller settings for both models are given in table 6.6. Figure 6.7 shows controller performance for the conditions of batch ID1; in both cases, the quality x^0_{n+1} is controlled well. The dynamic model shows some overshoot. The presence of overshoot was found to be independent of controller tuning, which indicates that it is the result of general model behavior, similar to the experimental results.

The controller gain K is negative for the static model. This means that in order to increase product quality, the reflux fraction R^* has to be decreased. This is in contradiction to what is expected physically and is caused by the fuzzy model for x^0_{n+1}. The consequent parameters with respect to R^* (which are a measure for the partial derivatives $\partial x^0_{n+1} \partial R^*$) are negative, which locally results in an increase of x^0_{n+1} if R^* is decreased.

Model	K	τ_I
Static	-50	0.1
Dynamic	100	0.1

Table 6.6: *PI-controller settings for* x^0_{n+1}

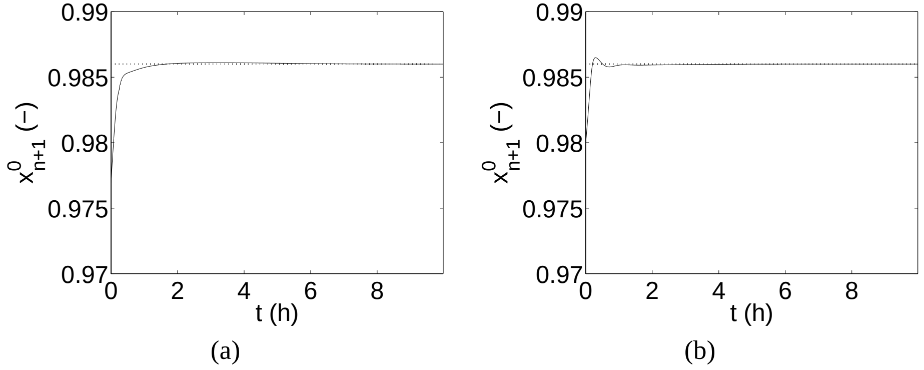


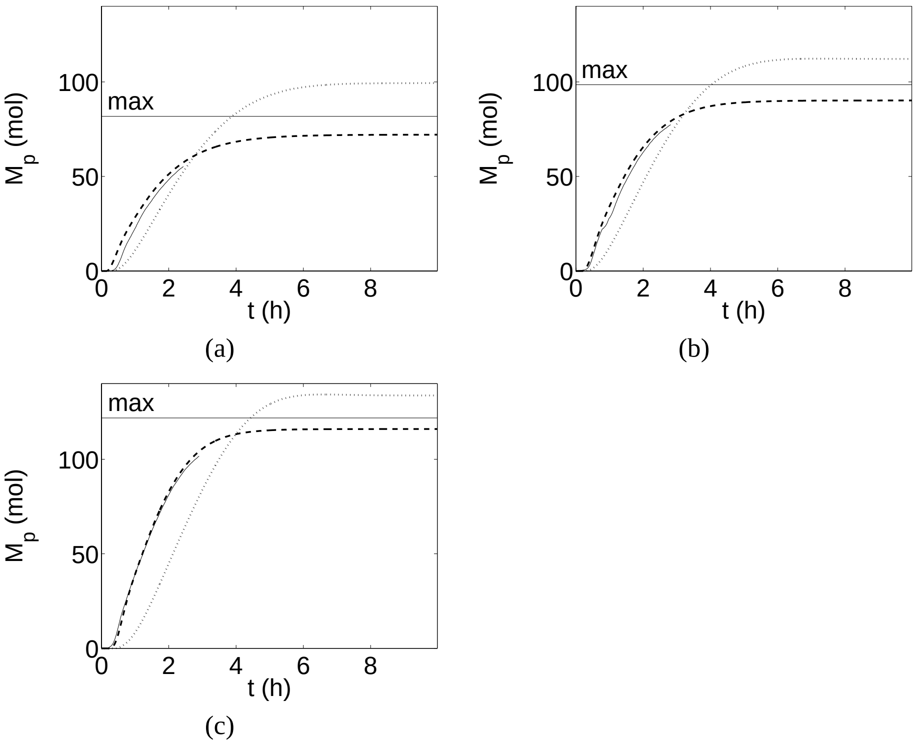
Figure 6.7: *Quality control results static (a) and dynamic (b) model, settings run ID1.*

The negative parameters are the result of the fit to the experimental data. The fuzzy model maps the reflux fraction, the column composition and the vapor flow to the product quality. With the exception of start-up, the product quality x^0_{n+1} is constant throughout the batch run. This means that the fuzzy model essentially describes a hyperplane with slope 0. Due to the noise that is present, the least squares fit results in a slightly negative slope with respect to R^*.

Although it was found that the controller of the static model achieves constant quality, the reflux fraction R^* is not manipulated in accordance with the experimental results. This can be observed from the production curves, which are shown figure 6.8. The duration of the batches during simulation was 10 h, which is longer than the actual experiments. This was done in order to simulate maximum exhaustion. The horizontal line indicates the theoretically maximum production (when all of the volatile component is recovered). On average, the reflux fraction of the static model is lower than the reflux fraction of the dynamic model, which results in production that is higher than the theoretical maximum.

If R^* changes, it takes some time before the overall effect on x^0_{n+1} is observed. This is characterized by the dominant time constant τ_x. The static fuzzy model does not incorporate this information. The result is that the product quality x^0_{n+1} is not correlated correctly with the inputs R^*, x_{col} and L_V. This can clearly be seen during start-up. Consider figure 6.2. In figure 6.2 (b), the reflux fraction R^* is approximately 1 during the first 0.25 h of the batch. At this moment, the product quality x^0_{n+1} is relatively low, as shown in figure 6.2 (a). The static fuzzy model correlates high reflux fractions with low product quality, which is not in accordance with what would be expected. In the dynamic model, the extra information that dominant the time constant provides is incorporated. Here, x^* is relatively high during the first 0.25 h, as shown in figure 6.5. High reflux fractions are correlated with high x^*. The result is that the dynamic model approximates the experimental situation much more accurately. In addition, the extrapolation behavior ($t_b > 3\ h$) is in accordance with the expectation that the maximum exhaustion is below its theoretically achievable value.

The dynamic model includes the top concentration dynamics using a first order approximation characterized by the dominant time constant. An alternative would be to derive the dynamic behavior of the product quality from experimental data and incorporate this behavior



(a)

(b)

(c)

Figure 6.8: *Production curves for the conditions of batch run ID1 (a), ID2 (b) and ID3 (c). Solid line experimental results, dotted line static model, dashed line dynamic model*

in the form of a dynamic fuzzy submodel. The structure of such a hybrid model is similar to the structure of the static model and omits the need for a first order approximation. However, as shown in chapter 5, the experimental data needs to contain sufficient information about the dynamics in order to obtain a good model. In this case, experimental data is limited and noisy. The incorporation of the dominant time constant can be viewed upon as the use of an additional source of information, that does provide sufficient information about the dynamics without the need for more experimental data.

The results show that the performance of the static model is acceptable if the desired behavior of R^* is imposed on the model. During model validation, the desired behavior of the reflux fraction R^* was given by measurements. During the construction of the production curves, the behavior of R^* was determined by a PI controller. Since the static model does not provide sufficient information about the dynamic behavior of the product quality, the values of R^* given by the PI-controller do not approximate the measurements, which results in unacceptable performance.

The simulation also shows the influence of the start-up of the column. Production starts when the quality controller is switched on and the reflux fraction decreases. The start-up behavior as described the dynamic model is similar to the experimental results.

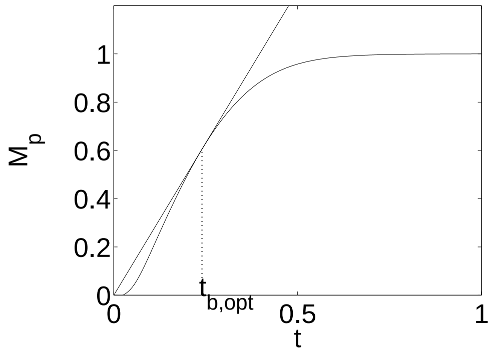


Figure 6.9: *Geometric determination of the optimal batch time*

6.3.2 Optimal batch time

To illustrate the application of the dynamic hybrid model, the optimal batch times for the five batches will be determined and compared with the experimental results. Consider the optimal batch time $t_{b,opt}$ as defined by the batch time for which the average production rate, given by

$$R_p = \frac{M_p}{t_b}, \tag{6.9}$$

is maximal. The optimal batch time can be obtained by simple geometric means, as illustrated in figure 6.9.

The hybrid model is used as a stand-alone simulator. This means that the model parameters are not changed for the different batch runs. Only the initial conditions $M_{col,0}$ and $x_{col,0}$ of the model are set to match the experiments. The initial conditions $L_{V,0}$ and $x^0_{n+1,0}$ were equal for all simulations and set to 120 mol and 0.98, respectively. Table 6.7 shows the experimental and simulation results.

The simulated maximum average production rates $R_{p,max}$ are in accordance with the measured results. The optimal batch times however, as predicted by the model, are shorter than the actual optimal batch times. The model predicts consistently increasing batch time for increasing $x_{col,0}$. This is not observed from the experimental results. The error is mainly caused by the differences in controller switching time. Figure 6.8 shows that the general trend of the simulated production curves match the experiments, but that production starts at

	Measurements		*Hybrid model*	
Run	$t_{b,opt}$ (h)	$R_{p,max}$ (mol/h)	$t_{b,opt}$ (h)	$R_{p,max}$ (mol/h)
ID1	1.33	25.0	1.10	28.1
ID2	1.65	32.0	1.30	34.8
ID3	1.57	42.5	1.55	42.6
VAL1	1.66	22.9	1.10	28.4
VAL2	1.42	32.8	1.25	34.0

Table 6.7: *Optimal batch times and maximum production rates*

different times.

In the simulations, controller switching is implemented according to the procedure that was used during the experiments. However, during the experiments, the procedure was performed manually and the actual switching times were not recorded. Variations in the execution of the procedure would result in different optimal batch times.

In addition, the initial conditions $x^0_{n+1,0}$ and $L_{V,0}$ and the parameters τ_V and $L_{V,ss}$ have a large influence on the behavior of x^0_{n+1} during start-up. Inaccuracies in these parameters influence the switching time of the controller and thus the optimal batch time. The performance of the hybrid model could be improved by investigating these parameters and their estimation. This requires additional experimentation.

6.4 Concluding remarks

The model presented in this chapter shows that the hybrid modeling approach has been successfully applied in a situation where limited experimental data is available. It was possible to design a relatively simple model, in which the dynamic characteristics are obtained from previous research and the static characteristics are based on experimental data. It was also possible to use the model to investigate batch operating conditions for distillation of a single cut. Results, however, can be improved if some of the model parameters can be determined more accurately.

The advantage of the simplified overall dynamic hybrid model is that it can describe the production of the batch column including part of the start-up of the column, without the need for a detailed description of internal column dynamics. The model structure is based on the simplified overall model proposed in Betlem (1997), but the description of the separation is derived from observed behavior without making a priori assumptions.

The hybrid model does not provide more information than is provided during model design. The experimental data describes the behavior of the column under constant quality control for one single product quality, which means that the hybrid model is only valid under these conditions. In addition, omitting relevant dynamic information in the model structure does not give problems if this information is imposed on the model; model validation, where the desired behavior of R^* was imposed by the measurements yielded acceptable performance of the static model, while the results during the construction of the production curves were unacceptable.

In general, in order to obtain good results, a process model needs to describe the most important dynamic characteristics of a process. The simulation results of the static and dynamic models illustrate that this is not different for hybrid models; since the fuzzy submodels are static, hybrid fuzzy-first principles models as presented in this work have the same dynamic characteristics as first principles models.

7

Conclusions and outlook

In the previous chapters, the analysis and design of hybrid fuzzy-first principles models have been illustrated. This chapter discusses the main conclusions with respect to the hybrid modeling procedure and hybrid model properties. In addition, some suggestions for future research are given.

7.1 Hybrid modeling

Hybrid fuzzy-first principles models, as presented in this work, consist of a framework of dynamic mass and energy balances, formulated in state space form, supplemented with algebraic and fuzzy equations. The fuzzy equations describe physical phenomena that are poorly understood or difficult to model using first principles. This is a serial modeling approach, which results in a model structure in which internal variables can be interpreted physically. This way, the model structure guarantees a certain level of transparency.

The main sources of information during hybrid model development are first principles, process data and human expertise. First principles are used to derive the model structure. Although fuzzy logic is extremely suitable to quantify expert knowledge, it is often difficult to integrate this knowledge in a predefined hybrid model structure. In the hybrid modeling context, expert knowledge is more suitable to provide structural information. For example, it can be used to identify the essential characteristics of a process. Process data can then be used to derive quantitative information.

The type of fuzzy model which is used is the TSK type. This type can be interpreted as a collection of local linear models. The operating range of these local models is determined by (a combination of) fuzzy sets. This type of fuzzy model is extremely suitable to describe highly nonlinear relations. In addition, many good identification algorithms exist for deriving TSK models from process data. Although linguistic fuzzy models can be interpreted better by humans, they require a more complex structure to describe the relations than TSK models. The advantage of using TSK type fuzzy models is reduced complexity in combination with structural transparency.

To build a hybrid model, the modeling problem is partitioned into several smaller and simpler problems, which can be solved by performing a series of sequential steps. This is an advantage over proposed global approaches, in which the modeling problem is approached as a whole. A structured modeling approach is presented, which consists of three phases. The steps of the modeling approach are performed sequentially and independently of each other. This provides modelers with the flexibility to use different or customized modeling tools. In the first phase, the model objective and quality requirements are formulated. The second phase involves the design of the model, which can be summarized as follows:

- Determine the hybrid model structure by characterizing process behavior and distinguish subprocess modeling problems, for which fuzzy submodels should be developed
- Based on this structure, acquire relevant process data
- Estimate any unmeasurable behavior of the subprocesses

- Identify the fuzzy submodels and determine other model parameters
- Integrate the fuzzy submodels in the hybrid model structure by optimizing their parameters with respect to the hybrid model output

In the final phase, the hybrid model is evaluated with respect to the model requirements. The model quality is determined by evaluating model performance, complexity, interpretability and process independence.

For the identification of the fuzzy submodels, three different classes of identification algorithms were compared: fuzzy clustering, neurofuzzy methods and genetic algorithms. Fuzzy clustering provides a better way to identify the fuzzy submodels than neurofuzzy methods or genetic algorithms do. This unsupervised learning approach requires less a priori model structure information and therefore is less sensitive to initialization. This makes the approach useful in situations where little a priori information about the modeled phenomenon is available.

During submodel integration, the parameters of the fuzzy models are optimized. The fuzzy models that result from the identification step are used as the starting point. Since the number of parameters of fuzzy models can be quite large, the optimization is focused on the parameters that have the most influence on the results. In general, these are the consequent parameters. In addition, the number of parameters to be optimized can be reduced by optimizing rule or model weights. Sensitivity analysis can help to determine the most relevant fuzzy models or fuzzy model parameters and gains insight in the optimization results, as well as the model behavior.

Hybrid models for three processes were developed. A simulated fed-batch bioreactor was used to investigate the use of different modeling tools. The hybrid model performs well. In addition, detailed analysis of the performance of the modeling tools is possible because of the simple model structure. To illustrate the modeling approach on a more complex system and to investigate hybrid model properties, a hybrid model for a simulated continuous pulp digester was developed. The model describes the process using a structure that represents characteristic physical phenomena and contains four fuzzy submodels. The simplifications result in some differences between the behavior of the hybrid model and the reference model. However, the hybrid model meets the quality requirements that were set. Finally, the modeling approach was used to develop a hybrid model for a experimental batch distillation column. The hybrid model describes the overall dynamic behavior of the column with four state equations and one fuzzy submodel. The behavior of the top quality is characterized by a dominant first order time constant and a fuzzy model that describes the quality forcing function. The hybrid model describes the experimental data well, including (part of) the start-up behavior.

7.2 Hybrid model properties

The use of fuzzy logic has an advantage over other black box techniques. The way fuzzy models are used in this work combines unsupervised learning with the ability to describe complex or poorly understood phenomena more transparently than other black box techniques, such as artificial neural networks. Fuzzy clustering yields fuzzy models with independent rules and which are constructed using little a priori information. This way, the fuzzy models are derived from the observed behavior without imposing a predefined model structure. A posteriori, it is possible to analyze and interpret the model structure in terms of characteristic operating regimes, complexity and the level of nonlinearity.

With respect to dynamic performance, complexity and interpretability, hybrid models are similar to first principles models. This is inherent to the hybrid model structure. Since the dynamic structure is based on first principles, the dynamic performance of hybrid and first principles models is comparable. In addition, the level of interpretability is comparable. In both types of models, the model structure provides information about the relations between the modeled physical phenomena. Physical interpretation on a more detailed level is only possible if the specific equations are based on first principles. If not, as is the case with empirical or fuzzy relations, interpretation is limited to a characterization of the behavior of the equations.

Static performance and process independence of hybrid models are comparable with fuzzy models. Depending on the number of fuzzy equations in a hybrid model, static performance is similar to the static performance of a fuzzy model. In both cases, the static properties are derived from observed behavior. In the case of the hybrid model, the model structure may influence static performance. Since a complete fuzzy model has a high number of parameters in relation to its model structure, it has more flexibility to describe the desired behavior than a hybrid model has. However, for the test cases discussed in this work, the hybrid models could achieve acceptable static performance.

Since fuzzy models are data driven, they are only valid in the operating regime that is represented by the identification data. The first principles part can only partly compensate for limited validity of the fuzzy models outside this operating regime. It also imposes a level of process dependence for hybrid models; process dependence is increased if the fuzzy part of the hybrid model is increased.

Similar to first principles models, hybrid fuzzy-first principles models can match desired dynamic behavior if the model structure represents the essential dynamic characteristics of the process. This was illustrated with the pulp digester case, where a simplified hybrid model matches the performance of a detailed first principles model. This was achieved by determining the essential characteristics of the process from a physical point of view, for given operating conditions. It was also illustrated with the batch distillation case, where hybrid model performance was improved by incorporating essential dynamic information. The use of fuzzy logic in hybrid models introduces flexibility, which enables the description of complex behavior with a predefined, interpretable overall model structure.

7.3 Suggestions for future research

In chapter 1, four different trends that in all likelihood will characterize the future development of intelligent systems were presented (Stephanopoulos and Han, 1996). As was shown in this thesis, hybrid modeling can be seen as part of two of these trends: the integration of multiple knowledge representations and the integration of processing methodologies. It seems therefore appropriate to discuss suggestions for future research in relation to these trends.

A process model is not often a goal in itself. The true merit of hybrid fuzzy-first principles modeling becomes apparent during extensive application of the proposed approach. The influence of the choices that were made for the general hybrid model structure and the modeling approach will then be clear. In this light, the use and relation of the different sources of information is of particular interest.

A frequently made statement is that modeling is more of a craft than it is a science. This illustrates the difficulties associated with turning data into knowledge. A modeler must have the ability to distinguish relevant from irrelevant information. In addition, he or she should be able to combine information that is represented in different ways. Structured modeling approaches, such as presented in this work, provide modelers with a framework that can be helpful in accomplishing this. However, the use and application of information, as well as the combination of information that is represented in different ways, is often based on experience.

Hybrid fuzzy-first principles models, to some extend, provide a way to combine information that is represented in the form of first principles, expert knowledge and process data. Expert knowledge can be used to determine essential characteristics, first principles knowledge is used to determine the relations between these characteristics and process data is used as a source of quantitative information. It would be interesting to investigate the relation and combination of these sources of information in more detail. This can provide a more methodological approach for the design of the hybrid model structure. A starting point would be to use insights in this matter from research fields such as data mining, knowledge engineering or psychology and adapt them for process modeling.

The use of hybrid models in the optimization of process operation is an interesting area of application. For these optimization problems, models need to be general (i.e. valid for a wide variety of operating conditions), structured and easy to calibrate (Cagan *et al.*, 1996). Hybrid models have a physically interpretable structure, are able to describe complex nonlinear behavior and are easier to build than detailed first principles models. An illustrative example of an application in optimization is presented in Simutis *et al.* (1997). Here, the influence of process models on increasing the benefit/cost ratio of model-supported optimization and control is addressed. The influence of augmenting a first principles model with neural networks for the optimization of the operation of a biochemical reactor is investigated. The structure of the resulting hybrid model is similar to the structure of hybrid models in this work.

A

Fuzzy logic

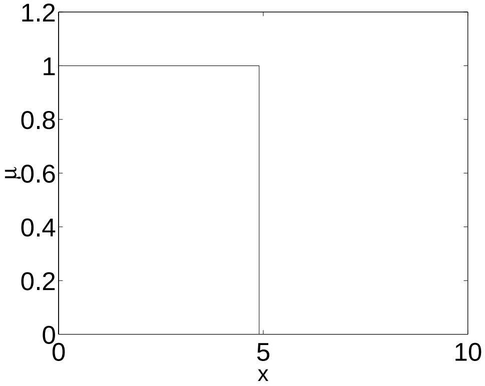


Figure A.1: *Membership function for a crisp set*

This appendix describes some basic concepts regarding fuzzy set theory and presents different structures of fuzzy models. Most of this appendix is based on Yager and Filev (1994), which gives a more detailed discussion about fuzzy logic.

A.1 Basic concepts

A.1.1 Fuzzy sets

In ordinary set theory, an ordinary or crisp subset is defined with respect to some universe of discourse X, which in itself is a crisp set. Such a subset is described with a characteristic function, indicating the membership of the subset. Let X be a universe of discourse and let S be a subset of X. The characteristic function can be defined as

$$\mu_S : X \rightarrow \{0, 1\} \tag{A.1}$$

such that for any element x of the universe, $\mu_S(x) = 1$ if x is a member of S and $\mu_S(x) = 0$ if x is not a member of S. Figure A.1 illustrates this.

A *fuzzy subset* is described by a characteristic function that differs from equation A.1 in that it is extended from the unit binary pair $\{0, 1\}$ to the unit interval $[0, 1]$:

$$\mu_S : X \rightarrow [0, 1] \tag{A.2}$$

Equation A.2 is called a *membership function*. The membership function describes to which degree an element is a member of the fuzzy subset. Let S be a fuzzy subset of the universe of discourse X. The membership of S for an element x is calculated with the membership function for S and is denoted the *membership grade* μ_S:

$$\mu_S = \mathrm{mf}_S(x) \tag{A.3}$$

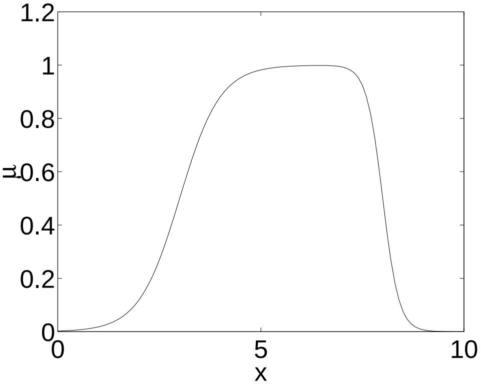


Figure A.2: *Double sigmoid membership function*

Commonly used shapes for membership functions are triangular, trapezoidal, gaussian or bell shapes. This work uses continuously differentiable double sigmoid membership functions, given by:

$$\mu = \frac{1}{1+\exp\left(-a_1(x-c_1)\right)}\frac{1}{1+\exp\left(-a_2(x-c_2)\right)} \tag{A.4}$$

in which a_1, c_1, a_2 and c_2 are the membership function parameters. Figure A.2 shows an example.

Fuzzy subsets are particularly useful for representing concepts with imprecise boundaries. Assume a domain of people X. Using a fuzzy subset to represent $tall$ people discards the restriction of having to categorize every person as a member of the set $tall$ or not, as is the case for a crisp subset. Instead, it is possible to make more subtle distinctions by using membership grades, such as ”not tall at all” or ”definitely tall” and thereby to more naturally represent the imprecise concept of being tall.

A.1.2 Operations on fuzzy sets

Fuzzy set operations are an extension of operations on crisp sets. Research in this area has been very extensive and much literature is available. Some important operations are discussed here.

Consider two fuzzy subsets A and B of X. The *union* of the fuzzy subsets A and B is a fuzzy subset C, denoted $C = A \cup B$, such that for each $x \in X$:

$$C(x) = \max\left(A(x), B(x)\right) = A(x) \vee B(x) \tag{A.5}$$

in which $\vee$ denotes the $\max$ operator. In essence, this is an extension of the logical OR operator. The counterpart of the union is the *intersection*. The intersection D of two subsets A and B, denoted $D = A \cap B$, is given by:

$$D(x) = \min\left(A(x), B(x)\right) = A(x) \wedge B(x) \tag{A.6}$$

in which $\wedge$ denotes the $\min$ operator. This is an extension of the logical AND operator. Another commonly used intersection operator is the product. For crisp sets, the result is the same as with the $\min$ operator. For fuzzy sets, results differ. In this work, the $\min$ operator is used.

A.1.3 Linguistic values and degree of truth

A fuzzy subset can be used to describe the meaning of a concept. In the above, a fuzzy subset of the length of a person was used to define $tall$. One use of this presentational ability of a fuzzy subset is to help define *linguistic values*. Assume V is a variable taking its value in the set X. The usual way to represent information about this variable is by statements of the form $V = x$, where x is some value in the set X. This idea can be extended to allow the value of V to assume some linguistic value. Thus if V describes a persons length, statements of the following form can be allowed:

$$V = tall \tag{A.7}$$

Using the membership function for $tall$, the *degree of truth* of the statement in equation A.7 can be calculated. When the premise

$$V = x \tag{A.8}$$

is given, and the statement given in equation A.7 is made, the degree of truth of this statement is calculated as the membership grade of x in $tall$.

The use of linguistic values essentially means the association of a fuzzy subset with the value of a variable and is very important in inferencing fuzzy models.

A.1.4 Defuzzification

In many applications of fuzzy techniques it may be necessary to transform a fuzzy subset, or a collection of subsets, to a crisp value. This is known as *defuzzification*. In general, two methods are applied: the Mean of Maxima (MOM) method and the Center Of Area (COA) method.

With the Mean of Maxima method, the crisp output is calculated as a mean of all values in the universe of discourse that have maximal membership grades:

$$y = \frac{1}{m} \sum_{y_i \in G} y_i \tag{A.9}$$

in which G is the set of elements which attain the maximum value of the membership grades μ_{F,y_i} of the fuzzy subset F and m is the cardinality of G.

The MOM method does not take the shape of the fuzzy subset into account. The Center Of Area method does not have this problem. COA defuzzification is calculated as follows:

$$y = \frac{\int y\mu_{F,y}dy}{\int \mu_{F,y}dy} \tag{A.10}$$

A.2 Fuzzy models

Fuzzy models, a representation of the essential features of a system by the apparatus of fuzzy set theory, basically fall into two categories, which differ fundamentally in their ability to represent different types of information. The first includes linguistic models, of which the Mamdani-type are the most common. These models are based on collections of IF-THEN rules with vague predicates and fuzzy reasoning.

The second category of fuzzy models is based on the Takagi-Sugeno-Kang (TSK) method of reasoning (Takagi and Sugeno, 1985)[1]. These models are formed by logical IF-THEN rules that have a fuzzy antecedent part and a functional consequent part. TSK models integrate the ability of linguistic models for qualitative knowledge representation with an effective potential for expressing quantitative knowledge as well.

A.2.1 Linguistic models

Generally speaking, linguistic models describe systems using a set of IF-THEN rules, in which the rules take the place of the usual set of equations used to characterize a system. The linguistic model is a knowledge-based system; it contains rules which incorporate inherently fuzzy or effectively fuzzifiable real-world knowledge. The decision making ability of the linguistic model depends on the existence of a rule-base and fuzzy reasoning mechanism.

Consider a double-input single-output system. Knowledge about this system can be encoded by a set of IF-THEN rules with two antecedent variables and one consequent variable:

$$\text{IF } u_1 = \text{mf}_{u_1,i} \text{ AND } u_2 = \text{mf}_{u_2,j} \text{ THEN } y = \text{mf}_{y,k} \tag{A.11}$$

in which the fuzzy sets are represented by the membership functions $mf_{*,*}$. The IF-part of a rule is called the *premise* or *antecedent* part of the rule, the THEN part is called the *consequent part.*

The number of rules and the structure depends on the knowledge that the model represents. The maximum number of rules of the fuzzy model is determined by the maximum number of combinations of the antecedent membership functions. Consider the rule in equation A.11. If 3 fuzzy sets for both u_1 and u_2 are available, the maximum number of rules for the fuzzy

[1] In the literature, fuzzy models based on the TSK method of reasoning are called TSK models, Sugeno models or TS models. They all denote the same type of model.

model is 9. If a fuzzy set occurs in more than one rule, the rules are *dependent*; changing the fuzzy set will affect the behavior of several rules. If a fuzzy set occurs only in one rule, the rules are *independent*.

A.2.2 Fuzzy inference

The following algorithm is used for calculation of the output that is inferred by an linguistic model, via the Mamdani method. Assume a linguistic model for a 2-by-1 system as given in equation A.11 with 2 independent rules:

$$\begin{array}{l}\text{IF } u_1 = \text{mf}_{u_1,1} \text{ AND } u_2 = \text{mf}_{u_2,1} \text{ THEN } y = \text{mf}_{y,1} \\ \text{IF } u_1 = \text{mf}_{u_1,2} \text{ AND } u_2 = \text{mf}_{u_2,2} \text{ THEN } y = \text{mf}_{y,2}\end{array} \tag{A.12}$$

Also assume a crisp input $u_1 = x_1^*$ and $u_2 = x_2^*$. The first step is to determine the influence the different rules of the model have on the output of the model. The influence of a rule i is measured by its degree of firing (DOF) $\tau_{DOF,i}$, which represents the degree of truth of the premise part of a rule, given the premise $u_1 = x_1^*$ and $u_2 = x_2^*$. The DOF is calculated as:

$$\tau_{DOF,i} = \text{mf}_{u_1,i}(x_1^*) \wedge \text{mf}_{u_2,i}(x_2^*) \tag{A.13}$$

The DOF of a rule can be used to determine the fuzzy set for the output of that rule, given the premise $u_1 = x_1^*$ and $u_2 = x_2^*$. This is the inference step:

$$\text{mf}_{y,i}^*(y) = \tau_{DOF,i} \wedge \text{mf}_{y,i} \tag{A.14}$$

The next step is to aggregate the inferred fuzzy sets $\text{mf}_{y,i}^*$ by using the $\max$ operator:

$$\text{mf}_y(y) = \wedge_{i=1}^{2} \text{mf}_{y,i}^* \tag{A.15}$$

The final step involves defuzzification of the aggregated fuzzy set for the output y, which results in a crisp output value. Figure A.3 shows the inference mechanism.

A.2.3 TSK models

A known disadvantage of the linguistic models is that they do not incorporate specific knowledge in an explicit form, if that kind of information is available but cannot be expressed within the framework of fuzzy set theory. This kind of information is often available in the form of mathematical equations. Sugeno and co-workers proposed an alternative form of fuzzy reasoning, which provides a possibility to incorporate such information. The TSK reasoning method is associated with a rule-base of a special format that is characterized with

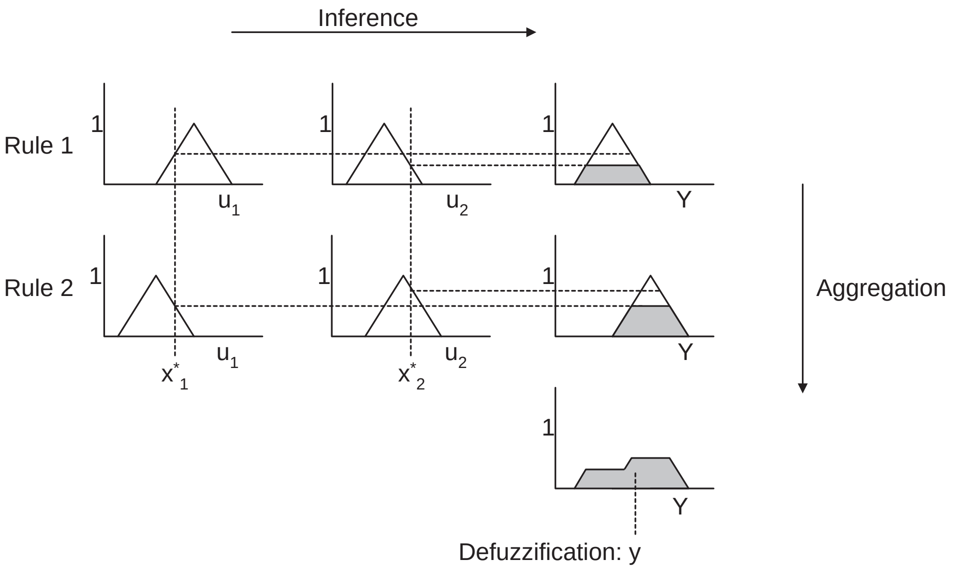


Figure A.3: *Fuzzy inference mechanism for linguistic models*

functional type consequence parts instead of the fuzzy consequence parts used in the linguistic model. For a 2-by-1 system with two independent rules this can be represented as follows:

$$\begin{array}{l} \text{IF } u_1 = \mathrm{mf}_{u_1,1} \text{ AND } u_2 = \mathrm{mf}_{u_2,1} \text{ THEN } y = a_1 u_1 + b_1 u_2 + c_1 \\ \text{IF } u_1 = \mathrm{mf}_{u_1,2} \text{ AND } u_2 = \mathrm{mf}_{u_2,2} \text{ THEN } y = a_2 u_1 + b_2 u_2 + c_2 \end{array} \tag{A.16}$$

In this example, the consequent part incorporates linear equations, but in essence, any function can be incorporated. The crisp output y, inferred by the fuzzy model, is defined as the weighted average of the crisp outputs y_i of each rule:

$$y = \frac{\sum_{i=1}^{2} \tau_{DOF,i} y_i}{\sum_{i=1}^{2} \tau_{DOF,i}} \tag{A.17}$$

The great advantage of TSK models is the power to describe complex technological processes. The problem is decomposed into simpler sub-systems, in many cases linear.

B

Bioreactor model

c_{Lm}	=	0.0084 h^{-1}
K	=	0.01 h^{-1}
K_I	=	1.0 g/l
K_L	=	0.05
K_p	=	0.0001 g/l
K_X	=	0.3
m_{xm}	=	0.029 h^{-1}
q_{pm}	=	0.004 h^{-1}
$Y_{p/s}$	=	1.2
$Y_{x/s}$	=	0.47
μ_m	=	0.11 h^{-1}

Table B.1: *Bioreactor model parameters*

This appendix describes the fed-batch bioreactor reference model which is used in chapter 4.

B.1 Model equations

The model describes four states: the biomass concentration $X\ (g/l)$, the substrate concentration $S\ (g/l)$, the product concentration $P\ (g/l)$ and the volume $V\ (l)$. The state equations are given by:

$$\frac{dX}{dt} = X(\mu - D - c_L) \tag{B.1}$$

$$\frac{dS}{dt} = -\sigma X + (S_f - S)D \tag{B.2}$$

$$\frac{dP}{dt} = q_p X - P(D + K) \tag{B.3}$$

$$\frac{dV}{dt} = F \tag{B.4}$$

in which $S_f\ (g/l)$ is the feed substrate concentration, $K\ (h^{-1})$ is the product decay constant.

Dilution rate:

$$D = \frac{F}{V} \tag{B.5}$$

Growth rate:

$$\mu = \frac{\mu_m S}{K_X X + 10} \tag{B.6}$$

Cell lysis rate:

$$c_L = \frac{c_{Lm} X}{K_L + X + 1} \exp(-S/100) \tag{B.7}$$

Product formation rate:

$$q_p = \frac{1.5 q_{pm} S X}{4K_p + XS\left(1 + \frac{S}{3K_I}\right)} \tag{B.8}$$

Variable	*Noise amplitude*
X	$0.2\ g/l$
S	$0.1\ g/l$
P	$0.1\ g/l$
V	$0\ g/l$
F	$0.005\ l/h$

Table B.2: *process and input noise amplitudes*

Substrate consumption rate:

$$\sigma = \frac{\mu}{Y_{x/s}} + \frac{q_p}{Y_{p/s}} + m_x \tag{B.9}$$

with

$$m_x = \frac{m_{xm} X}{X + 10} \tag{B.10}$$

Model parameters are given in table B.1.

For the simulations, a discrete version of the model is implemented. White noise of a certain amplitude is added to the model process state and input at time step k. Since the state of time step k is used to calculate the state at time step $k+1$, some correlation occurs. Noise amplitude settings are given in table B.2.

B.2 Notation

D	Dilution rate	(h^{-1})
F	Feed flow rate	(l/h)
K	Constant	(h^{-1})
K_L	Constant	$(-)$
K_X	Constant	$(-)$
P	Product concentration	(g/l)
S	Substrate concentration	(g/l)
S_F	Substrate concentration in the feed	(g/l)
V	Volume	(l)
X	Biomass concentration	(g/l)
$Y_{x/s}$	Constant	$(-)$
$Y_{p/s}$	Constant	$(-)$
c_L	Cell lysis rate	(h^{-1})
c_{Lm}	Constant	(h^{-1})
m_x	Maintenance energy	(h^{-1})
m_{xm}	Constant	(h^{-1})
q_p	Product formation rate	(h^{-1})
α	Net growth rate (hybrid model)	(h^{-1})
μ	Growth rate	(h^{-1})
μ_m	Constant	(h^{-1})
σ	Substrate consumption rate	(h^{-1})

C

Extended Purdue Model

Section	*Height* (m)	*Cross sectional area* S (m^2)
A	5.50	18.0
B	1.34	18.0
C	1.83	18.7
D	1.55	18.7
E	9.81	18.7
F	1.46	18.7
G	5.18	18.7
H	1.58	19.9
I	1.74	19.9
J	2.01	21.1

Table C.1: *EPM section dimensions*

This appendix describes the Extended Purdue Model. The model flowsheet, section model data flow diagram and model parameters are given. Other information can be found in Wisnewski *et al.* (1997).

C.1 Model flowsheet

The model flowsheet consists of a series of tubular reactors or ”sections” that are connected to form the digester model. Mixers, splitters and heaters are also added. The flowsheet is shown in figure C.1. Table C.1 shows section dimensions.

C.2 Section model

All model equations of the section model are listed here. Figure C.2 shows the context diagram of the section model, while figure C.3 shows the Data Flow Diagram for the section model. This DFD represents the physical effects that are modeled in relation to the states.

Mass balance solid phase:

$$\frac{\partial \rho_{s,i}}{\partial t} = -\frac{\phi_c}{S(1-\eta)}\frac{\partial \rho_{s,i}}{\partial z} + R_{s,i} \text{ for } i = 1, \ldots, 5 \tag{C.1}$$

Mass balance entrapped phase:

$$\frac{\partial \rho_{e,i}\varepsilon}{\partial t} = -\frac{\phi_c}{S(1-\eta)}\frac{\partial \rho_{e,i}\varepsilon}{\partial z} + \varepsilon D_{ce}(\rho_{f,i} - \rho_{e,i}) + R_{s,i} + \hat{\phi}_b \rho_{f,i} \text{ for } i = 1, \ldots, 6 \tag{C.2}$$

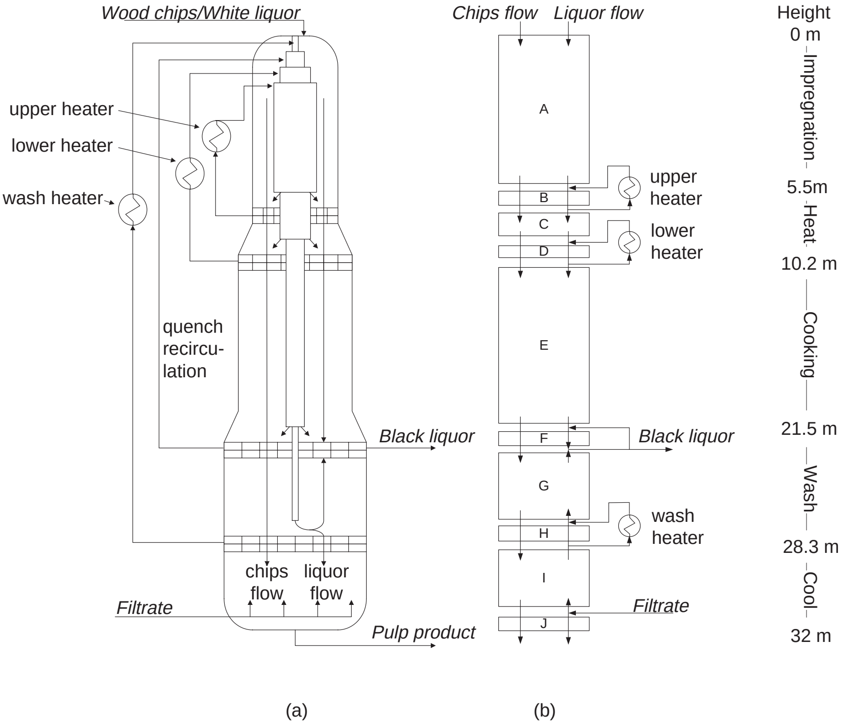


Figure C.1: *Digester unit operation (a) and model flowsheet (b)*

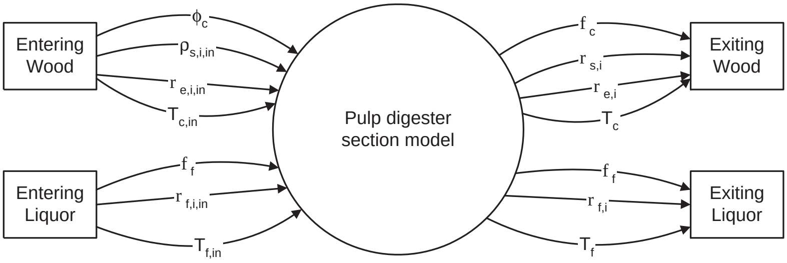


Figure C.2: *Digester section model context diagram*

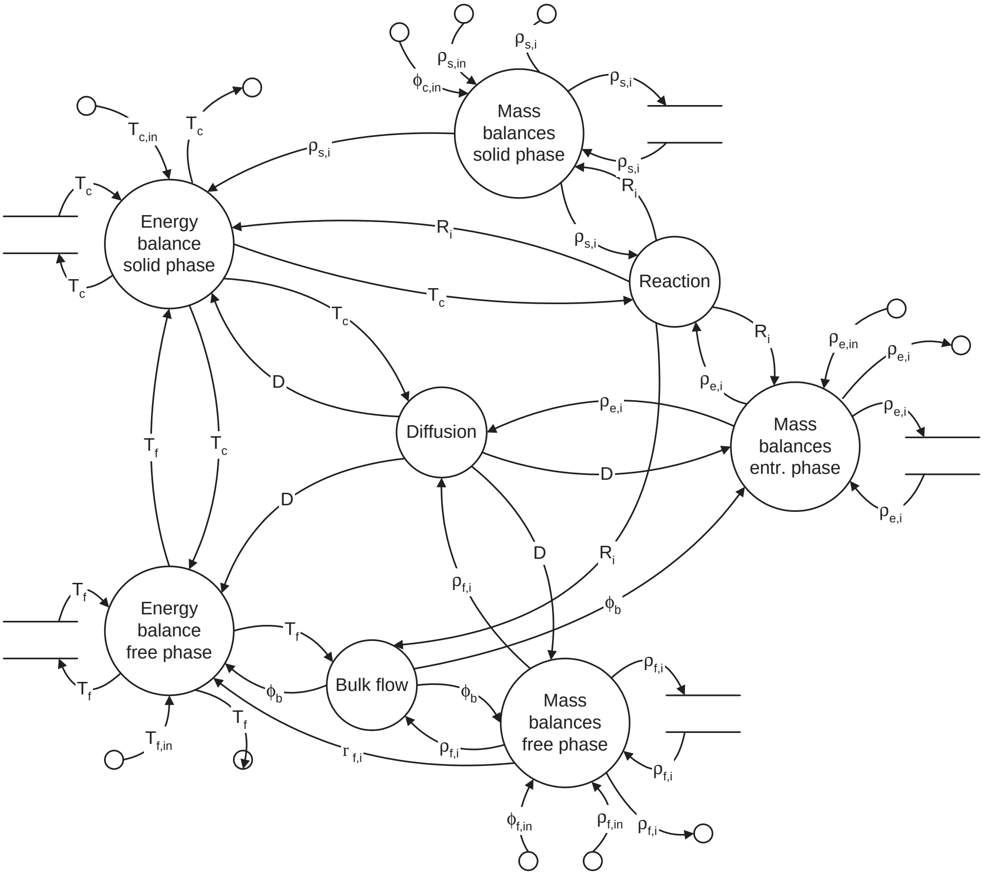


Figure C.3: *Digester section model Data Flow Diagram (b)*

Mass balance free liquor phase:

$$\frac{\partial \rho_{f,i}}{\partial t} = -\frac{1}{S\eta}\frac{\partial \rho_{f,i}\phi_f}{\partial z} + D_{cf}(\rho_{e,i} - \rho_{f,i}) \\ - \frac{1-\eta}{\eta}\hat{\phi}_b \rho_{f,i} \text{ for } i = 1, \ldots, 6 \quad \text{(C.3)}$$

Energy balance solid and entrapped phase:

$$\frac{\partial}{\partial t}(Cp_s M_s + Cp_e M_e \varepsilon)T_c = -\frac{\phi_c}{S(1-\eta)}\frac{\partial}{\partial z}(Cp_s M_s + Cp_e M_e \varepsilon)T_c \\ + \Delta H_R \sum_{i=1}^{5} R_{s,i} + U(T_f - T_c) + \hat{\phi}_b Cp_f M_f T_f + \varepsilon D_{ce} D_E \quad \text{(C.4)}$$

Energy balance free liquor phase:

$$\frac{\partial}{\partial t}(Cp_f M_f T_f) = -\frac{1}{S(\eta S)}\frac{\partial}{\partial z}(\phi_f Cp_f M_f T_f) + \frac{1-\eta}{\eta}U(T_c - T_f) \\ + \frac{1-\eta}{\eta}\hat{\phi}_b Cp_f M_f T_f + D_{cf} D_F \quad \text{(C.5)}$$

Reaction rate:

$$R_{s,i} = -e_f(k_{1,i}\rho_{e,1} + k_{2,i}\sqrt{\rho_{e,1}\rho_{e,3}})(\rho_{s,i} - \rho_{s,i}^{\infty}) \text{ for } i = 1, \ldots, 5 \quad \text{(C.6)}$$

Reaction rate constants:

$$k_{1,i} = A_{1,i} \exp\left(\frac{-E_{1,i}}{RT_c}\right) \text{ for } i = 1, \ldots, 5 \quad \text{(C.7)}$$

$$k_{2,i} = A_{2,i} \exp\left(\frac{-E_{2,i}}{RT_c}\right) \text{ for } i = 1, \ldots, 5 \quad \text{(C.8)}$$

Reaction stochiometry:

$$R_{e,i} = \sum_{j=1}^{5} b_{i,j} R_{s,j} \text{ for } i = 1, \ldots, 6 \quad \text{(C.9)}$$

Volumetric bulk flow:

$$\hat{\phi}_b = \frac{-\sum_{i=1}^{5} R_{s,i}}{\bar{\rho}_s} \text{ for } i = 1, \ldots, 5 \quad \text{(C.10)}$$

Diffusion coefficient:

$$D_{ce} = 6.1321\sqrt{T_c}\exp\left(\frac{-4870}{1.98T_c}\right) \quad \text{(C.11)}$$

Volumetric correction for diffusion coefficient free liquor phase:

$$D_{cf} = D_{ce}\frac{\varepsilon(1-\eta)}{\eta} \tag{C.12}$$

Net energy transported into woodchips per volume of diffusing mass:

$$D_E = T_p Cp_l \sum_{i=1}^{4}(\rho_{f,i} - \rho_{e,i}) + T_p Cp_s \sum_{i=5}^{6}(\rho_{f,i} - \rho_{e,i}) \tag{C.13}$$

in which

$$T_p = \begin{cases} T_f \text{ if } \rho_{f,i} > \rho_{e,i} \\ T_c \text{ if } \rho_{f,i} < \rho_{e,i} \end{cases} \tag{C.14}$$

$$D_F = -D_E \tag{C.15}$$

Porosity:

$$\varepsilon = 1 - \frac{\sum_{i=1}^{5}\rho_{s,i}}{\bar{\rho_s}} \tag{C.16}$$

Heat capacity entrapped phase:

$$Cp_e = Cp_l\frac{M_{el}}{M_e} + Cp_s\frac{M_{es}}{M_e} \tag{C.17}$$

Total density solid phase:

$$M_s = \sum_{i=1}^{5}\rho_{s,i} \tag{C.18}$$

Total density entrapped phase

$$M_e = M_{el} + M_{es} \tag{C.19}$$

Heat capacity free liquor phase:

$$Cp_f = Cp_l\frac{M_{fl}}{M_f} + Cp_s\frac{M_{fs}}{M_f} \tag{C.20}$$

Total density free liquor phase

$$M_f = M_{fl} + M_{fs} \tag{C.21}$$

Partial liquid components density entrapped phase

$$M_{el} = \rho_{water} + \sum_{i=1}^{4}\rho_{e,i} \tag{C.22}$$

Partial solid components density entrapped phase

$$M_{es} = \sum_{i=5}^{6} \rho_{e,i} \tag{C.23}$$

Partial liquid components density free liquor phase

$$M_{fl} = \rho_{water} + \sum_{i=1}^{4} \rho_{f,i} \tag{C.24}$$

Partial solid components density free liquor phase

$$M_{fs} = \sum_{i=5}^{6} \rho_{f,i} \tag{C.25}$$

Kappa number:

$$\kappa\# = \frac{\rho_{s,1} + \rho_{s,2}}{0.00153 \sum_{i=1}^{5} \rho_{s,i}} \tag{C.26}$$

Yield:

$$\gamma = \frac{\sum_{i=1}^{5} \rho_{s,i,exiting}}{\sum_{i=1}^{5} \rho_{s,i,entering}} \tag{C.27}$$

C.3 Heater model

In the hybrid model, the EPM upper and lower heater are lumped to form a ”preheater”. To make comparisons between the two models, the preheater has to provide the same amount of energy as the upper and lower heater do. The resulting outlet temperature depends on this energy, the composition of the flow and the flow rate. Therefore, in the EPM, the heaters are modeled using a simple static energy balance. The heater model can be described as follows:

$$\phi_{out} = \phi_{in} \tag{C.28}$$

$$Q_h = \phi_{in} Cp_{in} M_{in} (T_{out} - T_{in}) \tag{C.29}$$

$$M_{in} = M_{l,in} + M_{s,in} \tag{C.30}$$

in which

$$M_{l,in} = \rho_{water} + \sum_{i=1}^{4} \rho_{f,i,in}; \tag{C.31}$$

and

$$M_{s,in} = \sum_{i=5}^{6} \rho_{f,i,in}; \tag{C.32}$$

i	*Softwood (–)*	*Hardwood (–)*
1	0	0
2	0	0
3	0.71	0.65
4	0.25	0.25
5	0	0

Table C.2: *Non-reactive fractions of solid phase components*

$$Cp_{in} = Cp_l \frac{M_{l,in}}{M_{in}} + Cp_s \frac{M_{s,in}}{M_{in}}; \tag{C.33}$$

C.4 Typical operating values and parameters

The non-reactive portions of the solid components are related to the composition of the wood entering the first section of the digester and are the same in any following digester section. The non-reactive portions are independent of the exact wood composition, but do depend on the type of wood. The non-reactive portions are related to the densities of solid components entering the digester in the following way:

$$\rho_{s,i}^{\infty} = \alpha_{s,i}^{\infty} \rho_{s,i,in} \tag{C.34}$$

in which $\rho_{s,i}^{\infty}$ is the non-reactive portion of the component, $\alpha_{s,i}^{\infty}$ is the non-reactive fraction and $\rho_{s,i,in}$ is the initial density of the component. The fraction for the solid phase components are given in table C.2.

The stochiometric matrix b which links the reaction rates of the solid phase components to the reaction rates of the entrapped phase components is defined as:

$$b = \begin{bmatrix} \beta_{OHL} - 0.5\beta_{HSL} & \beta_{OHL} - 0.5\beta_{HSL} & \beta_{OHC} & \beta_{OHC} & \beta_{OHC} \\ -(\beta_{OHL} - 0.5\beta_{HSL}) & -(\beta_{OHL} - 0.5\beta_{HSL}) & -\beta_{OHC} & -\beta_{OHC} & -\beta_{OHC} \\ 0.5\beta_{HSL} & 0.5\beta_{HSL} & 0 & 0 & 0 \\ -0.5\beta_{HSL} & -0.5\beta_{HSL} & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 \end{bmatrix} \tag{C.35}$$

The stochiometric coefficients for the consumption of Effective Alkali and Hydrosulfide are stated in table C.3. Tables C.3 through C.6 show various other model parameters.

The implemented compaction profile is based upon (Wisnewski *et al.*, 1997), which uses different, fixed compaction values for each CSTR. In the PFR approach used in this work,

	Softwood	*Hardwood*
β_{OHL}	$0.166\ kg$ OH$/kg$ lignin	$0.21\ kg$ OH$/kg$ lignin
β_{OHC}	$0.395\ kg$ OH$/kg$ carbohydrate	$0.49\ kg$ OH$/kg$ carbohydrate
β_{HSL}	$0.039\ kg$ HS$/kg$ lignin	$0.05\ kg$ HS$/kg$ lignin

Table C.3: *Stochiometric coefficients*

	$(s,1)$	$(s,2)$	$(s,3)$	$(s,4)$	$(s,5)$
A_1 (hardwood)	0.3954	1.457E11	28.09	7.075	5.8267E3
A_1 (softwood)	0.2809	6.035E10	6.4509	1.5607	1.0197E4
A_2 (hardwood)	12.49	1.873	124.9	47.86	3.225E16
A_2 (softwood)	9.26	0.489	28.09	10.41	5.7226E16
E_1	29.3	115	34.7	25.1	73.3
E_2	31.4	37.7	41.9	37.7	167

Table C.4: *Pre-exponential factors and activation energies*

those compaction values are linearized along the spatial domain. The following relations describe the linear compaction profile. Two discontinuities exist at places in the digester where the diameter alters.

$$\eta = \begin{cases} -0.0135z + 0.6966 & \text{for } z \leq 10.9195 \\ -0.0122z + 0.7227 & \text{for } 10.9195 < z \leq 20.2502 \\ -0.0101z + 0.7002 & \text{for } 20.2502 < z \end{cases} \tag{C.36}$$

Parameter	*Value*		*Definition*
ϕ_c	1.3964	m^3/min	woodchip flow rate
$\phi_{f,in}$	2.3497	m^3/min	entering white liquor flow rate
ϕ_{uh}	7.2728	m^3/min	upper heater recirculation flow rate
T_{uh}	418.83	K	upper heater recirculation temperature
ϕ_{lh}	6.7785	m^3/min	lower heater recirculation flow rate
T_{lh}	438.64	K	lower heater recirculation temperature
ϕ_{quench}	0	m^3/min	quench recirculation flow rate
ϕ_{wash}	0.2067	m^3/min	wash zone recirculation flow rate
T_{wash}	363.4	K	wash heater temperature
ϕ_{cheat}	0.3514	m^3/min	cheater flow rate
ϕ_{cool}	0.5	m^3/min	cooling zone flow rate
ϕ_{filt}	1.6312	m^3/min	entering filtrate flow rate
T_{filt}	360.93	K	entering filtrate temperature
$T_{c,in}$	395	K	entering woodchip temperature
$T_{f,in}$	382	K	entering white liquor temperature
$\rho_{s,1,in}$	25.37	kg/m^3	high reactivity lignin density of entering woodchip
$\rho_{s,2,in}$	101.49	kg/m^3	low reactivity density of entering woodchip
$\rho_{s,3,in}$	270.53	kg/m^3	cellulose density of entering woodchip
$\rho_{s,4,in}$	12	kg/m^3	galactoglucomman density of entering woodchip
$\rho_{s,5,in}$	129.03	kg/m^3	araboxylan density of entering woodchip
$\rho_{e,1,in}$	0.01	kg/m^3	active effective alkali concentration in entrapped liquor phase of entering woodchip
$\rho_{e,2,in}$	0.01	kg/m^3	passive effective alkali concentration in entrapped liquor phase of entering woodchip
$\rho_{e,3,in}$	0.01	kg/m^3	active hydrosulfide concentration in entrapped liquor phase of entering woodchip
$\rho_{e,4,in}$	0.01	kg/m^3	passive hydrosulfide concentration in entrapped liquor phase of entering woodchip
$\rho_{e,5,in}$	0.01	kg/m^3	dissolved lignin concentration in entrapped liquor phase of entering woodchip
$\rho_{e,6,in}$	0.01	kg/m^3	dissolved carbohydrates concentration in entrapped liquor phase of entering woodchip
$\rho_{f,1,in}$	78.8	kg/m^3	active effective alkali concentration of entering white liquor flow
$\rho_{f,2,in}$	0	kg/m^3	passive effective alkali concentration of entering white liquor flow
$\rho_{f,3,in}$	13.3	kg/m^3	active hydrosulfide concentration of entering white liquor flow
$\rho_{f,4,in}$	0	kg/m^3	passive hydrosulfide concentration of entering white liquor flow
$\rho_{f,5,in}$	59.6	kg/m^3	dissolved lignin concentration of entering white liquor flow
$\rho_{f,6,in}$	59.6	kg/m^3	dissolved carbohydrates concentration of entering white liquor flow
$\rho_{f,1,filtr}$	4.8055	kg/m^3	active effective alkali concentration of entering filtrate flow
$\rho_{f,2,filtr}$	0.01	kg/m^3	passive effective alkali concentration of entering filtrate flow
$\rho_{f,3,filtr}$	0.01	kg/m^3	active hydrosulfide concentration of entering filtrate flow
$\rho_{f,4,filtr}$	0.01	kg/m^3	passive hydrosulfide concentration of entering filtrate flow
$\rho_{f,5,filtr}$	45	kg/m^3	dissolved lignin concentration of entering filtrate flow
$\rho_{f,6,filtr}$	45	kg/m^3	dissolved carbohydrates concentration of entering filtrate flow

Table C.5: *Standard operating parameters*

Parameter	*Value*		*Definition*
e_f	0.9	$-$	reaction rate multiplier
$\bar{\rho}_s$	1666	kg/m^3	density of solid wood material
ρ_{water}	999	kg/m^3	density of water
ΔH	-518	kJ/kg	heat of reaction
Cp_l	4.19	$kJ/(kg\ K)$	heat capacity liquor
Cp_s	1.47	$kJ/(kg\ K)$	heat capacity wood substance
R	0.0083145	$kJ/(mol\ K)$	universal gas constant
U	827	$kJ/(min\ m^3\ K)$	heat transfer coefficient

Table C.6: *Digester unit operation parameters*

C.5 Frequency analysis

Figures C.4-C.7 show the Bode diagrams for the Kappa number and the yield as a function of the lower heater heat output Q_{lh} and the liquor flow rate ϕ_f. Table C.7 shows the frequency analysis results; the amplitudes $\Delta\kappa\#$ and $\Delta\gamma$ are the gain for a 10 % and 7 % step change in Q_{lh} and ϕ_l, respectively. The amplitudes for a positive and a negative step change are presented. In the table, O denotes the order of the system response, τ the time constant and θ the dead time.

Input	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$	$\tau_{[input],\gamma}$	$\theta_{[input]}$	$\Delta\kappa\#\ (-)$	$\Delta\gamma\ (-)$
			(min)	(min)	(min)		
Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020

Table C.7: *Frequency response analysis results for EPM*

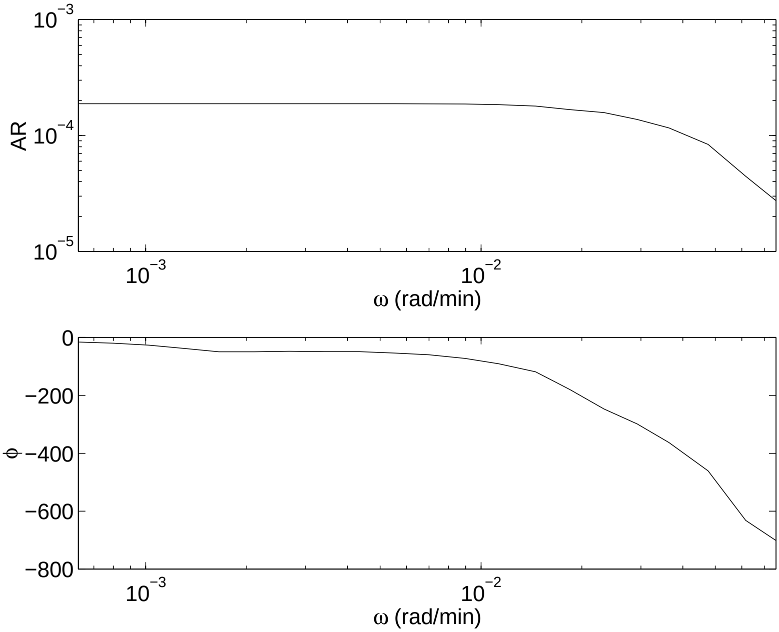


Figure C.4: *Bode diagram $\kappa\#$ as a function of Q_{lh}*

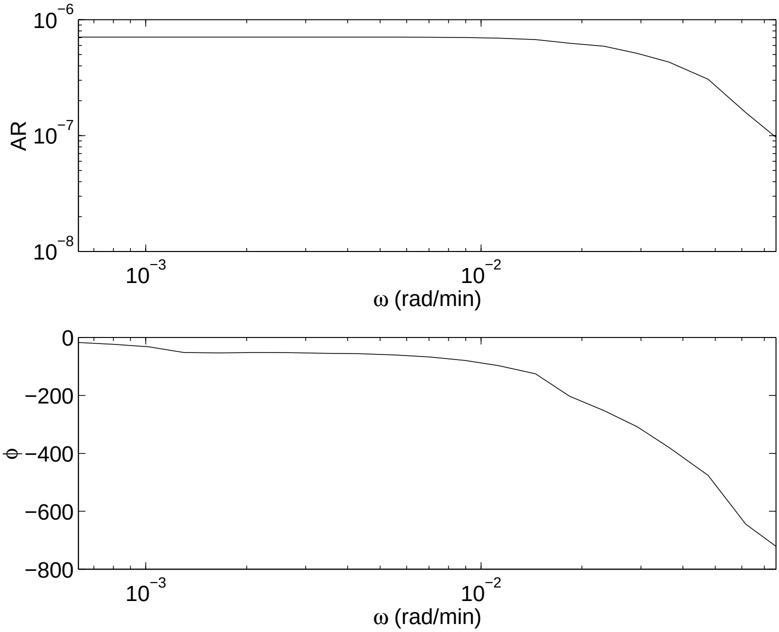


Figure C.5: *Bode diagram γ as a function of Q_{lh}*

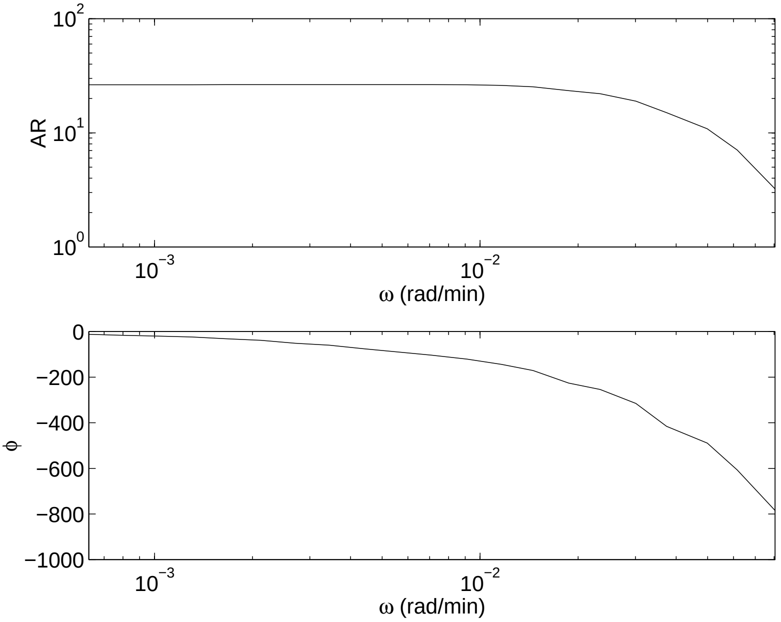


Figure C.6: *Bode diagram $\kappa\#$ as a function of ϕ_f*

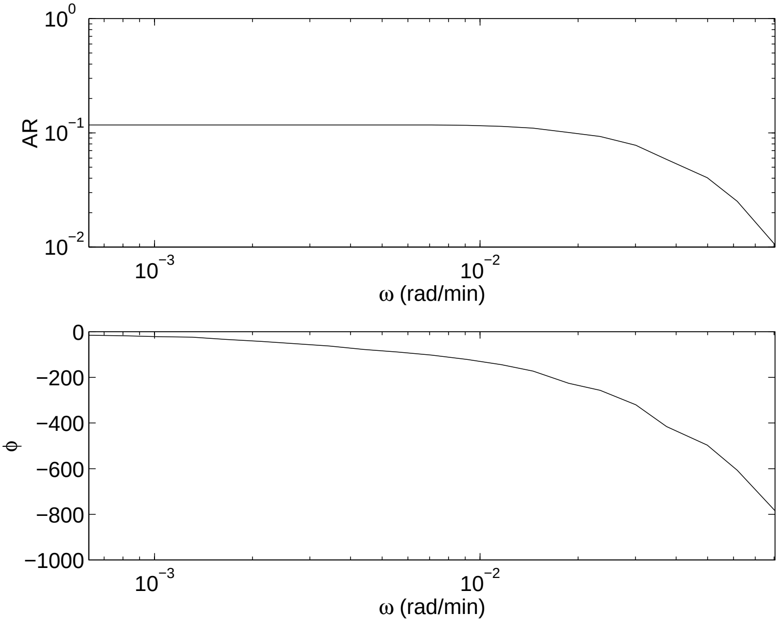


Figure C.7: *Bode diagram γ as a function of ϕ_f*

C.6 Notation

$A_{1,i}$	Arrhenius constant species i	$(m^3/kg\ min)$
$A_{2,i}$	Arrhenius constant species i	$(m^3/kg\ min)$
Cp_e	Heat capacity, entrapped phase	$(kJ/kg\ K)$
Cp_f	Heat capacity, free liquor phase	$(kJ/kg\ K)$
Cp_l	Heat capacity, liquor	$(kJ/kg\ K)$
Cp_s	Heat capacity, solid phase	$(kJ/kg\ K)$
D_E	Energy transported into wood chip	(kJ/m^3)
D_F	Energy transported into free liquor phase	(kJ/m^3)
D_{ce}	Diffusion coefficient, entrapped phase	(min^{-1})
D_{cf}	Diffusion coefficient, free liquor phase	(min^{-1})
$E_{1,i}$	Activation energy species i	(kJ/mol)
$E_{2,i}$	Activation energy species i	(kJ/mol)
M_e	Mass entrapped phase	(kg/m^3)
M_{el}	Mass liquor components, entrapped phase	(kg/m^3)
M_{es}	Mass solid components, entrapped phase	(kg/m^3)
M_f	Mass free liquor phase	(kg/m^3)
M_{fl}	Mass liquor components, free liquor phase	(kg/m^3)
M_{fs}	Mass solid components, free liquor phase	(kg/m^3)
M_{in}	Entering mass, heater	(kg/m^3)
$M_{l,in}$	Entering mass, liquor components, heater	(m^3/min)
M_s	Mass solid phase	(kg/m^3)
$M_{s,in}$	Entering mass, solid components, heater	(m^3/min)
Q_h	Heat supply heater	(kJ/min)
Q_{lh}	Heat supply lower heater	(kJ/min)
Q_{uh}	Heat supply upper heater	(kJ/min)
R	Universal gas constant	$(kJ/molK)$
$R_{e,i}$	Reaction rate species (e,i)	(m^3/min)
$R_{s,i}$	Reaction rate species (s,i)	(m^3/min)
S	Cross sectional area	(m^2)
T_c	Wood chip temperature	(K)
T_f	Free liquor phase temperature	(K)
T_{in}	Temperature entering flow rate, heater	(K)
T_{out}	Temperature exiting flow rate, heater	(K)
T_p	Temperature used in heat transfer calculation	(K)
U	Heat transfer coefficient	$(kJ/min\ m^3K)$
$\boldsymbol{b}$	Reaction stochiometry	$(-)$
e_f	Reaction rate multiplier	$(-)$
$k_{1,i}$	Kinetic constant species i	$(m^3/kg\ min)$
$k_{2,i}$	Kinetic constant species i	$(m^3/kg\ min)$
z	Location in the reactor, top = 0	(m)
ΔH_R	Reaction enthalpy	(kJ/kg)
$\alpha^{\infty}_{s,i}$	Non-reactive fraction species (s,i)	(kg/m^3)
β_*	Stochiometric coefficients	$(-)$
γ	Yield	$(-)$
ε	Porosity	$(-)$
η	Ratio free liquor volume - reactor volume	$(-)$
θ_Q	Dead time with respect to Q_h	(min)
θ_ϕ	Dead time with respect to ϕ_l	(min)
$\kappa\#$	Kappa number	$(-)$

$\rho_{e,i}$	Concentration species i, entrapped phase	(kg/m^3)
$\rho_{f,i}$	Concentration species i, free liquor phase	(kg/m^3)
$\rho_{s,i}$	Concentration species i, solid phase	(kg/m^3)
$\rho_{s,i}^{\infty}$	Non-reactive portion species (s,i)	(kg/m^3)
$\bar{\rho}_s$	Density of solid material	(kg/m^3)
ρ_{water}	Density of water	(kg/m^3)
$\tau_{[input],[output]}$	Time constant for $[output]$ with respect to $[input]$	(min)
$\hat{\phi}_b$	Bulk flow	(m^3/min)
ϕ_c	Wood chip flow rate	(m^3/min)
ϕ_{in}	Entering flow rate, heater	(m^3/min)
ϕ_l	Liquor phase flow rate	(m^3/min)
ϕ_{out}	Exiting flow rate, heater	(m^3/min)

D

Pulp Digester Hybrid Model

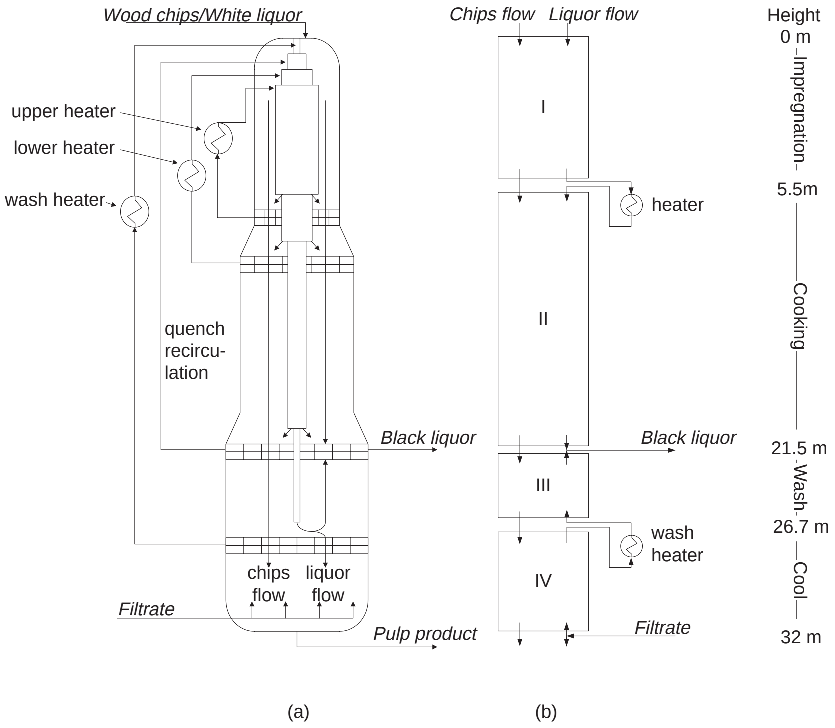


Figure D.1: *Digester unit operation (a) and hybrid model flowsheet (b)*

This appendix describes the hybrid model of the continuous pulp digester that is described in chapter 5. The model flowsheet, section model data flow diagram and model parameters are given.

D.1 Model flowsheet

The model flowsheet consists of a series of tubular reactors or ”sections” that are connected to form the digester model. Mixers, splitters and heaters are also added. The flowsheet is shown in figure D.1. Table D.1 shows section dimensions.

Section	*Height (m)*	*Cross sectional area S (m^2)*
I	5.50	18.0
II	15.99	18.7
III	5.18	18.7
IV	5.33	21.1

Table D.1: *Hybrid model section dimensions*

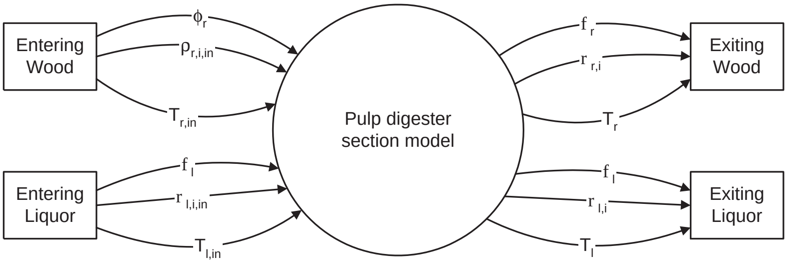


Figure D.2: *Hybrid model section context diagram*

D.2 Section model

All model equations of the section model are listed here. Figure D.2 shows the context diagram of the section model, while figure D.3 shows the Data Flow Diagram for the section model. This DFD represents the physical effects that are modeled in relation to the states.

Mass balance reaction phase:

$$\frac{\partial \rho_{r,i}}{\partial t} = -\frac{\phi_r}{S(1-\eta)}\frac{\partial \rho_{r,i}}{\partial z} + R_{r,i} \text{ for } i = 1,\ldots,3 \tag{D.1}$$

$$\frac{\partial \rho_{r,i}}{\partial t} = -\frac{\phi_r}{S(1-\eta)}\frac{\partial \rho_{r,i}}{\partial z} + R_{r,i} + D_{cr}(\rho_{l,i} - \frac{1}{\varepsilon}\rho_{r,i+3}) \text{ for } i = 4,\ldots,7 \tag{D.2}$$

The difference between equations D.1 and D.2 is that species $(r,1)$ to $(r,3)$ cannot diffuse to the liquor phase, while species $(r,4)$ to $(r,7)$ can. The porosity ε is introduced in the diffusion term to compensate for the lumping of the solid and entrapped phases. In the EPM, the concentrations of species $(r,4)$ to $(r,7)$ are based on the entrapped phase volume. In the hybrid model, they are based on the chip volume. Dividing by ε corrects this. An alternative is to build a fuzzy equation that calculates D as a function of T_r, $\rho_{l,i}$ and $\rho_{r,i+3}$, but introducing ε is simpler. The corrected concentration can be interpreted as an "effective concentration" that determines the driving force behind the diffusion. The porosity ε is defined in equation D.13.

Mass balance liquor phase:

$$\frac{\partial \rho_{l,i}}{\partial t} = -\frac{1}{S\eta}\frac{\partial \rho_{l,i}\phi_l}{\partial z} + D_{cl}(\frac{1}{\varepsilon}\rho_{r,i+3} - \rho_{l,i}) \quad \text{for} \quad i = 1,\ldots,4 \tag{D.3}$$

Energy balance reaction phase:

$$\frac{\partial}{\partial t}(Cp_r M_r T_r) = -\frac{\phi_r}{S(1-\eta)}\frac{\partial}{\partial z}(Cp_r M_r T_r) + \Delta H_R \sum_{i=1}^{3} R_{r,i} + U(T_l - T_r) + D_{cr} D_R \tag{D.4}$$

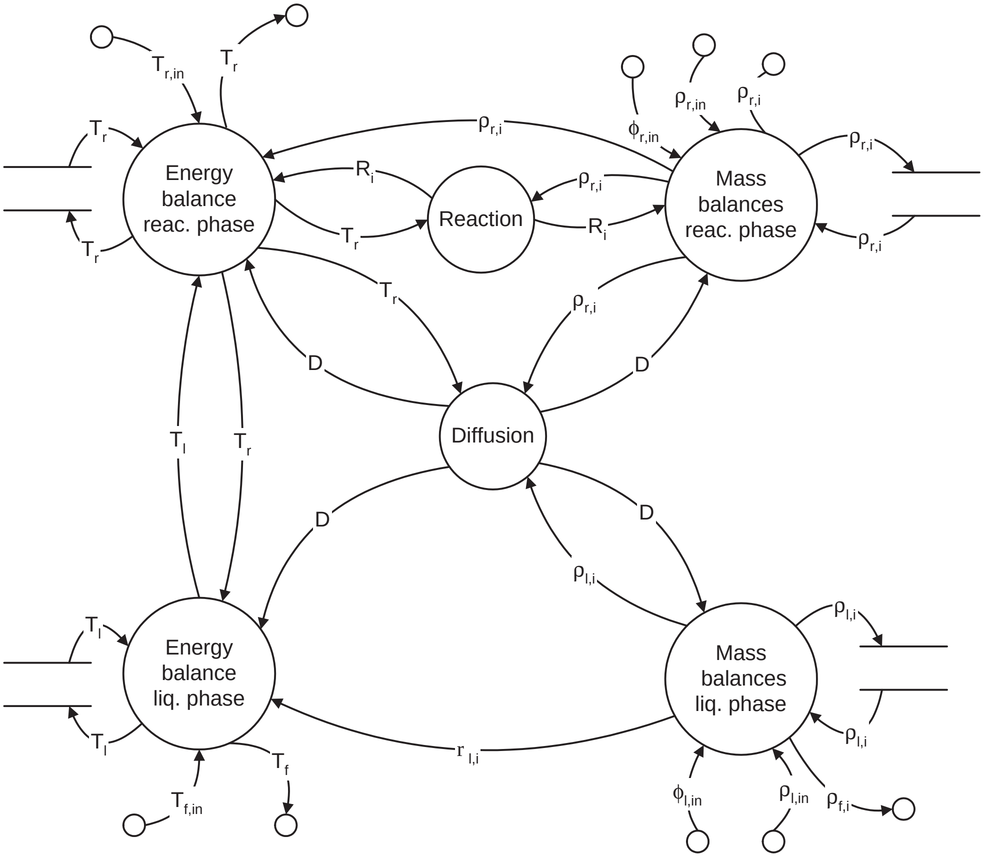


Figure D.3: *Hybrid model section Data Flow Diagram*

Energy balance liquor phase:

$$\frac{\partial}{\partial t}(Cp_l M_l T_l) \;=\; -\frac{1}{S(\eta S)}\frac{\partial}{\partial z}(\phi_l Cp_l M_l T_l) \;+\; \frac{1-\eta}{\eta}U(T_r \;-\; T_l) \;+\; D_{cl}D_L \quad \text{(D.5)}$$

Reaction rate:

$$R_{r,i} = f_{fuzzy}(\rho_{r,i}, \rho_{r,4}, T_r) \;\; \text{for} \;\; i = 1, \ldots, 3 \quad \text{(D.6)}$$

Reaction stochiometry:

$$R_{r,i} = \sum_{j=1}^{3} b_{i,j} R_{r,j} \;\; \text{for} \;\; i = 4, \ldots, 7 \quad \text{(D.7)}$$

Diffusion coefficient:

$$D_{cr} = f_{fuzzy}(T_r) \quad \text{(D.8)}$$

Volumetric correction for diffusion coefficient liquor phase:

$$D_{cl} = D_{cr}\frac{1-\eta}{\eta} \quad \text{(D.9)}$$

Net energy transported into woodchips per volume of diffusing mass:

$$D_R = T_p Cp_w \sum_{i=1}^{2}(\rho_{l,i} - \frac{1}{\varepsilon}\rho_{r,i+3}) + T_p Cp_s \sum_{i=3}^{4}(\rho_{l,i} - \frac{1}{\varepsilon}\rho_{r,i+3}) \quad \text{(D.10)}$$

in which

$$T_p = \begin{cases} T_l \text{ if } \rho_{l,i} > \rho_{r,i} \\ T_r \text{ if } \rho_{l,i} < \rho_{r,i} \end{cases} \quad \text{(D.11)}$$

$$D_L = -D_R \quad \text{(D.12)}$$

Porosity:

$$\varepsilon = 1 - \frac{\sum_{i=1}^{3} \rho_{r,i}}{\bar{\rho_s}} = \frac{V_e}{V_c} \quad \text{(D.13)}$$

Heat capacity reaction phase:

$$Cp_r = Cp_w \frac{M_{rl}}{M_r} + Cp_s \frac{M_{rr}}{M_r} \quad \text{(D.14)}$$

Total density reaction phase

$$M_r = M_{rl} + M_{rr} \quad \text{(D.15)}$$

with

$$M_{rr} = \sum_{i=1}^{3} \rho_{r,i} + \sum_{i=6}^{7} \rho_{r,i} \tag{D.16}$$

and

$$M_{rl} = \rho_{water} + \sum_{i=4}^{5} \rho_{r,i} \tag{D.17}$$

Heat capacity liquor phase:

$$Cp_l = Cp_w \frac{M_{ll}}{M_l} + Cp_s \frac{M_{lr}}{M_l} \tag{D.18}$$

Total density liquor phase

$$M_l = M_{ll} + M_{lr} \tag{D.19}$$

with

$$M_{ll} = \rho_{water} + \sum_{i=1}^{2} \rho_{l,i} \tag{D.20}$$

and

$$M_{lr} = \sum_{i=3}^{4} \rho_{l,i} \tag{D.21}$$

Kappa number:

$$\kappa\# = \frac{\rho_{r,1} + \rho_{r,2}}{0.00153 \sum_{i=1}^{3} \rho_{r,i}} \tag{D.22}$$

Yield:

$$\gamma = \frac{\sum_{i=1}^{3} \rho_{r,i,exiting}}{\sum_{i=1}^{3} \rho_{r,i,entering}} \tag{D.23}$$

D.3 Fuzzy models

The hybrid model of the pulp digester contains four fuzzy models:

$$R_{r,i} = f_{fuzzy}(\rho_{r,i}, \rho_{r,4}, T_r) \text{ for } i = 1 \ldots 3 \tag{D.24}$$

$$D_{cr} = f_{fuzzy}(T_r) \tag{D.25}$$

The models were identified with GK-clustering in combination with structure optimization. The weights of the rules were determined in the submodel integration step. Clustering settings and modeling results are shown in table D.2. Using the input-output data that was generated with the EPM, this results in the fuzzy models given below. The premise part membership functions are given in figures D.4 to D.7.

$$\text{IF } \rho_{r,1} = \text{mf}_i \text{ AND } \rho_{r,4} = \text{mf}_i \text{ AND } T_r = \text{mf}_i$$
$$\text{THEN } R_{r,1} = w_i^{R_{r,1}} y_i \text{ for } i = 1, \ldots, 4 \tag{D.26}$$

in which

$$y_i = p_{i,1}^{R_{r,1}} \rho_{r,1} + p_{i,2}^{R_{r,1}} \rho_{r,4} + p_{i,3}^{R_{r,1}} T_r + p_{i,4}^{R_{r,1}} \tag{D.27}$$

with

$$p^{R_{r,1}} = \begin{bmatrix} -0.0088 & -0.0011 & -0.0002 & 0.0933 \\ -0.0252 & -0.0019 & -0.0011 & 0.5478 \\ 0.0004 & -0.0059 & -0.0029 & 1.1060 \\ -0.0233 & -0.0065 & -0.0060 & 2.7791 \end{bmatrix} \quad w^{R_{r,1}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{D.28}$$

$$\text{IF } \rho_{r,2} = \text{mf}_i \text{ AND } \rho_{r,4} = \text{mf}_i \text{ AND } T_r = \text{mf}_i$$
$$\text{THEN } R_{r,2} = w_i^{R_{r,2}} y_i \text{ for } i = 1, \ldots, 6 \tag{D.29}$$

in which

$$y_i = p_{i,1}^{R_{r,2}} \rho_{r,2} + p_{i,2}^{R_{r,2}} \rho_{r,4} + p_{i,3}^{R_{r,2}} T_r + p_{i,4}^{R_{r,2}} \tag{D.30}$$

with

$$p^{R_{r,2}} = \begin{bmatrix} -0.0291 & -0.0154 & -0.0220 & 9.8500 \\ 0.0238 & -0.0022 & -0.0116 & 2.1300 \\ -0.0185 & -0.0367 & -0.1045 & 45.7436 \\ -0.0092 & -0.0085 & -0.0385 & 16.2819 \\ -0.0003 & -0.0042 & -0.0001 & 0.0000 \\ -0.0307 & -0.0235 & -0.0585 & 26.2124 \end{bmatrix} \quad w^{R_{r,2}} = \begin{bmatrix} 0.73 \\ 0.93 \\ 0.47 \\ 1.19 \\ 1.14 \\ 1.70 \end{bmatrix} \tag{D.31}$$

$$\text{IF } \rho_{r,3} = \text{mf}_i \text{ AND } \rho_{r,4} = \text{mf}_i \text{ AND } T_r = \text{mf}_i$$
$$\text{THEN } R_{r,3} = w_i^{R_{r,3}} y_i \text{ for } i = 1, \ldots, 6 \tag{D.32}$$

in which

$$y_i = p_{i,1}^{R_{r,3}} \rho_{r,3} + p_{i,2}^{R_{r,3}} \rho_{r,4} + p_{i,3}^{R_{r,3}} T_r + p_{i,4}^{R_{r,3}} \tag{D.33}$$

with

$$p^{R_{r,3}} = \begin{bmatrix} -0.0111 & -0.0245 & -0.0813 & 38.2907 \\ -0.0132 & -0.0364 & -0.0909 & 43.4828 \\ -0.0052 & -0.0227 & -0.0208 & 10.5408 \\ -0.0026 & -0.0114 & -0.0128 & 6.0335 \\ -0.0002 & -0.0041 & -0.0011 & 0.4765 \\ -0.0083 & -0.0154 & -0.0546 & 25.5889 \end{bmatrix} \quad w^{R_{r,3}} = \begin{bmatrix} 0.99 \\ 1.38 \\ 1.04 \\ 1.95 \\ 1.25 \\ 1.12 \end{bmatrix} \tag{D.34}$$

Fuzzy model	*Inputs*	γ_{dr}	*# data features*	k_0	γ_{cm}	*# rules*	$RMSE$
$R_{r,1}$	$\rho_{r,1}, \rho_{r,4}, T_r$	2	179	10	0.7	4	0.024
$R_{r,2}$	$\rho_{r,2}, \rho_{r,4}, T_r$	1	333	10	0.7	6	0.14
$R_{r,3}$	$\rho_{r,3}, \rho_{r,4}, T_r$	5	198	10	0.7	6	0.12
D_{cr}	T_r	1	36	5	0.2	2	0.026

Table D.2: *Fuzzy model identification settings and results*

$$\text{IF } T_r = \text{mf}_i \text{ THEN } D_{cr} = w_i^{D_{cr}} y_i \text{ for } i = 1, \ldots, 2 \tag{D.35}$$

in which

$$y_i = p_{i,1}^{D_{cr}} T_r + p_{i,2}^{D_{cr}} \tag{D.36}$$

with

$$p^{D_{cr}} = \begin{bmatrix} 0.0014 & -0.5092 \\ 0.0062 & -2.3676 \end{bmatrix} \; w^{D_{cr}} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{D.37}$$

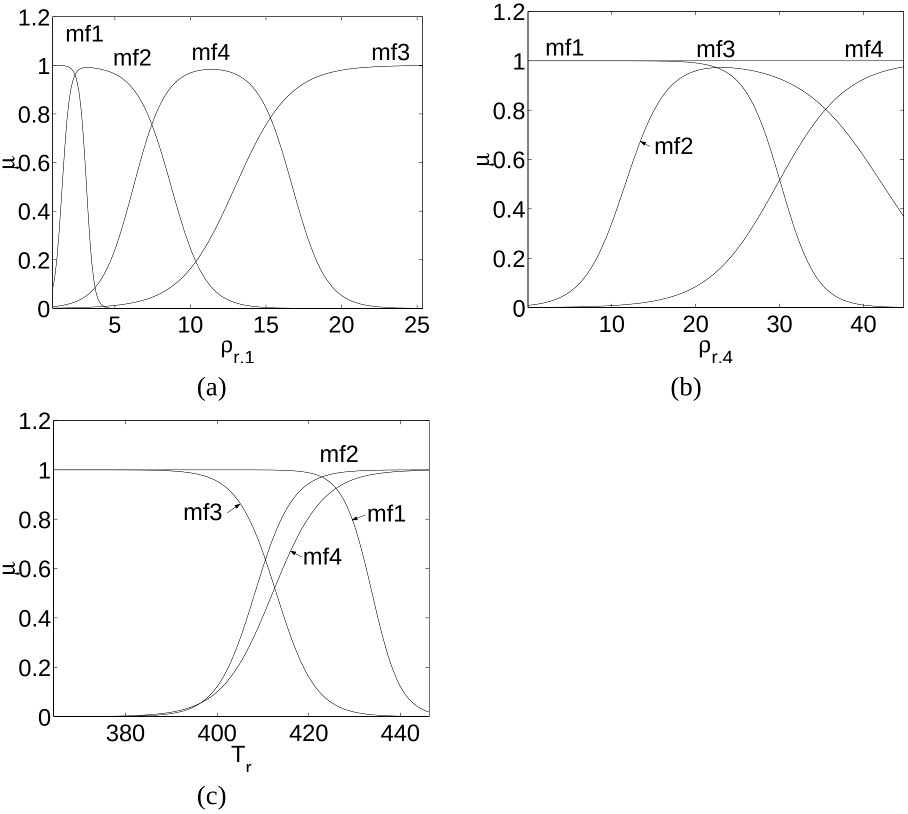
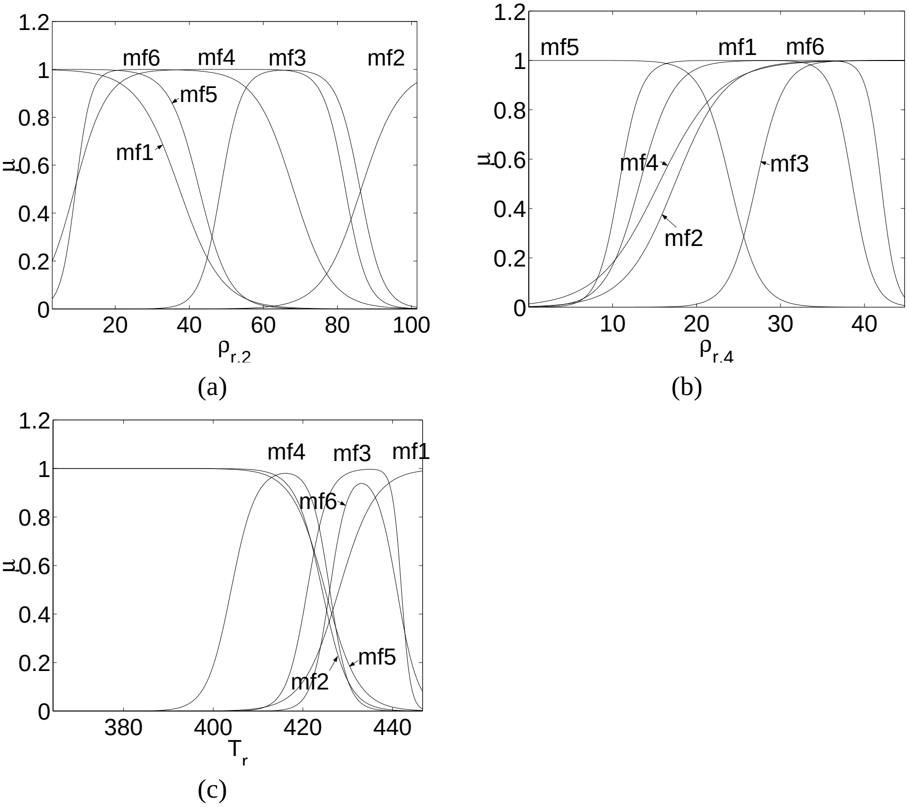


Figure D.4: *Premise part membership functions fuzzy model $R_{r,1}$*

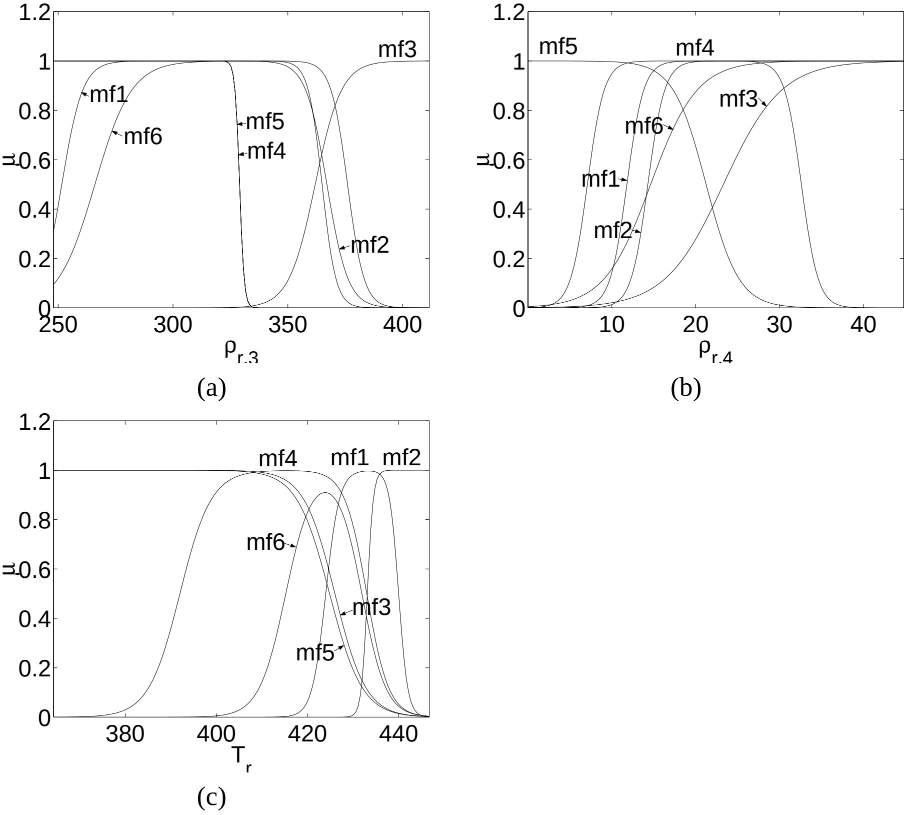


(a)

(b)

(c)

Figure D.5: *Premise part membership functions fuzzy model* $R_{r,2}$



(a)

(b)

(c)

Figure D.6: *Premise part membership functions fuzzy model* $R_{r,3}$

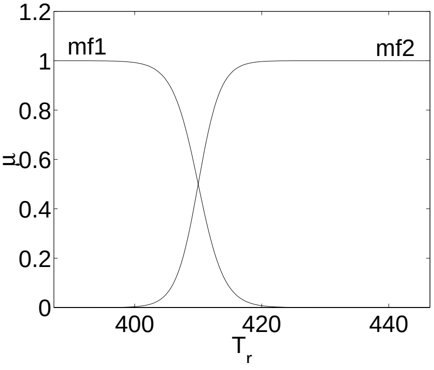


Figure D.7: *Premise part membership functions fuzzy model* D_{cr}

	Softwood	*Hardwood*
β_{OHL}	$.166\ kg$ OH$/kg$ lignin	$0.21\ kg$ OH$/kg$ lignin
β_{OHC}	$0.395\ kg$ OH$/kg$ carbohydrate	$0.49\ kg$ OH$/kg$ carbohydrate

Table D.3: *Stochiometric coefficients*

D.4 Heater model

In the hybrid model, the EPM upper and lower heater are lumped to form a ”preheater”. To make comparisons between the two models, the preheater has to provide the same amount of energy as the upper and lower heater do. The resulting outlet temperature depends on this energy, the composition of the flow and the flow rate. Therefore, in the hybrid model, the heaters are modeled using a simple static energy balance. The heater model can be described as follows:

$$\phi_{out} = \phi_{in} \tag{D.38}$$

$$Q_h = \phi_{in} Cp_{in} M_{in} (T_{out} - T_{in}) \tag{D.39}$$

$$M_{in} = M_{l,in} + M_{r,in} \tag{D.40}$$

in which

$$M_{l,in} = \rho_{water} + \sum_{i=1}^{2} \rho_{l,i,in} \tag{D.41}$$

and

$$M_{r,in} = \sum_{i=3}^{4} \rho_{l,i,in} \tag{D.42}$$

$$Cp_{in} = Cp_w \frac{M_{l,in}}{M_{in}} + Cp_s \frac{M_{r,in}}{M_{in}} \tag{D.43}$$

D.5 Typical operating values and parameters

The stochiometric matrix b which links the reaction rates of the reaction phase components to the reaction rates of the liquor phase components can be derived from equation C.35. Lumping species $(s,3)$ to $(s,5)$ corresponds with removing columns 4 and 5 from b and lumping species $(e,1)$ and $(e,3)$ corresponds with summing rows 1 and 3 of b. For the reaction products, rows 2 and 3 also have to be summed. This results in:

$$b = \begin{bmatrix} \beta_{OHL} & \beta_{OHL} & \beta_{OHC} \\ -\beta_{OHL} & -\beta_{OHL} & -\beta_{OHC} \\ -1 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{D.44}$$

The stochiometric coefficients for the consumption of Effective Alkali and Hydrosulfide are stated in table D.3. Tables D.4 through D.5 show various other model parameters.

Parameter	*Value*		*Definition*
ϕ_r	1.3964	m^3/min	reaction phase (woodchip) flow rate
$\phi_{l,in}$	2.3497	m^3/min	entering white liquor flow rate
Q_h	627099	kJ/min	Preheater energy input
ϕ_{quench}	0	m^3/min	quench recirculation flow rate
T_{wash}	365.133	K	wash heater temperature
ϕ_{filt}	1.6312	m^3/min	entering filtrate flow rate
T_{filt}	360.93	K	entering filtrate temperature
$T_{r,in}$	395	K	entering woodchip temperature
$T_{l,in}$	382	K	entering white liquor temperature
$\rho_{r,1,in}$	25.37	kg/m^3	high reactivity lignin density of entering woodchip (reaction phase)
$\rho_{r,2,in}$	101.49	kg/m^3	low reactivity lignin density of entering woodchip (reaction phase)
$\rho_{r,3,in}$	411.56	kg/m^3	carbohydrates density of entering woodchip
$\rho_{r,4,in}$	0.02	kg/m^3	active liquor concentration in entering reaction phase
$\rho_{r,5,in}$	0.02	kg/m^3	passive liquor concentration in entering reaction phase
$\rho_{r,6,in}$	0.01	kg/m^3	dissolved lignin concentration in entering reaction phase
$\rho_{r,7,in}$	0.01	kg/m^3	dissolved carbohydrates concentration in entering reaction phase
$\rho_{l,1,in}$	92.1	kg/m^3	active liquor concentration of entering white liquor flow
$\rho_{l,2,in}$	0	kg/m^3	passive liquor concentration of entering white liquor flow
$\rho_{l,3,in}$	59.6	kg/m^3	dissolved lignin concentration of entering white liquor flow
$\rho_{l,4,in}$	59.6	kg/m^3	dissolved carbohydrates concentration of entering white liquor flow
$\rho_{f,1,filtr}$	4.8055	kg/m^3	active liquor concentration of entering filtrate flow
$\rho_{f,2,filtr}$	0.02	kg/m^3	passive liquor concentration of entering filtrate flow
$\rho_{f,5,filtr}$	45	kg/m^3	dissolved lignin concentration of entering filtrate flow
$\rho_{f,6,filtr}$	45	kg/m^3	dissolved carbohydrates concentration of entering filtrate flow

Table D.4: *Standard operating parameters*

Parameter	*Value*		*Definition*
$\bar{\rho}_s$	1666	kg/m^3	density of solid wood material
ρ_{water}	999	kg/m^3	density of water
ΔH_R	-518	kJ/kg	heat of reaction
Cp_w	4.19	$kJ/(kg\ K)$	heat capacity liquor
Cp_s	1.47	$kJ/(kg\ K)$	heat capacity wood substance
U	827	$kJ/(min\ m^3\ K)$	heat transfer coefficient

Table D.5: *Digester unit operation parameters*

Model	*Input*	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$ $(min$	$\tau_{[input],\gamma}$ $(min$	$\theta_{[input]}$ (min)	$\Delta\kappa\#$ $(-)$	$\Delta\gamma$ $(-)$
EPM	Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
	ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020
Hybrid	Q_{lh}	1.8	1.9	40	45	84	-5.77, 6.40	-0.018, 0.018
model	ϕ_f	2.4	2.7	34	34	109	4.23, -3.69	0.014, -0.013

Table D.6: *Frequency response analysis results for hybrid model*

The implemented compaction profile is based upon (Wisnewski *et al.*, 1997), who uses different, fixed compaction values for each CSTR. In the PFR approach for the hybrid model, compaction is neglected. The ratio of the volume of the liquor phase and the volume of the reactor η is assumed to be constant. This means that the volumes of the reaction phase and the liquor phase do not change.

$$\eta = 0.54 \tag{D.45}$$

D.6 Frequency analysis

Figures D.8-D.11 show the Bode diagrams for the Kappa number and the yield as a function of the lower heater heat output Q_{lh} and the liquor flow rate ϕ_f. Table D.6 shows the frequency analysis results; the gains $\Delta\kappa\#$ and $\Delta\gamma$ are the gains for steps in Q_{lh} and ϕ_l. The steps size of the steps in ϕ_l was 7 %. The step size of the steps in Q_h was equivalent with a 10 % change in lower heater heat output Q_{lh} of the EPM. The gains for a positive and a negative step are presented. In the table, O denotes the order of the system response, τ the time constant and θ the dead time.

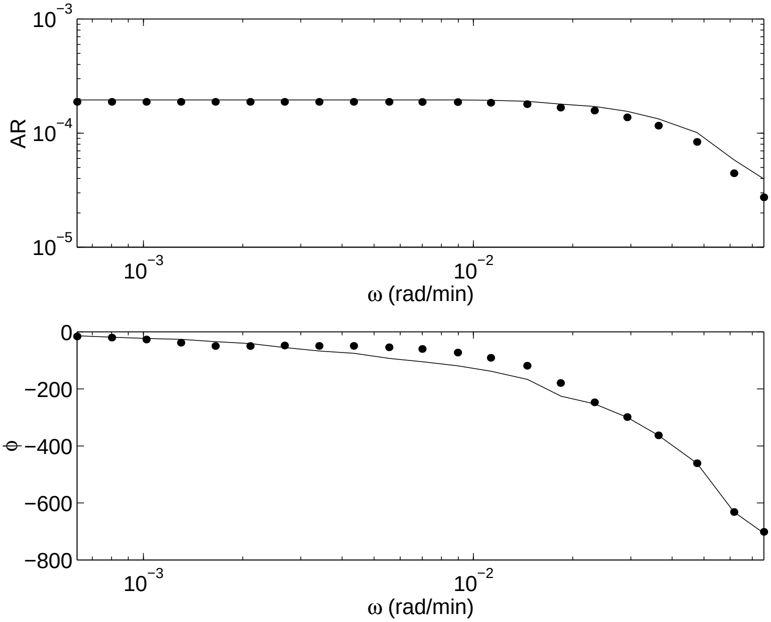


Figure D.8: *Bode diagram $\kappa\#$ as a function of Q_{lh}. Dots EPM, line hybrid model.*

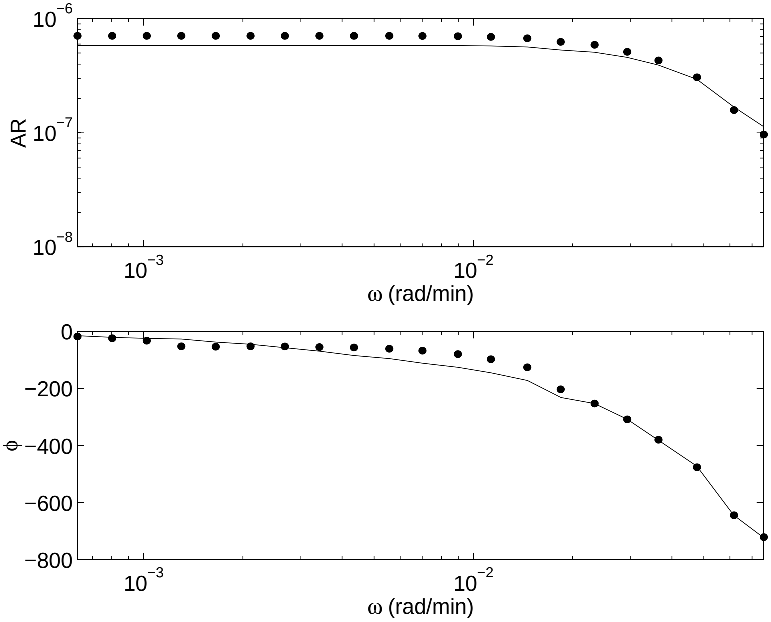


Figure D.9: *Bode diagram γ as a function of Q_{lh}. Dots EPM, line hybrid model.*

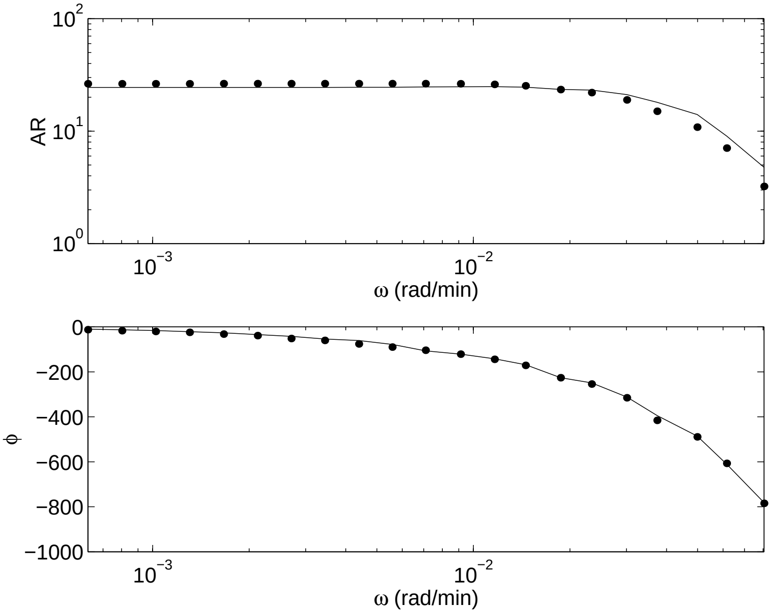


Figure D.10: *Bode diagram $\kappa\#$ as a function of ϕ_f. Dots EPM, line hybrid model.*

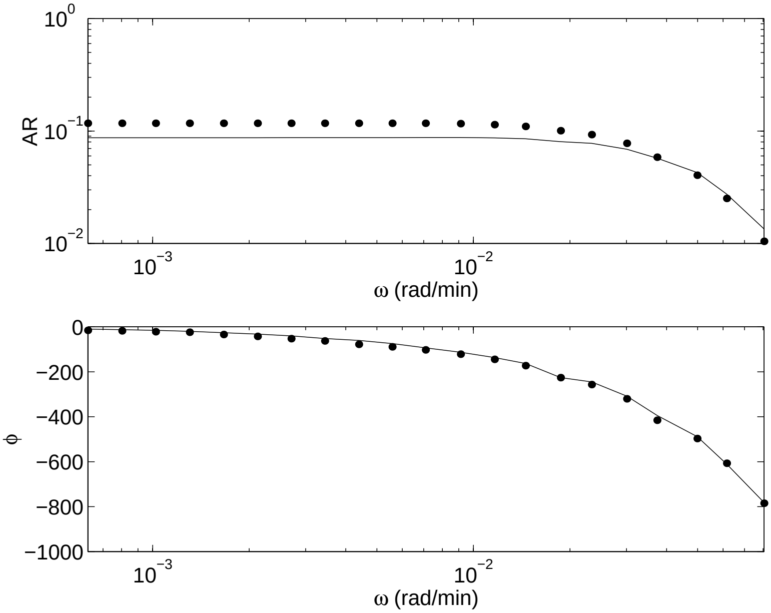


Figure D.11: *Bode diagram γ as a function of ϕ_f. Dots EPM, line hybrid model.*

D.7 Notation

Cp_l	Heat capacity, liquor phase	$(kJ/kg\ K)$
Cp_r	Heat capacity, reaction phase	$(kJ/kg\ K)$
Cp_w	Heat capacity, liquor	$(kJ/kg\ K)$
D_R	Energy transported into wood chip	(kJ/m^3)
D_L	Energy transported into liquor phase	(kJ/m^3)
D_{cr}	Diffusion coefficient, reaction phase	(min^{-1})
D_{cl}	Diffusion coefficient, liquor phase	(min^{-1})
M_{rl}	Mass liquor components, reaction phase	(kg/m^3)
M_{rr}	Mass reaction phase components, reaction phase	(kg/m^3)
M_l	Mass liquor phase	(kg/m^3)
M_{ll}	Mass liquor components, liquor phase	(kg/m^3)
M_{lr}	Mass reaction phase components, liquor phase	(kg/m^3)
M_{in}	Entering mass, heater	(kg/m^3)
$M_{l,in}$	Entering mass, liquor components, heater	(m^3/min)
M_r	Mass reaction phase	(kg/m^3)
$M_{r,in}$	Entering mass, reaction phase components, heater	(m^3/min)
Q_h	Heat supply heater	(kJ/min)
$R_{r,i}$	Reaction rate species (r,i)	(m^3/min)
S	Cross sectional area	(m^2)
T_r	Reaction phase temperature	(K)
T_l	Liquor phase temperature	(K)
T_{in}	Temperature entering flow rate, heater	(K)
T_{out}	Temperature exiting flow rate, heater	(K)
T_p	Temperature used in heat transfer calculation	(K)
U	Heat transfer coefficient	$(kJ/min\ m^3\ K)$
$\boldsymbol{b}$	Reaction stochiometry	$(-)$
z	Location in the reactor, top = 0	(m)
$p^{[*]}$	Consequent part parameters fuzzy models	
$w^{[*]}$	Rule weights fuzzy models	
ΔH_R	Reaction enthalpy	(kJ/kg)
β_*	Stochiometric coefficients	$(-)$
γ	Yield	$(-)$
ε	Porosity	$(-)$
η	Ratio liquor volume - reactor volume	$(-)$
θ_Q	Dead time with respect to Q_h	(min)
θ_ϕ	Dead time with respect to ϕ_l	(min)
$\kappa\#$	Kappa number	$(-)$
$\rho_{l,i}$	Concentration species i, liquor phase	(kg/m^3)
$\rho_{r,i}$	Concentration species i, reaction phase	(kg/m^3)
$\bar{\rho}_s$	Density of solid material	(kg/m^3)
ρ_{water}	Density of water	(kg/m^3)
$\tau_{[input],[output]}$	Time constant for $[output]$ with respect to $[input]$	(min)
ϕ_r	Reaction phase flow rate	(m^3/min)
ϕ_{in}	Entering flow rate, heater	(m^3/min)
ϕ_l	Liquor phase flow rate	(m^3/min)
ϕ_{out}	Exiting flow rate, heater	(m^3/min)

E

Pulp Digester Fuzzy Model

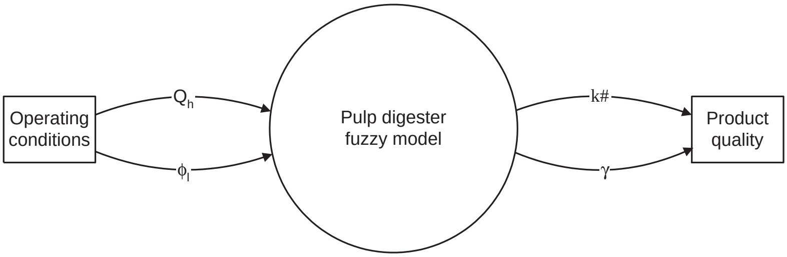


Figure E.1: *Hybrid model section context diagram*

This appendix describes the dynamic fuzzy model of the pulp digester. The fuzzy models, identification and validation data and the frequency analysis are presented.

E.1 Fuzzy models

The fuzzy model of the pulp digester describes the Kappa number ($\kappa\#$) and the yield (γ) at the reactor outlet as a function of the heat supply Q_h and the liquor flow ϕ_l by using two second order MISO models. Figure E.1 shows the context diagram of the complete model. Since the fuzzy model of the digester only consists of two fuzzy equations, no DFD was constructed.

The model equation structure is as follows:

$$\kappa\#_k = f_{fuzzy}(\kappa\#_{k-1}, \kappa\#_{k-2}, Q_{k-\theta_Q}, \phi_{l,k-\theta_l}) \tag{E.1}$$

$$\gamma_k = f_{fuzzy}(\gamma_{k-1}, \gamma_{k-2}, Q_{k-\theta_Q}, \phi_{l,k-\theta_l}) \tag{E.2}$$

in which $_k$ denotes the time step and θ_Q and θ_l denote the dead times with respect to Q_h and ϕ_l.

The models were identified with GK-clustering in combination with structure optimization. Clustering settings and model results are shown in table E.1. This results in the following fuzzy models:

$$\begin{aligned}\text{IF } \kappa\#_{k-1} = \text{mf}_i \text{ AND } \kappa\#_{k-2} = \text{mf}_i \text{ AND } Q_{h,k-\theta_Q} = \text{mf}_i \\ \text{AND } \phi_{l,k-\theta_\phi} = \text{mf}_i \text{ THEN } \kappa\#_k = y_i \text{ for } i = 1, \ldots, 3\end{aligned} \tag{E.3}$$

in which

$$y_i = p^\kappa_{i,1}\kappa\#_{k-1} + p^\kappa_{i,2}\kappa\#_{k-2} + p^\kappa_{i,3}Q_{h,k-\theta_Q} + p^\kappa_{i,4}\phi_{l,k-\theta_\phi} + p^\kappa_{i,5} \tag{E.4}$$

with

$$p^\kappa = \begin{bmatrix} 1.98 & -0.98 & -0.0046 & 0.0070 & 0.0230 \\ 1.98 & -0.98 & -0.0058 & 0.0067 & 0.0293 \\ 1.98 & -0.98 & -0.0068 & 0.0098 & 0.0303 \end{bmatrix} \tag{E.5}$$

Model	$\theta_Q\ (min)$	$\theta_\phi\ (min)$	k_0	γ_{cm}	*# rules*
$\kappa\#$	87	113	6	0.7	3
γ	87	113	6	0.4	5

Table E.1: *Settings and structure results fuzzy model*

and

$$\text{IF } \gamma_{k-1} = \text{mf}_i \text{ AND } \gamma_{k-2} = \text{mf}_i \text{ AND } Q_{h,k-\theta_Q} = \text{mf}_i$$
$$\text{AND } \phi_{l,k-\theta_\phi} = \text{mf}_i \text{ THEN } \gamma_k = y_i \text{ for } i = 1, \ldots, 5 \qquad \text{(E.6)}$$

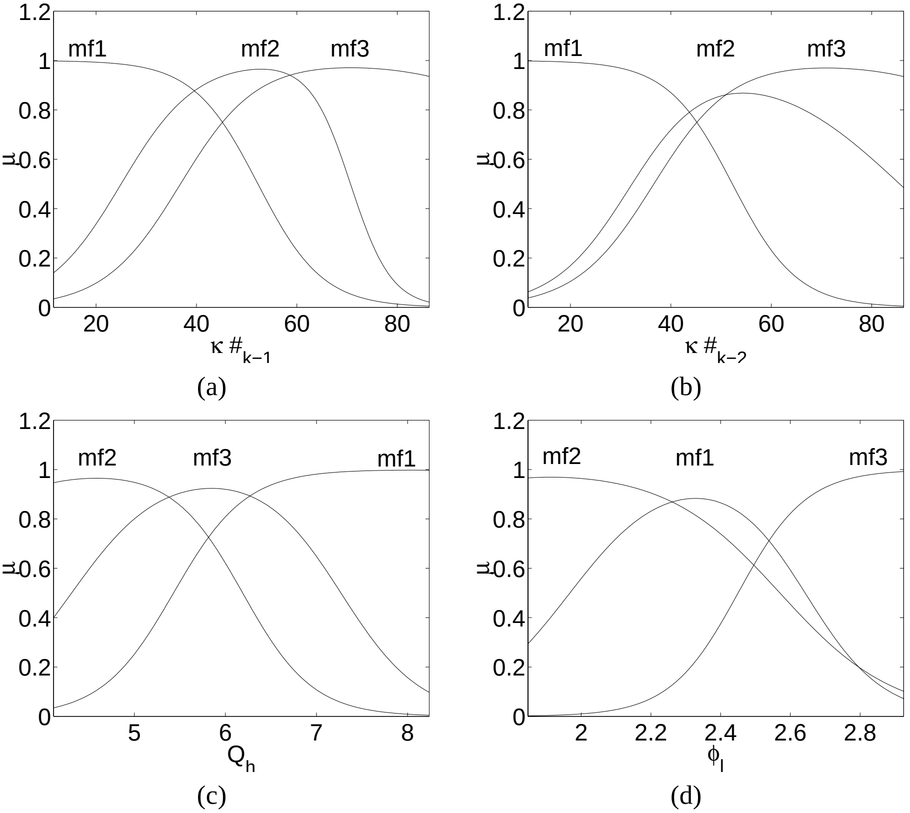
in which

$$y_i = p^\gamma_{i,1}\kappa\#_{k-1} + p^\gamma_{i,2}\kappa\#_{k-2} + p^\gamma_{i,3}Q_{h,k-\theta_Q} + p^\gamma_{i,4}\phi_{l,k-\theta_\phi} + p^\gamma_{i,5} \qquad \text{(E.7)}$$

with

$$p^\gamma = \begin{bmatrix} 1.98 & -0.98 & -0.000016 & 0.000020 & 0.000192 \\ 1.98 & -0.97 & -0.000017 & 0.000026 & 0.000206 \\ 1.98 & -0.97 & -0.000017 & 0.000023 & 0.000236 \\ 1.98 & -0.97 & -0.000026 & 0.000037 & 0.000342 \\ 1.98 & -0.98 & -0.000016 & 0.000034 & 0.000190 \end{bmatrix} \qquad \text{(E.8)}$$

The membership functions for the two fuzzy MISO models are presented in figures E.2 and E.2



(a) (b) (c) (d)

Figure E.2: *Premise part membership functions MISO model $\kappa\#$*

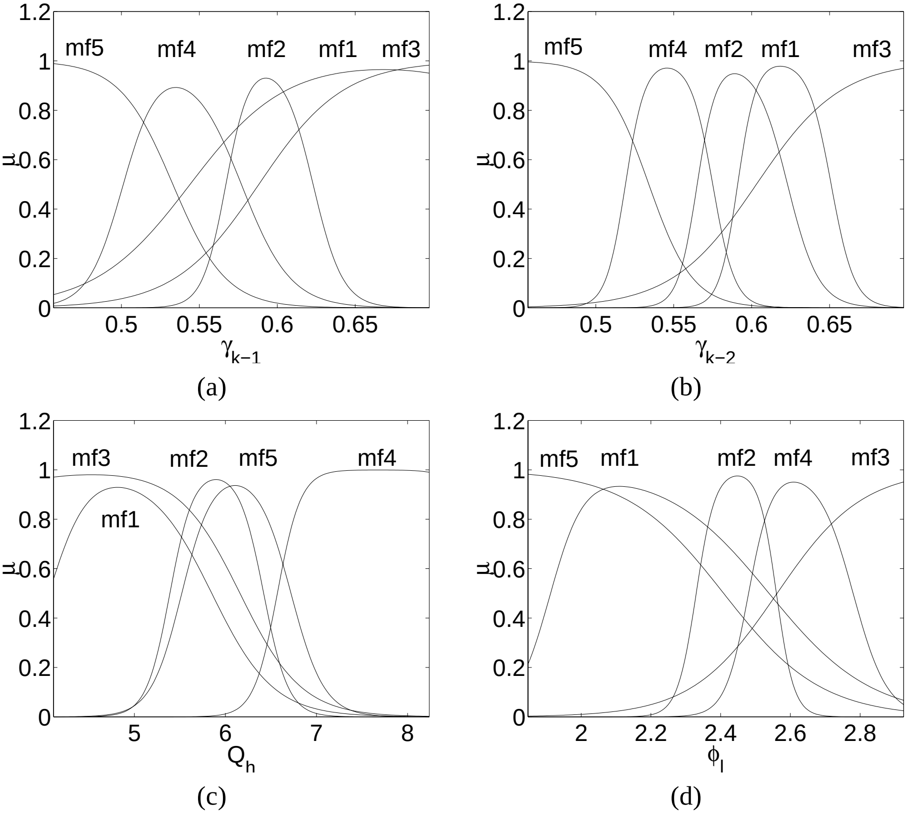
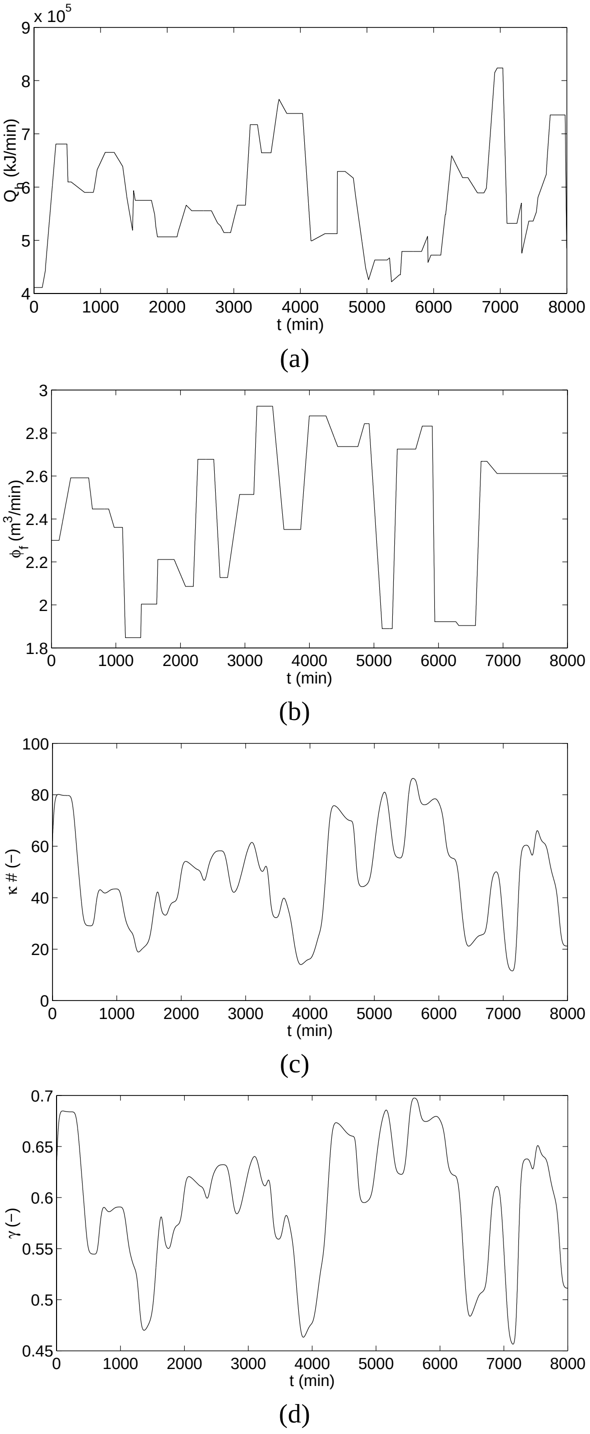


Figure E.3: *Premise part membership functions MISO model* γ

E.2 Identification and validation data sets

The identification and validation data sets were generated with the EPM using ramped-RMRI signals for Q_h (which consisted of the summed ramped-RMRI signals that for the lower heater heat output Q_{lh} and the upper heater heat output Q_{uh}) and the liquor flow rate ϕ_f. The identification data set is shown in figure E.4, the validation data set is shown in figure E.5. The value of Q_h is divided by a factor of $1e5$ to prevent membership function evaluation problems.



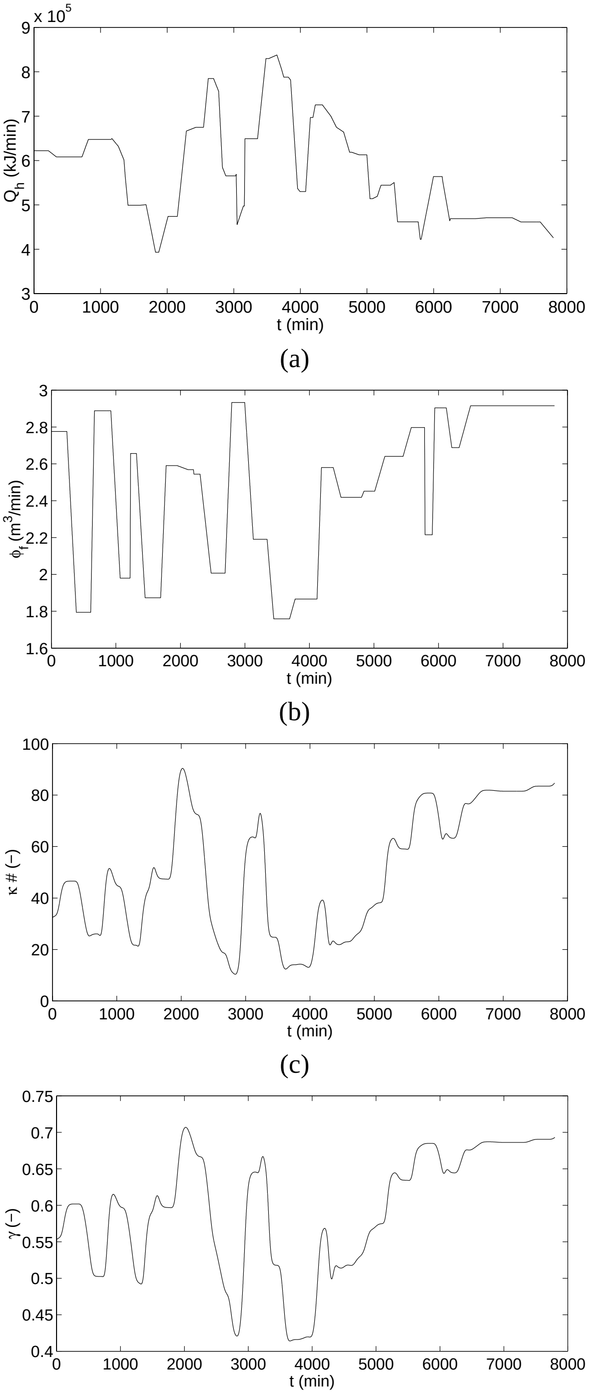
(a)

(b)

(c)

(d)

Figure E.4: *Identification data for Q_h (a), ϕ_l (b), $\kappa\#$ (c) and γ (d)*



(a)

(b)

(c)

(d)

Figure E.5: *Validation data for Q_h (a), ϕ_l (b), $\kappa\#$ (c) and γ (d)*

Model	*Input*	$O_{[input],\kappa}$	$O_{[input],\gamma}$	$\tau_{[input],\kappa}$ (min)	$\tau_{[input],\gamma}$ (min)	$\theta_{[input]}$ (min)	$\Delta\kappa\#$ $(-)$	$\Delta\gamma$ $(-)$
EPM	Q_{lh}	2.3	2.3	42	43	87	-5.51, 6.17	-0.022, 0.022
	ϕ_f	2.9	3.3	35	35	113	4.71, -4.06	0.019, -0.020
Fuzzy	Q_h	2.1	2.0	52	58	87	-4.54, 4.98	-0.018, 0.022
model	ϕ_l	2.0	2.0	50	50	113	3.70, -3.68	0.019, -0.020

Table E.2: *Frequency response analysis results for EPM*

E.3 Frequency analysis

Figures E.6-E.9 show the Bode diagrams for the Kappa number and the yield as a function of the lower heater heat output Q_{lh} and the liquor flow rate ϕ_f. Table E.2 shows the frequency analysis results; the gains $\Delta\kappa\#$ and $\Delta\gamma$ are the gains for steps in Q_{lh} and ϕ_l. The steps size of the steps in ϕ_l was 7 %. The step size of the steps in Q_h was equivalent with a 10 % change in lower heater heat output Q_{lh} of the EPM. The gains for a positive and a negative step are presented. In the table, O denotes the order of the system response, τ the time constant and θ the dead time.

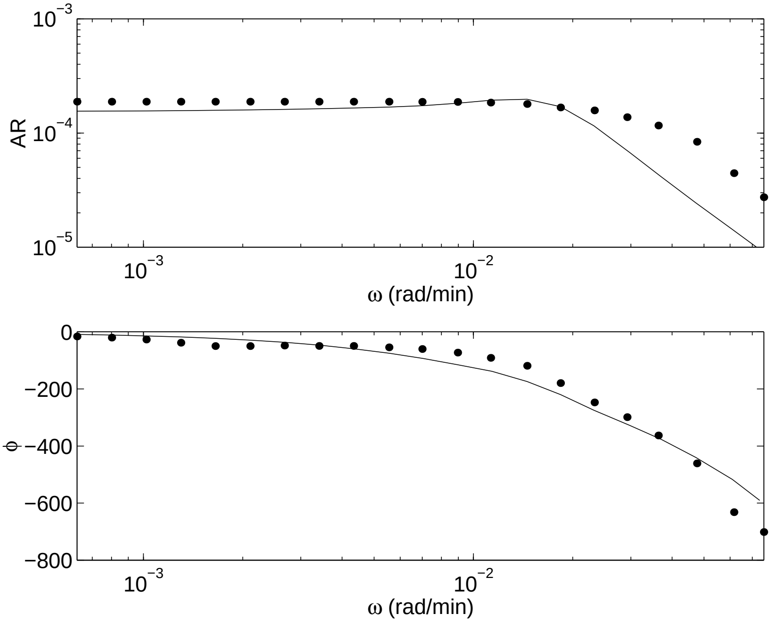


Figure E.6: *Bode diagram $\kappa\#$ as a function of Q_{lh}. Dots EPM, line fuzzy model.*

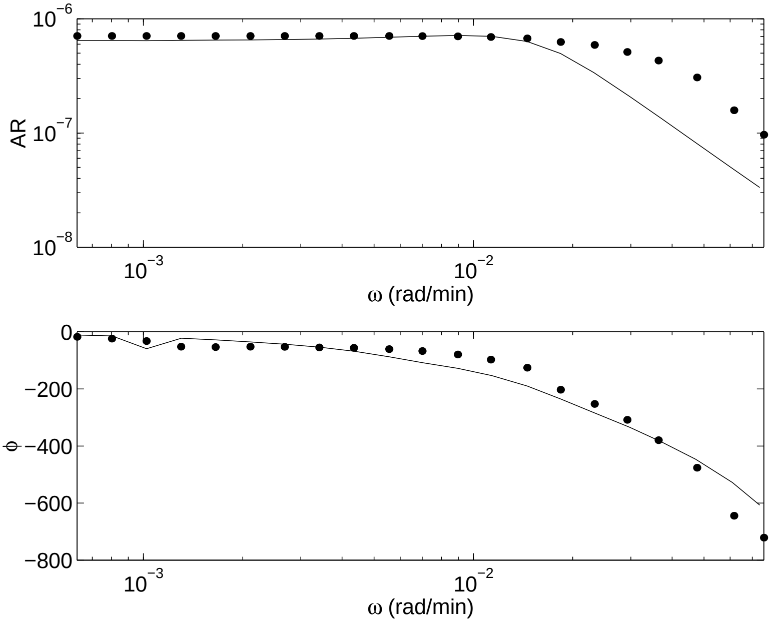


Figure E.7: *Bode diagram γ as a function of Q_{lh}. Dots EPM, line fuzzy model.*

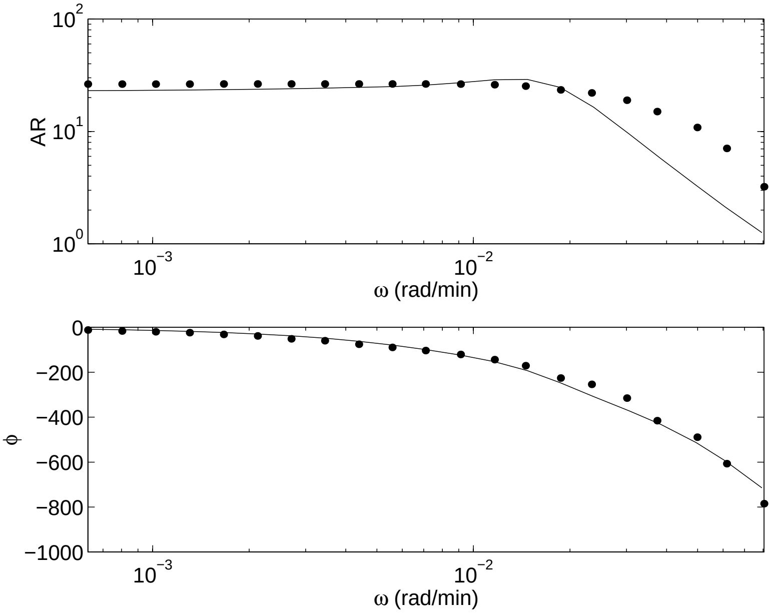


Figure E.8: *Bode diagram $\kappa\#$ as a function of ϕ_f. Dots EPM, line fuzzy model.*

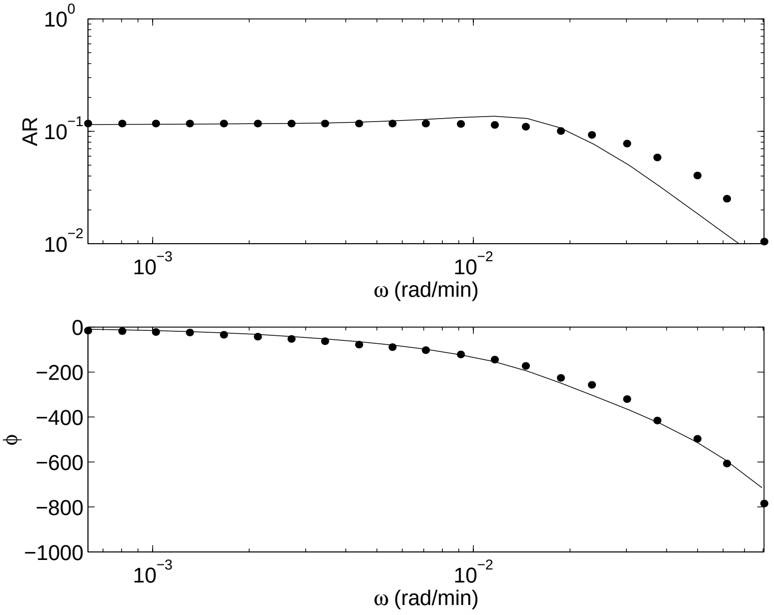


Figure E.9: *Bode diagram γ as a function of ϕ_f. Dots EPM, line fuzzy model.*

E.4 Notation

Q_h	Heat supply heater	(kJ/min)
$p^{[*]}$	Consequent part parameters fuzzy models	
$w^{[*]}$	Rule weights fuzzy models	
γ	Yield	$(-)$
θ_Q	Dead time with respect to Q_h	(min)
θ_ϕ	Dead time with respect to ϕ_l	(min)
$\kappa\#$	Kappa number	$(-)$
$\tau_{[input],[output]}$	Time constant for $[output]$ with respect to $[input]$	(min)
ϕ_l	Liquor phase flow rate	(m^3/min)

F

Distillation column hybrid models

Parameter	Value		Definition
τ_V	0.2	h	Column heating time constant
$L_{V,0}$	120	mol/h	Initial vapor flow
$L_{V,ss}$	166	mol/h	Steady state vapor flow approximation
$x^0_{n+1,sp}$	0.986	$-$	Product quality for which the model is valid

Table F.1: *Simplified overall static hybrid model parameters*

This appendix describes the simplified overall static and dynamic hybrid models. The data flow diagram and model parameters are given. In addition, the performance of the fuzzy submodel is presented.

F.1 Simplified overall static model

F.1.1 Model equations

The simplified overall static hybrid model is given by the following set of equations:

$$\frac{dM_{col}}{dt} = -L_V(1 - R^*) \tag{F.1}$$

$$M_{col}\frac{dx_{col}}{dt} = -L_V(1 - R^*)(x^0_{n+1} - x_{col}) \tag{F.2}$$

$$\frac{dL_V}{dt} = \frac{1}{\tau_V}(L_{V,ss} - L_V) \tag{F.3}$$

$$x^0_{n+1} = f_{fuzzy}(R^*, x_{col}, L_V) \tag{F.4}$$

$$M_p = M_{col,0} - M_{col} \tag{F.5}$$

Model parameters are given in table F.1. Figure F.1 shows the DFD of the model.

F.1.2 Fuzzy submodel

The premise part membership functions of the fuzzy submodel for x^0_{n+1} are given in figure F.2. The model can be represented as follows:

IF $R^* = \mathrm{mf}_i$ AND $x_{col} = \mathrm{mf}_i$ AND $L_V = \mathrm{mf}_i$

THEN $x^0_{n+1} = w_i^{x^0_{n+1}} y_i$ for $i = 1, \ldots, 3$ (F.6)

in which

$$y_i = p_{i,1}^{x^0_{n+1}} R^* + p_{i,2}^{x^0_{n+1}} x_{col} + p_{i,3}^{x^0_{n+1}} L_V + p_{i,4}^{x^0_{n+1}} \tag{F.7}$$

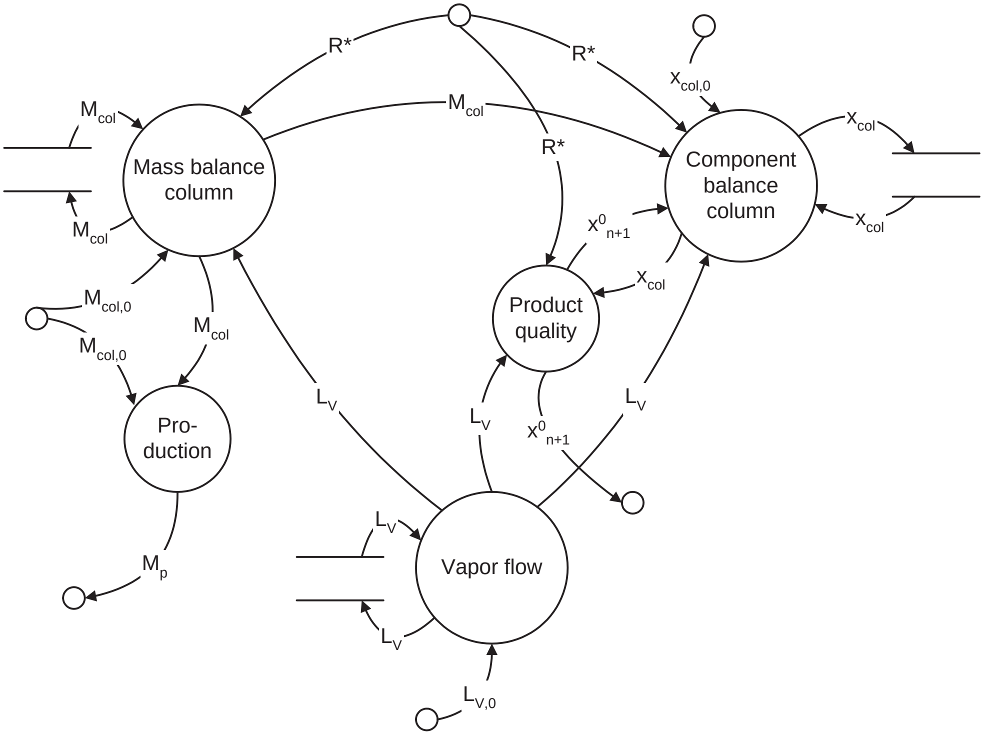
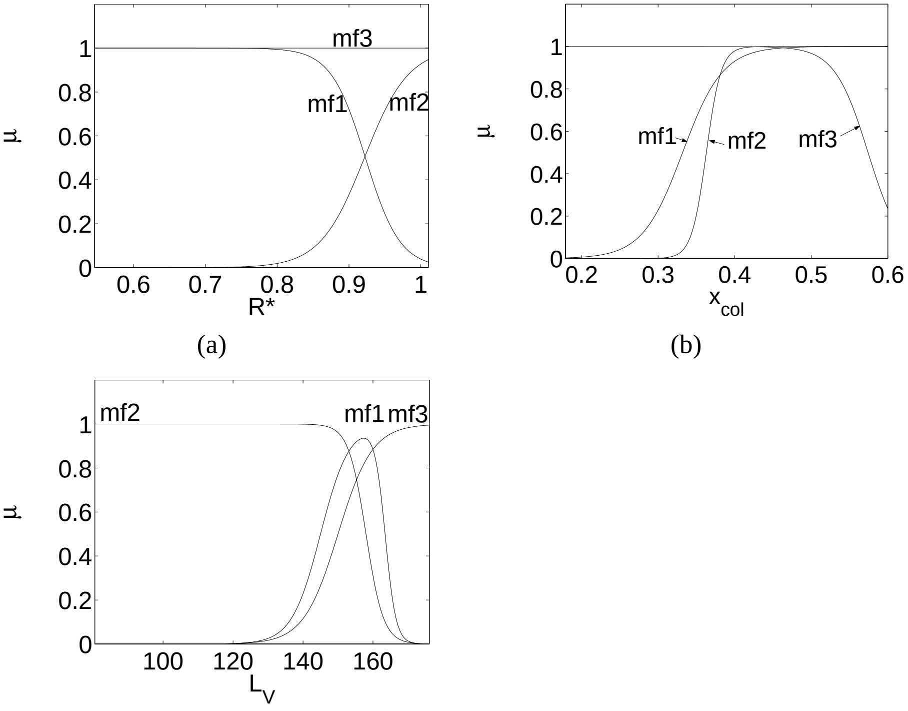


Figure F.1: *Simplified overall static model Data Flow Diagram*



(a)

(b)

(c)

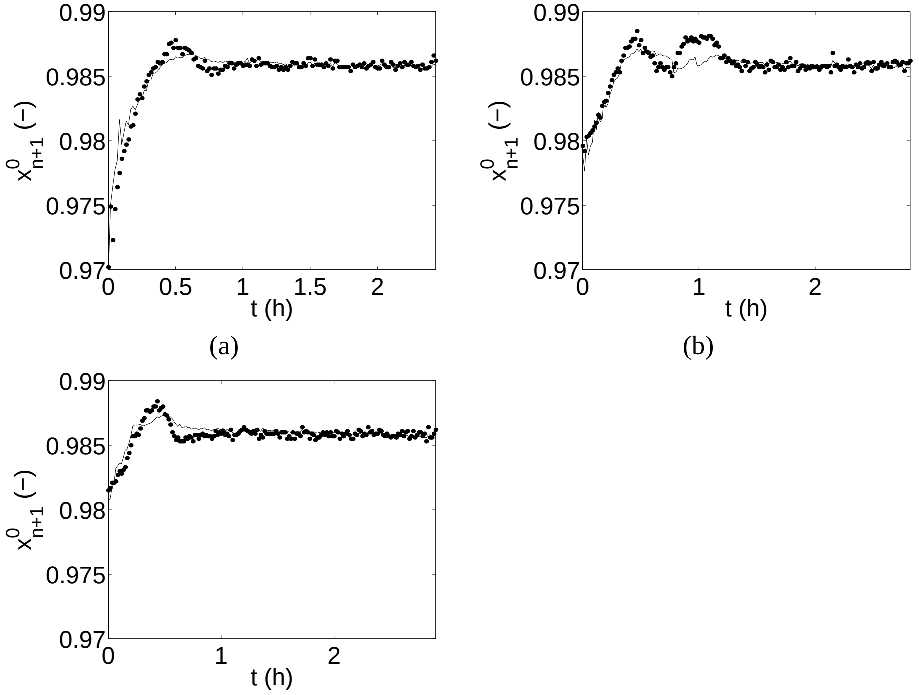
Figure F.2: *Premise part membership functions fuzzy model* x^0_{n+1}

with

$$p^{x^0_{n+1}} = \begin{bmatrix} -0.0034 & 0.0023 & -0.0000 & 0.9914 \\ -0.0315 & 0.0083 & 0.0002 & 0.9824 \\ -0.0056 & -0.0022 & -0.0000 & 0.9925 \end{bmatrix} \; w^{x^0_{n+1}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{F.8}$$

F.1.3 Fuzzy model performance

The performance of the fuzzy submodel for x^0_{n+1} with respect to the identification data is given in figure F.3.



(a)

(b)

(c)

Figure F.3: *Performance fuzzy model x^0_{n+1} with respect to identification data ID1 (a), ID2 (b) and ID3 (c)*

Parameter	*Value*		*Definition*
τ_V	0.2	h	Column heating time constant
$L_{V,0}$	120	mol/h	Initial vapor flow
$L_{V,ss}$	166	mol/h	Steady state vapor flow approximation
$x^0_{n+1,sp}$	0.986	$-$	Product quality for which the model is valid
τ_x	0.42	h	Dominant column time constant

Table F.2: *Simplified overall dynamic hybrid model parameters*

F.2 Simplified overall dynamic model

F.2.1 Model equations

The simplified overall dynamic hybrid model is given by the following set of equations:

$$\frac{dM_{col}}{dt} = -L_V(1 - R^*) \tag{F.9}$$

$$M_{col}\frac{dx_{col}}{dt} = -L_V(1 - R^*)(x^0_{n+1} - x_{col}) \tag{F.10}$$

$$\frac{dL_V}{dt} = \frac{1}{\tau_V}(L_{V,ss} - L_V) \tag{F.11}$$

$$\frac{dx^0_{n+1}}{dt} = \frac{1}{\tau_x}(x^* - x^0_{n+1}) \tag{F.12}$$

$$x^* = f_{fuzzy}(R^*, x_{col}, L_V) \tag{F.13}$$

$$M_p = M_{col,0} - M_{col} \tag{F.14}$$

Model parameters are given in table F.2. Figure F.4 shows the DFD of the model.

F.2.2 Fuzzy submodel

The premise part membership functions of the fuzzy submodel for x^* are given in figure F.5. The model can be represented as follows:

$$\text{IF } R^* = \text{mf}_i \text{ AND } x_{col} = \text{mf}_i \text{ AND } L_V = \text{mf}_i$$

$$\text{THEN } x^* = w_i^{x^0_{n+1}} y_i \text{ for } i = 1, \ldots, 3 \tag{F.15}$$

in which

$$y_i = p^{x^*}_{i,1}R^* + p^{x^*}_{i,2}x_{col} + p^{x^*}_{i,3}L_V + p^{x^*}_{i,4} \tag{F.16}$$

with

$$p^{x^*} = \begin{bmatrix} 0.0250 & 0.0182 & -0.0001 & 0.9767 \\ 0.0582 & -0.0477 & 0.0003 & 0.9151 \\ 0.0194 & 0.0147 & 0.0000 & 0.9613 \end{bmatrix} \quad w^{x^*} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{F.17}$$

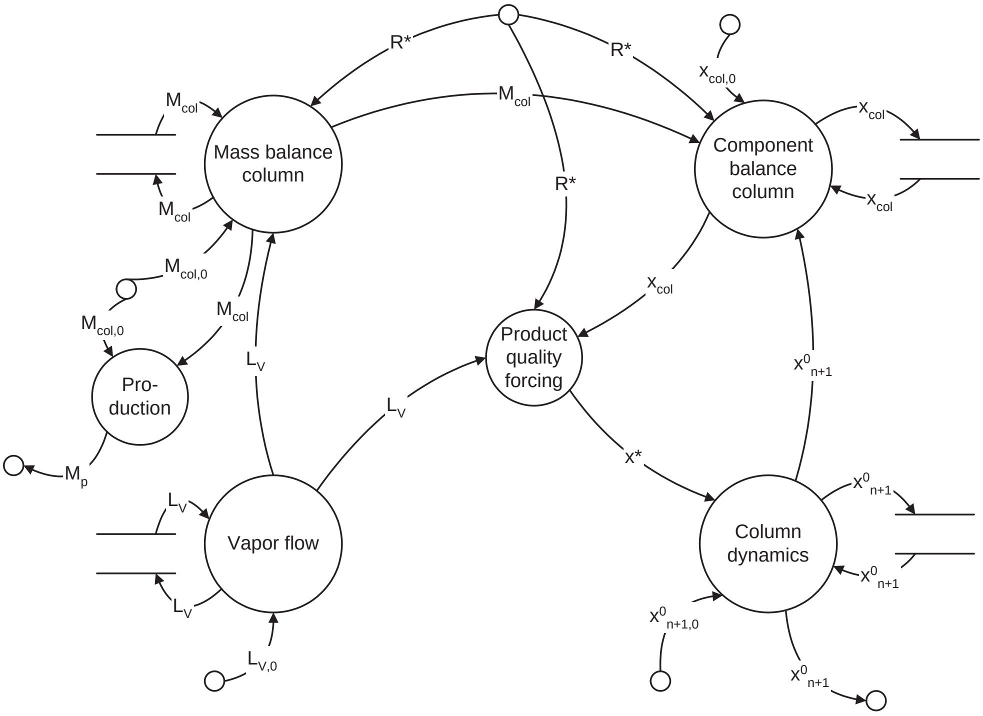
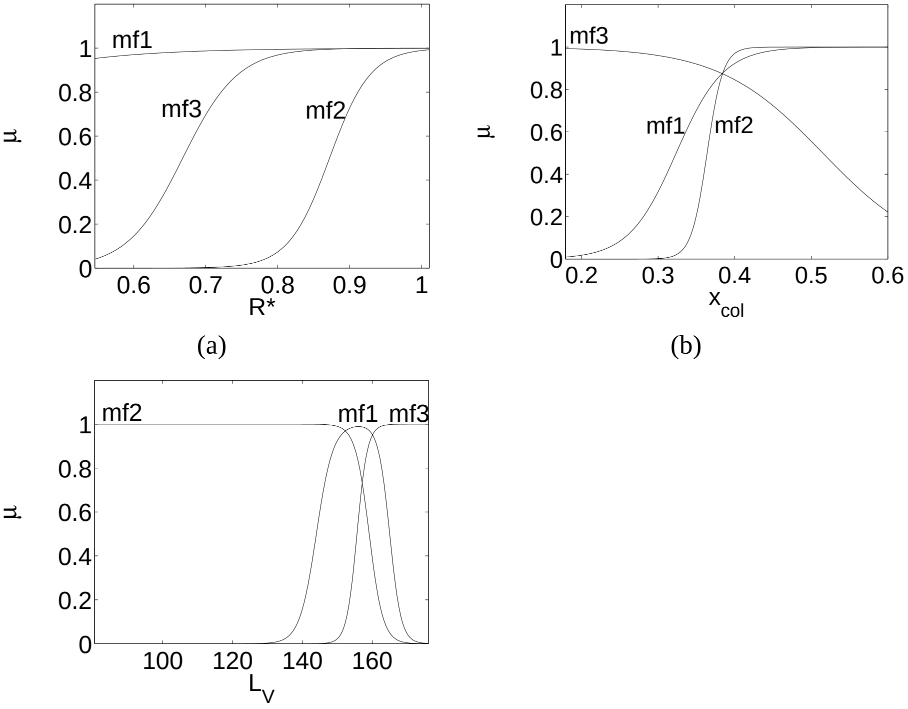


Figure F.4: *Simplified overall dynamic model Data Flow Diagram*

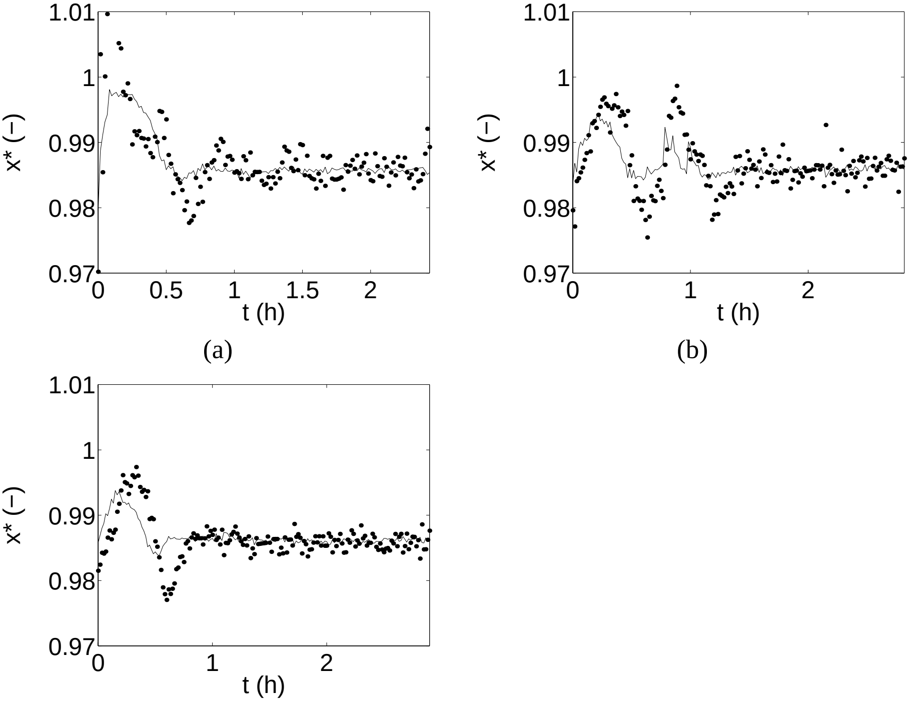


(a)

(b)

(c)

Figure F.5: *Premise part membership functions fuzzy model* x^*



(a)

(b)

(c)

Figure F.6: *Performance fuzzy model x^* with respect to identification data ID1 (a), ID2 (b) and ID3 (c)*

F.2.3 Fuzzy model performance

The performance of the fuzzy submodel for x^* with respect to the identification data is given in figure F.6.

F.3 Notation

L_V	Vapor flow after condensation	(mol/h)
$L_{V,ss}$	Pseudo steady state vapor flow	(mol/h)
M_{col}	Total mass column contents	(mol)
M_p	Product	(mol)
R^*	Reflux fraction	$(-)$
$p^{[*]}$	Consequent part parameters fuzzy models	
$w^{[*]}$	Rule weights fuzzy models	
x^*	Product quality forcing function	$(-)$
x_{col}	Fraction volatile component in the column	$(-)$
x_{n+1}^0	Fraction volatile component in product	$(-)$
τ_V	Time constant characterizing column heating	(h)
τ_x	Dominant time constant	(h)

Summary

Dynamic process modeling in chemical engineering is often based on a combination of first principles and empirical relations. These models are interpretable, in the sense that, by analyzing the model, there is a physical understanding of the process behavior. Many process models consist of a framework of mass, component and energy balances describing the essential process accumulation. Within this framework, phenomena such as chemical reaction or mass transfer can be described by static empirical relations. However, for many processes, information about - or understanding of - these phenomena can be limited. In addition, the behavior can be complex. This may result in difficulties during the derivation of empirical relations.

Hybrid fuzzy-first principles models can be a useful alternative in these situations. By combining fuzzy logic submodels with a physical model framework, hybrid fuzzy-first principles models are obtained that combine a high level of interpretability with the ability to deal with complex behavior. Hybrid fuzzy-first principles models are especially suited to describe highly nonlinear behavior over a large operating domain. Examples are models of batch or fed-batch processes, cyclic processes or distributed parameter processes, such as plug flow reactors.

This work deals with the development and analysis of hybrid fuzzy-first principles models. A model structure is proposed in which hybrid models consist of a framework of dynamic mass and energy balances, formulated in state-space form, supplemented with algebraic and fuzzy equations. The fuzzy equations describe physical phenomena that are poorly understood or difficult to model using first principles. The proposed structure guarantees a certain level of transparency, since internal variables and their behavior can be interpreted. The quality of hybrid models is assessed by evaluating dynamic performance, static performance, complexity, interpretability and process independence.

The main sources of information during hybrid model development are first principles, process data and human expertise. First principles are used to derive the model structure. Although fuzzy logic is extremely suitable to quantify expert knowledge, it is often difficult to integrate this knowledge in a predefined hybrid model structure. Therefore, expert knowledge is only used to provide structural information. Process data is used to determine model parameters and identify the fuzzy equations.

The type of fuzzy model which is used is the Takagi-Sugeno-Kang (TSK) type. This type can be interpreted as a collection of local linear models. TSK fuzzy models are extremely suitable for describing highly nonlinear relations. Although linguistic fuzzy models can be interpreted better by humans, they require a more complex structure to describe a relation than TSK models. The advantage of using TSK type fuzzy models is reduced complexity in combination with structural transparency.

For the development of hybrid models, a structured modeling approach is presented. This approach consists of several independent steps, divided into three phases. In the first phase, the model objective and quality requirements are formulated. The second phase involves the design of the model. In the design phase, the modeling problem is divided into several smaller problems, which can be solved sequentially: structure design, behavior estimation, identification and optimization. In the third phase, the hybrid model is evaluated with respect to the model requirements.

The hybrid model structure is determined by formulating process behavior hypotheses. These hypotheses describe the essential characteristics of the process behavior. Based on these hypotheses, basic model equations can be derived. In addition, the phenomena that will be described with fuzzy logic are distinguished.

Subsequently, the process data required for identification is obtained. Most fuzzy model identification techniques require input-output data. To acquire input-output data, appropriate experiments need to be designed. Estimation techniques can be used to estimate behavior which is not directly measurable. For this purpose, Kalman filtering and PI-estimation were compared. The PI-estimator is structurally similar to a PI feedback controller. Its performance matches the Kalman filter, but the PI-estimator is easier to set up.

Three different approaches for the identification of fuzzy models were compared: fuzzy clustering, which searches for linear subspaces in data, genetic algorithms, a probabilistic optimization technique which can be used to determine fuzzy model parameters and neuro-fuzzy methods, in which the fuzzy model is interpreted as an artificial neural network. All approaches yielded acceptable results. However, fuzzy clustering was preferred, since it requires less a priori structure information. This makes the approach useful in situations where little information about the modeled phenomenon is available.

The hybrid model is formed by combination of the fuzzy models and the physical model framework. Since the fuzzy models are identified individually, it may be necessary to optimize the fuzzy model parameters with respect to the model output in order to improve the overall hybrid model performance. The best results are obtained if the optimization is focused on the consequent parameters of the fuzzy model; these are the parameters of the local linear models. The number of parameters to be optimized can be reduced by optimizing rule or model weights.

The modeling approach was illustrated using three different cases. A simple simulated fed-batch reactor was used for the development of the approach. To analyze hybrid model properties, a hybrid model of a simulated continuous pulp digester was built and compared with a first principles model and a fuzzy model. A batch distillation column was used to illustrate hybrid modeling of an experimental setup.

The hybrid models have shown that the use of fuzzy logic in hybrid modeling introduces flexibility, which enables the description of complex behavior with a predefined, interpretable overall model structure. This is accomplished since the fuzzy submodels describe complex behavior in a transparent way without the need for an a priori fuzzy model structure. The need for such a structure would reduce flexibility. The result of this flexibility is that hybrid fuzzy-first principles models can match the desired behavior if the model structure represents the essential dynamic characteristics of the process.

Since process dynamics are modeled with a similar structure, dynamic performance, complexity and interpretability of hybrid models and first principles models are comparable. With respect to static performance and process independence, hybrid models are more comparable with fuzzy models. Depending on the number of fuzzy equations, the static performance of the hybrid model is based on observed behavior, similar to fuzzy models. The fuzzy equations in hybrid models are data driven, which imposes a level of process dependence. The fuzzy models are valid in the operating regime that is represented by the identification data. The first principles part can only partly compensate for limited validity of the fuzzy models outside this operating regime.

The result is that hybrid fuzzy-first principles models are useful in applications which focus on a specific installation and require dynamic models that are transparent and provide a general explanation of the process behavior.

Samenvatting

Binnen de chemische technologie is dynamisch modelleren vaak gebaseerd op een combinatie van fysische principes en empirische relaties. Deze modellen zijn interpreteerbaar: door het model te analyseren kan fysisch begrip verkregen worden van het gedrag van het model. Veel modellen bestaan uit een raamwerk van massa-, energie- en componentbalansen die de essentiële processaccumulatie beschrijven. Binnen dit raamwerk kunnen verschijnselen zoals chemische reactie en stofoverdracht beschreven worden met empirische relaties. Echter, voor veel processen kan informatie over - of inzicht in - deze verschijnselen beperkt zijn. Daarnaast kan het gedrag van het proces complex zijn, wat kan leiden tot moeilijkheden bij het opstellen van empirische relaties.

Hybride fuzzy/fysische modellen kunnen in deze situaties een bruikbaar alternatief zijn. Door fuzzy submodellen te combineren met een fysisch raamwerk worden hybride fuzzy/fysische modellen verkregen die een hoge mate van interpreteerbaarheid combineren met de mogelijkheid om complex gedrag te beschrijven. Hybride fuzzy/fysische modellen zijn in het bijzonder geschikt om niet-lineair gedrag binnen een groot operatiedomein te beschrijven. Voorbeelden zijn batchprocessen, cyclische processen of gedistribueerde parameter-processen, zoals buisreactoren.

Dit werk behandelt het ontwerp en analyse van de hybride fuzzy/fysische modellen. Een modelstructuur zal voorgesteld worden, waarin hybride modellen bestaan uit een raamwerk van dynamische massa- en energiebalansen, geformuleerd in de vorm van toestandsvergelijkingen, aangevuld met algebraïsche en fuzzy vergelijkingen. De fuzzy vergelijkingen beschrijven fysische verschijnselen die slecht begrepen worden of moeilijk op grond van fysische principes te modelleren zijn. De voorgestelde structuur garandeert een bepaalde transparantie omdat interne variabelen en hun gedrag geïnterpreteerd kunnen worden. De kwaliteit van hybride modellen wordt bepaald door de dynamische en statische prestatie, de complexiteit, de interpreteerbaarheid en procesonafhankelijkheid in kaart te brengen.

De belangrijkste bronnen van informatie voor de bouw van hybride modellen zijn fysische principes, procesdata en menselijke expertise. Fysische principes worden gebruikt om de modelstructuur te bepalen. Hoewel fuzzy logic uitermate geschikt is om expertkennis te kwantificeren, is het vaak moeilijk deze kennis in een vastgelegde modelstructuur te integreren. Daarom wordt expertkennis slechts gebruikt om modelstructuren aan te geven. Procesdata wordt gebruikt om modelparameters en de fuzzy submodellen te bepalen.

Het type fuzzy modellen dat gebruikt wordt is het Takagi-Sugeno-Kang (TSK) type. Dit type kan geïnterpreteerd worden als een verzameling van lokale lineaire modellen. TSK fuzzy modellen zijn uitermate geschikt om hoog niet-lineair gedrag te beschrijven. Hoewel linguïstische fuzzy modellen beter door mensen geïnterpreteerd kunnen worden, is er een meer complexe structuur nodig om een relatie te beschrijven dan bij gebruik van TSK fuzzy modellen. Het voordeel van het gebruik van TSK modellen is gereduceerde complexiteit in combinatie met een transparante structuur.

Voor de ontwikkeling van hybride modellen is een gestructureerde modelleringsaanpak gepresenteerd. Deze aanpak bestaat uit verschillende onafhankelijke stappen, verdeeld over 3 fasen. In de eerste fase worden de modeldoelstelling en kwaliteitseisen geformuleerd. De tweede fase betreft het ontwerp van het model. Het model wordt geëvalueerd in de derde fase in relatie tot de kwaliteitseisen.

De structuur van het hybride model wordt bepaald door gedragshypothesen te formuleren. Deze hypothesen beschrijven de essentiële karakteristieken van het procesgedrag. Uitgaande van deze hypothesen kunnen de basisvergelijkingen van het model afgeleid worden. Tevens worden de verschijnselen die met behulp van fuzzy logic beschreven worden onderscheiden.

Vervolgens wordt de benodigde procesdata vergaard. De meeste identificatietechnieken voor fuzzy modellen hebben ingangs- en uitgangsdata nodig. Om de data te verkrijgen moeten geschikte procesexperimenten ontworpen worden. Daarnaast kunnen schattingstechnieken gebruikt worden om gedrag te schatten dat niet direct meetbaar is. Voor deze toepassing zijn het Kalman filter en de PI-schatter vergeleken. De PI-schatter is structureel gezien vergelijkbaar met een PI-regelaar. De prestatie is vergelijkbaar met het Kalman filter, maar de PI-schatter is veel makkelijker op te stellen.

Voor de identificatie van de fuzzy submodellen zijn drie verschillende technieken vergeleken: fuzzy clustering, waarbij gezocht wordt naar lineaire subruimten, genetische algoritmen, een probabilistische optimalisatietechniek die gebruikt kan worden voor het bepalen van de parameters van een fuzzy model, en neurofuzzy methoden, waarbij het fuzzy model beschouwd wordt als een neuraal netwerk. Alle technieken leverde acceptabele resultaten. Echter, de voorkeur gaat uit naar fuzzy clustering omdat deze techniek minder a priori informatie over de structuur van het fuzzy model nodig heeft. Dit maakt de aanpak geschikt in situaties waar weinig informatie over het te modelleren fenomeen beschikbaar is.

Door de fuzzy modellen en het fysische raamwerk te combineren ontstaat het hybride model. Omdat de fuzzy modellen afzonderlijk bepaald worden, kan het nodig zijn om de parameters van de fuzzy modellen te optimaliseren om de prestatie van het hybride model te verbeteren. De beste resultaten worden bereikt als de DAN-parameters van de fuzzy modellen geoptimaliseerd worden; dit zijn de parameters van de lokale lineaire modellen. Het aantal parameters dat geoptimaliseerd moet worden kan gereduceerd worden door regel- of modelgewichten te optimaliseren.

De modelleringsaanpak is geïllustreerd op drie verschillende processen. Een simpele gesimuleerde fed-batch reactor is gebruikt voor de ontwikkeling van de aanpak. Voor de analyse van de eigenschappen van hybride modellen is een hybride model van een (gesimuleerde) continue houtverpulper gebouwd. Dit model is vergeleken met een fysisch model en een fuzzy model. Tenslotte is een batchdestillatiekolom gebruikt om hybride modellering van een experimentele opstelling te illustreren.

De hybride modellen hebben aangetoond dat het gebruik van fuzzy logic voor hybride modelleren flexibiliteit introduceert, zodat beschrijving van complex gedrag met een vooraf gedefinieerde, interpreteerbare modelstructuur mogelijk wordt. Dit wordt bereikt doordat de fuzzy submodellen complex gedrag op een transparante manier beschrijven, zonder dat voor de fuzzy modellen een a priori modelstructuur vastgelegd dient te worden. Dit zou de flexibiliteit verminderen. Het resultaat van de flexibiliteit is dat hybride fuzzy/fysische modellen het gewenste gedrag kunnen beschrijven mits de modelstructuur de essentiële dynamische karakteristieken van het proces representeert.

Omdat de dynamica binnen fysische en fuzzy modellen op dezelfde manier beschreven wordt, zijn de dynamische prestatie, complexiteit en interpreteerbaarheid van deze modellen vergelijkbaar. Voor wat betreft statische prestatie en procesonafhankelijkheid zijn hybride modellen meer vergelijkbaar met fuzzy modellen. Afhankelijk van het aantal fuzzy submodellen is de statische prestatie van hybride modellen gebaseerd op het waargenomen gedrag, vergelijkbaar met fuzzy modellen. De fuzzy elementen van hybride modellen zijn gebaseerd op procesdata, wat een mate van procesafhankelijkheid oplegt. De fuzzy submodellen zijn geldig in het operatiedomein dat de procesdata representeert. Het fysische gedeelte kan slechts gedeeltelijk compenseren voor beperkte geldigheid buiten dit domein.

Het resultaat is dat hybride fuzzy/fysische modellen geschikt zijn voor toepassingen die zich richten op een specifieke installatie en waarbij behoefte is aan dynamische modellen die transparant zijn en een algemene uitleg van het gedrag geven.

Dankwoord

Zo'n promotieproject is een hele onderneming. Het proefschrift dat er nu ligt had niet tot stand kunnen komen als ik niet bijgestaan was door een groep enthousiaste mensen, die ik dan ook graag wil bedanken.

Brian Roffel en Ben Betlem hebben mij tijdens het project begeleid. Brian, het idee voor het project kwam van jou. Al tijdens de beginperiode van mijn afstudeeropdracht wist je me warm te maken voor onderzoek en jij hebt me al snel de kans gegeven het project verder vorm te geven. Jouw pragmatische insteek en resultaatgericht denken hebben vele obstakels uit de weg geholpen en tot een aantal leuke resultaten geleid. Ben, jij was meer de filosoof en de discussies met jou zijn van groot belang geweest voor het bepalen van de richting. Het "afdwalen" tijdens onze gesprekken zorgde ervoor dat veel onderwerpen in een goede context geplaatst konden worden, ookal kwamen we vaak weer "terug bij af."

Het proefschrift had er nooit kunnen liggen zonder het vele werk dat verzet is door de grote groep afstudeerders. Naar onderwerp waren dit: Pieter Jansen van der Sligte (vastleggen ervaringskennis), Peter Deurenberg (TN, verwerking ervaringskennis), Henri Witteveen (parameterschatting, maar ook een jaar lang collega en geduchte tegenstander bij squash!), Coen Noppert (genetische algorithmen), Yücel Kök (Rijksuniversiteit Groningen, neurofuzzy systemen), Freek Stoffelen (TN, submodel integratie, de Pot van Freek), Maarten Jansen (modeleigenschappen), Simone Bijl (modeleigenschappen) en Barry Meddeler (modeleigenschappen). Jammergenoeg is het werk van Tobias op den Brouw niet direct in het proefschrift opgenomen. Tobias, hoewel de AWZI geen geschikte case voor het project bleek, heeft jou werk wel meer inzicht gegeven in de (on)mogelijkheden van het gebruik van industriële data en is op deze manier waardevol geweest voor het bepalen van de toepassingsmogelijkheden van hybride modelleren. Jongens, de verschillende onderwerpen hadden nooit behandeld kunnen worden zonder jullie inzet. Het werken met jullie heb ik altijd een van de mooiste aspecten van het promoveren gevonden. Ook waardeer ik het contact dat we naast het werk hadden of nog steeds hebben. Bedankt!

Ik wil de leden van de verschillende D-commissies eveneens bedanken voor hun bijdrage: Henk van den Beld† (CT), Ties Bos (CT), Herman Hemmes (TN), Hans Kuipers (CT), Sietse van der Meulen (TN), Ruud Visscher (Parenco BV), Sjoerd de Vries (TO), Theo de Vries (EL) en Jan Wattenberg (Parenco BV).

Mijn paranimfen Jan Jelle Sijbesma en Ralf Heijkants hebben mij met de laatste loodjes geholpen. Jongens, bedankt dat jullie me bij willen staan. Ook wil ik iedereen van de vakgroep bedanken voor de plezierige tijd de afgelopen jaren. Bartie, du wirst immer mein Süße sein.

Tot slot wil ik mijn ouders bedanken voor hun onvoorwaardelijke steun. Hoewel jullie niet altijd op de hoogte waren van waar ik me nu eigenlijk mee bezig hield, hebben jullie me altijd gestimuleerd en de kans gegeven om verder te leren. Zonder jullie had ik dit niet kunnen bereiken.

Bibliography

Alexandridis, A., H. Sarimveis and T. Retsina (2001). Modeling and control of continuous digesters using the pls methodology. In: *Pulp Digester Modeling and Control Workshop*. Annapolis (USA).

Atkinson, K. E. (1989). *Numerical analysis*. 2nd ed.. John Wiley and Sons. New York (USA).

Babuška, R. (1996). Fuzzy modeling and identification. Ph.d. thesis. Delft Technical University. Delft (The Netherlands).

Babuška, R. and H. B. Verbruggen (1994). Applied fuzzy modeling. In: *Proceedings IFAC Symposium on Artificial Intelligence in Real Time Control*. Valencia (Spain). pp. 61–66.

Babuška, R., H. J. L. Van Can and H. B. Verbruggen (1996). Fuzzy modeling of enzymatic penicillin-g conversion. In: *Preprints 13th IFAC world congress*. Vol. N. San Francisco (USA). pp. 479–484.

Back, T. and F. Kursawe (1995). Evolutionary algorithms for fuzzy logic: a brief overview. In: *Advances in fuzzy systems - applications and theory* (B. Bouchon-Meunier, R. R. Yager and L. A. Zadeh, Eds.). Vol. 4. pp. 3–10. World Scientific. Singapore.

Baranyi, P., I. Bavelaar, L. T. Koczy and A. Titli (1997). Inverse rule base of various fuzzy interpolation techniques. In: *Seventh IFSA World Congress*. Prague. pp. 121–126.

Baranyi, P., I. M. Bavelaar, R. Babuska, L. T. Koczy, A. Titli and H. B. Verbruggen (1998). A method to invert a linguistic fuzzy model. *International Journal of Systems Science* **29**(7), 711–721.

Bastin, G. and D. Dochain (1990). *On-line estimation and adaptive control of bioreactor*. Vol. 1 of *Process Measurement and Control*. Elsevier. Amsterdam.

Betlem, B. H. L. (1997). Hierarchical control of batch distillation. Ph.d. thesis. University of Twente. Enschede (The Netherlands).

Betlem, B. H. L. (2000). Batch distillation column low-order models for quality program control. *Chemical Engineering Science* **55**, 3187–3194.

Betlem, B. H. L., H. Krijnsen and H. Huijnen (1998). Optimal batch distillation control based on specific measures. *Chemical Engineering Journal* **71**, 111–126.

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press. New York.

Bezdek, J. C., C. Coray, R. Gunderson and J. Watson (1981). Detection and characterization of cluster substructure, ii. fuzzy c-varieties and convex combinations thereof. *SIAM Journal of Applied Mathematics* **40**(2), 339–357.

Boehm, B. W. (1973). Characteristics of software quality. Technical Report TRW-SS-73-09. Thompson Ramo Wooldridge Systems Group, Systems Engineering and Integration Division.

Bohlin, T. (1994). A csae study of grey box identification. *Automatica* **30**(2), 307–318.

Box, G. E. P. and G. M. Jenkins (1970). *Time series analysis; forecasting and control*. Holden-Day series in time series analysis. Holden-Day. San Francisco.

Box, G. E. P., W. G. Box and J. S. Hunter (1978). *Statistics for experimenters*. Wiley series in probability and mathematical statistics. John Wiley and Sons. New York (USA).

Branch, M. A., T. F. Coleman and Y. Li (1999). A subspace, interior and conjugate gradient method for large-scale bound-constrained minimization problem. *SIAM Journal on Scientific Computing* **21**(1), 1–23.

Bristol, E. H. (1966). On a new measure of interaction for multivariable process control. *IEEE Transactions on Automation and Control* **AC-11**, 133.

Brown, R. G. and P. Y. C. Hwang (1992). *Introduction to random signals and applied Kalman filtering*. 2nd ed.. John Wiley and Sons. New York (USA).

Byrd, R. H., R. B. Schnabel and G. A. Shultz (1988). Approximate solution of the trust region problem by minimization over two-dimansional subspaces. *Mathematical programming* **40**, 247–263.

Cagan, J., I. E. Grossman and J. Hooker (1996). Combining artificial intelligence and optimization in engineering design: a brief survey. *AIChE Symposium series* **92**(312), 110–118.

Caracotsios, M. and W. E. Stewart (1985). Sensitivity analysis of initial value problems with mixed ode's and algebraic equations. *Computers in Chemical Engineering* **9**(4), 359–365.

Crosby, P. B. (1979). *Quality is free: The art of making quality certain*. McGraw-Hill. New York (USA).

da Costa Sousa, J. M. (1998). A fuzzy approach to model-based control. Ph.d. thesis. Technical University Delft. Delft (The Netherlands).

da Costa Sousa, J. M., R. Babuska and H. B. Verbruggen (1997). Fuzzy predictive control applied to an air-conditioning system. *Control engineering practice* **5**(10), 1395–1406.

De Bruin, H. A. E. and B. Roffel (1996). A new identification method for fuzzy linear models of nonlinear dynamic systems. *Journal of Process Control* **6**(5), 227–293.

Dubois, D. and H. Prade (1993). Fuzzy sets: a survey of engineering applications. *Computers and chemical engineering* **17**(Supp), S373–S380.

Dufour, P., S. Bhartiya, P. S. Dhurjati and F. J. Doyle III (2001). A neural network approach for diagnosis in a continuous pulp digester. In: *Pulp Digester Modeling and Control Workshop*. Annapolis (USA).

Dunn, I. J., J. Heinzle, J. Ingham and J. E. Prenosil (1992). *Biological reaction engineering, principles, applications and modelling with PC simulation*. VCH Verlagsgesellschaft. Weinheim (Germany).

Dunn, J. C. (1974). A graph theoretic analysis for pattern classification via tamura's fuzzy relation. *IEEE Transactions on Systems, Man and Cybernetics* **4**(3), 310–313.

Edgar, T. F. and D. M. Himmelblau (1988). *Optimization of chemical processes*. McGraw-Hill. New York (USA).

Eykhoff, P. (1974). *System Identification*. John Wiley and Sons. London (UK).

Fischer, M. and R. Isermann (1996). Robust hybrid control based on inverse fuzzy process models. In: *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*. New Orleans (USA). pp. 1210–1216.

Funkquist, J. (1997). Grey-box identification of a continuous digetser - a distributed-parameter process. *Control Engineering Practice* **5**(7), 919–930.

Godasi, S. and A. Palazoglu (2001). Recursive identification of a wiener model to describe grade transitions in a continuous pulp digester. In: *Pulp Digester Modeling and Control Workshop*. Annapolis (USA).

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley. Reading, MA (USA).

Gonzalez, A. J. and D. D. Dankel (1993). *The Engineering of Knowledge-based Systems - Theory and Practice*. Prentice Hall. Engelwood Cliffs, NJ (USA).

Gupta, S., P.-H. Liu, S. A. Svoronos, R. Sharma, N. A. Abdel-Khalek, Y. Cheng and H. El-Shall (1999). Hybrid first-principles/neural networks model for column flotation. *AIChE Journal* **45**(3), 557–566.

Hart, A. (1992). *Knowledge acquisition for expert systems*. McGraw-Hill. New York (USA).

Hathaway, R. J. and J. C. Bezdek (1993). Switching regression models and fuzzy clustering. *IEEE transactions on Fuzzy Systems* **71**(3), 503–516.

Hertz, J., A. Krogh and R. G. Palmer (1991). *Intorduction to the theory of neural computation*. Addison-Wesley. Redwood (USA).

Holst, J., U. Holst, H. Madsen and H. Melgaard (1992). Validation of grey box models. In: *IFAC Adaptive systems in control and design processing*. Grenoble (France). pp. 53–60.

Hwang, G.-J. (1995). Knowledge acquisition for fuzzy expert systems. *International Journal of Intelligent Systems* **10**, 541–560.

Jahn, E. C. and R. W. Strauss (1992). Industrial chemistry of wood. In: *Riegel's Handbook of industrial chemistry* (E. R. Riegel and J. A. Kent, Eds.). 9th ed.. pp. 435–485. Van Nostrand Reinhold. New York (USA).

Jain, A. K. and R. C. Dubes (1988). *Algorithms for Clustering Data*. Prentice Hall. Englewood Cliffs, NJ (USA).

Jang, J.-S. R. (1993). Anfis: Adaptive-network -based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics* **23**(3), 665–685.

Jansen, M. (2000). Modeling and simulation of a continuous pulp digester. M.sc. thesis. University of Twente. Enschede (The Netherlands).

Jansen van der Sligte, P. (1999). Using expert knowledge in fuzzy modeling. M.sc. thesis. University of Twente. Enschede (The Netherlands).

Johansen, T. A. and B. A. Foss (1997). Operating regime based process modeling and identification. *Computers and chemical engineering* **21**(2), 159–176.

Jordan, M. I. and D. E. Rumelhart (1992). Forward models: Supervised learning with a distal teacher. *Cognitive Science* **16**(3), 307–354.

Kalman, R. E. (1960). A new approach to linear filtering and prediction theory. *Journal of Basic Engineering* **82D**, 35–45.

Kan, S. H. (1994). *Metrics and models in software quality engineering*. Addison-Wesley. Reading (USA).

Kavli, T. (1993). Asmod–an algorithm for adaptive spline modelling of observation data. *International journal of control* **58**(4), 947–968.

Kaymak, U. and R. Babuska (1995). Compatible cluster merging for fuzzy modeling. In: *FUZZ-IEEE/IFES'95*. Yokohama, Japan. pp. 897–904.

Kendall, M. and A. Stuart (1979). *The advanced theory of statistics*. Vol. 2. 4th ed.. Charles Griffin and company. London (UK).

KrishnaKumar, K., P. Gonsalves, A. Satyadas and G. Zacharias (1995). Hybrid fuzzy logic flight controller synthesis via pilot modeling. *Journal of Guidance, Control and Dynamics* **18**(5), 1098–1105.

Luyben, W. L. (1990). *Process modeling, simulation, and control for chemical engineers*. 2nd ed.. McGraw-Hill. New York (USA).

Marsh, S. (1994). *Fuzzy Logic Program*. Cortex Communications. Austin, TX (USA).

Nauck, D. and R. Kruse (1997). Function approximation by nefprox. In: *Second European Workshop on Fuzzy Decision Analysis and Neural Networks for Management, Planning and Optimization*. Dortmund, Germany.

Nelles, O. (1997). Lolimot - lokale, lineare modelle zur identifikation nichtlinearer, dynamischer systeme. *Automatisierungstechnik* **45**(4), 163–174.

Nonaka, I. and H. Takeuchi (1995). *The Knowledge-creating Company; How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press. New York.

Otto, O. and K. Hartmann (1996). Hybrides modell eines sichters auf der basis kunstlicher neuronaler netze. *Chemie Ingenieur Technik* **68**(12), 1578–1581.

Pal, N. R., K. Pal, J. C. Bezdek and T. A. Runkler (1997). Some issues in system identification using clustering. In: *The 1997 IEEE international conference on neural networks*. Houston (USA). pp. 2524–2529.

Piron, E., E. Latrille and F. Rene (1997). Application of artificial neural networks for crossflow microfiltration modelling: black box and semi-physical approaches. *Computers and chemical engineering* **21**(9), 1021–1030.

Porru, G., C. Aragonese, R. Baratti and A. Servida (2000). Monitoring of a co oxidation reactor through a grey-model based ekf observer. *Chemical engineering science* **55**, 331–338.

Psichogios, D. C. and L. H. Ungar (1992). A hybrid neural network-first principles approach to process modeling. *AIChE Journal* **38**(10), 1499–1511.

Qi, H., X.-G. Zhou, L.-H. Liu and W.-K. Yuan (1999). A hybrid neural netwrok-first principles model for fixed-bed reactor. *Chemical Engineering Science* **54**, 2521–2526.

Ramirez, W. F. (1994). *Process control and identification*. Academic Press. San Diego (USA).

Rausis, K. (1998). Verification des lois de commande floue. Ph.d. thesis. Ecole Polytechnique Federale de Lausanne. Lausanne (Switzerland).

Rippin, D. W. T. (1983). Simulation of single- and multiproduct batch chemical plants for optimal desing and operation. *Computers and chemical engineering* **7**(3), 137–156.

Roffel, B. and P. Chin (1987). *Computer control in the process industries.* Lewis Publishers. Chelsea, MI (USA).

Roubos, J. A., P. Krabben, M. Setnes, R. Babuska, J. J. Heijnen and H. B. Verbruggen (1999). Hybrid model development for fed-batch bioprocesses; combining physical equations with the metabolic network and black-box kinetics. In: *6th UK Workshop on Fuzzy Systems*. Brunel University, Uxbridge (UK). pp. 231–239.

Seinfeld, J. H. and L. Lapidus (1974). *Process modeling, estimation, and identification*. Vol. 3 of *Mathematical methods in chemical engineering*. Prentice Hall. Englewood Cliffs (USA).

Sestito, S. and T. S. Dillon (1994). *Automated Knowledge acquisition*. Prentice Hall. New York (USA).

Setnes, M. (1999). Complexity reduction methods for fuzzy systems design. In: *Fuzzy Logic Control Advances in Applications* (H. B. Verbruggen and R. Babuska, Eds.). pp. 87–111. World Scientific. Singapore.

Shinskey, F. G. (1984). *Distillation control for productivity and energy conservation*. 2nd ed.. McGraw-Hill. New York (USA).

Simutis, R., R. Oliveira, M. Manikowski, S. Feyo de Azevedo and A. Lubbert (1997). How to increase the performance of models for process optimization and control. *Journal of biotechnology* **59**, 73–89.

Smith, C. C. and T. J. Williams (1974). Mathematical modeling, simulation and control of the operation of kamyr continuous digester for kraft process. Technical Report 64. Purdue University, PLAIC.

Sneddon, I. N. (1976). *Encyclopaedic dictionary of mathematics for engineers and applied scientists*. Pergamon Press. Oxford (UK).

Sohlberg, B. (1998). *Supervision and control for industrial processes*. Springer Verlag. Berlin (Germany).

Stephanopoulos, G. (1984). *Chemical process control: an introduction to theory and practice*. Prentice Hall. Englewood Cliffs, NJ (USA).

Stephanopoulos, G. and C. Han (1996). Intelligent systems in process engineering: a review. *Computers and chemical engineering* **20**(6/7), 743–791.

Takagi, T. and M. Sugeno (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Man, Systems and Cybernetics* **15**(1), 116–132.

Tansley, D. S. W. and C. C. Hayball (1993). *Knowledge-Based Systems Analysis and Design - A KADS Developer's Handbook*. BNR Europe Ltd.. Hertfordshire (UK).

Thompson, M. L. and M. A. Kramer (1994). Modeling chemical processes using prior knowledge and neural networks. *AIChE Journal* **40**(8), 1328–1340.

Van Can, H. J. L., C. Hellinga, K. Ch. A. M. Luyben, J. J. Heijnen and H. A. B. Te Braake (1996). Strategy for dynamic process modeling based on neural networks in macroscopic balances. *AIChE Journal* **42**(12), 3403–3418.

Van Can, H. J. L., H. A. B. Te Braake, C. Hellinga, K. Ch. A. M. Luyben and J. J. Heijnen (1997). An efficient model development strategy for bioprocesses based on neural networks in macroscopic balances. *Biotechnology and Bioengineering* **54**(6), 549–566.

Van Can, H. J. L., H. A. B. Te Braake, S. Dubbelman, C. Hellinga, K. Ch. A. M. Luyben and J. J. Heijnen (1998). Understanding and applying the extrapolation properties of serial gray-box models. *AIChE Journal* **44**(5), 1071–1089.

Van Lith, P. F., H. Witteveen, B. H. L. Betlem and B. Roffel (2001). Multiple nonlinear parameter estimation using pi feedback control. *Control Engineering Practice* **9**(5), 517–531.

Venkatasubramanian, V. and S. H. Rich (1988). An object-oriented two-tier architecture for integrating compiled and deep-level knowledge for process diagnosis. *Computational Chemical Engineering* **12**, 903–921.

Westerterp, K. R., W. P. M. van Swaaij and A. A. C. M. Beenackers (1984). *Chemical reactor design and operation*. John Whiley and Sons. Chichester (UK).

Wisnewski, P. A., F. J. Doyle and F. Kayihan (1997). Fundamental continuous-pulp-digester model for simulation and control. *AIChE Journal* **43**(12), 3175–3192.

Yager, R. R. and D. P. Filev (1994). *Essentials of fuzzy modeling and control*. John Wiley and Sons. New York (USA).

Yang, M. S. (1993). A survey of fuzzy clustering. *Mathematical and computer modelling* **18**(11), 1–16.

Ying, H. (1994). Sufficient conditions on general fuzzy systems as function approximators. *Automatica* **30**(3), 521–525.

Yourdon, E. (1989). *Modern structured analysis*. Prentice Hall. Englewood Cliffs (USA).

Zadeh, L. A. (1994). Soft computing and fuzzy logic. *IEEE Software* **11**(6), 48–56.

Zhao, J., V. Wertz and R. Gorez (1994). A fuzzy clustering method for the identification of fuzzy models for dynamical systems. In: *9th IEEE International Symposium on Intelligent Control*. Columbus, Ohio (USA). p. 172.