A Grey-Box Identification Case Study -- The Åström–Bell Drum-Boiler Model

Sørlie, James A.; Eborn, Jonas

1997

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
A grey-box identification case study: The Åström–Bell drum-boiler model

James A. Sørlie
Jonas Eborn

Department of Automatic Control
Lund Institute of Technology
June 1997
Title and subtitle
A grey-box identification case-study: The Åström–Bell drum-boiler model

Abstract
This report describes a case-study involving parameter identification of a family of physical models for a drum-boiler, as described by Åström and Bell. The models are based on first principles, but they also include some grey-box parts where there is not complete physical knowledge. The case-study uses a series of unique open-loop data from Öresundsverket in Sweden.

The object of the case-study is to study the interaction between the modelling tool, OmSim, and the parameter optimization tool, IdKit. A newly developed interface for equation export using Maple is used to tie the tools together. The case-study also addresses the question of which of a set of given model structures is adequate for capturing the dynamics seen in the data. The results show that a previously published fourth-order model is the most powerful unfalsified model. A newer fifth-order model including time-delays is rejected since it does not add any reproducibility with respect to the data in the six datasets.

Key words
modeling, simulation, nonlinear models, model validation, parameter estimation.

Classification system and/or index terms (if any)

Supplementary bibliographical information

The report may be ordered from the Department of Automatic Control or borrowed through:
University Library 2, Box 3, SE-221 00 Lund, Sweden
Fax +46 46 222 44 22   E-mail ub2@ub2.lu.se
## Contents

1 Introduction ........................................ 1  
1.1 Report Outline .................................. 1  
1.2 Acknowledgment .................................. 3  

2 Model Definition ................................... 4  
2.1 Model Equations and Hypotheses ................. 5  
2.2 Model Structure Administration .................. 6  
2.3 Model Parameterization and Simulation Interface 6  
2.4 Signal Model Specification ....................... 8  
2.5 Equation Export and Code Generation .......... 9  
2.6 Closure ......................................... 10  

3 Simulation Testing .................................. 11  
3.1 Manual Parameter Tuning ......................... 11  
3.1.1 An Example: Balancing the Flux of Mass and Energy 11  
3.2 Parameter Sensitivities and Couplings ......... 13  
3.2.1 Couplings with Fluid Friction in the Down-Comers 13  
3.2.2 Over-Parameterization of the Drum Level Measurement Model 13  
3.2.3 Couplings due to Non-Equilibrium State Initialization 14  
3.2.4 Iterative State Initialization and the Drum Level Measurement Model (Revisited) 15  
3.3 Structure Determination ......................... 16  
3.3.1 An Example: Validity of Enthalpy Approximations in Energy Balances 16  

4 Parameter Optimization ............................ 19  
4.1 Design of the Optimization Trials .............. 19  
4.2 Calibration Factors and Steam Valve Nonlinearity 19  
4.3 Thermal Capacitance and Disposition of the Metal Mass 21  
4.4 $M_0$ Shrink-and-Swell Heuristics vs. a Fourth State Variable 21  
4.5 Variable Time Delay of the Bubble Flow Rate .... 24  
4.6 Friction Factor and Non-Equilibrium Initialization 24  
4.7 Summary ........................................ 24  

5 Conclusion ......................................... 27  

A Maple Derivations .................................. 28  
A.1 Preliminaries .................................... 28  
A.1.1 Approximation of Thermodynamic Property Data 28  
A.2 Second Order Structure ......................... 30  
A.2.1 Global Mass and Energy Balances ............. 30  
A.2.2 Explicit Second-Order State Equations ........ 31  
A.2.3 A Simplified Single-State Model .............. 34  
A.3 Third Order Structure ............................ 35  
A.3.1 A Drum Level Measurement Model ............. 36  
A.3.2 Vapor Mass Fraction at Risers Outlet as an Augmented Third State 38  
A.3.3 Approximation of the Total Vapor Volume Fraction in the Risers 42  
A.3.4 Mass Flow Rate through the Down-comers ..... 45  
A.3.5 Third-Order Equations with an Alternate Choice of State Variable 47  
A.3.6 An Augmented Reduced Two-State Model ....... 49  
A.3.7 Iterative Initialization of the Third State .... 51  
A.4 Fourth Order Structure ......................... 53
### CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>B</strong></td>
<td>Omola Definitions</td>
<td></td>
</tr>
<tr>
<td>B.1</td>
<td>Library Definitions</td>
<td>57</td>
</tr>
<tr>
<td>B.2</td>
<td>Model Structures</td>
<td>57</td>
</tr>
<tr>
<td>B.2.1</td>
<td>Reduced Order Definitions</td>
<td>66</td>
</tr>
<tr>
<td>B.2.2</td>
<td>Alternate State Realization</td>
<td>72</td>
</tr>
<tr>
<td>B.2.3</td>
<td>Time Delay Realizations</td>
<td>73</td>
</tr>
<tr>
<td>B.2.4</td>
<td>Iterative State Variable Initialization</td>
<td>74</td>
</tr>
<tr>
<td>B.3</td>
<td>Simulation Models</td>
<td>76</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>Annotated IdKit Sessions</td>
<td>77</td>
</tr>
<tr>
<td>C.1</td>
<td>On Batch Processing</td>
<td>81</td>
</tr>
<tr>
<td>C.2</td>
<td>Investigation of $\mathcal{M}_3$</td>
<td>85</td>
</tr>
<tr>
<td>C.3</td>
<td>Investigation of $\mathcal{M}_4$</td>
<td>93</td>
</tr>
<tr>
<td>C.4</td>
<td>Investigation of $\mathcal{M}_5$</td>
<td>102</td>
</tr>
<tr>
<td><strong>References</strong></td>
<td></td>
<td>111</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Six datasets from Eklund's Öresundsverk experiments.</td>
</tr>
<tr>
<td>2.1</td>
<td>Ideal physical model of a steam generation process.</td>
</tr>
<tr>
<td>2.2</td>
<td>Inheritance hierarchy of the boiler model structures.</td>
</tr>
<tr>
<td>2.3</td>
<td>The OmSim simulation interface to the experimental data.</td>
</tr>
<tr>
<td>2.4</td>
<td>Each simulation model specifies: (i) a boiler submodel structure and (ii) a signal-model mapping for equation export.</td>
</tr>
<tr>
<td>3.1</td>
<td>Typical results of iterative manual tuning.</td>
</tr>
<tr>
<td>3.2</td>
<td>Simulation experiment assessing the model's sensitivity to fluid friction.</td>
</tr>
<tr>
<td>3.3</td>
<td>Effect of non-equilibrium initialization of ( x_t ) on the drum level model.</td>
</tr>
<tr>
<td>3.4</td>
<td>Simulation experiments assessing the validity of the enthalpy approximation made in the energy balance derivations.</td>
</tr>
<tr>
<td>4.1</td>
<td>Optimization results for the calibration coefficients ( q_{c f}, q_{fcf} ) and ( k_s ) for each of the six datasets and three model structures.</td>
</tr>
<tr>
<td>4.2</td>
<td>Optimization results for the drum and risers metal masses ( \text{md} ) and ( \text{mr} ) for each of the six datasets, three model structures and three fixed values of ( k_f ).</td>
</tr>
<tr>
<td>4.3</td>
<td>Optimization results for the surface area of liquid in the drum ( A_d ) and heuristic parameters ( b_1 ) and ( b_2 ) in ( M_3 ) for each of the six datasets and three fixed values of ( k_f ).</td>
</tr>
<tr>
<td>4.4</td>
<td>Optimization results for the surface area of liquid in the drum ( A_d ), the heuristic parameter ( b_1 ) and ( V_{sd0} ) in ( M_4 ) and ( M_5 ), for each of the six datasets and three fixed values of ( k_f ). The standard deviation of the drum level prediction error ( \text{se}_{\text{p}} ) indicates the quality of each trial's result.</td>
</tr>
<tr>
<td>4.5</td>
<td>Optimization results from trial 2 with the initial state condition ( x_0 ) free, for each of the six datasets and six fixed values of ( k_f ). Log-log plot axes reveal the nature of the coupling.</td>
</tr>
<tr>
<td>A.1</td>
<td>Qualitative ( P-v ) diagram of the thermodynamic properties for saturated water. Shown in bold is the parabolic nature of the curve-fits used.</td>
</tr>
<tr>
<td>A.2</td>
<td>A simplified geometric representation of the risers.</td>
</tr>
</tbody>
</table>
List of Listings

3.1 NonEquilibriumInit.ocl—Simulation script for Figure 3.3. ............................................ 16
3.2 lhuMass.ocl—Simulation script for Figure 3.4. ................................................................. 18
B.1 library.ol—Base definitions used in much of the model library. ......................................... 57
B.2 BoilerIC.om—Interface class definition for the drum boiler flow model. ............................. 59
B.3 BoundedVariableGain.om—Definition of a bounded smooth nonlinear gain block. ............. 59
B.4 LowPassFilter.om—A first-order low pass filter block. ....................................................... 60
B.5 ProductJunction.om—A scalar signal multiplier block. ......................................................... 60
B.6 StaticGain.om—A constant linear gain block. ......................................................................... 60
B.7 SumJunction.om—A scalar signal summer block. ................................................................. 61
B.8 WienerProcess.om—A pure integration which, mathematically, yields a Wiener process if driven by continuous-time white Gaussian noise. ....................................................... 61
B.9 SignalModelMapping.om—The parameterized signal model names used in equation export to Maple. ......................................................................................................................... 61
B.10 SaturationMM.om—Medium model containing the thermodynamic properties of saturated water at a given pressure. .......................................................................................... 62
B.11 OresundSimlC.om—The simulation interface to the experimental data, including stochastic input and output error modeling. .............................................................. 63
B.12 Boiler2dFM.om—Definition of the second-order root model structure. ................................. 67
B.13 Boiler3dFM.om—Definition of the third-order model structure. ........................................... 70
B.14 Boiler4dFM.om—Definition of the fourth-order model structure. ......................................... 71
B.15 Boiler1dFM.om—Reduced first-order structure, assuming the total volume of water in the system is constant. ............................................................... 72
B.16 Boiler2dFM.om—An augmented, reduced second-order model structure. ............................ 72
B.17 Boiler3dFM.om—An augmented, reduced third-order model structure. ................................. 72
B.18 Boiler2dSWFM.om—An alternate state realization with \( V_{\text{sw}} \), the volume of water in the drum, as a state variable. .......................................................... 73
B.19 Boiler3dFM.om—A variation of the third-order structure that includes realizations (pure and approximate) of a time-delayed signal \( q_{\text{cd}} \), the total condensation flow rate. ................................. 74
B.20 Boiler4dFM.om—A variation of the fourth-order structure that includes realizations (pure and approximate) of a time-delayed signal \( q_{\text{cd}} \), the vapor flux across the liquid surface in the drum. ........................................ 75
B.21 Boiler5dFM.om—For equation export, a definition equivalent to Boiler4dFM.om, but with only the Padé(0,1) delay approximation. ......................................................... 75
B.22 Boiler3dFM.om—Derivation of the third-order model structure including an iterative initialization of \( x_0(0) \) based on first-order Taylor series approximations. .................................................... 76
B.23 Oresund2d.om—Simulation model definition for the second-order model structure, plus parameterization information for equation export. ....................................................... 77
B.24 Oresund3d.om—The derived third-order simulation model including over-written inherited information for equation export. .......................................................... 77
B.25 Oresund4d.om—Simulation model definition for the fourth-order model structure; the parameter \( b_2 \) is replaced by \( V_{\text{cd}} \) in the equation export parameter mapping. ................................. 78
B.26 Oresund1d.om—Simulation model for the reduced first-order model structure. .................. 78
B.27 Oresund2d.om—Simulation model for the augmented, reduced second-order model structure. ................................................................................................................................. 78
B.28 Oresund3d.om—Simulation model for the reduced third-order model structure. ................. 79
B.29 Oresund3dWd.om—Simulation model for the model structure with an alternate state realization. ............................................................................................................................. 79
B.30 Oresund3d.om—Simulation model for the third-order model structure with alternate time-delay realizations of \( q_{\text{cd}} \). ................................................................. 79
<table>
<thead>
<tr>
<th>LIST OF LISTINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.31 Oresund4.om—Simulation model for the fourth-order model structure with alternate time-delay realizations of $q_{ed}$.</td>
</tr>
<tr>
<td>B.32 Oresund5.om—Simulation model for the fourth-order model structure with a Padé(0,1) approximation of the time-delayed $q_{ed}$.</td>
</tr>
<tr>
<td>C.1 cm_setv—UNIX shell script created by IdKit's macro facility.</td>
</tr>
<tr>
<td>C.2 cs_writep0—Command script that saves the parameter values as an IdKit macro and OmSim OCL script.</td>
</tr>
<tr>
<td>C.3 cs_setp0.cf—Command script used to encapsulate the IdKit command setp0.</td>
</tr>
<tr>
<td>C.4 cs_setp0.x0—Command script used to encapsulate the IdKit command setp0.</td>
</tr>
<tr>
<td>C.5 cs_setm—Command script used to encapsulate the IdKit command setm.</td>
</tr>
<tr>
<td>C.6 cs_setp—Command script used to encapsulate the IdKit command setp.</td>
</tr>
<tr>
<td>C.7 M3_Exp—Command script used to investigate the third-order model structure.</td>
</tr>
<tr>
<td>C.8 M4_Exp—Command script used to investigate the fourth-order model structure.</td>
</tr>
<tr>
<td>C.8 M5_Exp—Command script used to investigate the fourth-order model structure.</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Overview of the model structures defined using OMOLA. ................. 7
2.2 Overview of the model parameterization and prior knowledge. ........ 7
3.1 The manually tuned calibration factors used as starting point for optimization. ... 12
4.1 Mean values of some of the parameters for the six datasets. ............. 23
C.1 Summary of the IdKit command macros and command scripts. ........... 82
C.2 Optimization results for $M_6$ with $k_f = 0.001$ fixed. .......... 87
C.3 Optimization results for $M_6$ with $k_f = 0.005$ fixed. .......... 89
C.4 Optimization results for $M_6$ with $k_f = 0.01$ fixed. .......... 91
C.5 Optimization results for $M_6$ with $k_f = 0.001$ fixed. .......... 96
C.6 Optimization results for $M_6$ with $k_f = 0.005$ fixed. .......... 98
C.7 Optimization results for $M_6$ with $k_f = 0.01$ fixed. .......... 100
C.8 Optimization results for $M_6$ with $k_f = 0.001$ fixed. .......... 105
C.9 Optimization results for $M_6$ with $k_f = 0.005$ fixed. .......... 107
C.10 Optimization results for $M_6$ with $k_f = 0.01$ fixed. .......... 109
Chapter 1

Introduction

At the outset, the goals of this project were loosely specified as follows:

- Prototype the code-generation interface between LTH's OmSim simulation environment and KTH's IDKit software for grey-box identification as proposed in [1].

- Use these tools to investigate what improvements can be gained via parameter optimization of the deterministic thermo-physical drum boiler model structures developed by R. Bell and K.J. Åström [2-5].

Results addressing the first point are presented in a companion report [6]. This report documents the results of the identification case study that ensued the second point. Additional motivation for the investigation came from the following desires:

- to demonstrate the utility of OMOLA in defining multiple models as proposed in [1],

- to gain practical experience using the identification tools with real industrial data.

In particular, the availability of six sets of open-loop data [7, see Figure 1.1] provides a unique opportunity to assess the structural fidelity of the deterministic models by considering parameter reproducibility.

Parameter Reproducibility

Bohlin [8, 9] discusses a number of approaches in gauging the performance of model structures. The most stringent is what he calls parameter reproducibility. This is a measure of the variation of parameter optimization results derived from a number of independent datasets. A reasonable premise for such a comparison requires that we limit consideration to only those parameters which are expected to be constant.

A test of this severity is deemed appropriate for the family of model structures under consideration because of its deterministic, thermodynamic basis. Most of the parameters have physical meaning and several are physical constants, e.g. the metal masses. Furthermore, parameter reproducibility provides an analysis tool for assessing the validity of structural hypotheses. Through such an investigation, we can answer questions like: “Are we data modeling or actually modeling the system?”

1.1 Report Outline

The report tries to give a fair presentation of large body of work done over a long period of time, starting during the autumn of 1996. Chapter 2 gives a brief description of the drum-boiler model [5, 12], which is the subject of the case-study. Details are given about the different model structures and hypotheses used in the parameter optimization trials. Chapter 2 also gives information on how the equation export from OMOLATO IDKit was done and how the limitations in this export mechanism affects the model definition. Chapter 3 describes initial simulation trials to assess the quality of the hypotheses and the identifiability of the parameters. In Chapter 4 the results from the parameter optimization trials are explained and finally, Chapter 5 gives some conclusions on what was learned from the case-study.

Details on derivations, model definition and parameter optimization trials are given in the appendices.
Figure 1.1: Six datasets from Eklund's Öresundsverk experiments [7]. Three controlled signals were perturbed separately under both full (90%) and partial (50%) load conditions.
1.2 Acknowledgment

The work was supported in part by Sydkraft AB under the auspices of Project 391, Modelling and Control of Energy Systems.
Chapter 2

Model Definition

The model structures studied in this report are a result of work by Professors Karl Johan Åström and Rodney Bell [2–5]. To verify their derivations, we used MAPLE (software for computer algebra) to rederive and study the modeling equations. The output of the MAPLE worksheets is presented in Appendix A. Appendix B provides an alternate summary of the modeling equations, i.e. the OMOLA definitions for the family of model structures.

The outline for this chapter is as follows: In the following section, we summarize Åström and Bell's results and suggest a few alternate hypotheses to some of their heuristic models. Section 2.2 surveys the model structures and how OMOLA simplified their definition and administration. Section 2.3 summarizes the parameterization of the model equations and the OMOLA simulation interface. The later defines collectively the calibration constants associated with the experimental data. Sections 2.4 and 2.5 discuss respectively the prerequisites and procedure of exporting the model equations from OMSIM [10] and generating C-code used by IdtKit [6, 11]. Finally, Section 2.6 makes some closing remarks regarding the model definition stage of this project.

![Figure 2.1: Ideal physical model of a steam generation process. Shown are the global control volume (I), and three component control volumes for the risers (II), the down-comers (III) and the liquid-zone of the drum (IV).](image-url)
2.1 Model Equations and Hypotheses

An idealized physical model for the system is shown in Figure 2.1. Steam vapor is vented from the drum with flowrate \( q_v \). Feed-water enters the drum in a sub-cooled liquid state with flowrate \( q_{fw} \) and temperature \( T_{fw} \). Combustion of fuel gives rise to the heat flowrate \( Q \) into the risers. Steam vapor is generated by channeling the liquid phase from the drum through the down-comers/risers circuit. The flowrate into this circuit \( q_{dc} \) is driven by the density gradient caused by the phase change in the risers. At the risers outlet, the two-phase mixture is characterized by the mass flowrate \( q_r \) and vapor mass-fraction \( x_r \).

The fundamental modeling simplification is that the two phases of water inside the system are everywhere in a saturated thermodynamic state. With this assumption, all thermodynamic properties can be characterized by one independent variable. The drum pressure \( P \) is chosen to be this key state variable since it is the most globally uniform variable in the system. Another key assumption is an instantaneous and uniform thermal equilibrium between water and metal everywhere. This simplifies including thermal capacitance effects.

Indicated in Figure 2.1 are the boundaries of four thermodynamic control volumes. Mass and energy balances for the global control volume (c.v. I) yield two state equations; see Section A.2. The state variables are pressure \( P \) and the total volume of liquid water in the system \( V_{sw} \). By combining the mass and energy balances for c.v. II to eliminate the flowrate \( q_r \), a third state equation is derived with the vapor mass-fraction \( x_r \) as state variable; see Section A.3. By considering fluid friction in c.v. III, a fluid momentum balance establishes the flowrate \( q_{dc} \). A combination of the mass and energy balances for c.v. IV yields a fourth state equation with state variable \( V_{sd} \), the volume of steam vapor below the liquid surface; see Section A.4. Assembled in matrix notation, the fourth-order model structure (i.e. a set of parameterized implicit differential state equations) is:

\[
\begin{align*}
\mathcal{M}_4 & : \begin{bmatrix}
e_{11} & e_{12} & 0 & 0 \\
e_{21} & e_{22} & 0 & 0 \\
e_{31} & e_{32} & e_{33} & 0 \\
e_{41} & e_{42} & e_{43} & e_{44}
\end{bmatrix} \begin{bmatrix}
\frac{\partial}{\partial t} V_{sw} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r \\
\frac{\partial}{\partial t} V_{sd}
\end{bmatrix}
= \begin{bmatrix}
q_{fw} - q_s \\
Q + h_{fw} q_{fw} - h_s q_s + \Delta_l \\
Q - (h_s - h_w) x_r q_{dc} + \Delta_{II} \\
\frac{V_{sd} - V_0}{\tau_{sd}} + \frac{(h_{fw} - h_w) q_{fw} + \Delta_{IV}}{\rho_s (h_s - h_w)}
\end{bmatrix}
\end{align*}
\]

The elements of the coefficient matrix \( e_{11}, e_{12}, e_{21}, \) etc., are state dependent. They are all listed in Appendix A on pages 41 and 55. On the right, \( \Delta_l, \Delta_{II} \) and \( \Delta_{IV} \) represent under-modeling, i.e. unmodeled energy interactions (nominally taken to be zero). The initial state conditions are parameterized \([V_{sw}^0, P^0, x_r^0, V_{sd}^0]^T\). In addition to these, the model involves seven physical parameters: metal masses \( m_d, m_r, m_{sd} \), volumes \( V_d, V_r, V_{sd} \), and a fluid friction coefficient in the down-comers \( h_f \). Known constants are the specific heats \( C_{fw} \) and \( C_p \) for the feed-water and metal respectively. Finally, note that the third-order model structure \( \mathcal{M}_3 \) corresponds exactly to the first three state equations in Equation 2.1.

For the purpose of level control, Åström and Bell [5] proposed the following measurement model for the level in the drum (see Section A.3.1):

\[
l = \frac{V_{sw} + V_{sd}}{A_d} - L_0.
\]

The level variation \( \delta l \) is caused by variations in the volumes of liquid in the drum \( V_{sw} \) and the steam below the surface \( V_{sd} \). This model introduces two additional physical parameters: \( A_d \), the drum’s cross-sectional area at the nominal level, and \( L_0 \), nominal level offset. The aim of including variation in \( V_{sd} \) is to capture the level dynamics known as the “shrink-and-swell” effect [5]. Note that we amend Equation 2.2 in Section 3.2.

To assess the necessity of including the fourth state equation in \( \mathcal{M}_4 \), we shall investigate parameter optimization of both third and fourth-order model structures. In the third-order structure \( \mathcal{M}_3 \), the state variable \( V_{sd} \) in equation Equation 2.2 is replaced with an instantaneous value. Engineering judgment suggests several approximations for its value:

\[
V_{sd} = \begin{cases}
0 & \text{hypothesis 0,} \\
b_1 + b_2 (q_{fw} - q_s) & \text{hypothesis 1,} \\
b_1 x_r q_r / \rho_s + b_2 (q_{fw} - q_s) & \text{hypothesis 2,} \\
b_1 x_r q_r / \rho_s + b_2 q_{dc} / \rho_s & \text{hypothesis 3,} \\
b_1 x_r q_r / \rho_s + b_2 q_{dc} / \rho_s & \text{hypothesis 4.}
\end{cases}
\]

\( V_{sd}(t) = V_{sd}(t) - V_{dc} - (1 - \alpha_r(t)) V_r \), where \( \alpha_r \) is the total volume fraction of steam in the risers, i.e. \( \alpha_r = V_{sr} / V_r \). An approximation with form \( \alpha_r = \text{fcn}(P, x_r) \) is given in [3, 4]; see also Section A.3.3.

\[1\]
The fourth-order structure $\mathcal{M}_4$ instead involves a similar set of hypotheses for the bubble-residence time-constant $\tau_{sd}$. The heuristics for the values which have been tested are:

$$
\begin{align*}
\tau_{sd} &= \begin{cases} 
 b_1 \rho_s \frac{(V_d - V_{ud})}{q_s} & \text{hypothesis 0}, \\
 b_1 \rho_s \frac{(V_d - V_{ud} - V_{sd})}{q_s} & \text{hypothesis 1}, \\
b_1 \rho_s \frac{(2V_{sd} - V_{sd})}{x_r/q_r} & \text{hypothesis 2}, \\
b_1 \rho_s \frac{V_{sd} - x_r}{q_r} & \text{hypothesis 3}.
\end{cases}
\end{align*}
$$

(2.4)

These hypotheses are programmed in Listings B.12 and B.14 using the technique proposed in [1] for programming multiple realizations. Note that $b_1$ and $b_2$ are "grey-box" parameters to be optimized.

Bell and Áström have also proposed a fifth order model structure [12]. This was motivated by the desire to capture an additional "rapid-swell" phenomena present in some of the measured data (see the non-minimum phase behavior in the level variation of experiments A and F in Figure 1.1). The extra dynamics model the transport delay of steam vapor passing from the risers to the drum surface. The transport delay with time constant $\tau_{sd}$ is approximated by a Padé(0,1) approximation:

$$
\phi(-\tau_{sd}) = \frac{1}{1 + \tau_{sd} s} + O(s^2)
$$

Augmenting the state space representation of this approximation yields the following set of implicit differential equations:

$$
\begin{align*}
\mathcal{M}_5 : & \\
\begin{bmatrix}
e_{11} & e_{12} & 0 & 0 & 0 \\
e_{21} & e_{22} & 0 & 0 & 0 \\
0 & e_{32} & e_{33} & 0 & 0 \\
e_{41} & e_{42} & e_{43} & e_{44} & 0 \\
0 & 0 & 0 & 0 & \tau_{sd}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial}{\partial t} V_{wt} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r \\
\frac{\partial}{\partial t} V_{sd} \\
\frac{\partial}{\partial t} q_{sd}
\end{bmatrix} &= \\
\begin{bmatrix}
q_{fw} - q_s \\
Q + h_{fw} q_{fw} - h_s q_{th} + \Delta I \\
Q - (h_s - h_{aw}) x_r q_{de} + \Delta II \\
\frac{V_{sd} - V_{sd}^2}{\tau_{sd}} + \frac{(b_{h_s - h_{aw})} q_{fw} + \Delta \rho}{\rho_0 (h_s - h_{aw})} + \frac{\tau_{sd} q_{de}}{\rho_0} \\
x_r q_r - q_{sd}
\end{bmatrix}
\end{align*}
$$

(2.5)

The physical interpretation of the fifth state variable, $q_{sd}$, is the vapor mass flow rate crossing the liquid surface. Note the appearance of $q_r$ in the right-hand side of equations. The mass balance for the risers (see Section A.3.2) gives the following expression:

$$
q_r = q_{de} - \frac{\partial}{\partial t} M_r = \text{fcn} (\frac{\partial}{\partial t} P, \frac{\partial}{\partial t} x_r, \ldots).
$$

In order to maintain compatibility with the definitions for the coefficients $e_{42}$ and $e_{43}$ in $\mathcal{M}_4$, we chose to leave $q_r$ on the right-hand side (rather than bringing all time-derivative dependencies to the left-hand side). Consequently, derivation of explicit state equations (ODE’s) is more complex than the bottom row of the coefficient matrix in Equation 2.5 suggests. Fortunately OMsSim automates this task using Cramer’s rule, affording us the luxury of the more compact definition as implicit state equations.

### 2.2 Model Structure Administration

As alluded to above, OMOL greatly simplified the task of defining and administering model structures. Table 2.1 and Figure 2.2 on the facing page provide an overview of the model structures we have programmed. Besides the three model structures $\mathcal{M}_0, \mathcal{M}_4,$ and $\mathcal{M}_5$, we defined and tested a number of alternative structures. Not all have been (or can be) used in parameter optimization. Listings for most are given in Appendix B.

### 2.3 Model Parameterization and Simulation Interface

The parameter definitions are concentrated in the OMOL classes: OresundSimIC and Boiler2PM (see Listings B.11 and B.12 respectively). Table 2.2 on the next page summarizes the model parameterization and rates qualitatively our prior knowledge of the parameter values. Using a scale from...
2.3 Model Parameterization and Simulation Interface

<table>
<thead>
<tr>
<th>Model Structure</th>
<th>OMOLA Class</th>
<th>State Variables</th>
<th>Delay Var's</th>
<th>Descriptive Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_2$</td>
<td>Boiler2FM</td>
<td>$V_{ut}$, $P$, $x_r$, $V_{sd}$, $V_{wd}$</td>
<td>$q_{ct}$, $q_{sd}$</td>
<td>second order</td>
</tr>
<tr>
<td>$M_3$</td>
<td>Boiler3FM</td>
<td>+ +</td>
<td></td>
<td>third order</td>
</tr>
<tr>
<td>$M_4$</td>
<td>Boiler4FM</td>
<td>+ + + +</td>
<td></td>
<td>fourth order</td>
</tr>
<tr>
<td>$M_{dr}$</td>
<td>Boiler1rFM</td>
<td>+</td>
<td></td>
<td>reduced second order</td>
</tr>
<tr>
<td>$M_d$</td>
<td>Boiler2rFM</td>
<td>+ + + +</td>
<td></td>
<td>reduced third order</td>
</tr>
<tr>
<td>$M_{dr}$</td>
<td>Boiler3rFM</td>
<td>+ + +</td>
<td></td>
<td>reduced fourth order</td>
</tr>
<tr>
<td>$M_{sd}$</td>
<td>Boiler3wdFM</td>
<td>+ + +</td>
<td>+</td>
<td>alternate state realization</td>
</tr>
<tr>
<td>$M_{d2}$</td>
<td>Boiler3iFM</td>
<td>+ + +</td>
<td></td>
<td>iterative initialization</td>
</tr>
<tr>
<td>$M_{d3}$</td>
<td>Boiler3dFM</td>
<td>+ + + +</td>
<td></td>
<td>multiple delay realizations</td>
</tr>
<tr>
<td>$M_{d4}$</td>
<td>Boiler4dFM</td>
<td>+ + + +</td>
<td>+</td>
<td>multiple delay realizations</td>
</tr>
<tr>
<td>$M_5$</td>
<td>Boiler5FM</td>
<td>+ + + +</td>
<td>+</td>
<td>Padé(0,1) delay approx.</td>
</tr>
<tr>
<td>$M_6$</td>
<td>Boiler5aFM</td>
<td>+ + + +</td>
<td>+</td>
<td>Padé(1,1) delay approx.</td>
</tr>
<tr>
<td>$M_7$</td>
<td>Boiler6FM</td>
<td>+ + + +</td>
<td>+</td>
<td>Padé(1,2) delay approx.</td>
</tr>
</tbody>
</table>

Table 2.1: Overview of the model structures defined using OMOLA.

![Inheritance hierarchy of the boiler model structures.](image)

Figure 2.2: Inheritance hierarchy of the boiler model structures.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>OMOLA</th>
<th>Units</th>
<th>Description</th>
<th>Prior Knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_{scf}$</td>
<td>$q_{scf}$</td>
<td>kg/s/ton/hr</td>
<td>steam flow rate calibration factor</td>
<td>light grey</td>
</tr>
<tr>
<td>$q_{f}$</td>
<td>$q_{f}$</td>
<td>MW/ton/hr</td>
<td>fuel flow rate calibration factor</td>
<td>grey</td>
</tr>
<tr>
<td>$\sigma_d$</td>
<td>$\sigma_d$</td>
<td>kg/cm²</td>
<td>standard deviation of pressure output error</td>
<td>grey</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>$\sigma_r$</td>
<td>mm</td>
<td>standard deviation of drum level output error</td>
<td>grey</td>
</tr>
<tr>
<td>$m_d$</td>
<td>$m_d$</td>
<td>kg</td>
<td>metal mass of the drum</td>
<td>light grey</td>
</tr>
<tr>
<td>$m_r$</td>
<td>$m_r$</td>
<td>kg</td>
<td>metal mass of the risers</td>
<td>light grey</td>
</tr>
<tr>
<td>$A_d$</td>
<td>$A_d$</td>
<td>m²</td>
<td>cross-sectional area of the drum</td>
<td>light grey</td>
</tr>
<tr>
<td>$k_f$</td>
<td>$k_f$</td>
<td>–</td>
<td>fluid friction factor in the down-comers</td>
<td>light grey</td>
</tr>
<tr>
<td>$k_s$</td>
<td>$k_s$</td>
<td>kg/s/MPa</td>
<td>compressibility coefficient in the steam valve</td>
<td>light grey</td>
</tr>
<tr>
<td>$L_0$</td>
<td>$L_0$</td>
<td>m</td>
<td>slack factor in the drum level model</td>
<td>light grey</td>
</tr>
<tr>
<td>$b_1$</td>
<td>$b_1$</td>
<td>–</td>
<td>slack factor in $V_{sd}$ and $\tau_{sd}$ hypotheses</td>
<td>light grey</td>
</tr>
<tr>
<td>$b_2$</td>
<td>$b_2$</td>
<td>m³/kg/s</td>
<td>slack factor in $V_{sd}$ hypotheses; $V_{sd}$ in $M_4$ &amp; $M_5$</td>
<td>light grey</td>
</tr>
<tr>
<td>$\delta V_{ut}$</td>
<td>$\delta V_{ut}$</td>
<td>m³</td>
<td>perturbation in the initial condition of $V_{ut}$</td>
<td>light grey</td>
</tr>
<tr>
<td>$P^0$</td>
<td>$P^0$</td>
<td>MPa</td>
<td>initial condition of the drum pressure $P$</td>
<td>light grey</td>
</tr>
<tr>
<td>$x_r^0$</td>
<td>$x_r^0$</td>
<td>–</td>
<td>initial condition of the vapor mass fraction $x_r$</td>
<td>light grey</td>
</tr>
</tbody>
</table>

Table 2.2: Overview of the model parameterization and prior knowledge.

Dept. of Automatic Control
J. Sarlø, J. Eborn
LUTFD2/TFRT-7563-SE
Lund Institute of Technology
1998.08.14
black to white, black indicates little or no help from engineering physics. At the opposite end of the spectrum, white indicates a high degree of certainty.

Figure 2.3 shows the simulation interface to the five measured inputs: two steam flows, feed-water flow, feed-water temperature and fuel flow. The interface with real data necessitates conversion factors; the simulation schematic shows several. Most uncertain is the calibration of the heat input $Q$. Because the chemical energy content of the fuel is known to vary, this gain has been probabilistically modeled with a nominal value $q_{f/I}$ and a known, bounded range $q_{f/m}$. More certain are the calibrations of the steam mass flowrates $q_{s_1}$ and feed-water mass flowrate $q_{f/w}$. We shall consider the later a known constant, assuming liquid flow measurements are more precise than vapor flow measurements. The conversion of steam flowrate is complicated by the fact that the measurements are volume flowrates, which need to be converted to mass flowrate. Since steam is a compressible medium, the conversion is pressure dependent. This can be compensated for by including a compressibility factor, $k_s$, which can also be seen as a "steam valve nonlinearity". Similar to [7] this has been modeled as

$$q_s = q_{s_1}(q_{s1} + q_{s2}) + k_s(p - p^0) \quad (2.6)$$

The compressibility factor takes into account that at higher pressures the density of steam is higher and thus the mass flow rate is higher. These choices give $q_{f/I}$, $q_{s_1}$ and $k_s$ as additional parameters for optimization.

In addition to the stochastic modeling of the gain $q_{f/I}$, the simulation interface includes simple stochastic input and output-error models. The focus of this study is parameter optimization in a deterministic setting. Accordingly, only the instantaneous output-error models will be investigated here. In a possible continuation of this work, the input-error models could be used to investigate the effects of under-modeling, i.e. $\Delta I$ etc.

### 2.4 Signal Model Specification

A requirement for the automatic code generation is a name mapping that defines inputs, outputs etc. For the OMSim IDKTr interface developed as part of this project (see [6]) this was solved by requiring a `SignalModelMapping` submodel be included in the simulation model. Listing B.9 provides the template. This class defines the vector variables and a parameter matrix of a standard nonlinear stochastic state space model structure (see [1, 11]):

$$\frac{dx}{dt} = f(t, x_t, u_t, \theta) + w(t, \theta) \quad \text{and} \quad y_t = h(t, x_t, u_t, \theta) + v(t, \theta).$$

The `SignalModelMapping` class is used by including it in the simulation model, and defining the vector dimensions and element-wise contents. This constitutes a name mapping from the models internal
hierarchical names to the standard state space vectors $u$, $v$, $w$, $x$ and $y$, as well as the parameter matrix $\theta$.

The row-by-row organization of the matrix $\theta$ is important. When using IDKIT, the user fixes or frees entire rows at a time—it is not possible to fix and free elements row-wise. Only parameters for which simultaneous optimization is essential should be grouped row-wise. Furthermore, IDKIT uses the same numerical scaling factor for all elements of a row.

The second-order simulation model of\textit{resund2} shown above and listed on page 77 provides an example of the signal model mapping. Lines 8–19 define the vector variable mappings. The first four rows of the parameter matrix mapping (lines 24–27) collect those parameters which we have no intentions of optimizing yet we do not wish to hard-wire their values into the automatically generated C-code. The remaining rows (lines 28–46) map the parameter names which we (potentially) wish to optimize. Note the correspondence between these lines of OMOLA code and Table 2.2. No parameters were grouped row-wise because of the need for distinct scaling factors.

2.5 Equation Export and Code Generation

The process of turning OMOLA model equations into C-code functions that can be used by the parameter optimization routines in IDKIT consists of two steps:

(i) exporting sorted model equations from OMSIM, and then
(ii) invoking a shell script that translates the equations and uses Maple to generate C-code.

These steps and also limitations concerning the use of conditional statements in the model are described fully in [6]. For the sake of self-containment, a short description follows.

Exporting Omola Equations

Th IDKIT model interface requires a set of nonlinear ordinary differential equations in explicit form. In our case we defined the model in an implicit differential equation form. By configuring OM$\text{SIM}$ to use Cramer's rule during the instantiation the model is automatically transformed to explicit ODE form. This is set by choosing the “Config$\rightarrow$Options” menu and marking the “Cramer's rule for manipulation” option before starting the simulator. We also need to mark “Log to file” option to create the log-file.

To generate the sorted equations we use the OM$\text{SIM}$ simulators “Flat model” and “Event part” debug output. This writes all the continuous equations as well as parameter equations and event equations to the log-file. In the event part, only initialization events are allowed. This restriction is
necessary since the code generation script (described below) is designed only to handle continuous time models.

Code Generation Script

The model equations captured in the log-file are processed by running the Unix shell-script Code-Gen.sh in the same directory as the log-file. This shell-script automatically performs the steps of:

(i) OMSIM log file translation,
(ii) MAPLE post-processing of the equations, and
(iii) MAPLE code generation.

The script takes no arguments and, if successful, it creates two files in the working directory:

model.c — a subroutine module for use with IdKit,
Smodel.c — a SIMULINK S-function for MATLAB.

Event Equation Limitations

It is important to note the limitations of the code generation script. With regard to the event equations, i.e. the “Event part” of the log-file, we mentioned that only initialization events are supported. A further limitation is that these events cannot form an iterative sequence of equations.

To convey this restriction, we compare the initialization programming in the model structures \( M_2 \) and \( M_{3i} \). The latter is a special iteratively-initialized derivation of \( M_3 \). The motivation behind this is discussed in Section 3.2.4. The sequence is derived in Section A.3.7. Note that \( M_{3i} \) inherits the events of \( M_3 \).

Considering \( M_2 \) first, observe that lines 124 and 142 in Listing B.12 involve the same equation:

\[
\text{new}(V_{\text{wt}}) := 1/2*V_d + (1-ar_0)*V_r + V_{dc} + dV_{\text{wt}}0;
\]

In contrast, the \( M_{3i} \) equations:

\[
\text{new}(x_r) := x_r0;
\]

\[
\text{new}(x_r) := -ar/dardx + x_r/2 + (1/2 hc/dardx/qdc)*sqrt((hc*qdc*(4*hc*qdc*ar*(ar-3*dardx*xr)) + hc*qdc*(dardx*xr)^2 + 8*Q*ar*dardx) );
\]

programmed in lines 28 and 12–14 of Listings B.13 and B.22 respectively constitute an iterative sequence.

The cause of the restriction is simple. The code generation script translates the OMSIM equations into MAPLE assignment statements. For an iterative sequence, this results in repeated assignments to the same name (i.e. the same left-hand-side symbol). When loaded into MAPLE, no iterative substitution takes place. Instead, only the last assignment survives.

2.6 Closure

Since the definition of the model structures in OMOLA constituted a substantial amount of time, it seems relevant to recap some of issues raised in the process. Questions raised while programming the OMOLA model definitions are:

- How important is the thermal capacitance effects of the metal mass? Where is it most important to include these effects in the modeling equations?

- What are the consequences of using enthalpy instead of internal energy in deriving the energy balances? How valid is this approximation?

Answers to these questions can be found through simulation testing and parameter optimization, the topics of the following two chapters.
Chapter 3

Simulation Testing

More than anything else, this case study has demonstrated the importance of good simulation tools. When conceptualizing a methodology for grey-box identification (see e.g. [1, 13]) it is easy to over-emphasize the importance of statistical methods. One should not under-estimate what can be learned directly through iterative simulation and analysis. This synthesis of information is what we refer collectively to as "simulation testing." Indeed, the knowledge gained through simulation testing can be essential for achieving success with statistical methods. Conveying this lesson is one reason we include this chapter.

The outline for the chapter is as follows: Section 3.1 discusses the role that manual parameter tuning has played in this case study. As a tutorial example, we consider calibration of the model to the experimental data. Section 3.2 presents a qualitative assessment of parameter sensitivities, couplings and over-parameterization. Through simulation testing, we were able to identify where this occurs in the modeling equations. These findings are largely responsible for the iterative refinements of the model structures described in the preceding chapter. Section 3.3 demonstrates how simulation testing can answer questions of model structure, i.e. structure determination. Specifically, we answer the question concerning the validity of replacing internal energy with enthalpy in the derivation of energy balance.

3.1 Manual Parameter Tuning

Manual tuning of parameters is important for a number of reasons. Most obvious is the need to determine values for those parameters that cannot be established from construction data. To even begin a simulation let alone parameter optimization, we need a functional set of numeric values for the parameters. By functional, we mean in the sense that numerical integration is not driven unstable. In case study, manual experimentation was important for two reasons:

(i) to establish good "guesses" suitable for initializing computer-aided tuning, and
(ii) to "feel out" parameter sensitivities as well as couplings between parameters.

With computer-aided tuning, we mean of course parameter optimization. For iterative methods of parameter estimation, e.g. as implemented in IdKrr, good initial guesses greatly aid in speeding convergence to the optimum. Prior knowledge of the parameter sensitivities, however rudimentary, guides the choice of which parameters to fix as constants and which to free for optimization. In fact one does this in stages, freeing more and more parameters sequentially. Ideally, the model becomes more "tuned" to the data at each stage. We will elaborate on this procedure in the next chapter. For now, we concern ourselves tutorially with manual tuning.

3.1.1 An Example: Balancing the Flux of Mass and Energy

To begin, recall the nature of the global mass balance. Accumulation equals the flux in minus the flux out i.e. the feed-water flow rate $q_{fw}$ minus the vented steam flow rate $q_v$. In terms of Eklund's measured signals and calibration factors, we have:

$$
\frac{dM}{dt} = q_{fw} \mu_0 - q_v \mu_1 \\
= \text{fcn}(u, \theta) \neq \text{fcn}(x, u, \theta).
$$
Simulation Testing

Here \( x, u \) and \( \theta \) represent generically the state variables, measured inputs and parameter respectively. The last line above emphasizes the pure integral nature of the system. System stability requires a net balance of the mass flux. The same can be said of the global energy balance, even though it is not purely integral in nature, i.e.:

\[
\frac{dU}{dt} = q_{\text{in}}u_2 + h_{ws}q_{\text{in}}u_3 - h_sq_{\text{in}}u_1 = \text{fcn}(x, u, \theta).
\]

Recall in Section 2.3 we chose based on the accuracy of liquid and vapor flow measurements (i) to fix \( q_{\text{in}} \) and (ii) to leave \( q_{\text{in}} \) free for parameter optimization. Clearly this choice was essential for identifiability.

Similarly, we left free \( q_{\text{in}} \), the calibration factor relating heat transfer to the measured fuel flow rate.

Engineering know-how guides the manual tuning of \( q_{\text{in}} \) and \( q_{\text{in}} \). We know Eklund’s experiments were conducted under near-equilibrium operating conditions. Logically, one must first balance the inlet and outlet mass flows \( q_{\text{in}} \) and \( q_{\text{in}} \). This is done by adjusting \( q_{\text{in}} \). The state variable \( V_{\text{in}} \) is fairly representative of the total mass in the system. Once the net mass flux is balanced, one can balance the energy flux by adjusting \( q_{\text{in}} \). The state variable \( P \) is a good indicator of the total energy in the system.

Figure 3.1 shows results of manually calibrating \( q_{\text{in}} \) and \( q_{\text{in}} \) to the dataset from experiment E. Starting with \( q_{\text{in}} = 0.26 \) and \( q_{\text{in}} = 5.6 \), the following adjustments are shown in Figure 3.1(a):

\[
q_{\text{in}} : 0.26 \rightarrow 0.27 \rightarrow 0.265 \rightarrow 0.267 \rightarrow 0.266 \rightarrow 0.2665
\]

With the mass flux more or less balanced, the following adjustments were made:

\[
q_{\text{in}} : 5.6 \rightarrow 5.61 \quad q_{\text{in}} : 0.2665 \rightarrow 0.267 \quad q_{\text{in}} : 5.61 \rightarrow 5.62 \rightarrow 5.615
\]

These are shown in Figure 3.1(b). In this plot, the measured pressure signal is shown as a dashed line. Note that manual tuning (by mortals) is best limited to tweaking one parameter at a time.

<table>
<thead>
<tr>
<th>Regime</th>
<th>Experiment</th>
<th>( q_{\text{in}} )</th>
<th>( q_{\text{in}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>full load</td>
<td>E</td>
<td>0.267</td>
<td>5.612</td>
</tr>
<tr>
<td>partial load</td>
<td>A, B</td>
<td>0.253</td>
<td>5.575</td>
</tr>
<tr>
<td>partial load</td>
<td>J</td>
<td>0.253</td>
<td>6.205</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>0.253</td>
<td>6.121</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>0.257</td>
<td>6.121</td>
</tr>
</tbody>
</table>

Table 3.1: The manually tuned calibration factors used as starting point for optimization.

Table 3.1 summarizes the manually tuned values that were used as initial guesses in computer-aided optimization trials. Note that because of changes in Eklund’s experimental conditions, e.g.

Dept. of Automatic Control
LUTFD2/TFRT-7563-SE
J. Sørlie, J. Eborn
Lund Institute of Technology
1998.08.14
3.2 Parameter Sensitivities and Couplings

With nonlinear model structures, it is difficult to assess the global nature of a manually tuned parameter. As demonstrated in the preceding example, changes in one parameter affect the tuning of others. Nevertheless with a little ingenuity it is possible to get a qualitative feel for parameter sensitivities.

In tuning calibration factors for example, the effects of thermal capacitance were not included. This was done by setting the metal masses equal to zero. (see lines 80–82, and 117 in Listing B.12, lines 21 and 12 in Listings B.13 and B.14 respectively). This increases the models sensitivity to \( q_{c,\ell} \) and \( q_{r,\ell} \), and greatly affects the perceived results in manually tuning them. Including thermal capacitances effectively slows down the dynamics of the energy balance. Graphically one can imagine the numerical explosions (due to imbalance in mass flux) shown in Figure 3.1(a) on the facing page elapsing at a slower pace. That the thermal energy effects affect the mass balance is evidence of the implicit couplings between the global mass and energy balances; cf. page 33 in Section A.2.

3.2.1 Couplings with Fluid Friction in the Down-Comers

The most uncertain parameter in the model is \( k_f \), the coefficient of the fluid friction. This parameter enters the modeling equations via \( q_{dc} \), the mass flow rate entering the down-comers. In both formulations of \( q_{dc} \) proposed by Åström and Bell [3, 12], \( k_f \) enters functionally in the denominator under a square root:

\[
q_{dc} \propto \frac{1}{\sqrt{k_f}}.
\]

The precise relations are derived via momentum balances in Section A.3.4 and programmed in lines 154–170 in Listing B.12.

To establish qualitatively the model’s sensitivity with respect to \( k_f \) via simulation testing, we perturbed \( k_f \) and looked for effects in the simulated response. Figure 3.2 shows the most notable signal couplings. These can be summarized as follows:

\[
\downarrow k_f \quad \Rightarrow \quad \uparrow q_{dc}, \quad \downarrow x_r, \quad \downarrow \alpha_r.
\]

For a quasi-steady rate of heat transfer to the risers, increasing the mass flow rate through the risers (\( q_{dc} \)) decreases the vapor generation (\( x_r \) and \( \alpha_r \)). Thus, the model agrees qualitatively with our physics-based engineering intuition.

The last coupling, between \( k_f \) and \( \alpha_r \), is significant in the third-order model structure proposed in [3, 4]. Recall in Equation 2.2 that to capture the shrink-and-swell effect, Åström and Bell incorporated the volume of vapor below the surface, \( V_{sd} \). In order to decouple \( V_{sd} \) from \( k_f \), we inserted the multiplicative grey-box parameter \( b_1 \) to Åström and Bell’s instantaneous heuristic (see Equation 2.3 and Section A.3.1). In essence, we tested:

\[
V_{sd}(t) = b_1 \alpha_r(t)V_r.
\]

This adds an extra degree of freedom to the measurement model, allowing magnification of the dynamics contained in \( \alpha_r \), independent of the value of \( k_f \).

3.2.2 Over-Parameterization of the Drum Level Measurement Model

When choosing what parameters to optimize, over-parameterization is an important issue. This is very common in model structures based on first principles. For example, thermal energy storage in metal depends on \( mC_p \) i.e. a product of two constants. These parameters cannot be estimated independently. More subtle over-parameterizations are easily overlooked.

In this study we set out to estimate the friction factor, \( k_f \). It was deemed impossible since \( k_f \) seemed mainly to affect the static offset in the drum level. To account for this static error, we introduced the parameter \( L_0 \) in Equation 2.2 on pg. 5. Further scrutiny of the original level model
Simulation Testing

Figure 3.2: A simulation test that assesses the model’s sensitivity to the coefficient of fluid friction. Note that $k_f$ modulates not only the mean value, but also the peak-to-peak variation of the other signals.

revealed a subtle over-parameterization. Eliminating $L_0$ from the equation led to the following variational measurement model:

$$\delta l(t) = \frac{\left( V_{wd}(t) - V_{wd}(0) \right) + \left( V_{sd}(t) - V_{sd}(0) \right)}{A_d}. \quad (3.2)$$

In essence, we replaced the nominal drum level $L_0$ with a function of $V_{wd}(0)$ and $V_{sd}(0)$ which are themselves functions of the initial state conditions. This indirect parameterization of $L_0$ agrees logically with the physical assumption of near-equilibrium operation during the experiments. Unfortunately, the over-parameterization was only part of the problem and its elimination does not affect the identifiability of $k_f$. What the variational level model helped clarify is the dependency of the level model on the initial state conditions.

3.2.3 Couplings due to Non-Equilibrium State Initialization

Much to our chagrin, after switching from Equation 2.2 to Equation 3.2, drum level simulations still showed a static offset. More simulation testing revealed the source of the error: non-equilibrium initialization of the third state $x_r(t)$. Thus there is a coupling between $k_f$ and the initial state condition $x_0^3$.

From Figure 3.2, we know the effect changing $k_f$ has on $x_r$. Figure 3.3 shows the effect of perturbing $x_0^3$. Summarizing this later figure, we have:

$$\uparrow x_0^3 \quad \Rightarrow \quad \downarrow V_{wd}(0), \quad \uparrow V_{wd}(t), \quad \downarrow V_{sd}(0), \quad \downarrow \delta l(t).$$

The initialization of $x_r$ affects the state initialization of $V_{wd}$ through the non-linear relationship for $\alpha_r$. This is best explained by looking at the OMOLA code — look for $ar0$ in lines 124 and 142 in

Dept. of Automatic Control
J. Solie, J. Eborn
Lund Institute of Technology
3.2 Parameter Sensitivities and Couplings

Figure 3.3: Effect of non-equilibrium initialization on the drum level model: For any fixed value of \( k_f \), a non-equilibrium initialization of state \( x_r(t) \) causes static error in \( \delta l(t) \).

Listing B.12. The first-order transients seen in \( V_{ad}(t) \) and \( V_{sd}(t) \) are the dynamic response due to the non-equilibrium initialization of \( V_{ad} \). As before this is best understood by examining the ÖMÖLA code — the discrete initialization sequence programmed in lines 120–143 of the above mentioned listing. We note that repeating lines 130–137 in this sequence as a fourth event (timed after the Start event) has no effect on the simulation results shown in Figure 3.3.

The implication of the coupling is this: In order to maintain a near-equilibrium initialization, a change in \( k_f \) requires a corresponding change in \( x_r^0 \) i.e.:

\[ \uparrow k_f \quad \text{requires} \quad \uparrow x_r^0. \]

The exact correspondence is analytically characterized by the equilibrium condition provided by the third state equation (see Equation 2.1 or Section A.3.2):

\[ -h_e(0) x_r^0 q_{dc}(0) + Q(0) = 0. \]  \hfill (3.3)

Including the parameter \( x_r^0 \) in a model is thus an over-parameterization. Any attempts at optimizing \( k_f \) require the simultaneous adjustment of \( x_r^0 \). Simultaneous unconstrained optimization of the coupled parameters is an ill-posed problem, hence \( k_f \) becomes unidentifiable. Finally, we note that because the coupling is transmitted through both \( V_{ad} \) and \( V_{sd} \) (see Equation 3.2), the problem does not go away by switching to the fourth-order model structure.

3.2.4 Iterative State Initialization and the Drum Level Measurement Model (Revisited)

The coupling between \( x_r^0 \) and \( k_f \) motivated us to investigate parameterized initializations for the state \( x_r \). Theoretically, Equation 3.3 could be used to parameterize \( x_r^0 = \text{fn}(P^0, Q(0), k_f) \). The nonlinear

Dept. of Automatic Control
J. Sørlie, J. Eborn

Lund Institute of Technology

LUTFD2/TFRT–7563–SE
1998.08.14
dependency of \( \eta_{dc} \) on \( x_r \) however prohibits a closed-form solution for \( x_r^2 \). An approximate iterative solution is derived in Section A.3.7 and encoded in Listing B.22; see pages 51 and 76 respectively.

As discussed in Section 2.5, an iterative sequence of discrete events cannot be translated to C-code by the MAPLE code generation tools [6]. Consequently we cannot use the iterative solution in conjunction with IDRT. Instead, we must account for any non-equilibrium state initialization. To do this we reintroduced the \( L_0 \) parameter into the drum level model. Our final revised variational measurement model is:

\[
dl(t) = \left( \frac{\nu_{ud}(t) - \nu_{ud}(0)}{A_d} \right) + \left( \frac{\nu_{ad}(t) - \nu_{ad}(0)}{A_d} \right) + L_0.
\]

Both Equations 2.2 and 3.4 appear in the OMOL model definition as multiple realizations; see lines 151–152 in Listing B.12.

### 3.3 Structure Determination

To complete this chapter, we present an example showing how questions of model structure can be answered simply and definitively through simulation testing. The example also demonstrates the utility of good simulation tools.

#### 3.3.1 An Example: Validity of Enthalpy Approximations in Energy Balances

The basic assumption is that internal energy \( u \) is well approximated by enthalpy \( h \) in the derivations of the compartmental energy balances. Since \( h = u + P/\rho \) by definition, this is equivalent to assuming \( u \gg P/\rho \). For a given operating regime, it is easy to inspect this relationship through simulation. It is however desirable to verify such a question by actually having both realizations of the model structure available for testing. This is especially true for complex nonlinear systems. Recall we have defined the model in implicit form and this leads to extremely complex nonlinear equations when put in explicit form. Furthermore, we would ideally like to have some quantitative feel for how the approximation (or rather, that which we are neglecting) enters into the equations.

By running the derivations in Appendix A with both the exact and approximate substitutions, i.e.:

\[
u = h - \frac{P}{\rho} \quad \text{and} \quad \nu \approx h,
\]

we established that the approximation affects only the coefficients \( e_{22}, e_{32} \) and \( e_{42} \). In each of these expressions, one or more "-1" components appear in the exact derivation. With a little thought, this makes sense: \( e_{22}, e_{32} \) and \( e_{42} \) are the coefficients of the time derivative of pressure \( P \) in each of the state equations. Note that energy and enthalpy do not enter into the derivation of the first state equation, hence \( e_{22} \) is not affected by the approximation. To summarize matters, we collect the OMOL programming of \( e_{22}, e_{32} \) and \( e_{42} \):

#### Listing 3.1: NonEquilibriumInit.cci—Simulation script for Figure 3.3 using dataset E and the third-order model structure (\( M_d \)); cf. Listing B.24 on pg. 77.

```c
1 BEGIN
2 Simulator s();
3 s.reset();
4 Model m(s);
5 m.qscf := 0.267366 ; n.Boiler.ku := 8.12786 ;
6 m.qctf := 5.65107 ; n.Boiler.kl := 0.67;
7 m.Signal := 0.177821 ; n.Boiler.x0 := 0 ;
8 m.Signal5 := 13.099 ; n.Boiler.b1 := 1 ;
9 m.Boiler.m := 0 ; n.Boiler.b2 := 0 ;
10 m.Boiler.ar := 345208 ; n.Boiler.dWx0 := 0 ;
11 m.Boiler.Ad := 26.4135 ; m.Boiler.FO := 10.6089 ;
12 m.Boiler.kf := 0.01 ; m.Boiler.xr0 := 0.110198 ;
13 a.start;
14 s.reset;
15 m.Boiler.xr0 := 0.10 ;
16 a.start;
17 s.reset;
18 m.Boiler.xr0 := 0.15 ;
19 a.start;
20 ENO;
```
3.3 Structure Determination

VectorVar ISA Std::VectorVar WITH
  EPS TYPE STATIC Real := 2^-52; % IEEE floating point
  abs TYPE Column[n] := abs(value);
  logabs TYPE Column[n] := ln(abs(value*EPS*ones(n,1)))/ln(10);
  sum TYPE Real;
  sum = ones(1,n)*value;
END;

e22 ISA DrumBoiler::VectorVar WITH
  n = 7;
  value = [Vst*[(hs*drsdp; rs*dsdp; -1*Ihu)];
            Vwt*[hw*dwdp; rw*dhp; -1*Ihu]; (md+mr+mdc)*Cp*dTsdp];
END;

e32 ISA DrumBoiler::VectorVar WITH
  n = 8;
  value = [Vr*[ (1-ar)*rw+drwp; (1-ar)*hr*drwp; -1*Ihu];
            (rs + (rw-rs)*x)*hc*darp ]; mr*Cp*dTsdp];
END;

e42 ISA DrumBoiler::VectorVar WITH
  n = 6;
  value = [ Vsd*[ 1/rs*drsdp; 1/hc*dsdp];
            1/rs/hc*[Vsd+rw*dhp; msd*Cp*dTsdp];
            -Ihu/rs/hc*[ Vd+Vsd; Vr*P*darp]]; % [m3/MPa]
END;

See respectively Listings B.1 and B.12-B.14. The constant \( \text{Ihu} \) serves as a binary valued switch: when equal to one, the exact derivations are active; when zero, the approximation is active. Note the value semantics of the vector variables: the `sum` attribute is actually what enters the state equations (e.g. `e22.sum*dot(P)`).

To quantitatively assess the contribution of these "\(-1\)" terms, we devised a simple test. Notice how both the terms in question and the thermal capacitance effects and the enter linearly into the left-hand side of the state equations (look for `mdot` in the mentioned listings). By first neglecting thermal effects (letting \( m = 0 \)) and then including it in a token sense (letting \( m = 1 \)), we can compare quantitatively the magnitudes of the various components. A simulation script for this is shown in Listing 3.2.

Figure 3.4 shows the results of this simulation experiment using logarithmic scales. In each of the three plots, the token thermal capacitive effects are several orders of magnitude greater than the `Ihu`-switched components. These are in turn many orders of magnitude greater than IEEE floating point epsilon, \( \text{EPS} = 2.2204e-16 \). Nevertheless, since the unit of the metal mass is kilograms, realistic values (\( m \approx 10^6 \)) are many orders of magnitude greater than the token value \( m = 1 \). In other words, the model is many orders of magnitude more sensitive to the thermal capacitive effects than the effects of the approximative enthalpy derivation. Based on this quantitative physical insight, we can safely proceed using the simplifications prescribed by Bell and Åström.
Figure 3.4: Simulation tests show the model is far more sensitive to the effects of thermal storage (i.e., metal masses) than to effects of approximating internal energy with enthalpy in the energy balance derivations.

Listing 3.2: lhuMass.ocl—Simulation testing script investigating model sensitivity to the approximative enthalpy derivation.
Chapter 4

Parameter Optimization

The provisional goal of this project was to see what improvements might be gained through parameter optimization. Our ambitions evolved into an investigation of structural fidelity due to the availability of multiple datasets. As discussed in Chapter 1, parameter reproducibility provides a quite stringent analysis tool.

The outline for this chapter is as follow: Section 4.1 discusses the design of the optimization trials and, in doing so, surveys the entire investigation. The remaining sections traverse through the results of the sequence of optimization trials.

4.1 Design of the Optimization Trials

A complete derivation of the modeling equations and their programming in OMOLA is given in Appendices A and B. Summarizing the model definition, we have third, fourth and fifth-order model structures $M_3$, $M_4$ and $M_5$ with the parameterization shown in Table 2.2.

Figure 2.3 presents the OMOLA model of the system and the simulation interface definition. The latter includes the connection to the five measured inputs: two steam flows, feed-water flow, feed-water temperature and fuel flow. An interface with real data necessitates conversion factors, these are also indicated in Figure 2.3. Most uncertain is the calibration of the heat input $Q$. Because the chemical energy content of the fuel is known to vary, this gain was modeled probabilistically with a nominal value $q_{fr}$. More certain are the calibration factors of the steam mass flowrates $q_s$ and feed-water mass flowrate $q_{fwater}$. For identifiability, one of these must be fixed. We took the later to be a known constant, assuming liquid flow measurements are more precise than vapor flow measurements.

Deterministic identification requires some form of an output error model. The simplest possible is additive discrete-time white noise. With this choice, the parameters $\sigma_4$ and $\sigma_5$ shown in Figure 2.3 represent the standard deviations of the drum pressure and level simulation errors. The magnitudes of $\sigma_4$ and $\sigma_5$ serve as gauges of the quality of the deterministic model.

4.2 Calibration Factors and Steam Valve Nonlinearity

Preliminary identification trials revealed three parameters to be essential in calibrating the model to the data. By this we mean reducing the simulation error sufficiently so that meaningful conclusions could be drawn from subsequent investigations. The three key parameters are $q_{sat}$, $q_{f0}$ and $k_s$. Figure 4.1 shows the optimization results for their calibration for each of the six datasets, three model structures and three fixed values of $k_f$ (0.001, 0.005 and 0.01). The trials start with $k_s = 0$ and manually tuned values for $q_{sat}$ and $q_{f0}$.

The standard deviation of the drum pressure simulation error clearly indicates improvements in the tuning. Note the effect of the fixed non-zero $k_s$ value in the fourth trial—the unified decrease in $\sigma_4$ clearly demonstrates the importance of steam valve non-linearities. Note also the mixed results in the fifth trial, where $k_s$ was free for optimization, and the again unified decrease of $\sigma_4$ in the sixth trial—in the latter, thermal capacitance of the metal mass $m_r$ was included.

Based on the functional dependence of $M_5$ on $k_f$, we did not expect to see any affect of its variation. Interestingly, Figure 4.1 shows the three batches of data lying on top of each other, for all three model structures. This indicates for the entire family of models, the calibration to the data is independent of the value of $k_f$. 

19
Parameter Optimization

Figure 4.1: Optimization results for the calibration coefficients $q_{\theta f}$, $q_{\iota f}$ and $k_s$ for each of the six datasets and three model structures.
4.3 Thermal Capacitance and Disposition of the Metal Mass

In the derivations of all energy balances, we included terms for thermal capacitance based on the metal mass of the drum, down-comers and risers: \( m_d, m_{dc} \) and \( m_r \). For identifiability, one of these must be fixed. Based on physical considerations, we took \( m_{dc} \) to be zero. Figure 4.2 shows the optimization results for \( m_d \) and \( m_r \). Trials 6–10 on \( \mathcal{M}_5 \) and 17–21 on \( \mathcal{M}_4 \) were designed as follows:

(i) \( m_r \) free for unbounded search from \( m_r = 0 \),
(ii) \( m_d \) free for unbounded search from \( m_d = 0 \),
(iii) both free for unbounded search from \( m_r = m_d = 0 \),
(iv) both free for constrained search \( (m_r, m_d > 0) \) from \( m_r = m_d = 1e5 \),
(v) both free for unbounded search from preceding results.

The last variant was intended to check the global nature of the optimization result.

In Figure 4.2, the standard deviation \( \sigma_5 \) for trials 6 vs. 7 and 17 vs. 18 suggest the effects of \( m_r \) are more appropriate than those of \( m_d \). Fortunately this agrees with engineering intuition. Trials 8, 10, 19 and 21 are interesting since they involve two-degrees of freedom. In \( \mathcal{M}_4 \) this led to negative values for \( m_d \) in all cases except datasets B and G (feed water flow perturbation). For these experiments, local minima are apparent since the \( m_r \) change sign from trial 8 to 10.

The results for \( \mathcal{M}_4 \) appear to be more plagued of local minima. Beyond a preference for \( m_d \) instead of \( m_r \), in datasets E and J (steam flow perturbation), little can be inferred from these results. This is especially unfortunate since it is in this model structure that the functional dependencies upon \( k_f \) become more pronounced (\( V_{ad} \) is dependent on \( q_{ad} \) which is a function of \( k_f \)). In \( \mathcal{M}_4 \) we would hope for some identifiability w.r.t. \( k_f \), but nothing can be inferred from Figure 4.2.

The occurrence of negative values in Figure 4.2, although physically unrealistic, is interesting. A negative mass equates to a negative energy flux in the energy balances, i.e. an energy loss. In effect, optimization is telling us that the data would "prefer" inclusion of other phenomena, e.g. thermal radiation losses. This can be construed as data fitting, but the data is an important and viable source of information. Finally, these results agree with our preliminary investigations of stochastic input disturbance models (the \( \Delta \)'s in Figure 2.3).

4.4 \( \mathcal{M}_3 \) Shrink-and-Swell Heuristics vs. a Fourth State Variable

In \( \mathcal{M}_5 \), the volume of steam vapor in the drum below the liquid surface \( V_{ad} \) is heuristically modeled as \( V_{ad} = b_1 a_r V_r + b_2 q_{ad}/\rho_s \). In \( \mathcal{M}_4 \), this signal is the fourth state variable. When using grey-box heuristics, one should investigate the model's reproducibility [9]. Are we simply "data fitting" or have we truly captured the physics of the system? Trials 11–14 shown in Figure 4.3 were designed to test the reproducibility of the heuristics w.r.t. the six datasets. The trials were specified as follows:

(i) \( m_d, m_r \) or both free, plus \( A_d \) free and \( b_1 = b_2 = 0 \),
(ii) \( m_d, m_r \) or both free, plus \( A_d \) free and \( b_1 = 1, b_2 = 0 \),
(iii) \( m_d, m_r \) or both free, plus \( A_d \) and \( b_1 \) free, \( b_2 = 0 \),
(iv) \( m_d, m_r \) or both free, plus \( A_d, b_1 \) and \( b_2 \) free.

Comparing trials 11 and 12 \( (b_1 = 0 \rightarrow 1) \) in Figure 4.3, the standard deviation \( \sigma_5 \) shows improvements only for datasets E and J (steam flow perturbations). In all other cases, incorporating the \( b_1 \)-heuristic actually degrades the model's performance. In trial 13, fitting \( b_1 \) decreases \( \sigma_5 \) uniformly w.r.t. the six datasets. The price for the improvement is the spread in the value of \( b_1 \). Additionally, negative values contradict the physics behind the heuristic. This is a good example of grey-box data fitting—the heuristic is of little use if we seek deterministic reproducibility.

Trial 14 tests the heuristic involving \( q_{ad} \), a flow rate designated the "total condensation flow rate" in [3]. Simulation testing\(^1\) showed that this signals dynamics transmit the excitation of both \( q_t \) and \( Q \). It is perturbations of these inputs that gives rise to the shrink-and-swell phenomenon. Clearly, the spread in the optimization results indicates, again, data fitting.

The only conclusion we can really draw from these trials is the relative insensitivity of the metal mass results (see Figure 4.2) to these parameter variations. The difference between the time constants of thermal effects and the drum level dynamics would explain this. Finally we address the justification for adding the fourth state variable—it is with regard to reproducibility that we truly gain something. Virtually all standard deviations \( \sigma_5 \) results for \( \mathcal{M}_4 \) (trials 16–30 shown in Figure 4.2) lie near the best achieved results for \( \mathcal{M}_5 \) (shown in Figure 4.3).

\(^1\)Actually, simulation testing showed a time delayed version of \( q_{ad} \) to be ripe for the basis of a heuristic model.

Dept. of Automatic Control  
Lund Institute of Technology  
J. Särle, J. Ekorn  
1998.08.14
Figure 4.2: Optimization results for the drum and risers metal masses $m_d$ and $m_r$ for each of the six datasets, three model structures and three fixed values of $k_f$. 
4.4 $M_4$ Shrink-and-Swell Heuristics vs. a Fourth State Variable

Figure 4.3: Optimization results for the surface area of liquid in the drum $A_d$ and heuristic parameters $b_1$ and $b_2$ in $M_4$ for each of the six datasets and three fixed values of $k_f$.

<table>
<thead>
<tr>
<th>Dataset:</th>
<th>E</th>
<th>A</th>
<th>B</th>
<th>J</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_{ef}$</td>
<td>0.257</td>
<td>0.255</td>
<td>0.257</td>
<td>0.257</td>
<td>0.252</td>
<td>0.255</td>
</tr>
<tr>
<td>$q_{fe}$</td>
<td>5.652</td>
<td>5.622</td>
<td>5.659</td>
<td>6.241</td>
<td>6.086</td>
<td>6.073</td>
</tr>
<tr>
<td>$k_s$</td>
<td>8.109</td>
<td>7.714</td>
<td>9.493</td>
<td>4.583</td>
<td>4.028</td>
<td>5.118</td>
</tr>
<tr>
<td>$m_{req}$ [tons]</td>
<td>337</td>
<td>442</td>
<td>409</td>
<td>408</td>
<td>346</td>
<td>614</td>
</tr>
<tr>
<td>$A_d$</td>
<td>28.31</td>
<td>25.02</td>
<td>29.07</td>
<td>26.80</td>
<td>25.02</td>
<td>24.87</td>
</tr>
<tr>
<td>$C$</td>
<td>0.3570</td>
<td>0.3879</td>
<td>0.3750</td>
<td>0.3559</td>
<td>0.3647</td>
<td>0.3778</td>
</tr>
<tr>
<td>$D$</td>
<td>0.5691</td>
<td>0.5545</td>
<td>0.5559</td>
<td>0.3293</td>
<td>0.3188</td>
<td>0.3248</td>
</tr>
</tbody>
</table>

Table 4.1: Mean values of some of the parameters for the six datasets. (Estimation “outliers” have been removed.) $C$ and $D$ are fitted parameters for the lines in Figure 4.5, $x_0^* = Dk_f^2$. 
4.5 Variable Time Delay of the Bubble Flow Rate

Both model structures \( M_4 \) and \( M_5 \) involve a heuristic model of the variable time delay constant \( \tau_{sd} \). In Section 2.1, four different hypotheses were established for this signal. In their programming, we re-used the grey-box parameter \( b_1 \) as a multiplicative constant. The fifth state in \( M_5 \) comes from the Padé\((0,1)\) approximation of the pure time delay.

A goal of the study was to assess which of the four heuristics works best. In trials 22, 25, 27, 29 (\( M_4 \)) and 33, 36, 39, 42 (\( M_5 \)), the grey-box parameter \( b_1 \) was held fixed (\( b_1 = 0.7 \) or 1.0 depending on the hypothesis). In each of the subsequent trials, \( b_1 \) was free for optimization. Additionally, in trials 35, 38, 41, 44 (\( M_5 \)), the initial condition \( V_{sd}^0 \) was free for search. From the results for \( \sigma_5 \) shown in Figure 4.2, it would appear that something is to be gained especially for datasets A and F (fuel flow perturbation). Unfortunately, the spread in values of \( b_1 \) indicates we are once again data fitting. Furthermore, the effect of tuning is to reduce the nominal value of \( \tau_{sd} \). This aggravates the problem of numerical stiffness associated with the use of Padé approximations.

Note that optimization results which reduce \( \tau_{sd} \) are essentially removing the time delay's effect. The detuning that we observed could be explained by the numerical stiffness. We used a fixed integration time step of 1 second—the data sample period is 10 seconds. Also, the number of degrees of freedom may also have been too great—between 8 and 10 parameters were being fit simultaneously. More investigation would be required to sort out this issue.

4.6 Friction Factor and Non-Equilibrium Initialization

Unfortunately, based on the optimization results shown in Figures 4.1, 4.2 & 4.3, little can be conclusively said about the uncertain friction factor \( k_f \). The best guideline for its value remains the approximate knowledge of the flow rate \( q_{dc} \). In Section 3.2 we established the coupling between \( k_f \) and the initial state condition \( x_0 \). Optimization results shown in Figure 4.5 verify the nature of this coupling.

4.7 Summary

The parameter optimization trials resulted in a vast amount of data, all of the optimization results are listed in Appendix C. It is not a trivial task to draw conclusions from this data. Preliminary results of this case-study were reported in [16]. The main difference to the results here is that the model in that article did not include the steam compressibility, \( k_s \), in Equation 2.6 and that the results only covered the partial load data, datasets J, F and G. The inclusion of the compressibility factor, \( k_s \), made the pressure dynamics in the model much better and the AIC values for comparable experiments in Appendix C are 20% lower than in [16].

However, the main conclusions remain; model structure \( M_4 \) has better reproducibility compared to \( M_5 \). For some hypotheses \( M_5 \) can also give low AIC values, but this is seen as data-fitting since the spread of the grey-box parameters \( b_1 \) and \( b_2 \) is large over the different data-sets.

Additionally, in this study we also compared the model structure \( M_5 \) to \( M_4 \). The added complexity in \( M_5 \) can be rejected although the AIC values for some cases are slightly lower than for the \( M_4 \) structure. The difference in AIC is not consistent over the different datasets, and the spread in \( b_1 \) again indicates data-fitting. For the same reasons it is difficult to conclusively choose one of the \( \tau_{sd} \)-hypotheses in \( M_4 \), although the results in [16] as well as those for some of the datasets in this study indicate that hypothesis 2 in Equation 2.4 is to be preferred.

Mean values of some model parameters over all model structures are given in Table 4.1 on pg. 23. As in [16], we find that the metal mass is higher than the nominal value of 300 tons, but the values here are closer to the nominal due to the inclusion of steam compressibility. The extreme value, \( m_{tot}=614 \) tons, for data-set G is probably due to low excitation of the pressure dynamics. The compressibility factor, \( k_s \), is larger at higher pressures, as can be expected.
Figure 4.4: Optimization results for the surface area of liquid in the drum $A_d$, the heuristic parameter $b_1$ and $V_{d}^0$ in $M_4$ and $M_6$, for each of the six datasets and three fixed values of $k_f$. The standard deviation of the drum level prediction error $\sigma_b$ indicates the quality of each trial's result.
Figure 4.5: Optimization results from trial 2 with the initial state condition $x^0_r$ free, for each of the six datasets and six fixed values of $k_f$. Log-log plot axes reveal the nature of the coupling.
Chapter 5

Conclusion

Summary of Conclusions

Our experiences can be summarized as follows:

- Investigations of the third-order model structure $\mathcal{M}_3$ show that, in addition to the calibration factors $q_{f_r}$ and $q_{n_r}$, the compressibility coefficient $k_s$ and metal mass $m_r$ (or alternately $m_d$) are essential for calibrating the model to the data. Furthermore, the pressure dependency of $k_s$ indicates that it should itself be parameterized as a function of the operating point $P^0$. The uncertainty in $k_f$ proves to be not so consequential. The fidelity of the entire family of models appears to be quite insensitive to this parameter.

- Comparisons of the three main model structures, $\mathcal{M}_5$, $\mathcal{M}_4$ and $\mathcal{M}_6$, show that the extra steam volume state in $\mathcal{M}_4$ adds reproducibility. On the other hand, the delay state in $\mathcal{M}_5$ does not give any conclusively better results. Thus, it can be concluded that $\mathcal{M}_4$ is the most powerful unfalsified model.

- Simulation testing and deterministic identification suggest the shrink-and-swell effect will require more complex modeling than the simple time delay model. The simple model is best suited to datasets A and F (fuel flow perturbations). Reproducibility suffers in the other cases. Better heuristics for the time-varying time delay $\tau_{sd}$ based possibly on the signal $q_{sd}$ would be worth investigating.

- In view of the problems in estimating time delays, more investigations should be made into developing identification methods suited for systems with time delays.

- Finally, our experience indicates that an identification investigation of stochastic models would be useful. By considering state disturbances, more information (about where to improve the model) may be extracted from the data.
Appendix A

Maple Derivations

A.1 Preliminaries

Begin by initializing the Maple computational engine to guarantee a fresh start, and loading a few Maple libraries. The first provides support for linear algebra, while the second contains several procedures useful in manipulating the derived expressions. For documentation, type ?linalg or ?student in Maple.

> restart;
> with(linalg):
> with(student):

Also define some utility procedures; type ?operators[functional] for documentation. These will be used throughout.

> REVERSE := eqn -> rhs(eqn) = lhs(eqn):
> STRIP := eqn -> sort(subs(
    map( f->f=op(0,f), indets(eqn,anyfunc(sr.ring) ) ), eqn )):

To show their usage, we make a quick demonstration. The first simply reverses the order of a relation statement. The second is more for cosmetic purposes; it strips expressions of their time-dependency making them easier to look at when type-set.

> x(t) = y(t) + z(t);

\[ x(t) = y(t) + z(t) \]

> REVERSE(*);

\[ y(t) + z(t) = x(t) \]

> STRIP(*);

\[ x = y + z \]

A.1.1 Approximation of Thermodynamic Property Data

To set the scene for the derivations, we recount the approximate relations given in [3] for the thermodynamic properties of saturated water:

> h[s] = a01+(a11+a21*(P(t)-10))*(P(t)-10) ;
> rho[s] = a02+(a12+a22*(P(t)-10))*(P(t)-10) ;
> h[w] = a03+(a13+a23*(P(t)-10))*(P(t)-10) ;
> rho[w] = a04+(a14+a24*(P(t)-10))*(P(t)-10) ;
> T[sat] = a05+(a15+a25*(P(t)-10))*(P(t)-10) :

The last command stores the derived result in a Maple data object known as a table. The table name, eq, and mnemonic indices provide a convenient system for saving results for later recall. Returning to the property data, Figure A.1 attempts to motivate the use of...
Preliminaries

Figure A.1: Qualitative $P-v$ diagram of the thermodynamic properties for saturated water. Shown in bold is the parabolic nature of the curve-fits used.

Second-order polynomials by illustrating the region of validity of the above approximations. In the derivations, time-derivatives of the properties will be required.

Using the chain-rule for differentiation, these derivatives can be expressed in terms of the following differentials:

\[
\begin{align*}
\frac{\partial}{\partial t} h_s(t) &= ((2 P(t) - 20) a_{11} + a_{11}) \left( \frac{\partial}{\partial t} P(t) \right), \\
\frac{\partial}{\partial t} \rho_s(t) &= ((2 P(t) - 20) a_{12} + a_{12}) \left( \frac{\partial}{\partial t} P(t) \right), \\
\frac{\partial}{\partial t} h_w(t) &= ((2 P(t) - 20) a_{13} + a_{13}) \left( \frac{\partial}{\partial t} P(t) \right), \\
\frac{\partial}{\partial t} \rho_w(t) &= ((2 P(t) - 20) a_{14} + a_{14}) \left( \frac{\partial}{\partial t} P(t) \right), \\
\frac{\partial}{\partial t} T_{sat}(t) &= ((2 P(t) - 20) a_{15} + a_{15}) \left( \frac{\partial}{\partial t} P(t) \right)
\end{align*}
\]

Treat the feed water as an incompressible fluid, $\rho_{fw}(t) = \rho_w(t)$. Then by definition, the enthalpy of the feed water, $h_{fw}(t) = u_{fw}(t) + P(t) v(t)$ can be expressed:

\[
h_{fw}(t) = c_{fw} T_{fw}(t) + \frac{P(t)}{\rho_w(t)}
\]
A.2 Second Order Structure

In this section, we shall derive the modeling equations with pressure, $P(t)$, and total volume of the liquid water phase, $V_{wl}(t)$, as the physical state variables. The key assumptions are:

A1: the water inside the system is everywhere in a saturated thermodynamic state;
A2: instantaneous uniform thermal equilibrium between metal and water;
A3: homogeneous mixing of liquid in the drum.

With the first assumption, all thermodynamic properties can be characterized by one independent variable. This makes it possible to use the approximations given in previous section. With the second assumption, capacitive thermal effects of the metal can be included. The third assumption is necessary since the feed water enters the system in a sub-cooled liquid state.

The section continues with the derivation of a single-state model, i.e. a model reduction of the two-state structure. By making the additional assumption:

A4: the drum level controller actions appear to be instantaneous when compared with the dynamics of the system,

the dynamics of the second state, $V_{st}(t)$, can be neglected. Models with such approximations can be motivated for the purpose of control design.

A.2.1 Global Mass and Energy Balances

Consider conservation of mass applied to the global control volume shown in Figure 2.1 (c.v.I, page 3). At any time, the total mass of water in the system is equal to the mass of the two phases combined.

$M(t) = M_{wl}(t) + M_{st}(t)$

Assuming saturation conditions throughout the system, the mass of each phase is given by the density and total volume of the respective phase.

$M_{wl}(t) = \rho_{wl}(t)\; V_{wl}(t)$
$M_{st}(t) = \rho_{st}(t)\; V_{st}(t)$

The time rate of change of mass in the system is equal to the feed water mass flow rate in, minus the steam mass flow rate out of the system,

$\frac{\partial}{\partial t} M(t) = q_{fw}(t) - q_{st}(t)$

Substituting expressions, we arrive at the global mass balance:

$\frac{\partial}{\partial t} (\rho_{wl}(t)\; V_{wl}(t) + \rho_{st}(t)\; V_{st}(t)) = q_{fw}(t) - q_{st}(t)$

Note that the definition is made in terms of the “inert” form of the differential operator; see ?Diff for documentation.

Now consider the energy stored in the system. At any time, the total energy is equal to the internal energy of the two phases, plus the thermal energy stored in the metal.

$U(t) = U_{wl}(t) + U_{st}(t) + U_{T}(t)$

Again, assuming saturation conditions throughout, as well as an instantaneous thermal equi-
librium between the metal and saturated water, we have:

$$U_{uw}(t) = \rho_u(t) u_w(t) V_{uw}(t), U_{us}(t) = \rho_s(t) u_s(t) V_{us}(t), U_T(t) = M_T c_p T_{sat}(t)$$

The time rate of change of energy in the system is equal to the flux in minus the flux out.

$$\frac{\partial}{\partial t} U(t) = Q(t) + h_{fw}(t) q_{fw}(t) - h_s(t) q_s(t) + \Delta_{sys}(t)$$

The thermal heat flow, $Q(t)$, comes from the combustion of fuel. The second and third terms represent the energy flux due to the feed water in and the steam flow out respectively; see Figure 2.1 on page 3. The last right-hand term, $\Delta_{sys}(t)$, is the under-modeling associated with our assumptions. Practically, it serves as a placeholder so that its effects, when filtered through subsequent derivations, will be clear. Substituting definitions into the left-hand side, we have:

$$\frac{\partial}{\partial t} (\rho_u(t) u_w(t) V_{uw}(t) + \rho_s(t) u_s(t) V_{us}(t) + M_T c_p T_{sat}(t)) =$$

$$Q(t) + h_{fw}(t) q_{fw}(t) - h_s(t) q_s(t) + \Delta_{sys}(t)$$

This change of variables simplifies the programming in OMOL as well as subsequent derivations where the mass and energy balances are combined (to eliminate flux terms on the right-hand side). Our final formulation of the global energy balance is:

$$\frac{\partial}{\partial t} \left( \rho_u(t) \left( u_w(t) - \frac{P(t)}{\rho_u(t)} \right) + \rho_s(t) \left( u_s(t) - \frac{P(t)}{\rho_s(t)} \right) \right) =$$

$$Q(t) + h_{fw}(t) q_{fw}(t) - h_s(t) q_s(t) + \Delta_{sys}(t)$$

This change of variables simplifies the programming in OMOL as well as subsequent derivations where the mass and energy balances are combined (to eliminate flux terms on the right-hand side). Our final formulation of the global energy balance is:

$$\frac{\partial}{\partial t} \left( \rho_u(t) \left( u_w(t) - \frac{P(t)}{\rho_u(t)} \right) + \rho_s(t) \left( u_s(t) - \frac{P(t)}{\rho_s(t)} \right) \right) =$$

$$Q(t) + h_{fw}(t) q_{fw}(t) - h_s(t) q_s(t) + \Delta_{sys}(t)$$

Using the student library's value operator, we now evaluate the inert diff operator. Dis-
tributing the differential operator yields:

\[
(r_w(t) - \rho_s(t)) \left( \frac{\partial}{\partial t} V_{\text{wt}}(t) \right) + \left( \frac{\partial}{\partial P} \rho_w(t) \right) V_{\text{wt}}(t) + \left( \frac{\partial}{\partial t} \rho_s(t) \right) V_{\text{st}}(t) = q_{\text{fw}}(t) - q_s(t)
\]

By the first assumption, all thermodynamic property data can be expressed in terms of pressure. Using the chain-rule, we can express the time-derivatives of these quantities in terms of the state derivative.

\[
\left( \frac{\partial}{\partial P} \rho_w \right) V_{\text{wt}}(t) + \left( \frac{\partial}{\partial P} \rho_s \right) V_{\text{st}}(t) + (r_w(t) - \rho_s(t)) \left( \frac{\partial}{\partial t} V_{\text{st}}(t) \right) = q_{\text{fw}}(t) - q_s(t)
\]

Inspecting this result, we see the left-hand side of the balance equation is a linear combination of the state derivatives. The coefficients of the time-derivatives are:

\[
e_{11} = \text{STRIP}\left( \text{coeff\left(lhs\left(\cdot\right), \text{Diff\left(P, t\right)}\right)} \right);
\]

\[
e_{12} = \text{STRIP}\left( \text{coeff\left(lhs\left(\cdot\right), \text{Diff\left(P, t\right)}\right)} \right);
\]

Here we have used our "cosmetic" utility to strip the time dependency from the expressions. Continuing, we use these coefficients to simplify the implicit form of the mass balance:

\[
e_{12} \left( \frac{\partial}{\partial t} P \right) + e_{11} \left( \frac{\partial}{\partial t} V_{\text{wt}} \right) = q_{\text{fw}} - q_s
\]

This implicit formulation is in fact what we shall program in OMOLA. To derive the explicit formulation, we need to first repeat the above steps, operating on the energy balance. As before, begin by eliminating the total volume of steam and then evaluating the inert differential operator.
ties in terms of pressure. Applying the chain-rule yields:

\[ \begin{align*}
& \text{subs( } \text{diff( h(s)(t),t) = Diff( h(s),P) * Diff(P,t),} \\
& \text{diff( h[w](t),t) = Diff( h[w],P) * Diff(P,t),} \\
& \text{diff(rho(s)(t),t) = Diff(rho(s),P) * Diff(P,t),} \\
& \text{diff(rho[w](t),t) = Diff(rho[w],P) * Diff(P,t),} \\
& \text{diff(T[sat](t),t) = Diff(T[sat],P) * Diff(P,t),} \\
& \text{diff(P(t),t) = Diff(P,t),} \\
& \text{diff(V[wt](t),t) = Diff(V[wt],t), " )::}
\end{align*} \]

As with the mass balance, the left-hand side is a linear combination of the time-derivatives. The state dependent coefficients are:

\[ e21 = -h_s \rho_s + h_w \rho_w \]

\[ e22 = \text{STRIP( coeff(lhs(")), Diff(V[wt],t)) ;} \]

\[ e22 = M_c \rho \left( \frac{\partial}{\partial P} T_{sat} \right) + \left( h_w \left( \frac{\partial}{\partial P} \rho_w \right) + \rho_w \left( \frac{\partial}{\partial P} h_w \right) - 1 \right) V_{ut} + \left( h_s \left( \frac{\partial}{\partial P} \rho_s \right) + \rho_s \left( \frac{\partial}{\partial P} h_s \right) - 1 \right) V_{ut} \]

Using these definitions, the global energy balance reduces to:

\[ \begin{align*}
& \text{subs( REVERSE("), REVERSE("), STRIP(")) ;} \\
& \text{e22 \left( \frac{\partial}{\partial t} P \right) + e21 \left( \frac{\partial}{\partial t} V_{ut} \right) = -h_s q_s + h_w q_w + \Delta_{p+e} + Q} \\
& \text{> eq[GB] := ["", ",", ","];} \\
& \text{> eq[coeff2] := table( [eq[GB][2..3], eq[GB][2..3]]) ;} \\
& \text{Again, it is this implicit formulation of the state equations that we shall program in OMOL. To better understand the nature of the couplings present in the model, we need to examine the explicit formulation. To do this we first collect the implicit equations using matrix notation:} \\
& \text{> \&*( matrix(2,2, [e11,e12,e21,e22]),} \\
& \text{> matrix(2,1, [Diff(V[wt],t), Diff(P,t)]) )} \\
& \text{> = matrix(2,1, [rhs(eq[GB][2]), rhs(eq[GB][2]))]);} \\
& \text{[ e11 e12 ] \&* \left[ \begin{array}{c}
\frac{\partial}{\partial t} V_{ut} \\
\frac{\partial}{\partial t} P
\end{array} \right] = \left[ \begin{array}{c}
q_w - q_s \\
-h_s q_s + h_w q_w + \Delta_{p+e} + Q
\end{array} \right] \\
& \text{> eq[sys2] := ";
\]

Thanks to the linear structure of the left-hand side, we can solve explicitly for the state derivatives. Inverting the coefficient matrix reveals the, in this case, full coupling present in the equations:

\[ \text{> inverse(op(1,1,eq[sys2])) ;} \]

\[ \left[ \begin{array}{c}
e22 \\
e21
\end{array} \right] = \left[ \begin{array}{c}
e12 \\
e11
\end{array} \right] \left[ \begin{array}{c}
e22 \left( q_w - q_s \right) + e12 \left( h_s q_s - h_w q_w - \Delta_{p+e} - Q \right) \\
e11 \left( h_s q_s - h_w q_w - \Delta_{p+e} - Q \right) + e21 \left( q_w - q_s \right)
\end{array} \right] \]

Solving algebraically for the state derivatives, we obtain the explicit formulation:

\[ \text{> op(2,1,eq[sys2]) =} \]

\[ \text{> map(collect, evalm( \&* rhs(eq[sys2]) ), [e11,e12,e21,e22]));} \]
This clearly shows the mixing of the two balance equations. Also, note the presence of the under-modeling place-holder \( \Delta p_{gb}(t) \) in both state equations.

A practical advantage of first-principle mechanistic models is the physical interpretation of the parameters and signals involved. This significance can be exploited in guiding attempts at model reduction. This is what we shall do next.

### A.2.3 A Simplified Single-State Model

The simplification is based on the following postulate. Assume the controller which regulates the drum water level is well tuned. Additionally, assume the control actions are "fast" compared with the dynamics of the rest of the system. If the variation in the water level in the drum is small, then variation in \( V_{st}(t) \) will also be small. This equates to the following approximation in the mathematical model:

\[
\frac{\partial}{\partial t} V_{st}(t) = 0
\]

Given that \( V_{st}(t) + V_{at}(t) = V_T \), the assumption also requires:

\[
\frac{\partial}{\partial t} V_{at}(t) = 0
\]

Note that this relation does not imply a constant spatial distribution of steam vapor.

Now, we have two balance equations involving the time-derivative of the remaining state variable, \( P(t) \). By combining them, we will hopefully cancel some of the complexity in the coefficient matrix, \( e \). Inspecting the expressions for \( e12 \) and \( e22 \), we see that either \( h_s(t) \) or \( h_w(t) \) will lead to cancellations when multiplied by the mass balance and subtracted from the energy balance. Note that using \( h_s(t) \) has the undesirable side effect of canceling one of the plants controlled inputs, namely the steam mass flow rate, \( q_s(t) \). For this reason, we choose to use \( h_w(t) \) and combining the balance equations yields:

\[
\left( \frac{\partial}{\partial t} \rho_s(t) \right) h_s(t) + \left( \frac{\partial}{\partial t} h_s(t) \right) \rho_s(t) - \left( \frac{\partial}{\partial t} P(t) \right) - h_w(t) \left( \frac{\partial}{\partial t} \rho_s(t) \right) \right) V_{st}(t)
\]

\[
+ \left( \left( \frac{\partial}{\partial t} h_w(t) \right) \rho_w(t) - \left( \frac{\partial}{\partial t} P(t) \right) \right) V_{at}(t) + M_T c_p \left( \frac{\partial}{\partial t} T_{sat}(t) \right) =
\]

\[
= -h_w(t) + \rho_w(t) q_s(t) + \rho_s(t) q_s(t) + Q(t) + \Delta p_{sat}(t)
\]

Note that choosing \( h_w(t) \) as the multiplier eliminates the time-derivative of \( \rho_w(t) \); investigating this terms qualitative behavior and contribution to the second-order equations is one way to assess the validity of the simplification. As before, we need to express everything in terms of the state variable. This begins with the chain-rule expansion of the time-derivatives:

\[
\text{subs}(\text{diff}(\text{h}[s](t),t) = \text{Diff}([\text{h}[s],P]*\text{Diff}(P,t),
\text{diff}(\text{h}[w](t),t) = \text{Diff}([\text{h}[w],P]*\text{Diff}(P,t),
\text{diff}(\rho_s[s](t),t) = \text{Diff}(\rho_s[s],P)*\text{Diff}(P,t),
\text{diff}(T[sat](t),t) = \text{Diff}(T[sat],P)*\text{Diff}(P,t),
\text{diff}(P(t),t) = \text{Diff}(P,t), " )):
\]

Next we recognize the enthalpy of vaporization, \( h_s(t) = h_s(t) - h_w(t) \), and make a simplifying change of variables before collecting the coefficient of the time-derivative:

\[
\text{changevar}(\text{h}[s](t) - \text{h}[w](t) = \text{h}[c](t), " ):
\text{el} = \text{STRIPE}(\text{coeff}(\text{h}[s](t), \text{Diff}(P,t)));
\]

\[
el = M_T c_p \left( \frac{\partial}{\partial P} T_{sat} \right) + \rho_s \left( \frac{\partial}{\partial P} h_s \right) - 1 \right) V_{st} + \rho_s \left( \frac{\partial}{\partial P} h_s \right) h_s - 1 \right) V_{st}
\]

The reduced first-order model is:
The focus of this section is refinement of the state equations. We begin by deriving a simple model of how the liquid level inside the drum varies. This is of interest because of a non-minimum phase behavior known as the "shrink and swell" effect: When subject to a control action, the drum level momentarily responds in the counter-intuitive direction, either rising or falling. Physically, the phenomenon is a function of the steam vapor that exits the risers and bubbles up through the liquid in the drum. Variations in drum pressure lead to surges in the rate the bubble rise to the surface, as well as to a kind of inverse cavitation effect—the bubbles collapse, condensing to saturated liquid. Capturing this behavior in the model is one of our chief objectives. The heuristics of this measurement model motivate the refinement of the state equations, i.e. the augmentation of an additional state.

In the second subsection, the variable that will be augmented is the mass fraction of saturated vapor at the outlet of the risers, \( x_s(t) \). The idea is to better characterize the quantity of the vapor bubbling up through the liquid in the drum. A linear combination of the mass and energy balances for the risers gives us the augmented third state equation.
A key quantity in both the definition of the measurement model and the derivation of the mass and energy balances turns out to be the total volume fraction of steam in the risers, $\alpha_r(t)$. This is defined intuitively as follows:

$$\alpha_r(t) = \frac{V_{rt}(t)}{V_r}$$

By assuming an approximate spatial distribution of the two phases in the risers is known, it is possible to approximate this quantity analytically as a function of the state variables, specifically:

$$\alpha_r(t) = \text{fcn}(P(t), x_r(t))$$

The approximation is derived in the third subsection.

In the fourth subsection, we consider another key variable in the augmented state equation: the mass flow rate entering the risers from the down-comers, $q_{dc}(t)$. This value will be established by considering the steady-state fluid momentum balances for the down-comers and risers.

This section concludes by presenting an alternate choice of state variables and revisiting model reduction. This provides two additional model structures with distinctly different structural couplings.

### A.3.1 A Drum Level Measurement Model

A simple model relating the variations in drum level to fluid volume is obtained by ignoring the drum’s geometric design. Consider approximating it with a constant “equivalent” area, $A_d$. Then the relation between the level and volume of water in the drum is:

$$A_d (l_0 + \delta_l(t)) = V_{wd}(t)$$

We have expressed the drum level as a set-point $l_0$ plus the variation $\delta_l(t)$ about the set-point.

In various formulations, Åström and Bell [3-5] have postulated the “shrink and swell” effect to be a function of the volume of steam vapor below the surface, $V_{sd}(t)$. The vapor comes from the risers and bubbles to the surface. Incorporating the volume of this vapor in the drum level model, we have:

$$V_{wd}(t) + V_{sd}(t) = A_d (l_0 + \delta_l(t))$$

Solving explicitly for the variation in the drum level gives:

$$\delta_l(t) = -l_0 + \frac{V_{wd}(t) + V_{sd}(t)}{A_d}$$

In terms of the state variable, $V_{wd}(t)$, the volume of water in the drum is given by:

$$V_{wd}(t) = V_{wd}(t) - V_{dc} - V_{wr}(t)$$

$$V_{wd}(t) = V_{ud}(t) - V_{dc} - V_{wr}(t)$$
> eq[Vrd] := ";
Note the volume of water in the down-comers is constant since we assume they contain only the liquid phase. The last term in preceding equation corresponds to the volume of saturated liquid in the risers. Note that \( V_r = V_{sr}(t) + V_{wr}(t) \). The partitioning between phases of the total volume can be expressed by introducing the time-varying total volume fraction of steam in the risers, \( \alpha_r(t) \).

\[
\alpha_r(t) = \frac{V_{sr}(t)}{V_r}
\]

An approximate expression for this quantity is derived in Section A.3.3. For now it suffices to work with this exact analytic definition. Observe that:

\[
V_{sr}(t) = \alpha_r(t) V_r
\]

\[
V_{wr}(t) = (1 - \alpha_r(t)) V_r
\]

\[
\text{eq[Vsr]} := ";
\]
\[
\text{eq[Vwr]} := ";
\]

It remains to define \( V_{sd}(t) \), the volume of bubbles below the surface. In [3, 4], Åström and Bell reason that:

\[
V_{sd}(t) = \alpha_r(t) V_r
\]

This states the volume of steam vapor bubbling to the surface is instantaneously equal to the volume of steam vapor in the risers, \( V_{sr}(t) \). This relationship is believed to be a heuristic based on the observation that the steam flow rate exiting the risers is roughly equal to the flow vented from the drum. Because this is a heuristic, and because it is likely that some steam condenses due to mixing with feed water in a sub-cooled state, we chose to add an additional degree of freedom to the model by introducing the constant, \( b_1 \). The relationship is then:

\[
V_{sd}(t) = b_1 \alpha_r(t) V_r
\]

\[
\text{eq[Vsd]} := ";
\]

A larger heuristic step would be to let \( V_{sd}(t) \) to be a function of the controlled inputs which evoke the shrink/swell effect. For now the simplicity of a constant suffices. Substituting the definitions above yields the following model for the drum level variation:

\[
\delta(t) = -l_0 + \frac{((1 + b_1) \alpha_r(t) - 1) V_r + V_{sd}(t) - V_{de}}{A_d}
\]

This is the measurement model we shall use in conjunction with the first, second and third-order state equations. In [5], the volume of steam below the surface, \( V_{sd}(t) \), is augmented as a state variable. The measurement model is then simply:

\[
\delta(t) = -l_0 + \frac{(-1 + \alpha_r(t)) V_r + V_{sd}(t) - V_{de} + V_{sd}(t)}{A_d}
\]

Augmentation of a fourth state variable is the focus of Section A.4. We now proceed with the
derivation of the third state equation.

A.3.2 Vapor Mass Fraction at Risers Outlet as an Augmented Third State

The extra state equation is derived in much the same way the first-order state equation was derived in Section A.2. We begin by defining the mass and energy balances. Then, by taking a linear combination of the two equations, we eliminate one of the key unknowns in the model, \( q_r(t) \), the mass flow rate exiting the risers. Note that once the augmented state equations are solved, this flow rate effectively becomes a known value—it is a function of the state variables and their derivatives. With regard to the “shrink and swell” effect, this is important because together \( q_r(t) \) and \( x_r(t) \) characterize the rate the steam vapor bubbles enter the drum from the risers.

Consider the thermodynamic control volume for the risers, c.v. II, shown in Figure 2.1. At any time, the total mass is equal to the mass of the two phases combined.

\[
M_r(t) = M_{w,r}(t) + M_{s,r}(t)
\]

As before, the mass of each phase is given by the saturation density and total volume of the respective phase:

\[
M_{w,r}(t) = \rho_{w}(t) V_{w,r}(t), \quad M_{s,r}(t) = \rho_{s}(t) V_{s,r}(t)
\]

The time rate of change of mass is equal to the mass flow rate entering from the down-comers, minus the mass flow rate of the wet mixture exiting at the top:

\[
\frac{\partial}{\partial t} M_r(t) = q_{dc}(t) - q_r(t)
\]

Substituting expressions yields the risers mass balance:

\[
\frac{\partial}{\partial t} \left( \rho_{w}(t) (1 - \alpha_r(t)) V_r + \rho_{s}(t) \alpha_r(t) V_r \right) = q_{dc}(t) - q_r(t)
\]

Now consider the energy balance for the control volume. As before, the energy stored at any time is:

\[
U_r(t) = U_{w,r}(t) + U_{s,r}(t) + U_T(t)
\]

Assuming saturation conditions and instantaneous thermal equilibrium between metal and water, the components are:

\[
U_{w,r}(t) = \text{subs}( \rho_{w}(t) V_{w,r}, \text{subs}( \rho_{w}(t) u_{w}(t) V_{w,r}(t) ) ),
\]

\[
U_{s,r}(t) = \text{subs}( \rho_{s}(t) V_{s,r}(t) ),
\]

\[
U_T(t) = M_r c_p T_{sat}(t)
\]

\[
U_{w,r}(t) = \rho_{w}(t) u_{w}(t) (1 - \alpha_r(t)) V_r, U_{s,r}(t) = \rho_{s}(t) u_{s}(t) \alpha_r(t) V_r, U_T(t) = M_r c_p T_{sat}(t)
\]

Note that we can investigate the effects of the thermal storage term included here by setting the metal mass, \( M_r \), to zero. The time rate of change of energy storage equals the flux in minus the flux out.

\[
\frac{\partial}{\partial t} U_r(t) = Q(t) + h_u(t) q_{dc}(t) - h_r(t) q_r(t) + \Delta_{\text{sh}}(t)
\]

Here, \( Q(t) \) is the thermal heat flow coming from the combustion process. We have assumed
the fluid state entering from the down-comers is a saturated liquid, i.e. in precisely the same state the fluid entered the down-comers. The third flux component involves the enthalpy of the saturated mixture at the outlet of the risers, \( h_r(t) \); this is defined below. The last term on the right represents under-modeling. As with the global energy balance, this term serves as a place-holder in subsequent derivations; again its nominal value will be zero. Substitution yields the risers' energy balance:

\[
\frac{\partial}{\partial t} \left( \rho w(t) \left( h_w(t) - \frac{P(t)}{\rho w(t)} \right) \right) (1 - \alpha_r(t)) V_r + \rho w(t) \left( h_w(t) - \frac{P(t)}{\rho w(t)} \right) \alpha_r(t) V_r + M_r c_p T_{sat}(t) \]

\[
= Q(t) + h_w(t) q_{de}(t) - h_r(t) q_r(t) + \Delta_{reb}(t)
\]

As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, \( q_r(t) \), the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{eq[reb]} := *;
\]

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]

Taking the linear combination yields:

\[
\text{As mentioned already, the augmented state equation is derived by taking a linear combination of the two balance equations. A key step is choosing the multiplier. For model reduction in Section 4.2, we chose to eliminate complexity in coefficient of the time-derivative. Here we choose to eliminate an unknown, } q_r(t), \text{ the mass flow rate exiting the risers. A quick look at the right-hand sides of the balance equations tells us that the multiplier is the enthalpy at the risers' exit:}
\]

\[
h_r(t) = h_w(t) + h_c(t) \alpha_r(t)
\]

This is a function of the augmented state, \( x_r(t) \), and the thermodynamic properties of saturated water. To facilitate simplification, the definition is made in terms of the enthalpy of vaporization:

\[
h_c(t) = h_s(t) - h_w(t)
\]
Note the expansion of the derivative of \( \alpha_r(t) \) involves partial derivatives with respect to both state variables \( P(t) \) and \( x_r(t) \). Along with the approximation for \( \alpha_r(t) \) itself, these partial derivatives will be derived in the following subsection. Now, proceeding with the job of simplifying and collecting the result:

As with the global balances, the left-hand side is a linear combination of the time-derivatives. Extracting the state dependent coefficients yields:

Using these definitions, the risers' combined mass-energy balance reduces to:
Now, collecting the augmented state equations in matrix form, we can derive the explicit formulation of the state derivatives. The implicit system of equations has form:

\[
\begin{bmatrix}
e11 & e12 & 0 \\
e21 & e22 & 0 \\
0 & e32 & e33 \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial}{\partial t} V_{st} \\
0 \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r \\
\end{bmatrix}
= \begin{bmatrix}
q_v - q_s \\
-\Delta_{vet} + Q \\
-h_s q_s + h_w q_{fe} + \Delta_{vet} + Q \\
-h_c x_r q_{dc} + Q + \Delta_{vet} \\
\end{bmatrix}
\]

Summarizing, the elements of the coefficient matrix are:

\[
eq \text{table}[(e11, e12, e21, e22, e32, e33)])
\]

Inverting the coefficient matrix yields:

\[
\text{inverse}(\text{op}(\text{lin}[\text{eq}(\text{sys3})])))
\]

Thanks to the linear structure on the left-hand side, we can solve explicitly for the state derivatives. The explicit state space formulation is:

\[
\text{lin}[\text{solve}(\text{op}(\text{eq}(\text{sys3})), \text{rhs}([\text{eq}(\text{sys3})])))
\]

\[
\text{op}(\text{lin}[\text{map}([\text{collect}, *, [e11, e12, e21, e22, e32]])
\]

\[
\begin{bmatrix}
e11 & e12 & 0 \\
e21 & e22 & 0 \\
e32 & e33 \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial}{\partial t} V_{st} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r \\
\end{bmatrix}
= \begin{bmatrix}
-\frac{\partial}{\partial t} (e11 e12 + (e11 e12 + e21) e22) \\
-\frac{\partial}{\partial t} (e11 e22 + e12 e21) \\
-\frac{\partial}{\partial t} (e11 e32 + e12 e31) \\
\end{bmatrix}
\]

\[
\text{Dept. of Automatic Control}
\]

\[
\text{J. Sørlie, J. Eborn}
\]

\[
\text{LUTFD2/TFR-7563-SE}
\]

\[
\text{Lund Institute of Technology}
\]

\[
1998.8.14
\]
Note the functional dependency on the under-modeling place-holders: $\Delta_{geb}(t)$ and $\Delta_{reb}(t)$. The former appears in all three state equations, while the latter appears only in the augmented state equation. Rearranging this equation makes matters more clear:

```maple
> lhs(*)[3,1] = collect( combine(**[3,1]),
> [Delta[geb], Delta[reb], Q, q[s], q[fw], q[dc], x[r], h[c]]);
> lhs(*) = map(simplify, rhs(*)), [e32, e11, e21]);
```

\[
\frac{\partial}{\partial t} x_r = -\quad \frac{e11 e32 \Delta_{reb}}{e33} + \frac{\Delta_{reb} e11 Q e32}{e33} - \frac{e32 (e11 h_r - e21) q_s}{e33 (e11 e22 - e12 e21)} - \frac{\Delta_{reb} e12}{e33} - \frac{h_c x_r q_dc}{e33}
\]

This structure is important if we attempt to represent the under-modeling using stochastic processes. The linear structure will lead to an identifiability problem if we attempt to simultaneously estimate the statistics of $\Delta_{geb}(t)$ and $\Delta_{reb}(t)$.

Before proceeding with the derivations of $\alpha_r(t)$ and its partial derivatives, we mention that explicit formulation of the state equations is not the wisest formulation for numerical implementation. The coefficient matrix of the time-derivatives clearly shows the original state equations are decoupled i.e. independent of the augmented equation. Utilizing this decoupling, a better formulation for numerical implementation is:

```maple
> lhs(****) = extend(submatrix(rhs(****),1..2,1..1)), 1, 0,
> (rhs(eq[sys]))[3,1] = e32*Diff(P(t),e33);
```

\[
\begin{bmatrix}
\frac{\partial}{\partial t} V_{at} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r
\end{bmatrix} = \begin{bmatrix}
(-h_s q_s + hfw qfw + \Delta_{geb} + Q) e12 + (-qfw + q_s) e22 \\
e11 e22 - e12 e21 \\
(-h_s q_s + hfw qfw + \Delta_{reb} + Q) e11 + (-qfw + q_s) e21 \\
e11 e22 - e12 e21 \\
-h_c x_r q_dc + Q + \Delta_{reb} - e32 \left( \frac{\partial}{\partial t} P \right) \\
e33
\end{bmatrix}
\]

In implementing this, the third computation is a function of the result of the second. Thus, the ordering of the computations is crucial.

We mention that this is automatically the formulation that is derived by the compilers of today's modern acausal model definition languages, e.g. OMOLA, DYMOLA and MODELICA. We now proceed with derivations of signals used above which we have not yet defined:

(i) $\alpha_r(t)$, the total volume fraction of steam vapor in the risers and its partial derivatives w.r.t. the state variables;

(ii) $q_{dc}(t)$, the mass flow rate passing through the down-comers.

### A.3.3 Approximation of the Total Vapor Volume Fraction in the Risers

In Section A.2, the gross nature of the analysis allowed us to ignore the spatial distribution of the two phases. Our aim here is to encapsulate in a function of the augmented state variable, $x_r(t)$, the effects of the spatial distribution in the risers. The function which does this is the total volume fraction of steam vapor in the risers. Its definition is intuitively:

```maple
> alpha[r](t) = V[ar](t)/V[r];
\]

\[
\alpha_r(t) = \frac{V_{ar}(t)}{V_r}
\]

In [3, 4], Åström and Bell developed an approximate expression for this quantity in terms of the state variables, i.e.:

```maple
> lhs(*) = f(P(t), x[r](t));
\]

\[
\alpha_r(t) = f(P(t), x_r(t))
\]

The basis of the approximation is the following assumption: assume the distribution of the
liquid and vapor phases spatially from the top to bottom of the risers is known. The mass fraction of steam vapor at any point is defined:

\[ x(t, \xi) = \frac{m_s(t, \xi)}{m_s(t, \xi) + m_w(t, \xi)} \]

Here \( \xi \) is the normalized spatial coordinate ranging from 0 at the bottom to 1 at the top of the risers. Based on results of finite-element modeling, Åström and Bell found the mass fraction of vapor varies linearly from the bottom to the top of the risers. Expressed in terms of the outlet conditions at the top, a model of the spatial distribution is:

\[ x(t, \xi) = x(t, x_f) \]

By definition, the specific volume fraction of steam vapor is the specific volume of the vapor component divided by the specific volume of the saturated mixture:

\[ \alpha(t, \xi) = \frac{v_s(t) x(t, \xi)}{v_s(t) + v_c(t) x(t, \xi)} \]

Expressed in terms of density, the definition becomes:

\[ \alpha(t, \xi) = \frac{x(t, \xi) \rho_u(t)}{\rho_u(t) x(t, \xi) + \rho_s(t)} \]

In the definition, we have made a change of variables: \( \rho_u(t) = \rho_u(t) - \rho_s(t) \); this greatly facilitates simplification of the approximation which will be based on this expression.

Figure A.2: A simplified geometric representation of the risers.

The relationship between the total and specific volume fractions of steam vapor is derived by integrating over the spatial variable \( \xi \). Consider a differential volume slice and the geometry shown in Figure A.2. The specific volume fraction has the following interpretation:

\[ dV = A \, L \, d\xi \]

\[ dV_s(t, \xi) = \alpha(t, \xi) dV(t, \xi) \]
This corresponds to the volume of the vapor component in the differential slice shown in Figure 4.2. Assuming the area $A_r$ is constant, geometrical relations lead to:

$$dV_{sr}(t,\xi) = \alpha(t,\xi) dV_r,$$

Integrating this over the range of the spatial variable yields the total volume of steam vapor in the risers:

$$V_{sr}(t) = \int_0^1 \alpha(t,\xi) V_r \, d\xi$$

Hence, the total volume fraction of steam in the risers is:

$$\alpha_r(t) = \int_0^1 \alpha(t,\xi) \, d\xi$$

Combining the approximate expression for the spatial distribution with the definition of the specific volume fraction gives:

$$\alpha_r(t) = \int_0^1 \frac{x_r(t) \rho_w(t)}{\rho_w(t) x_r(t) \xi + \rho_s(t)} \, d\xi$$

Evaluating the integral yields:

$$\alpha_r(t) = \frac{\rho_w(t) \ln \left( \frac{\rho_s(t)}{\rho_s(t) + \rho_w(t) x_r(t)} \right)}{x_r(t) \rho_w(t)^2} + \frac{\rho_w(t)}{\rho_w(t)}$$

This expression is the approximation for the total steam volume fraction in the risers. Because of our assumption of saturation conditions, the densities are functions of pressure. Thus we have $\alpha_r(t) = f(P(t), x_r(t))$. In [12] a change of variables is made:

$$\eta(t) = \rho_w(t) x_r(t) \rho_s(t)$$

Applying this change, we arrive at their expression for $\alpha_r(t)$:

$$\alpha_r(t) = \frac{\rho_w(t)(-\ln(1 + \eta(t)) + \eta(t))}{\rho_w(t) \eta(t)}$$

The augmented state equation requires partial derivatives of this expression with respect to
the state variables. Below we verify the expressions given in [12].

\[
\begin{align*}
\text{The partial with respect to the outlet mass fraction is:} \\
\frac{\partial}{\partial \eta} \rho_{o}(t) = -\frac{\rho_u(t) \ln \left( \frac{1}{1 + \eta(t)} \right)}{\rho_{o}(t) \eta(t)^2} - \frac{\rho_{o}(t)}{\rho_{s}(t) \eta(t) (1 + \eta(t))}
\end{align*}
\]

The partial with respect to pressure is more complicated. To verify the known expression, we proceed by eliminating terms from the right-hand side.

\[
\frac{\rho_{w}(t)^2}{\rho_{o}(t) \left( \frac{\partial}{\partial \rho} \rho_{s} \right) - \rho_{s}(t) \left( \frac{\partial}{\partial \rho} \rho_{w} \right)}
\]

Finally, we put the terms that we moved to the left-hand side back on the right. This yields the desired expression:

\[
\begin{align*}
\text{A.3.4 Mass Flow Rate through the Down-comers} \\
\text{The one-dimensional momentum balance for a pipe of length } L \text{ can be written:} \\
L \frac{\partial}{\partial t} q = A_i P_i - A_o P_o - F + V \rho_s g_c \\
\text{Apply the momentum balance to the down-comer tubes as well as the riser tubes:} \\
\text{Ldc} \frac{\partial}{\partial t} \rho_d(t) = A_{dc} P(t) - A_{dc} (P(t) + \rho_r(t) g_c L_r) - F + V_{dc} \rho_w(t) g_c
\end{align*}
\]
Here \( \rho_s(t) \) is the average density in the risers. This can be expressed in terms of the total vapor volume fraction \( a_s(t) \) as follows:

\[
\rho_s(t) = (1 - \alpha_s(t)) \rho_w(t) + \alpha_s(t) \rho_s(t);
\]

\[
\rho_t(t) = (1 - \alpha_t(t)) \rho_w(t) + \alpha_t(t) \rho_v(t).
\]

By combining the two momentum balance equations and then assuming \( L_{dc} = L_r = L \), we obtain a differential equation for the mass flow in the down-comer riser loop:

\[
2L \left( \frac{\partial}{\partial t} \rho_{dc}(t) \right) = (V_{dc} + V_r) \rho_r(t) \rho_v(t) - 2F
\]

We observe that the volumes on the right hand side can be combined to form the total volume in the down-comer riser loop:

\[
2L \left( \frac{\partial}{\partial t} \rho_{dc}(t) \right) = V_{tot} \rho_r(t) \rho_v(t) - 2F
\]

To find the static solution to this equation we substitute an expression for the frictional losses and solve for the constant mass flow:

\[
P = \frac{1}{2} \xi \rho_w \rho_v \rho_{dc}
\]

\[
\xi = \frac{1}{2} \frac{V_{tot} \rho_v}{V_r}
\]

Inserting this gives us hypothesis 2 (line 166 in Listing B.12) for the down-comer flow:

\[
q_{dc}^2 = 2 \frac{g \rho_r(t) \rho_v(t) V_r \rho_w A_{dc}}{k}
\]

The primary hypothesis (line 165 in Listing B.12) is taken from the previous Bell-Åström articles where they used a "lumped" friction factor \( k \) instead of the dimensionless factor used in this derivation.

\[
k = k_f g \rho_v A_{dc}
\]
A.3 Third Order Structure

\[ q_{dc}(t) = \sqrt{2} \sqrt{\alpha_r(t) \rho_w(t) V_r \over k_f} \]

This removes the dependence on \( A_{dc} \), so as to have one less physical parameter in the model.
In the implementation of these hypotheses for this study \( A_{dc} \) was calculated to make \( k \) equal to \( k_f \) since we wanted to have the same friction factor in both hypotheses. This gives \( A_{dc} \approx 10^{-4} \) which is physically unrealistic.

A.3.5 Third-Order Equations with an Alternate Choice of State Variable

In [3–5] the volume of liquid water in the drum, \( V_{wd}(t) \), was chosen to be a state variable instead of the total volume of water, \( V_{wt}(t) \). Effectively, this is a linear coordinate transformation so little appears to be gained with the switch. For the third-order state equations, the transformation actually destroys the decoupling present in the original formulation (page 42).
Because we are dealing with nonlinear equations, we include it in the study for thoroughness.
The change of variables is based on the following relation:

\[ V_{wt}(t) = V_{wd}(t) + V_{dc} + \alpha_r(t) V_r \]

To transform the state equations, we need the corresponding expression relating the time derivatives. Straight-forward differentiation yields:

\[ \frac{\partial}{\partial t} V_{wt} = \left( \frac{\partial}{\partial t} V_{wd} \right) - \left( \left( \frac{\partial}{\partial P} \alpha_r \right) \left( \frac{\partial}{\partial t} P \right) + \text{Diff}(\alpha_r, x_r) \left( \frac{\partial}{\partial t} x_r \right) \right) V_r \]

Now recall the definition of the global mass balance:

\[ \text{eq[GMB]}[1]; \]

\[ e12 \left( \frac{\partial}{\partial P} \right) + e11 \left( \frac{\partial}{\partial t} V_{wd} \right) = q_{fw} - q_s \]

Making the change of variables yields:

\[ \text{subs( eq[dVwt], eq[GMB]}[1]); \]

\[ \text{collect(*, \{Diff(V_{wd}, t), \text{Diff}(P,t), \text{Diff}(x[r], t)\});} \]

\[ e11 \left( \frac{\partial}{\partial t} V_{wd} \right) + \left( e12 - e11 \left( \frac{\partial}{\partial P} \alpha_r \right) V_r \right) \left( \frac{\partial}{\partial t} P \right) - e11 \text{Diff}(\alpha_r, x_r) \left( \frac{\partial}{\partial t} x_r \right) V_r = q_{fw} - q_s \]

As usual, we collect the coefficients for the matrix formulation:

\[ e11a = \text{coeff(lhs(*), Diff(V_{wd}, t)}), \]

\[ e12a = \text{coeff(lhs(*), Diff(P,t)}), \]

\[ e13a = \text{coeff(lhs(*), Diff(x[r], t})); \]

\[ e11a = e11e12a = e12 - e11 \left( \frac{\partial}{\partial P} \alpha_r \right) V_r, e13a = -e11 \text{Diff}(\alpha_r, x_r) V_r \]

The alternate formulation of the global mass balance is then:
Next we repeat these steps, applying them to the global energy balance:

\[ \text{eq[GBa][1]} \]

\[ e22 \left( \frac{\partial}{\partial t} P \right) + e21 \left( \frac{\partial}{\partial t} V_{wd} \right) = -h_{s} q_{s} + h_{fu} q_{fu} + \Delta_{vph} + Q \]

Substituting and collecting yields the coefficients:

\[ \text{eq[GBa]} := \left[ \left\{ \begin{array}{c}
eq \text{powsubs( REVERSE(["[3]"],[""]) :}
\text{subs( map(REVERSE, ["[2],["[1]"],[""]) ," \right) :}
eq \text{subs( map(REVERSE, ["[2],["[1]"],[""]) ,"} \right) ;
\end{array} \right. \right\} \]

\[ \text{eq[GBa]} := \left[ \left\{ \begin{array}{c}
eq \text{powsubs( REVERSE(["[3]"],[""]) :}
\text{subs( map(REVERSE, ["[2],["[1]"],[""]) ,"} \right) ;
\end{array} \right. \right\} \]

\[ e21a = e21, e22a = e22 - e21 \left( \frac{\partial}{\partial t} \alpha_{r} \right) V_{r}, e23a = -e21 \text{Diff} (\alpha_{r}, x_{r}) V_{r} \]

The alternate formulation of the global energy balance is then:

\[ \text{eq[GBa]} := \left[ \left\{ \begin{array}{c}
eq \text{powsubs( REVERSE(["[3]"],[""]) :}
\text{subs( map(REVERSE, ["[2],["[1]"],[""]) ,"} \right) ;
\end{array} \right. \right\} \]

Collecting these results in matrix form, we have:

\[ \text{eq[sys3a]} := \text{table(} \left[ \begin{array}{ccc}
eq e11a & e12a & e13a \\
eq e21a & e22a & e23a \\
eq 0 & e22 & e33 \end{array} \right] \left[ \begin{array}{c}
\frac{\partial}{\partial t} V_{wd} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_{r} \end{array} \right] = \left[ \begin{array}{c}
eq q_{fu} - q_{s} \\
eq -h_{s} q_{s} + h_{fu} q_{fu} + \Delta_{vph} + Q \\
eq -h_{c} x_{r} q_{dc} + Q + \Delta_{vph} \end{array} \right] \right) \]

where the elements of the coefficient matrix are:

\[ \text{eq[coef3a]} := \text{table(} \left[ \begin{array}{ccc}
eq e11a & e12a & e13a \\
eq e21a & e22a & e23a \\
eq 0 & e22 & e33 \end{array} \right] \left[ \begin{array}{c}
\frac{\partial}{\partial t} V_{wd} \\
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_{r} \end{array} \right] = \left[ \begin{array}{c}
eq q_{fu} - q_{s} \\
eq -h_{s} q_{s} + h_{fu} q_{fu} + \Delta_{vph} + Q \\
eq -h_{c} x_{r} q_{dc} + Q + \Delta_{vph} \end{array} \right] \right) \]
and the coefficients $e_{11}, e_{12}, e_{21}$ and $e_{22}$ are the same as before (page 33). Now with the non-zero elements $e_{13}$ and $e_{23}$, all three state equations are coupled. This is most evident in the coefficient matrix inverse:

$$
\begin{bmatrix}
-e_{22}e_{33} + e_{23}e_{32} & e_{12}e_{33} - e_{13}e_{32} & -e_{12}e_{23} - e_{13}e_{22} \\
e_{21}e_{33} & e_{11}e_{33} & e_{11}e_{23} - e_{13}e_{21} \\
e_{21}e_{32} & e_{11}e_{32} & e_{11}e_{22} - e_{12}e_{21}
\end{bmatrix}
$$

Thus, we can still solve explicitly for the time-derivatives, but the result is very complex. The explicit state space formulation for this alternate choice of basis is:

$$
\text{linsolve( op(1,1hs(eq[sys3a])), rhs(eq[sys3a])) = }
\text{map(collect, \star, \{e_{11}, e_{12a}, e_{13a}, e_{21a}, e_{22a}, e_{23a}, e_{32}, e_{33}\});}
$$

Note the functional dependency on the under-modeling place-holders: $\Delta_{p(t)}$ and $\Delta_{r(t)}$ now appear simultaneously in all three state equations. This essentially makes differentiating their effects in an error analysis impossible.

### A.3.6 An Augmented Reduced Two-State Model

We close out this section by revisiting model reduction. Recall the basis for the single-state model was the following assumption:

$$
\frac{\partial}{\partial t} V_{\text{ed}} = 0;
$$

It is immediately clear that we can augment the risers combined mass-energy balance to the one-state model. In matrix form, this yields:

$$
\text{&* ( matrix(2,2,[e_{1},0,e_{32},e_{33}]),}
\text{matrix(2,1,[Diff(P,t), Diff(x[t],t)]))}
\text{= matrix(2,1,[rhs(eq[sys1r]), rhs(eq[RCB][1])]);}
$$

Note the functional dependency on the under-modeling place-holders: $\Delta_{p(t)}$ and $\Delta_{r(t)}$ now appear simultaneously in all three state equations. This essentially makes differentiating their effects in an error analysis impossible.
Inverting the coefficient matrix yields:

\[
\begin{bmatrix}
\frac{1}{e1} & 0 \\
\frac{-e32}{e1 e33} & \frac{1}{e33}
\end{bmatrix}
\]

Solving for the explicit state space formulation yields:

\[
\begin{bmatrix}
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r
\end{bmatrix} =
\begin{bmatrix}
-q_u^a h_c + \Delta_{yst} - q_u^b h_w + h_f w q_{fw} + Q \\
\frac{e1}{e32} e32
\end{bmatrix}
\]

\[
\begin{bmatrix}
\frac{\partial}{\partial t} P \\
\frac{\partial}{\partial t} x_r
\end{bmatrix} =
\begin{bmatrix}
-q_u^a h_c + \Delta_{yst} - q_u^b h_w + h_f w q_{fw} + Q \\
\frac{e1}{e32} e32
\end{bmatrix}
\]

Finally, we verify that the preceding sections change in state variables has no effect on the derivation of the reduced one-state model. This is intuitively obvious, but easy to check. Begin by assessing the effect of the assumption behind the model reduction on the state variable transformation:

\[
\text{Diff}(V[w], t) = 0;
\]

\[
\text{Diff}(V[w], t) = \text{solve}(\text{subs}(*, \text{eq}(V[w], t)), \text{Diff}(V[w], t));
\]

Recall the expressions for the alternate coefficients:

\[
e11a = e11, e12a = e12 - e11 \left( \frac{\partial}{\partial t} \alpha_r \right) V_r, e13a = -e11 \text{Diff}(\alpha_r, x_r) V_r
\]
each other out.

\[ e_{12} \left( \frac{\partial}{\partial t} P \right) = q_{u} - q_{s} \]

The same result holds for the global energy balance:

\[ e_{21}a = e_{21}, e_{22}a = e_{22} - e_{21} \left( \frac{\partial}{\partial P} \alpha_{r} \right) V_{r}, e_{23}a = -e_{21} \text{Diff} (\alpha_{r}, x_{r}) V_{r} \]

Hence, the combination of the two balance will lead to the same result, regardless of the choice for the first state variable.

### A.3.7 Iterative Initialization of the Third State

Load procedure definitions for computing multivariate Taylor series expansions.

\[ \text{readlib} (\text{mtaylor}) \]

Begin by deriving a linear approximation of the non-linear down-comer flow rate relation derived in Section A.3.4. The approximation is obtained by taking a Taylor series expansion in \( \alpha_{r} \) and keeping only the first two terms of the series. The expansion is made about the nominal value \( \alpha_{r0} \) which, for simulation purposes, we shall take as the initial value of \( \alpha_{r} \).

\[ q_{dc} = \sqrt{2} \frac{\alpha_{r} \rho_{ua} V_{r}}{k_f} \]

\[ \text{lhs} (\text{eq}) = \text{mtaylor} (\text{rhs}(\text{eq}), \alpha_{r} = \alpha_{r0}, 2) ; \]

\[ q_{dc} = \sqrt{2} \frac{\alpha_{r0} \rho_{ua} V_{r}}{k_f} + \frac{1}{2} \frac{\sqrt{2} \frac{\alpha_{r0} \rho_{ua} V_{r}}{k_f}}{\alpha_{r0}} (\alpha_{r} - \alpha_{r0}) \]

For notational purposes, and to facilitate symbolic manipulation, we introduce the following parameterization:

\[ q_{dc0} = \text{subs} (\alpha_{r} = \alpha_{r0}, \text{rhs}(\text{eq})) ; \]

\[ q_{dc0} = \sqrt{2} \frac{\alpha_{r0} \rho_{ua} V_{r}}{k_f} \]

The linear approximation is thus:

\[ \text{powsubs} (\text{reverse}(\text{eq}), \text{eq}) ; \]

\[ q_{dc} = q_{dc0} + \frac{1}{2} \frac{(\alpha_{r} - \alpha_{r0}) q_{dc0}}{\alpha_{r0}} \]

The augmented third state equation gives us the following condition for equilibrium:
In terms of the above approximation we have:

\[-h_c x_r q_{dc} + Q = 0\]

We save this relation for use later. Next, we repeat the above steps in deriving a linear approximation for the nonlinear relationship between \(\alpha_r\) and \(x_r\).

\[\text{tmp} := \ast;\]
\[\text{STRIP}\left(\text{eq}\left[\text{alpha}[r](t)\right]\right);\]

\[\alpha_r = \frac{\rho_u \ln \left( \frac{\rho_s}{\rho_s + \rho_u x_r \sigma^2} \right) \rho_s}{x_r \sigma^2} + \frac{\rho_u}{\rho_u \sigma^2}\]

\[\text{ar}0 = \text{subs}(x[r]=\text{x}0,\text{rhs}(\ast));\]

\[\text{ar}0 = \frac{\rho_u \ln \left( \frac{\rho_s}{\rho_s + \rho_u \sigma^2} \right) \rho_s}{x_r \sigma^2} + \frac{\rho_u}{\rho_u \sigma^2}\]

\[\text{powsubs}\left(\text{REVERSE}(\ast),\ast\right);\]

\[\alpha_r = \frac{\rho_u \ln \left( \frac{\rho_s}{\rho_s + \rho_u \sigma^2} \right) \rho_s}{x_r \sigma^2} + \frac{\rho_u}{\rho_u \sigma^2}\]

\[\text{powsubs}\left(\text{REVERSE}(\ast),\ast\right);\]

\[\text{alpha}[r] = \text{ar}0 + \text{dards}\left(x[r] - \text{x}0\right);\]

Combining the two linear approximations in the state equilibrium relationship gives:

\[-h_c x_r \left( q_{dc}0 + \frac{1}{2} \frac{dards\left(x-r - \sigma^2\right) q_{dc}}{ar^0} \right) + Q = 0\]

The result of the Taylor series approximations is a quadratic expression in terms of the state.
Fourth Order Structure

This we can solve analytically, yielding an expression for the initial state condition \( x_r^0 \).

\[
\begin{align*}
x_r^0 &= \left\{ -\frac{ar}{dardx} + \frac{1}{2} x_r + \frac{1}{2} \left( h_c qdc0 \right)
\right.
\end{align*}
\]

\[
\begin{align*}
\left( 4h_c qdc0 ar^2 - 4h_c qdc0 ar dardx x_r + h_c qdc0 dardx^2 x_r^2 + 8 dardx Q ar^2 \right)
\end{align*}
\]

\[
\begin{align*}
\left( \frac{1}{2} / \left( h_c dardx qdc0 \right), -\frac{ar}{dardx} + \frac{1}{2} x_r + \frac{1}{2} \left( h_c qdc0 \right)
\right.
\end{align*}
\]

\[
\begin{align*}
\left( 4h_c qdc0 ar^2 - 4h_c qdc0 ar dardx x_r + h_c qdc0 dardx^2 x_r^2 + 8 dardx Q ar \right)^{1/2} \left( h_c dardx qdc0 \right)
\end{align*}
\]

Using OMMSIM, we determined that the first of the two solutions is the correct form to base an iterative initialization upon. Iteration is necessary because the right hand side of the quadratic solution depends upon \( x_r^0 \). To facilitate programming the iteration in OMOLA, we make the following variable changes and then print out the equation in a textual format that we "clip and paste" into the OMOLA model definition; see Listing B.22 on page 76.

\[
\begin{align*}
\text{subs}\left( ar=ar, h[c]=h_c, qdc0=qdc, rhs(*)[1] \right);
\end{align*}
\]

\[
\begin{align*}
-\frac{ar}{dardx} + \frac{1}{2} x_r + \frac{1}{2} \left( h_c qdc \right)
\end{align*}
\]

\[
\begin{align*}
\left( 4h_c qdc ar^2 - 4h_c qdc ar dardx x_r + h_c qdc dardx^2 x_r^2 + 8 dardx Q ar \right)^{1/2} \left( h_c dardx qdc \right)
\end{align*}
\]

\[
\begin{align*}
\text{latex procrastination}
\end{align*}
\]

A.4 Fourth Order Structure

\[
\begin{align*}
\text{latex procrastination}
\end{align*}
\]
> subs(eq[u], "");
> h[x](t) = h[w](t) + h[c](t)*x[r](t);

\[ h_x(t) = h_w(t) + h_c(t) x_r(t) \]

> simplify(subs("", "");
> collect("", [q[sd](t), q[fw](t)]);
> changevar(-h[s](t)+h[w](t)=-h[c](t), "");
> collect("", [h[c](t), q[fw](t)]);
> changevar(-q[sd](t)+q[x](t)*x[r](t)=-rho[s](t)*dV[sd](t)/tau[sd], "");

\[
\frac{\partial}{\partial t} (V_{sd}(t) h_s(t) \rho_s(t) - V_{sd}(t) P(t) + V_{wd}(t) h_w(t) \rho_w(t) - V_{wd}(t) P(t) + m_{sd} C_p T_{sat}(t)) \\
- h_w(t) \left( \frac{\partial}{\partial t} (\rho_s(t) V_{sd}(t) + \rho_w(t) V_{wd}(t)) \right) = \\
- \rho_s(t) dV_{sd}(t) h_s(t) + \left( -h_w(t) + h_{fw}(t) \right) q_{fw}(t) + \Delta IV
\]

> simplify(value("");

\[
\frac{\partial}{\partial t} V_{sd}(t) h_s(t) \rho_s(t) + V_{sd}(t) \left( \frac{\partial}{\partial t} h_s(t) \right) \rho_s(t) + V_{sd}(t) h_s(t) \left( \frac{\partial}{\partial t} \rho_s(t) \right) \\
- \left( \frac{\partial}{\partial t} V_{sd}(t) \right) P(t) - V_{sd}(t) \left( \frac{\partial}{\partial t} P(t) \right) + V_{wd}(t) \left( \frac{\partial}{\partial t} h_w(t) \right) \rho_w(t) - \left( \frac{\partial}{\partial t} V_{wd}(t) \right) P(t) \\
- V_{wd}(t) \left( \frac{\partial}{\partial t} P(t) \right) + m_{sd} C_p \left( \frac{\partial}{\partial t} T_{sat}(t) \right) - h_w(t) \left( \frac{\partial}{\partial t} \rho_s(t) \right) V_{sd}(t) \\
- h_w(t) \rho_s(t) \left( \frac{\partial}{\partial t} V_{sd}(t) \right) = \\
- \rho_s(t) dV_{sd}(t) h_s(t) + \rho_s(t) \tau_{sd} h_{fw}(t) \rho_s(t) \Delta IV \tau_{sd}
\]

> subs(eq[Vwr], eq[Vwd]);

\[ V_{wd}(t) = V_{wt}(t) - V_{dc} - (1 - \alpha_r(t)) V_r \]

> diff("", t);

\[ \frac{\partial}{\partial t} V_{wd}(t) = \left( \frac{\partial}{\partial t} V_{wt}(t) \right) + \left( \frac{\partial}{\partial t} \alpha_r(t) \right) V_r \]

> subs("", "");

\[
\frac{\partial}{\partial t} V_{sd}(t) h_s(t) \rho_s(t) + V_{sd}(t) \left( \frac{\partial}{\partial t} h_s(t) \right) \rho_s(t) + V_{sd}(t) h_s(t) \left( \frac{\partial}{\partial t} \rho_s(t) \right) \\
- \left( \frac{\partial}{\partial t} V_{sd}(t) \right) P(t) - V_{sd}(t) \left( \frac{\partial}{\partial t} P(t) \right) + V_{wd}(t) \left( \frac{\partial}{\partial t} h_w(t) \right) \rho_w(t) \\
- \left( \left( \frac{\partial}{\partial t} V_{sd}(t) \right) \left( \frac{\partial}{\partial t} \alpha_r(t) \right) V_r \right) P(t) - V_{wd}(t) \left( \frac{\partial}{\partial t} P(t) \right) + m_{sd} C_p \left( \frac{\partial}{\partial t} T_{sat}(t) \right) \\
- h_w(t) \left( \frac{\partial}{\partial t} \rho_s(t) \right) V_{sd}(t) - h_{fw}(t) \rho_s(t) \left( \frac{\partial}{\partial t} V_{sd}(t) \right) = \\
- \rho_s(t) dV_{sd}(t) h_s(t) + q_{fw}(t) \tau_{sd} h_{fw}(t) - \Delta IV \tau_{sd}
\]
\[
\begin{align*}
(\frac{\partial}{\partial t} V_{sd}) & \quad h_s(t) \rho_s(t) + V_{sd}(t) \left( \frac{\partial}{\partial P} h_s \right) \left( \frac{\partial}{\partial t} \rho_s \right) + V_{sd}(t) h_s(t) \left( \frac{\partial}{\partial P} \rho_s \right) \left( \frac{\partial}{\partial t} P \right) \\
& \quad - \left( \frac{\partial}{\partial t} V_{sd} \right) P(t) - \left( \frac{\partial}{\partial P} \rho_s \right) + V_{sd}(t) \left( \frac{\partial}{\partial P} h_w \right) \left( \frac{\partial}{\partial t} \rho_s \right) + V_{sd}(t) h_w(t) \left( \frac{\partial}{\partial P} \rho_s \right) \left( \frac{\partial}{\partial t} P \right) \\
& \quad + m_{sd} C_p \left( \frac{\partial}{\partial P} T_{sat} \right) \left( \frac{\partial}{\partial t} P \right) - h_w(t) \left( \frac{\partial}{\partial P} \rho_s \right) \left( \frac{\partial}{\partial t} P \right) V_{sd}(t) - h_w(t) \rho_s(t) \left( \frac{\partial}{\partial t} V_{sd} \right) \\
& \quad - \rho_s(t) dV_{sd}(t) h_s(t) + q_{fu}(t) \tau_{sd} h_w(t) - q_{fu}(t) \tau_{sd} h_{fu}(t) - \Delta IV \tau_{sd}
\end{align*}
\]

> collect( *, [Diff(V[wt],t), Diff(P,t), Diff(x[r],t), Diff(V[ad],t),
> V[ad](t), Diff(rho[s],P), rho[s](t), V[wd](t),
> tau[ad], q[fw](t)])
> changevar( h[s](t) = h[c](t) )
> simplify( */rho[s](t)/h[c](t) )
> collect( *, [Diff(V[wt],t), Diff(P,t), Diff(x[r],t), Diff(V[ad],t),
> V[ad](t), Diff(rho[s],P), rho[s](t), V[wd](t),
> tau[ad], q[fw](t)])
> eq[cvIV] := "
> e42 = STRIP( coeff(lhs(eq[cvIV]), Diff(P,t) ) )

\[
e_{42} = \left( \frac{\partial}{\partial P} \rho_s \right) + \left( \frac{\partial}{\partial t} \rho_s \right) h_s(t) \frac{1}{\rho_s h_s} \left( \frac{\partial}{\partial P} h_s \right) V_{sd}
\]

> powsubs( REVERSE("), STRIP( eq[cvIV] ) )
> e43 = STRIP( coeff(lhs(eq[cvIV]), Diff(x[r],t) ) )

\[
e_{43} = - \frac{\text{Diff}(\alpha_r, x_r) V_s P}{h_c \rho_s}
\]

> powsubs( REVERSE("), " )
> e44 = STRIP( coeff(lhs(eq[cvIV]), Diff(V[ad],t) ) )

\[
e_{44} = 1 - \frac{P}{h_c \rho_s}
\]

> powsubs( REVERSE("), " )
> e41 = STRIP( coeff(lhs(eq[cvIV]), Diff(V[wt],t) ) )

\[
e_{41} = - \frac{P}{h_c \rho_s}
\]
\[
\left( \frac{\partial}{\partial t} V_{wt} \right) e411 + e42 \left( \frac{\partial}{\partial t} P \right) + \left( \frac{\partial}{\partial t} \tau_r \right) e43 + e44 \left( \frac{\partial}{\partial t} V_{ad} \right) = \\
- \frac{dV_{sd}}{\tau_{sd}} + \frac{(h_c - h_{fs}) g_{fs}}{h_c} + \frac{\Delta JV}{\rho_s}
\]
Appendix B

Omala Definitions

B.1 Library Definitions

```plaintext
LIBRARY DrumBoiler;
USES k2db, K2TerminalLib;
USES Std, StdComp;

%% $Id: library.ol,v 1.18 1997/10/06 19:39:59 sorliej Exp$

MatrixVar ISA Std::MatrixVar WITH
  EPS TYPE STATIC Real := 2^(-52); %% IEEE floating point
  abs TYPE Matrix [m,n] := abs(value);
  logabs TYPE Matrix [m,n] := ln(abs(value+Eps*ones(m,n))/ln(10));
  rowsum TYPE Column [n];
  colsum TYPE Row [n];
  rowsum = value*ones(n,1);
  colsum = ones(1,n)*value;
END;

VectorVar ISA Std::VectorVar WITH
  EPS TYPE STATIC Real := 2^(-52); %% IEEE floating point
  abs TYPE Column[n] := abs(value);
  logabs TYPE Column[n] := ln(abs(value+Eps*ones(n,1))/ln(10));
  sum TYPE Real;
  sum = ones(1,n)*value;
END;

TimeDelay ISA Variable WITH
  u, T ISA Variable;
  x ISA Std::VectorVar;
END;

PureDelay ISA TimeDelay WITH
  %% Implements the Pade(0,1) approximation of a pure time delay:
  %% Y(s) 1
  %% U(s) 1 + sT
  value := delay(u,T,u);
  X.n := 0;
END;

Pade01 ISA TimeDelay WITH
  %% Implements the Pade(0,1) approximation of a pure time delay:
  %% Y(s) 1
  %% U(s) 1 + sT
  x.n := 1;
  % equations:
  % x + T*x' = u;
  value := x;
  initialization:
  Init, Relinit ISAN Event;
  OnEvent Init DO
  new(x) := u; % Initialize state assuming an equilibrium state.
  schedule(ReInit,0.0); % Fire immediately after Omsim's init-solver.
END;
```

Listing B.1: library.ol—Base definitions used in much of the model library.

57
OnEvent ReInit DO
new(x) := u;
END;
END;

Pade11 ISA TimeDelay WITH
%% Implement the Pade(1,1) approximation of a pure time delay:
%%
%% \( Y(s) \) = \( \exp(-sT) \) \( \frac{1 - sT/2}{1 + sT/2} \)
%%
%% \( U(s) \) = \( \frac{1 + 2sT + (sT)^2/2}{1 + sT/2} \)
%%
x.n := 1;

equations:
x + T/2*x' = 2*u;
value := x - u;
initialization:
Init, ReInit ISAN Event;
OnEvent Init DO
new(x) := 2*u;
END;
OnEvent ReInit DO
new(x) := u;
END;

Pade12 ISA TimeDelay WITH
%% Implement the Pade(1,2) approximation of a pure time delay:
%%
%% \( Y(s) \) = \( \exp(-sT) \) \( \frac{3 - 3sT}{1 + sT/2 + (sT)^2/2} \)
%%
%% \( U(s) \) = \( \frac{1 + 2sT + (sT)^2/2}{1 + sT/2 + (sT)^2/2} \)
%%
x.n := 2;
equations:
x[1]' = x[2];
T^2/2*x[2]' = u - 3*x[1] - 2*T*x[2];
value := 3*x[1] - T*x[2];
initialization:
Init, ReInit ISAN Event;
OnEvent Init DO
new(x) := [u/3;0];
END;
OnEvent ReInit DO
new(x) := [u/3;0];
END;

Pade22 ISA TimeDelay WITH
%% Implement the Pade(2,2) approximation of a pure time delay:
%%
%% \( Y(s) \) = \( \exp(-sT) \) \( \frac{3 - 3sT}{1 + sT/2 + (sT)^2/12} \)
%%
%% \( U(s) \) = \( \frac{1 + 2sT + (sT)^2/2}{1 + sT/2 + (sT)^2/12} \)
%%
x.n := 2;
equations:
x[1]' = x[2];
T^2/12*x[2]' = u - x[1] - T/2*x[2];
value := u - T*x[2];
initialization:
Init, ReInit ISAN Event;
OnEvent Init DO
new(x) := [u;0];
END;
OnEvent ReInit DO
new(x) := [u;0];
END;

Listing B.1: library.ocl (continued).
B.1 Library Definitions

Listing B.2: BoilerIC.om—Interface class definition for the drum boiler flow model.

Listing B.3: BoundedVariableGain.om—Definition of a bounded smooth nonlinear gain block.

Dept. of Automatic Control
Lund Institute of Technology
J. Sørlie, J. Ëborn

LUTFD2/TRF3-7563-SE
1998.8.14
Definitions

LowPassFilter ISA Base::Model WITH

1. LowPassFilter ISA Base::Model WITH
2. \#ID: LowPassFilter.om,v 1.2 1997/06/27 08:54:29 sørlei Exp $
3. Graphic ISA super::Graphic WITH bitmap TYPE String := "LowPassFilter"; END;
4. terminals:
5. T1 ISA Base::SimpleInput WITH
6. Graphic ISA super::Graphic WITH invisible := 1; x_pos := 0; y_pos := 150; END;
7. END;
8. T2 ISA Base::SimpleOutput WITH
9. Graphic ISA super::Graphic WITH invisible := 1; x_pos := 400; y_pos := 150; END;
10. END;
11. parameter:
12. omega ISA Base::Parameter WITH
13. default := 1.0; % Bandwidth equals the inverse of the filter time constant
14. END;
15. variables:
16. x ISA Base::Variable;
17. dW TYPE Real;
18. connections:
19. T1 = dW;
20. T2 = x;
21. equations:
22. $x' + \omega x = \sqrt{2*\omega}dW \quad \text{scaled for unit variance}$
23. END;

Listing B.4: LowPassFilter.om—A first-order low pass filter block.

ProductJunction ISA Base::Model WITH

1. ProductJunction ISA Base::Model WITH
3. Graphic ISA super::Graphic WITH bitmap TYPE String := "ProductJunction";
4. END;
5. terminals:
6. T1 ISA Base::SimpleInput WITH
7. Graphic ISA super::Graphic WITH
8. x_pos := 0; y_pos := 150; invisible := 1;
9. END;
10. END;
11. T2 ISA Base::SimpleInput WITH
12. Graphic ISA super::Graphic WITH
13. x_pos := 200; y_pos := 0; invisible := 1;
14. END;
15. END;
16. T3 ISA Base::SimpleOutput WITH
17. Graphic ISA super::Graphic WITH
18. x_pos := 400; y_pos := 150; invisible := 1;
19. END;
20. equation:
21. T3 = T1*T2;
22. END;

Listing B.5: ProductJunction.om—A scalar signal multiplier block.

StaticGain ISA StdComp::StaticGain WITH

1. StaticGain ISA StdComp::StaticGain WITH
2. \#ID: StaticGain.om,v 1.2 1997/06/27 09:02:56 sørlei Exp $
3. Graphic ISA super::Graphic WITH bitmap TYPE String := "StaticGain";
4. END;
5. T1 ISA Super::T1 WITH
6. Graphic ISA Super::Graphic WITH invisible := 1; END;
7. END;
8. T2 ISA Super::T2 WITH
9. Graphic ISA Super::Graphic WITH invisible := 1; END;
10. END;
11. END;

Listing B.6: StaticGain.om—A constant linear gain block.
Listing B.7: SumJunction.om—A scalar signal summer block.

Listing B.8: WienerProcess.om—A pure integration which, mathematically, yields a Wiener process if driven by continuous-time white Gaussian noise.

Listing B.9: SignalModelMapping.om—The parameterized signal model names used in equation export to Maple.
Definitions

% A medium model describing the thermodynamic properties of saturated water/steam.
% Author: Jonas Eborn
% Assumptions: medium state is pressure, static
% Model use: inside boiler models
% Model type: medium model. (Saturated Water/Steam)
% Units: pressure [MPa] (10 bar = 1 MPa, 1 Pa = 1 N/m²)
% density [kg/m³] (1 kg/m³ = 1e-3 g/cm³)
% enthalpy [MJ/kg]
% temperature [degC]

% $Id: SaturationMM.com,v 1.10 1997/06/27 08:56:49 soerliej Exp$

icon:
Graphic ISA super::Graphic WITH
bitmap TYPE String := "SaturationMM";

END;
terminals:
Min ISA Base::RecordTerminal WITH
Graphic ISA super::Graphic WITH x_pos := 0.0; y_pos := 150.0; END;
END;
Mout ISA Base::RecordTerminal WITH
Graphic ISA super::Graphic WITH x_pos := 400.0; y_pos := 150.0; END;
hs, hw ISA K2TerminalLib::EnthalpyTC;
rs, rw ISA Base::SimpleTerminal;
Tw ISA K2TerminalLib::TemperatureTC;
dhdp, dwdp, dsdp, dwpd, dTdpd ISA Base::SimpleTerminal;
END;

parameters:
a01 TYPE STATIC Real := 2.729486;
a11 TYPE STATIC Real := -1.899208;
a21 TYPE STATIC Real := -1160.8;
a02 TYPE STATIC Real := 53.1402;
a12 TYPE STATIC Real := 7.673;
a22 TYPE STATIC Real := 0.36;
a03 TYPE STATIC Real := 1.401586;
a13 TYPE STATIC Real := 4.933998;
a23 TYPE STATIC Real := -880.0;
a04 TYPE STATIC Real := 691.35;
a14 TYPE STATIC Real := -18.672;
a24 TYPE STATIC Real := -0.0603;
a05 TYPE STATIC Real := 310.6;
a15 TYPE STATIC Real := 0.521;
a25 TYPE STATIC Real := -0.33;
equations:
mout.hm = a01+(a11+a21*(Min.p-10))*(Min.p-10);
mout.dhdp = a12+a22*(Min.p-10);
mout.rs = a02+(a12+a22*(Min.p-10))*(Min.p-10);
mout.drdp = a12+a22*(Min.p-10);
mout.hw = a03+(a13+a23*(Min.p-10))*(Min.p-10);
mout.dwdp = a13+a23*(Min.p-10);
mout.rw = a04+(a14+a24*(Min.p-10))*(Min.p-10);
mout.dwpd = a14+a24*(Min.p-10);
mout.E = a05+(a15+a25*(Min.p-10))*(Min.p-10);
mout.dLsd = a15+a25*(Min.p-10);
END;

Listing 8.10: SaturationMM.com—Medium model containing the thermodynamic properties of saturated water at a given pressure.
B.1 Library Definitions

Listing B.11: OresundSimIC.com—The simulation interface to the experimental data, including stochastic input and output error modeling.
Definitions

1.1.1: OresundSimC.om (continued).

Listing B.11: OresundSimC.om (continued).
null
B.2 Model Structures

See the inheritance hierarchy in Figure 2.2 on page 6.

Dept. of Automatic Control
J. Sarlie, J. Eborn
Lund Institute of Technology

LISTING B.11: OresundSimIC.om (continued).
Listing B.12: Boiler2FM.om—Definition of the second-order root model structure.
ordinary parameters;
% initial conditions
13* dVwc0 ISA Parameter WITH default := 0.0; END; [m3]
14* P0 ISA Parameter WITH default := 8.72; END; [kPa]
% constants
76* AD ISA Parameter WITH default := 20; END; [m2]
77* Vd ISA Parameter WITH default := 40; END; [m3]
78* Vr ISA Parameter WITH default := 0.0; END; [m3]
79* Vdc ISA Parameter WITH default := 10; END; [m3]
80* sp ISA Parameter WITH default := 2.565; END; [kg]
81* sr ISA Parameter WITH default := 2.065; END; [kg]
82* kp ISA Parameter WITH default := 0.005; END; [kg]
83* kS ISA Parameter WITH default := 0.0; END; [kgs/MPa]
84* x10 ISA Parameter WITH default := 0.0; END; [-]
85* xro ISA Parameter WITH default := 0.05; END; [-]
86* b1 ISA Parameter WITH default := 1.0; END; [m, -, n]
87* b2 ISA Parameter WITH default := 0.0; END; [m3/sec/kg]
88* L0 ISA Parameter WITH default := 0.0; END; [m]

class parameters:
% To eliminate the complexity due to switches in the exported equations
% fix the switch parameters as constants, thereby allowing the compiler
% to eliminate them from the equation set (i.e. simplify them).
% Sw0, Sw1, Sw2, Sw3 ISA Parameter WITH default := 1; END;
95* Sw0 := 1; Sw1 := 1; Sw2 := 2; Sw3 := 4;

state_equations:
100* 1. Global mass balance
101* 2. Global energy balance
102 Maxvar ISA DrumBoiler::MatrixVar WITH
103* m = n; n := 2;
104* value = [e[1,1]*Vwc', e[1,2]*p';
105* e[2,1]*Vwc', e[2,2]*p';
106* rowsum[1] = qfw - qv;
107* rowsum[2] = Q + qfw*hfw - qv*hs + Delta_gebi;
108* END;
109* e TYPE Matrix[2,2] := [rw-rs, rw-ns; e12.sum; e22.sum];
110* e12 ISA DrumBoiler::VectorVar WITH
111* n := 2;
112* value := [Vwt*drwdp; Vwt*drwpd];
113* END;
114* e22 ISA DrumBoiler::VectorVar WITH
115* n := 7;
116* value := [Vwt*hw*drwpd; rw*drwpd; -1*thu;
117* Vwt*hw*drwpd; rw*drwpd; -1*thu; (md+mr+mdc)*Cp*dTsdp];
118* END;
119* initialization:
120* Init, ReInit, Start ISAN Event;
121* OnEvent Init DO
122* % Initialize state variables.
123* new(Vwc) := 1/2*Vd + (1-arc0)*Vr + Vdc + dVwc0;
124* new(P) := P0;
125* schedule(ReInit,0.0); % fire immediately after OmSim's init-solver
126* END;
127* OnEvent ReInit DO
128* % Initialize variables dependent upon the initial conditions
129* new(arc0) := ar;
130* new(qct0) := qct;
131* new(qfw0) := qfw;
132* new(qg0) := qg;
133* new(qs0) := qs;
134* new(rs0) := rs;
135* new(rw0) := rw;
136* new(Wd0) := Vd0*Vd + (1-arc0)*Vr + Vdc + dVwc0;
137* schedule(Start,0.0); % fire immediately after re-initialization.
138* END;
139* OnEvent Start DO
140* % Re-Initialize state initializations dependent upon these variables.
141* new(Vwc) := 1/2*Vd + (1-arc0)*Vr + Vdc + dVwc0;
142* END;

Listing B.12: Boiler2FM.om (continued).
B.2 Model Structures

Listing B.12: Boiler2FM.om (continued).

---

Dept. of Automatic Control  
J. Sarlie, J. Eborn  
Lund Institute of Technology  
1998.8.14
Listing B.13: Boiler3FM.om—Definition of the third-order model structure.
Listing B.14: Boiler4FM.om—Definition of the fourth-order model structure.
B.2.1 Reduced Order Definitions

```plaintext
1 Boiler1rFM ISA DrumBoiler::Boiler2FM WITH
2 || $Id: Boiler1rFM.om,v 1.3 1997/06/27 08:14:19 sorslej Exp $
3 4 model_reduction:
5  Vwt' = 0.0;
6 7 state_equations:
8  % 1. Combined global mass and energy balances
9 10 Mxdot ISA DrumBoiler::MatrixVar WITH
11  % Over-write inherited matrix definition
12 m,n; n=1;
13  value = (e[2,2] - hw*e[1,2])*p';
14  rowsum[1] = Q + (hfw-hw)*qtw - hc*qs + Delta_geb;
15  END;
16  END;
17 END;
```

**Listing B.15:** Boiler1rFM.om—Reduced first-order structure, assuming the total volume of water in the system is constant.

```plaintext
1 Boiler2rFM ISA DrumBoiler::Boiler3FM WITH
2 || $Id: Boiler2rFM.om,v 1.3 1997/06/27 08:15:06 sorslej Exp $
3 4 model_reduction:
5  Vwt' = 0.0;
6 7 state_equations:
8  % 1. Combined global mass and energy balances
9 10 Mxdot ISA DrumBoiler::MatrixVar WITH
11  % Over-write inherited matrix definition
12 m,n; n=1;
13  value = (e[2,2] - hw*e[1,2])*p';
14  rowsum[1] = Q + (hfw-hw)*qtw - hc*qs + Delta_geb;
15  END;
16  END;
17 END;
```

**Listing B.16:** Boiler2rFM.om—An augmented, reduced second-order model structure.

```plaintext
1 Boiler3rFM ISA DrumBoiler::Boiler4FM WITH
2 || $Id: Boiler3rFM.om,v 1.3 1997/06/27 08:21:23 sorslej Exp $
3 4 model_reduction:
5  Vwt' = 0.0;
6 7 state_equations:
8  % 1. Combined global mass and energy balances
9 10 Mxdot ISA DrumBoiler::MatrixVar WITH
11  % Over-write inherited matrix definition
12 m,n; n=1;
13  value = (e[2,2] - hw*e[1,2])*p';
14  rowsum[1] = Q + (hfw-hw)*qtw - hc*qs + Delta_geb;
15  END;
16  END;
17 END;
```

**Listing B.17:** Boiler3rFM.om—An augmented, reduced third-order model structure.
B.2.2 Alternate State Realization

Listing B.18: Boiler3wdFM.om—An alternate state realization with $V_{wd}$, the volume of water in the drum, as a state variable.
B.2.3 Time Delay Realizations

Listing B.19: Boiler3dFM.om—A variation of the third-order structure that includes realizations (pure and approximate) of a time-delayed signal $q_{ct}$, the total condensation flow rate.
8.2 Model Structures

Listing B.20: Boiler4dFM.om—A variation of the fourth-order structure that includes realizations (pure and approximate) of a time-delayed signal $q_{sd}$, the vapor flux across the liquid surface in the drum.

Listing B.21: Boiler5FM.om—For equation export, a definition equivalent to Boiler4dFM.om, but with only the Padé(0,1) delay approximation.
B.2.4 Iterative State Variable Initialization

Listing B.22: Boiler3iFM.om—Derivation of the third-order model structure including an iterative initialization of $z_r(0)$ based on first-order Taylor series approximations.
B.3 Simulation Models

Listing B.23: Oresund2.om—Simulation model definition for the second-order model structure, including parameterized signal model definitions for equation export (cf. Listing B.9).

```plaintext
Listing 8.23: Oresund2.om-Simulation model definition for the second-order model structure, including parameterized signal model definitions for equation export (cf. Listing 8.9)

Oresund2 ISA DrumBoiler::OresundSimIC WITH

Boiler ISA DrumBoiler::Boiler2PM;

equation_export:
SMX ISA DrumBoiler::SignalModelMapping WITH

input_signals:
U.n := 5; U := [qs1; qs2; qf; qfw; Tw];
U0.n := 5; U0 := [qs10; qs20; qf0; qfw0; Tw0];
Wy.n := 4; Wy := [Ch1; Ch2; Ch3; Ch4];
Wy.n := 2; Wy := [DN1; DN2];
output_signals:
Y.n := 2; Y := [dp; dl];

state_signals:
X ISA Std::VectorVar WITH
n := 6;

parameter_matrix_mapping:
PM ISA Std::MatrixVar WITH
m := 23;

END;
END;
END;
```

Listing B.24: Oresund3.om—The derived third-order simulation model including overwritten inherited information for equation export.

```plaintext
Listing 8.24: Oresund3.om-The derived third-order simulation model including overwritten inherited information for equation export.

Oresund3 ISA DrumBoiler::OresundSimIC WITH

Boiler ISA DrumBoiler::Boiler3PM;

equation_export:
SMX ISA DrumBoiler::SignalModelMapping WITH

X ISA Std::VectorVar WITH
n := 7;

value := [LPF1.x; Wp1.W; Wp2.W; Wp3.W; Boiler.Vwt; Boiler.p; Boiler.xr];

END;
END;
END;
```

Dept. of Automatic Control
J. Sørlie, J. Eborn
Lund Institute of Technology
LUTFD2/TFRT-7563-SE
1998.8.14
Omola Definitions

Listing B.25: Oresund4.om—Simulation model definition for the fourth-order model structure; the parameter b2 is replaced by Vs0 in the equation export parameter mapping.

Listing B.26: Oresund1r.om—Simulation model for the reduced first-order model structure.

Listing B.27: Oresund2r.om—Simulation model for the augmented, reduced second-order model structure.

Dept. of Automatic Control
J. Sørlie, J. Eborn
Lund Institute of Technology
1998.8.14
Listing B.28: Oresund3r.om—Simulation model for the reduced third-order model structure.

Listing B.29: Oresund3wd.om—Simulation model for the model structure with an alternate state realization.

Listing B.30: Oresund3d.om—Simulation model for the third-order model structure with alternate time-delay realizations of $q_{dt}$.

Listing B.31: Oresund4d.om—Simulation model for the fourth-order model structure with alternate time-delay realizations of $q_{st}$. 
Listing B.32: OresundS.om—Simulation model for the fourth-order model structure with a Padé(0,1) approximation of the time-delayed $q_{sd}$.
Appendix C

Annotated IdKit Sessions

This appendix presents the details of the computer-aided parameter optimization investigations of the third, fourth and fifth-order model structures ($M_3$, $M_4$, and $M_5$) respectively. The first section provides background information on the IdKit tools and explains how they were used as building blocks to create project specific tools. The three subsequent sections document the investigations using these tools. Each investigation consists of command script that was run for each of the six data sets; the results of these runs are presented in tabular form. To investigate the influence of the unidentifiable friction factor, all computations were repeated using three fixed values: $k_f = \{0.001, 0.005, 0.01\}$.

C.1 On Batch Processing

A few words on the IdKit tools and the user interface are necessary. For readers familiar with Ljung's System Identification Tool Box (SITB) [14], it may be useful to draw conceptual parallels between IdKit and the SITB.

We begin with the concept of a project. Computer-aided system identification involves two things: data and a model structure. An IdKit project is organized in a file-system directory. The data is stored in an ASCII text file. The equations of the model structure are encoded in another text file, as C-language subroutines. The status of the project is recorded in a binary database file (in a hidden sub-directory). The IdKit database is analogous to the SITB's theta object. Both are associated to a particular model structure and are "massaged" by the commands of the tool set.

In contrast to the SITB which is implemented as functions in Matlab, the IdKit tools are implemented as UNIX commands. Furthermore, rather than having a set of command-line arguments, the IdKit tools are designed around a "question and answer" dialog [15]. IdKit's commands may be executed either interactively or in batch mode. In the former, the user responds to a series of prompts; in the later, the answers to the prompts are embedded in a command script, i.e. a UNIX shell script. Batch mode operation is the basis of the macro facility provided by IdKit; see [11]. An example of an IdKit macro is shown in listing C.1 on the next page. Specifically note that batch mode operation makes use of so-called "here documents" inside UNIX shell scripts. The syntax of here-documents is as follows:

```
command << delimiter
...here-document...  (the standard input to the UNIX command)
delimiter
```

The point to be made is this: although IdKit lacks a command-line argument oriented user interface (which the SITB has), it is possible to construct one. Such an interface is highly desirable to automate repetitive and laborious tasks. Table C.1 on the following page summarizes the set of command macros and command scripts that were developed to aid the investigations. Listings C.2, C.5 and C.6 are included for illustrative purposes. The first provides a simple example of interfacing between IdKit and OmSim. In essence, the IdKit tools are building blocks which are encapsulated inside project specific tools.
## Table C.1: Summary of project specific command macros and scripts (cm and cs resp.) written to aid in automating the investigation.

<table>
<thead>
<tr>
<th>Script Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cm.mkdbase</td>
<td>Create or reinitialize an IdKit project directory's database.</td>
</tr>
<tr>
<td>cm.setv</td>
<td>Setup for ALMP validation.</td>
</tr>
<tr>
<td>cs.format</td>
<td>Convert raw ASCII data to IdKit's binary file format.</td>
</tr>
<tr>
<td>cs.setd</td>
<td>Set numerical integration time quantum and select dataset.</td>
</tr>
<tr>
<td>cs.setf</td>
<td>Setup for ML parameter optimization.</td>
</tr>
<tr>
<td>cs.setlm</td>
<td>Enter the signal model structure into the project database.</td>
</tr>
<tr>
<td>cs.setp</td>
<td>Set the free parameter set and corresponding search constraints.</td>
</tr>
<tr>
<td>cs.setp0</td>
<td>Set fixed parameters values and initial guesses of free parameters.</td>
</tr>
<tr>
<td>cs.setp0.cf</td>
<td>Set values of calibration factors.</td>
</tr>
<tr>
<td>cs.setp0.u0</td>
<td>Set values of initial input conditions.</td>
</tr>
<tr>
<td>cs.setp0.x0</td>
<td>Set values of initial state conditions.</td>
</tr>
<tr>
<td>cs.sets</td>
<td>Enter scales and ranges of signals in the model.</td>
</tr>
<tr>
<td>cs.writep0</td>
<td>Save parameter settings as an IdKit macro and an OmSim OCL script.</td>
</tr>
<tr>
<td>cs.writep02ocl</td>
<td>Save parameter values as an OmSim OCL script.</td>
</tr>
<tr>
<td>cs.writep2ocl</td>
<td>Save parameters resulting from a search as an OmSim OCL script.</td>
</tr>
</tbody>
</table>

Listing C.1: cm.setv—UNIX shell script created by IdKit's macro facility.

```bash
1 setv > /dev/null <<endsetv
2 # Validation procedures:
3 # 1: ALMP Method
4 # 4: Akaike Information Criterion
5 # Validation Procedure:
6 $1
7 # New data since fitting? (No=0, Yes=1) Accepted risk type 1:
8 0
9 endsetv
```

Listing C.2: cs.writep0—Command script that saves the parameter values as an IdKit macro and OmSim OCL script; the former also contains the specification of free parameters.

```bash
1 #!/bin/sh
2 #============================================================================
3 if test # -eq 1
4 then
5   echo "$1": ERROR. no filename given on the command line!
6 else
7   echo "$1": "$2": no filename given on the command line!
8   exit 0
9 fi
10 if
11   echo "$1": "$2": no filename given on the command line!
12   exit 0
13 fi
14 setp /dev/null
15 setp > /dev/null
16 writep
17 $1
18 if
19   echo "$1": "$2": no filename given on the command line!
20   exit 0
21 fi
22 if
23   echo "$1": "$2": no filename given on the command line!
24   exit 0
25 fi
26 if
27   echo "$1": "$2": no filename given on the command line!
28   exit 0
29 fi
30 if
31   echo "$1": "$2": no filename given on the command line!
32   exit 0
33 fi
34 if
35   echo "$1": "$2": no filename given on the command line!
36   exit 0
37 fi
38 if
39   echo "$1": "$2": no filename given on the command line!
40   exit 0
41 fi
```

### Dept. of Automatic Control
LUTFD2/TFRT-7563-SE
J. Serlie, J. Eborn
Lund Institute of Technology
1998.8.14
Listing C.3: `cs.setp0.cf`—Command script for IdKit’s `setp0` command; the dataset specified on the command line affects lines 40 and 42 of the “here-document”.

Listing C.4: `cs.setp0.x0`—Command script for IdKit’s `setp0` command; the dataset specified on the command line affects lines 70 and 72 of the “here-document”.

---

J. Serlin, J. Eborn  
Lund Institute of Technology  
1998.8.14
Listing C.5: cs.sem — Command script for IdKit's setm command; the model order specified on the command line affects the "here-document" input in lines 77, 106 and 124.
Listing C.6: cs_setup—A more elaborate example of shell-script programming which encapsulates IdKit's command for partitioning the parameter set into fixed and free subsets, as well as specifying constraints for the parameter search.

C.2 Investigation of $M_3$
Listing C.7: M3_Exp—Command script used to investigate the third-order model structure.
### (a) Experiment E, full load, perturbed steam flow.

<p>| | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### (b) Experiment A, full load, perturbed fuel flow.

<p>| | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### (c) Experiment B, full load, perturbed feed-water flow.

<p>| | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table C.2: Optimization results for $M_3$ with $k_f = 0.001$ fixed.

<p>| | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Dept. of Automatic Control**
J. Sjölie, J. Eborn
Lund Institute of Technology
1999.8.14

**LUTFDF2/TFRT-7563-SE**
(d) Experiment J, partial load, perturbed steam flow.

<table>
<thead>
<tr>
<th>id</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>1047.27</td>
<td>101.27</td>
<td>923.20</td>
<td>1003.48</td>
<td>1027.90</td>
<td>1040.35</td>
<td>1052.60</td>
<td>1064.85</td>
<td>1077.20</td>
<td>1089.57</td>
<td>1101.93</td>
<td>1114.28</td>
<td>1126.63</td>
<td>1138.98</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td>0.0174</td>
<td></td>
</tr>
<tr>
<td>Fx</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td></td>
</tr>
</tbody>
</table>

(e) Experiment F, partial load, perturbed fuel flow.

<table>
<thead>
<tr>
<th>id</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td>420.79</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td>0.0292</td>
<td></td>
</tr>
</tbody>
</table>

(f) Experiment G, partial load, perturbed feed-water flow.

Table C.2: $M_3, k_f = 0.001$ (continued).
### Table C.3: Optimization results for $M_3$ with $k_f = 0.005$ fixed.

<table>
<thead>
<tr>
<th>$f$</th>
<th>$M_3$</th>
<th>$M_4$</th>
<th>$M_5$</th>
<th>$M_6$</th>
<th>$M_7$</th>
<th>$M_8$</th>
<th>$M_9$</th>
<th>$M_{10}$</th>
<th>$M_{11}$</th>
<th>$M_{12}$</th>
<th>$M_{13}$</th>
<th>$M_{14}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$\beta$</td>
<td>$\gamma$</td>
<td>$\delta$</td>
<td>$\epsilon$</td>
<td>$\zeta$</td>
<td>$\eta$</td>
<td>$\theta$</td>
<td>$\iota$</td>
<td>$\kappa$</td>
<td>$\lambda$</td>
<td>$\mu$</td>
<td>$\nu$</td>
</tr>
</tbody>
</table>

#### (a) Experiment E, full load, perturbed steam flow.

#### (b) Experiment A, full load, perturbed fuel flow.

#### (c) Experiment B, full load, perturbed feed-water flow.
(d) Experiment J, partial load, perturbed steam flow.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>qefi</td>
<td>0.255</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td></td>
</tr>
<tr>
<td>qefj</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td></td>
</tr>
<tr>
<td>r12</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td></td>
</tr>
<tr>
<td>r14</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td></td>
</tr>
</tbody>
</table>

(e) Experiment F, partial load, perturbed fuel flow.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>qefi</td>
<td>0.255</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td>0.251</td>
<td></td>
</tr>
<tr>
<td>qefj</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td></td>
</tr>
<tr>
<td>r12</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td></td>
</tr>
<tr>
<td>r14</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td></td>
</tr>
</tbody>
</table>

(f) Experiment G, partial load, perturbed feed-water flow.

Table C.3: $M_3$, $k_f = 0.005$ (continued).
### Table C.4: Optimization results for $M_3$ with $k_f = 0.01$ fixed.

<table>
<thead>
<tr>
<th>$k_f$</th>
<th>$E_{\text{perturbed}}$</th>
<th>$E_{\text{flow}}$</th>
<th>$E_{\text{fixed}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.02</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.03</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.04</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.05</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.06</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.07</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.08</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.09</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.10</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.11</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.12</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.13</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.14</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.15</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.16</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.17</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.18</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.19</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.20</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.21</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.22</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.23</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.24</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.25</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.26</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.27</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.28</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.29</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.30</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.31</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.32</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.33</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.34</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.35</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.36</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.37</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.38</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.39</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.40</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.41</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.42</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.43</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.44</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.45</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.46</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.47</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.48</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.49</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
<tr>
<td>0.50</td>
<td>4.69</td>
<td>4.69</td>
<td>4.69</td>
</tr>
</tbody>
</table>
(d) Experiment J, partial load, perturbed steam flow.

(e) Experiment F, partial load, perturbed fuel flow.

(f) Experiment G, partial load, perturbed feed-water flow.

| M4, k_f | 0.01 |

---

**Table C.4:** $M_4, k_f = 0.01$ (continued).

---

**Dept. of Automatic Control**

**LUTFD2/TFRT-7563-SE**

**J. Sarrie, J. Eborn**

---

**Lund Institute of Technology**

---

**1998.8.14**
C.3 Investigation of $\mathcal{M}_4$
Perfom the AIMP test with the Vfd interface.

case "ldm_set" in
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
  0   * cs_setp -q qscf qfcf Sigm4 Sigm5 d=1 mr=1 Ad ke Lo Vfd0;
C.3 Investigation of M4

Listing C.8 M4.Exp (continued).
(a) Experiment E, full load, perturbed steam flow.

(b) Experiment A, full load, perturbed fuel flow.

(c) Experiment B, full load, perturbed feed-water flow.

| Table C.5: Optimization results for $M_4$ with $k_f = 0.001$ fixed. |
(d) Experiment A, partial load, perturbed steam flow.

<table>
<thead>
<tr>
<th>q (10^5 g/s)</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>qeff,10^5 g/s</td>
<td>122,191,17,2,3,22,315</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mdc,10^5 g/s</td>
<td>9.40,34,11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T:JSP,10^3 g/s</td>
<td>10.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>delta,10^5 g/s</td>
<td>1.1,2,0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vq,10^5 g/s</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td></td>
</tr>
<tr>
<td>vcf,10^5 g/s</td>
<td>0.2504</td>
<td>0.4003</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td></td>
</tr>
<tr>
<td>sigmax</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sigmax</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(e) Experiment F, partial load, perturbed fuel flow.

<table>
<thead>
<tr>
<th>q (10^5 g/s)</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>qeff,10^5 g/s</td>
<td>122,191,17,2,3,22,315</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mdc,10^5 g/s</td>
<td>6.60,38,15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T:JSP,10^3 g/s</td>
<td>10.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>delta,10^5 g/s</td>
<td>1.1,2,0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vq,10^5 g/s</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.2505</td>
<td></td>
</tr>
<tr>
<td>vcf,10^5 g/s</td>
<td>0.2504</td>
<td>0.4003</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td>0.3414</td>
<td></td>
</tr>
<tr>
<td>sigmax</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sigmax</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table C.5: M4, k_f = 0.001 (continued).
### (a) Experiment E, full load, perturbed steam flow.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>8.40</td>
<td>3.94</td>
<td>11.08</td>
<td>21.24</td>
<td>50.47</td>
<td>140.65</td>
<td>242.60</td>
<td>377.22</td>
<td>599.02</td>
<td>979.20</td>
<td>1366.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARV</td>
<td>1.02</td>
<td>1.86</td>
<td>2.98</td>
<td>4.74</td>
<td>7.80</td>
<td>13.58</td>
<td>25.88</td>
<td>42.20</td>
<td>69.72</td>
<td>125.40</td>
<td>199.10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>3.06</td>
<td>5.05</td>
<td>8.43</td>
<td>14.06</td>
<td>23.49</td>
<td>42.20</td>
<td>75.88</td>
<td>125.40</td>
<td>209.10</td>
<td>336.87</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>8.28</td>
<td>13.74</td>
<td>22.47</td>
<td>39.26</td>
<td>65.05</td>
<td>117.49</td>
<td>217.75</td>
<td>352.40</td>
<td>585.02</td>
<td>928.70</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.08</td>
<td>0.10</td>
<td>0.12</td>
<td>0.14</td>
<td>0.16</td>
<td>0.18</td>
<td>0.20</td>
<td>0.22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.08</td>
<td>0.10</td>
<td>0.12</td>
<td>0.14</td>
<td>0.16</td>
<td>0.18</td>
<td>0.20</td>
<td>0.22</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### (b) Experiment A, full load, perturbed fuel flow.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>8.40</td>
<td>3.94</td>
<td>11.08</td>
<td>21.24</td>
<td>50.47</td>
<td>140.65</td>
<td>242.60</td>
<td>377.22</td>
<td>599.02</td>
<td>979.20</td>
<td>1366.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARV</td>
<td>1.02</td>
<td>1.86</td>
<td>2.98</td>
<td>4.74</td>
<td>7.80</td>
<td>13.58</td>
<td>25.88</td>
<td>42.20</td>
<td>69.72</td>
<td>125.40</td>
<td>199.10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>3.06</td>
<td>5.05</td>
<td>8.43</td>
<td>14.06</td>
<td>23.49</td>
<td>42.20</td>
<td>75.88</td>
<td>125.40</td>
<td>209.10</td>
<td>336.87</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>8.28</td>
<td>13.74</td>
<td>22.47</td>
<td>39.26</td>
<td>65.05</td>
<td>117.49</td>
<td>217.75</td>
<td>352.40</td>
<td>585.02</td>
<td>928.70</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.08</td>
<td>0.10</td>
<td>0.12</td>
<td>0.14</td>
<td>0.16</td>
<td>0.18</td>
<td>0.20</td>
<td>0.22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.08</td>
<td>0.10</td>
<td>0.12</td>
<td>0.14</td>
<td>0.16</td>
<td>0.18</td>
<td>0.20</td>
<td>0.22</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### (c) Experiment B, full load, perturbed feed-water flow.

### Table C.6: Optimization results for $M_4$ with $k_f = 0.005$ fixed.
Table C.6: $M_4$, $k_f = 0.005$ (continued).
(a) Experiment E, full load, perturbed steam flow.

(b) Experiment A, full load, perturbed fuel flow.

(c) Experiment B, full load, perturbed feed-water flow.

Table C.7: Optimization results for $M_4$ with $k_f = 0.01$ fixed.
C.3 Investigation of $M_4$  

The document contains tables and equations related to the investigation of $M_4$. It appears to be a technical report or a research paper, possibly discussing automatic control systems or related topics.

**Table C.7: $M_4$, $k_f = 0.01$ (continued)**

LUTFD2/TFRT-7563-SE  
Lund Institute of Technology  
J. Sarlie, J. Eborn

1998.8.14
C.4 Investigation of $\mathcal{M}_5$
Listing C.8: M5.Exp—Command script used to investigate the fourth-order model structure.
Listing C.8 M5.Exp (continued)
Table C.8: Optimization results for $M_3$ with $k_f = 0.001$ fixed.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
<th>$\text{MSE}, \text{SSE}, \text{R}^2$, $\text{MAE}$</th>
<th>Ref.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-1124.04</td>
<td>985.713</td>
<td>n.e.</td>
<td>-1104.03</td>
<td>986.153</td>
<td>n.e.</td>
<td>-1123.38</td>
<td>983.950</td>
<td>n.e.</td>
<td>-1124.04</td>
<td>985.713</td>
<td>n.e.</td>
</tr>
<tr>
<td>B</td>
<td>-1210.02</td>
<td>705.165</td>
<td>n.e.</td>
<td>-1211.38</td>
<td>705.165</td>
<td>n.e.</td>
<td>-1210.56</td>
<td>705.165</td>
<td>n.e.</td>
<td>-1210.02</td>
<td>705.165</td>
<td>n.e.</td>
</tr>
<tr>
<td>C</td>
<td>-989.38</td>
<td>579.626</td>
<td>n.e.</td>
<td>-989.38</td>
<td>579.626</td>
<td>n.e.</td>
<td>-989.38</td>
<td>579.626</td>
<td>n.e.</td>
<td>-989.38</td>
<td>579.626</td>
<td>n.e.</td>
</tr>
</tbody>
</table>

(c) Experiment B, full load, perturbed feed-water flow.
Table C.8: $M_s k_l = 0.001$ (continued).

(d) Experiment C, partial load, perturbed fuel: water flow.

(e) Experiment D, partial load, perturbed steam flow.
Table C.9: Optimization results for $M_5$ with $k_f = 0.005$ fixed.

<table>
<thead>
<tr>
<th>Dept. of Automatic Control</th>
<th>LUTFD2/TFRT-7563-SE</th>
</tr>
</thead>
</table>
Table C.10: Optimization results for $M_5$ with $k_f = 0.01$ fixed.
### Table C.10: \( M_5, k_f = 0.01 \) (continued)

<table>
<thead>
<tr>
<th>( k_f )</th>
<th>( M_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>0.04</td>
<td>0.04</td>
</tr>
<tr>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>0.06</td>
<td>0.06</td>
</tr>
<tr>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>0.08</td>
<td>0.08</td>
</tr>
<tr>
<td>0.09</td>
<td>0.09</td>
</tr>
<tr>
<td>0.10</td>
<td>0.10</td>
</tr>
<tr>
<td>0.11</td>
<td>0.11</td>
</tr>
<tr>
<td>0.12</td>
<td>0.12</td>
</tr>
<tr>
<td>0.13</td>
<td>0.13</td>
</tr>
<tr>
<td>0.14</td>
<td>0.14</td>
</tr>
<tr>
<td>0.15</td>
<td>0.15</td>
</tr>
<tr>
<td>0.16</td>
<td>0.16</td>
</tr>
<tr>
<td>0.17</td>
<td>0.17</td>
</tr>
<tr>
<td>0.18</td>
<td>0.18</td>
</tr>
<tr>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>0.20</td>
<td>0.20</td>
</tr>
</tbody>
</table>

**Dept. of Automatic Control**  
J. Sarlén, J. Ekborn  
Lund Institute of Technology  
1998.8.14
References


