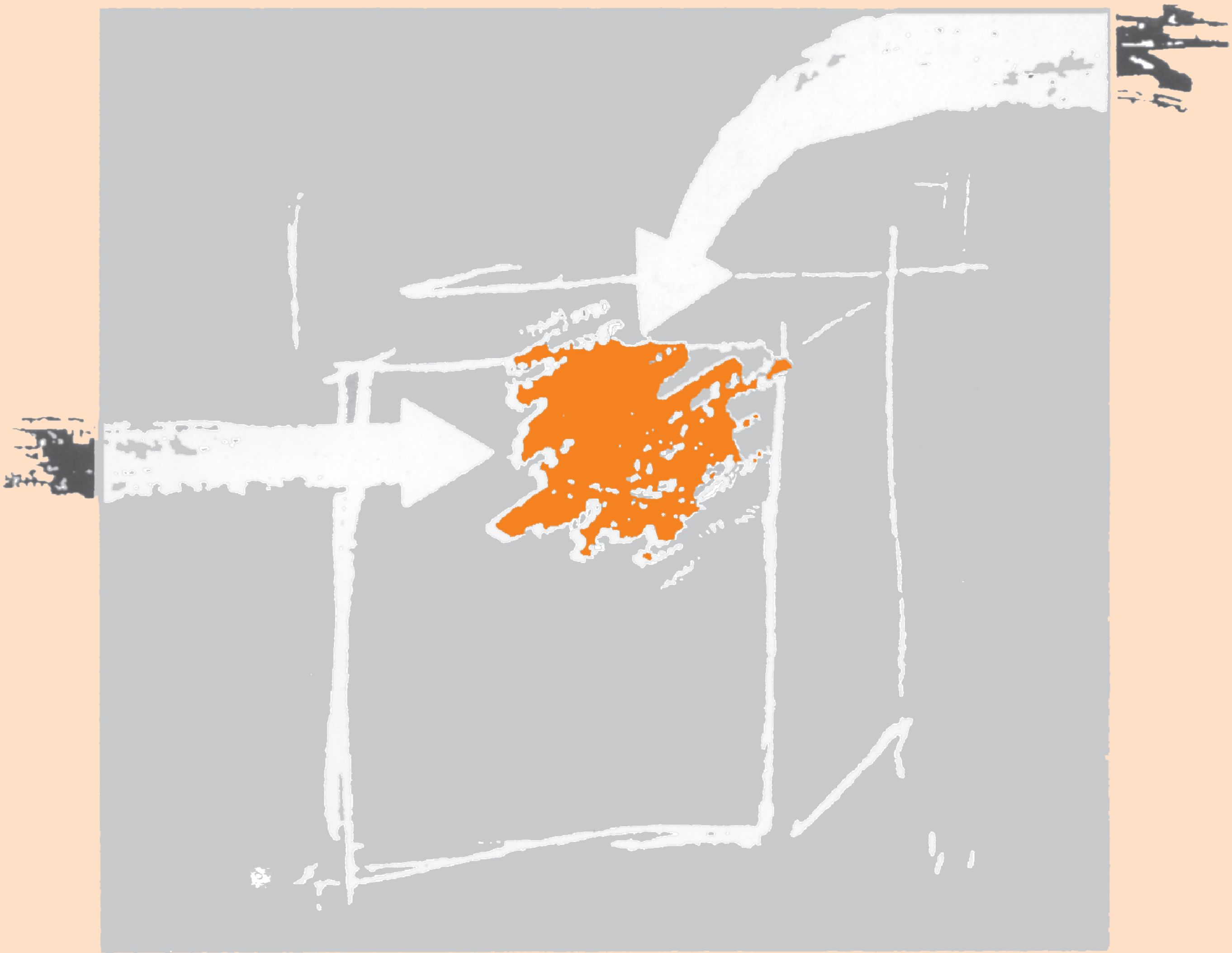
Colección Documentos Ciemat

MODELING AND SIMULATION OF TWO-PHASE FLOW EVAPORATORS FOR PARABOLIC-TROUGH SOLAR THERMAL POWER PLANTS



JAVIER BONILLA CRUZ
LUIS JOSÉ YEBRA MUÑOZ
SEBASTIÁN DORMIDO BENCOMO
EDUARDO ZARZA MOYA

PSA

GOBIERNO
DE ESPAÑA

MINISTERIO
DE ECONOMÍA
Y COMPETITIVIDAD

Ciemat
Centro de Investigaciones
Energéticas, Medioambientales
y Tecnológicas

MODELING AND SIMULATION OF TWO-PHASE FLOW EVAPORATORS FOR PARABOLIC-TROUGH SOLAR THERMAL POWER PLANTS

JAVIER BONILLA CRUZ

LUIS JOSÉ YEBRA MUÑOZ

SEBASTIÁN DORMIDO BENCOMO

EDUARDO ZARZA MOYA

PSA

Es propiedad:

EDITORIAL CIEMAT
Avda. Complutense, 40
28040-MADRID
2013

Catálogo general de publicaciones oficiales
http://www.060.es

Depósito Legal: M-28846-2013
ISBN: 978-84-7834-705-6
NIPO: 721-13-044-9

El CIEMAT no comparte necesariamente las opiniones y juicios expuestos en este documento, cuya responsabilidad corresponde únicamente a los autores.

Modeling and Simulation of Two-phase Flow Evaporators for Parabolic-Trough Solar Thermal Power Plants

Javier Bonilla Cruz
Luis José Yebra Muñoz
Sebastián Dormido Bencomo
Eduardo Zarza Moya

Plataforma Solar de Almería

Centro de Investigaciones Energéticas,
Medioambientales y Tecnológicas

2013

To my girlfriend, Rocío,
my parents, Cecilio and M$^{\underline{a}}$ Angeles,
and my brother, Carlos.

Javier Bonilla Cruz

Research is the process of going up alleys
to see if they are blind.

Marston Bates

Acknowledgments

The authors of this book would like to express their gratitude in the following paragraphs to all the people that have contributed, to a greater or lesser extent, to the development of this book, because it would not have been possible without their invaluable support. Please forgive us if we forget to mention someone in the following lines.

We gratefully acknowledge the funding received towards this book and work in the Automatic Control Group at Plataforma Solar de Almería (PSA) from Centro de Investigaciones Energéticas MedioAmbientales y Tecnológicas (CIEMAT).

We are indebted to the reviewers of this book, Dr. Lidia Roca, Dr. Jose Domingo Álvarez, Alberto de la Calle and Dr. Maria Isabel Roldán, for their corrections, advice and recommendations which contributed to improving this manuscript.

We consider it an honor and a privilege to work with Prof. François E. Cellier, for his reviews, advice and sharing work. This first author is also very grateful for the invitation to work at Eidgenössische Technische Hochschule (ETH) Zurich and the chance to attend to mathematical modeling of physical systems lectures.

At CIEMAT - PSA, we are very grateful to all the personnel, but specially to our colleagues at the Automatic Control Group – Dr. Lidia Roca, Alberto de la calle, Juan Luis Rivas and David Argüelles – for the discussions, advice and shared work in different projects. My gratitude is also extended to Dr. Loreto Valenzuela, for her help in obtaining experimental data to validate the models.

At Centro de Investigación de Energía SOLar (CIESOL), we would like to express our deeper appreciation to Dr. Manuel Pérez, Dr. Jose Domingo Álvarez, Dr. Manuel Pasamontes, Dr. Maria del Mar Castilla, Luis Castillo and Ricardo Silva for our discussions, jokes and shared coffees and lunches.

At Universidad de ALmería (UAL), we would like to thank the members of the Automatic Control, Electronics and Robotics Group (TEP-197), especially Dr. Manuel Berenguel, Dr. Francisco Rodríguez Dr. Jose Luis Guzmán, Dr. Ramón Gonzalez, Dr. Andrzej Pawlowski and Jorge Antonio Sánchez. We shared discussions, advice and work with them. We wish to thank Dr. Luis Iribarne for his help, recommendations and suggestions regarding reading materials about cubic splines.

We wish to thank José Ignacio Zapata and Dr. John Pye from the Australian National University (ANU), for their interest in our work, sharing results and knowledge about switching moving boundary models.

We really appreciate the know-how, recommendations and advice provided by Dr. Francesco Casella from the Politecnico di Milano about chattering in discretized homogeneous two-phase flow models and the object-oriented design of switching moving boundary models.

The first author would like to say thank you very much to his family and friends. Special mention goes to his parents, Cecilio and Mª Angeles, his brother Carlos and his girlfriend Rocío who always encouraged and supported him in this endeavor.

Abstract

The main goal of this work was to contribute in modeling horizontal two-phase flow boiling channels (evaporators) for parabolic-trough solar thermal power plants. The evaporator model is essential for the design of control schemes. Computational fluid dynamics were applied in order to model and discretize the evaporator.

Finite volume and moving boundary models were explored in this work. However, some issues in the dynamic simulation of finite volume models, i.e. chattering, did not allow for taking full advantage of dynamic simulations. Furthermore, none of the current moving boundary models considered dynamic switching between all possible flow configurations in evaporators for parabolic-trough solar thermal power plants.

The developed evaporator models were obtained from physical principles. The object-oriented equation-based modeling paradigm, which was considered for the design of the dynamic evaporator models, contributed to model maintenance, reusability and decoupling. The equation-based modeling paradigm increased reusability further, by not fixing the causality and thus making the models suitable for a wider range of experiments and simulations.

In order to validate the evaporator models, experimental data from a direct steam generation parabolic-trough solar thermal power plant was used – the DIrect Solar Steam (DISS) facility owned by Centro de Investigaciones Energéticas MedioAmbientales y Tecnológicas (CIEMAT)-Plataforma Solar de Almería (PSA), a Spanish government research and testing center.

This work makes contributions to the field of modeling and simulation of dynamical systems. A chattering study of finite volume homogeneous two-phase flow dynamic models is presented. The general solutions employed to solve the chattering problem were analyzed, and particular approaches were implemented: *Mean Densities* and the *Heuristic* approaches, enabling discretized models to be simulated effectively. New mathematical moving boundary models were developed in order to support dynamic switching between all possible flow configurations for evaporators and condensers, and an equation-based object-oriented library was also implemented to test the integrity and stability of the moving boundary models.

Approaches to solve chattering in finite volume models and new moving boundary models were validated against experimental data taken from the DISS facility. For this purpose, different DISS models were implemented, by considering finite volume and moving boundary models. Some unknown parameters were calibrated using a multi-objective genetic algorithm. Finally, these DISS models were compared against experimental data from the DISS facility in terms of accuracy and performance. In order to facilitate these tasks, simulation and calibration frameworks were defined.

All the developed DISS models, composed of finite volume and moving boundary evaporator models, were exposed to a wide range of operating conditions and disturbances, proving that they can take full advantage of dynamic simulations and can be used for the design, testing and validation of advanced control systems.

Contents

List of Figures

List of Tables

List of Acronyms

ABM	Adams Bashforth Moulton
ACS	Automatic Control System
ACSL	Advanced Continuous Simulation Language
ANU	Australian National University
API	Application Programming Interface
ARB	Architecture Review Board
BDF	Backward Differentiation Formula
CDS	Central Differencing Scheme
CFD	Computational Fluid Dynamics
CIEMAT	Centro de Investigaciones Energéticas MedioAmbientales y Tecnológicas - Research Center for Energy, Environment and Technology
CIESOL	Centro de Investigación de Energía SOLar - Solar Research Center
CPU	Central Processing Unit
CRS	Central Receiver, heliostat field or power tower System
CSSL	Continuous System Simulation Language
CSV	Comma-Separated Value
CUDA	Compute Unified Device Architecture
CV	Control Volume
DAE	Differential Algebraic Equation
DAS	Data Acquisition System
DASSL	Differential Algebraic System SoLver
DE	Dish/Engine system
DISS	DIrect Solar Steam
DISTOR	DIrect STORage
DNI	Direct Normal Irradiance
DSG	Direct Steam Generation
DSI	Direct Solar Irradiance
DSR	Direct Solar Radiation
DUKE	DUrchlaufKonzept - Entwicklung und erprobung - Once-through Concept Development and Demonstration
EOO	Equation-based Object-Oriented
ERD	Entity-Relationship Diagram
ETH	Eidgenössische Technische Hochschule - Swiss Federal Institute of Technology

FB	Flash Boiler
FDM	Finite Difference Method
FE	Finite Element
FEM	Finite Element Method
FFC	Friction Factor Correlation
FMI	Functional Mockup Interface
FMU	Functional Mockup Unit
FOU	First-Order Upwind
FVM	Finite Volume Method
GCC	GNU Compiler Collection
GNU	GNU is Not Unix
GUI	Graphical User Interface
HDS	Hybrid Differencing Scheme
HE	Heat Exchanger
HEM	Homogeneous Equilibrium Model
HRS	High Resolution Scheme
HTC	Heat Transfer Correlation
HTF	Heat Transfer Fluid
IA	Incidence Angle (θ)
IAM	Incidence Angle Modifier $K(\theta)$
IAPWS	International Association for the Properties of Water and Steam
IEC	International Electrotechnical Commission
IF97	Industrial Formulation 1997
INDITEP	INtegration of the DIrect steam generation Technology for Electricity Production
InHEM	InHomogeneous Equilibrium Model
IPH	Industrial Process Heat
ITEA	Information Technology for European Advancement
JCR	Journal Citation Reports
LF	Linear Fresnel system
M&S	Modeling and Simulation
MBM	Moving Boundary Model
MIT	Massachusetts Institute of Technology
MSL	Modelica Standard Library
NMF	Neutral Model Format

NSGA	Nondominated Sorting Genetic Algorithm
ODE	Ordinary Differential Equation
OO	Object-Oriented
PDE	Partial Differential Equation
PLDS	Power-Law Differencing Scheme
PRE	Percentage Relative Error (ϵ_r)
PSA	Plataforma Solar de Almería - Almería Solar Platform
PTC	Parabolic-Trough Collector
PTCN	Parabolic-Trough CoNcentrator
RHS	Right-Hand Side
SCi	Simulation Council inc.
SDK	Standard Development Kit
SI	le Système International d'unités - International System of Units
TTSE	Tabular Taylor Series Expansion
UAL	Universidad de ALmería - University of Almería
UDS	Upwind Differencing Scheme
UNED	Universidad Nacional de Educación a Distancia - Spanish National University of Distance Education

List of Abbreviations

§	Section
Ab.	Abbreviation
Abr.	Absorber tube
Appx.	Appendix
Chap.	Chapter
Chaps.	Chapters
Coef.	Coefficient
D	Dry
Der.	Derivative
Eq.	Equation
Eqs.	Equations
F	Flooded
Fig.	Figure
Figs.	Figures
G	General
Inl.	Inlet
Int.	Integration
List.	Listing
Max.	Maximum
Min.	Minimum
Outl.	Outlet
Part.	Partial
SC	Sub-Cooled
SH	SuperHeated
Spec.	Specific
Swit.	Switching
Tab.	Table
TP	Two-Phase

Nomenclature

Latin Letters

Symbol	Description	Units (SI)
A	Area	m^2
C	Concentration ratio	–
c	Speed of sound	m/s
c_p	Specific heat capacity	$\mathrm{J/(kg \cdot K)}$
d	Diameter	m
D_h	Hydraulic diameter	m
F_e	Soiling factor	–
f	Focal distance	m
F_f	Friction force	N
f_f	Fanning friction factor	–
F_{fg}	Friction and gravitational forces	N
F_g	Gravitational force	N
f_m	Moody or Darcy-Weisbach friction factor	–
g	Gravitational acceleration	$\mathrm{m/s}^2$
G_b	Direct solar irradiance	$\mathrm{W/m}^2$
$\dot{H}$	Enthalpy flow rate	W
h	Specific enthalpy	J/kg
k	Thermal conductivity	$\mathrm{W/(m \cdot K)}$
l	Length	m
m	Mass	kg
$\dot{m}$	Mass flow rate	kg/s
n	Number of control volumes	–
P	Power	W
p	Pressure	Pa
P_p	Pressure-volume work	J
$\dot{Q}$	Heat flow rate	W
$\dot{q}$	Heat flux per length	W/m
r	Reflectivity	–
S	Slip ratio	–
S_a	Surface area	m^2
S_c	Circumference	m
S_{up}	Speedup	–
T	Temperature	K
t	Time	s
U	Internal energy	J
U_{loss}	Loss heat transfer coefficient	$\mathrm{W/(m^2 \cdot K)}$

u	Specific internal energy	J/kg
V	Volume	m^3
W	Work	J
w	Velocity	m/s
x	Static quality	–
$\dot{x}$	Vapor or flow quality	–
y	Vertical spatial coordinate	m
z	Horizontal spatial coordinate	m

Greek Letters

Symbol	**Description**	**Units (SI)**
α	Convective heat transfer coefficient	$W/(m^2 \cdot K)$
α_i	Inclination angle	rad
α_a	Absorptivity	–
β	Acceptance angle	rad
θ	Incidence angle	rad
ϵ_r	Percentage relative error	–
ϵ	Roughness	m
η	Efficiency	–
γ	Void fraction	–
γ_i	Intercept factor	–
λ	Parameters of a model	
μ	Variable of a model	
ν	Specific volume	m^3/kg
ω	Weight in calibration	–
ψ	Intensive thermodynamic property	
ϕ	Rim angle	rad
ρ	Density	kg/m^3
σ	Stefan-Boltzmann constant	$W/(m^2 \cdot K^4)$
τ	Transmissivity	–
ε	Emissivity	–
ξ	Threshold	

Superscripts

Symbol	**Description**
χ'	χ of the saturated liquid
χ''	χ of the saturated vapor
χ^+	Cumulative value of χ
$\bar{\chi}$	Mean χ value

χ^*	Maximum χ value

Subscripts

Symbol	**Description**
$0°$	0° incidence angle
A	Area
a	Left boundary
b	Right boundary
abr	Absorber steel pipe
amb	Ambient
ap	Aperture
cov	Glass cover
$diss$	Dissipative
geo	Geometrical
he	Heat exchanger
ht	Heat transfer
htf	Heat transfer fluid
i	Inner
int	Interface
l	Local
mix	Mixture
min	Minimum
o	Outer
opt	Optical
out	Outlet
par	Parallel
ptc	Parabolic-trough collector
$ptcn$	Parabolic-trough concentrator
sc	Sub-cooled liquid
seq	Sequential
sh	Superheated vapor
sun	Sun
the	Thermal
θ	Incidence angle losses
tp	Two-phase flow
v	Volumetric
w	Pipe wall

I

Introduction

1

Introduction and Document Structure

The best structure will not guarantee results or performance. But the wrong structure is a guarantee of failure.

Peter F. Drucker

Abstract: The context, scope and goals of this work, together with the book structure, are presented in this introductory chapter. The scientific contributions derived from it are also enumerated, together with the research projects which funded it.

1.1 Context and Scope

Energy makes the world go around. Industry, daily life, transportation, communications – all are powered by energy. Industrial developments and population grow have been exponentially increasing our need for energy in order to maintain our energy-dependent lifestyles (cf. Fig. 1.1). Fossil fuels are currently our major power supplies, with market share ranging from 87%, according to British Petroleum (2012), to 76%, according to Renewable Energy Policy Network for the 21st Century (2010). However, rising costs, environmental issues and concerns about sustainability are presently encouraging investment and research into alternative sources that may provide clean, renewable, sustainable and efficient energy (Bloomberg New Energy Finance, 2011). Biofuel, biomass, geothermal, solar, tidal, wave and wind energies are currently being researched as the first concerns of governments around the world (International Energy Agency, 2011a).

About 93% of solar energy may be theoretically used (Winter et al., 2003), which makes it an appealing source of energy. Furthermore, the International Energy Agency (2011b) states that: "*The development of affordable,*

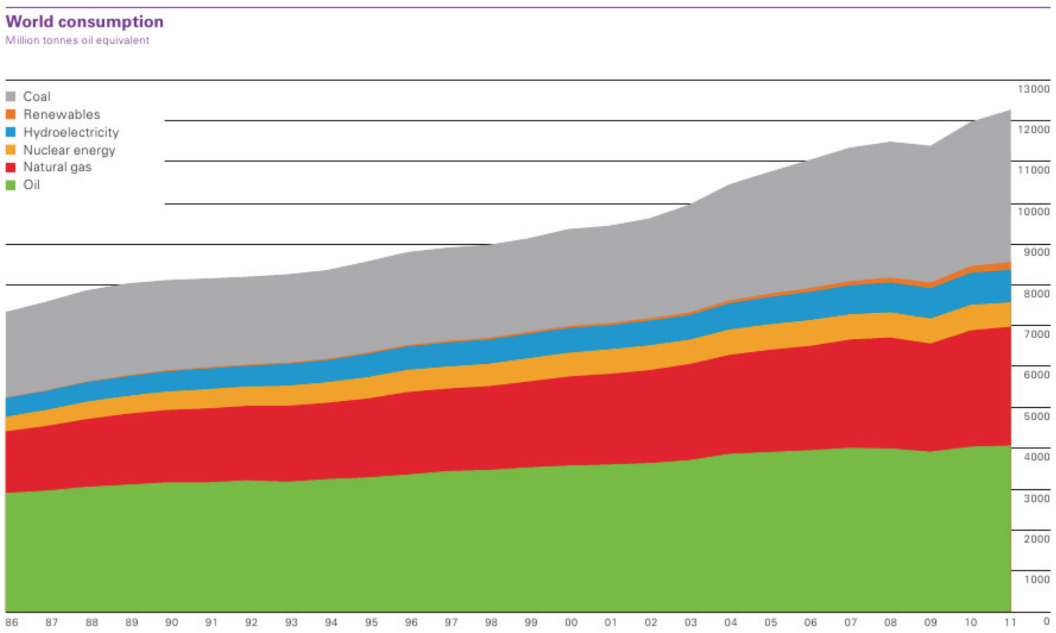


Figure 1.1: World energy consumption trend from 1986 to 2011 (British Petroleum, 2012)

inexhaustible and clean solar energy technologies will have huge longer-term benefits. It will increase countries' energy security through reliance on an indigenous, inexhaustible and mostly import-independent resource, enhance sustainability, reduce pollution, lower the costs of mitigating climate change, and keep fossil fuel prices lower than otherwise. These advantages are global. Hence, the additional costs of the incentives for early deployment should be considered learning investments; they must be wisely spent and need to be widely shared".

The two major solar energy technologies currently available are solar photovoltaics and solar thermal energy. The easier and most efficient way of storing heat through solar thermal energy, and therefore dispatchability on demand, is one of the most important differences between solar thermal energy and other renewable energies, e.g. photovoltaics or wind (Romero-Alvarez and Zarza, 2007). This makes solar thermal energy more appropriate for large-scale energy production.

Converting from solar heat to mechanical work is limited by the Carnot efficiency, and therefore it should be transferred to a Heat Transfer Fluid (HTF) at high temperature to achieve maximum conversion rates. Although the Sun is a source of high temperature, the distance from the Sun to the Earth, coupled with the Earth's atmosphere, limit solar irradiance available for terrestrial use. For this reason, optical concentration devices are needed to achieve maximum solar conversion with minimal heat loss; this is known as 'concentrating solar thermal technology'. Concentrating solar plants represent the most economical form of solar technology for electricity production, according to an independent study promoted by the World Bank (Enermodal Engineering Limited, 1999). Furthermore, there is a proven potential for further cost reduction (Romero-Alvarez and Zarza, 2007). Additionally, concentrating solar power plants do not produce CO_2, and most solar field materials and structures can be recycled (Fernández-García et al., 2010), thus making them suitable for maintaining standards of living without compromising environmental issues.

Four concentrating solar technologies are presently available: Parabolic-Trough Collectors (PTCs), Linear Fresnel systems (LFs), Central Receiver, heliostat field or power tower Systems (CRSs) and Dish/Engine systems (DEs). PTCs are by far the most mature solar concentrating technologies, as demonstrated commercially (Fernández-García et al., 2010). They can be used for a wide range of applications including electricity production, Industrial Process Heat (IPH) applications, water and space heating, air-conditioning and refrigeration, irrigation water pumping, desalination and detoxification.

A PTC is basically a parabola-shaped mirror which concentrates Direct Solar Radiation (DSR) on the absorber tube, also called a boiling channel or an evaporator, when phase changes occur, which is located in the parabola's focal line through which a HTF is pumped to acquire thermal energy from solar radiation. Parabolic-trough solar thermal power plants commonly use thermal oil as the HTF. This technology has been continuously improved since its first commercial implementation; however, this design can be optimized in order to produce superheated steam directly in the PTC field by means of using water as HTF; this technology is known as Direct Steam Generation (DSG). DSG presents some important challenges due to phase changes that affect the HTF: high-pressure, high-temperature elements, temperature gradients, process stability and controllability – all of these practical aspects have been studied in detail in Zarza (2000) and Valenzuela et al. (2004). On the other hand, DSG increases overall system efficiency while reducing investment costs by eliminating the thermal oil previously used as the HTF. Furthermore, DSG has many advantages: no danger of fire, suppression of contaminants, simplification and reduction of the plant and it improves performance due to the reduction of parasitic load.

Reliable control systems are required for parabolic-trough solar thermal power plants in order to maintain desirable operating conditions in the case of changes in solar radiation, because in solar concentrating technologies, the energy source – solar radiation – is not controllable. Over the last twenty years, a great deal of effort has been put into developing efficient control systems for parabolic-trough solar thermal power plants (Camacho et al., 1997, 2012). In PTCs, the outlet fluid temperature is controlled by the inlet mass flow rate of the HTF. These are non-linear dynamical systems which may be influenced by several disturbances. DSG also contributes to complicating the control scheme due to HTF phase changes. For these reasons, advanced Automatic Control Systems (ACSs) are required for this kind of system (Camacho et al., 1997).

The system model acquires more relevance when considering advanced control schemes – not only accurate but fast dynamic models are required for the development of advanced ACSs, e.g. model-based control. Dynamic models also provide a better understanding of the system and allow for experimentation with systems which might not exist, might be dangerous for people or resources, might be too expensive or might have dynamics that are too fast or too slow to be studied properly. Additionally, and in the context of real-time simulations, dynamic models can be used for training plant operators.

When considering the modeling of complex physical systems, and this is the case for parabolic-trough solar thermal power plants, the equation-based and Object-Oriented (OO) modeling paradigms are important aspects to take into account. The OO modeling paradigm contributes to model maintenance, reusability and decoupling (Cellier et al., 1996). The equation-based modeling paradigm improves reusability even more, as it does not fix causality, thus making the models suitable for a wider range of experiments and simulations.

This work sits within the realm of Equation-based Object-Oriented (EOO) modeling and the dynamic simulation of parabolic-trough solar thermal power plants considering DSG, where the HTF is a two-phase flow water-steam mixture. The boiling channel (evaporator), where the two-phase flow is heated, is the key element in the modeling process.

The particular parabolic-trough solar thermal power plant considered in this work is the DIrect Solar Steam (DISS) facility owned by Centro de Investigaciones Energéticas MedioAmbientales y Tecnológicas (CIEMAT) - Plataforma Solar de Almería (PSA), a Spanish government research and testing center. The selected EOO modeling language is Modelica. A previous DISS Modelica model was developed in Yebra (2006). However, some issues in dynamic simulation, i.e. chattering, did not allow for taking full advantage of dynamic simulations. Additionally, there is a need to define simulation and calibration frameworks, and to explore other discretization methods for the evaporator, which may lead to accurate and fast simulations.

1.2 Goals

The main goal of this work is to contribute to modeling processes in order to obtain accurate and fast two-phase flow dynamic evaporator models, and to perform dynamic simulations for parabolic-trough solar thermal power plants considering DSG. These models are intended to be used for the design, testing and validation of advanced ACSs. In order to achieve the main goal, the following tasks must be fulfilled.

1. To study and analyze numerical issues within the previous DISS Modelica model which are associated with the evaporator model, i.e. chattering. Provide solutions and implement different approaches to solve this practical problem in order to take full advantage of dynamic simulations.

2. To update the previous DISS Modelica model developed in Yebra (2006), making used of the latest advances in the EOO Modelica modeling language, in order to contribute to the maintenance, reusability and decoupling of the model and its components.

3. To develop new evaporator models considering alternative discretization schemes, besides the Finite Volume Method (FVM) applied in the previous DISS Modelica model. The moving boundary formulation seems to be appropriate when fast computation is required, which is the case when models for advanced ACSs are considered. Mathematical models and their corresponding implementation in a Modelica library, following the EOO modeling paradigm, should be developed by taking into account all possible flow configurations within the evaporators. Stability and integrity tests should also be developed to guarantee the robustness and accuracy of the models. It would be also advisable to develop condenser models, taking advantage of the EOO modeling paradigm, for future developments.

4. To develop a simulation framework for the DISS facility. The simulation framework should facilitate the whole simulation process, e.g. setting inputs, parameters and initial conditions. Furthermore, this framework should consider experimental data from the DISS facility as inputs into the model and outputs in order to compare simulation results.

5. To develop an automated and efficient calibration framework for the DISS Modelica model, considering experimental data from the DISS facility.

6. To validate the DISS Modelica model and the developed evaporator models against experimental data from the DISS facility, considering steady-state and transient predictions to guarantee the accuracy of the models.

1.3 Document Structure

This book is organized across several chapters. An abstract at the beginning of each chapter provides an outline of its content, while summary and conclusion sections are included in all the chapters, except in the first and last chapters. Scientific contributions, related to each chapter, are also enumerated. A brief description of each chapter is presented in the following list.

Chap. 1 - Introduction and Document Structure. The goals, scope, context and document structure of this book are presented in this chapter. The scientific contributions derived from it are also enumerated, together with the research projects which funded it.

Chap. 2 - Modeling & Simulation. In this chapter, Modeling and Simulation (M&S) concepts are reviewed. The benefits of using dynamic simulations are also detailed, and mathematical models and simulations are classified. The evolution of M&S is then reviewed. Furthermore, the modeling paradigms, modeling language, the M&S tool and numerical solver used in conjunction with the dynamic simulations performed in this work are described.

Chap. 3 - Solar Thermal Power Plants based on Parabolic-Trough Collectors and Direct Steam Generation. This chapter provides an introduction to solar concentrating technologies, especially to PTCs and DSG. The applications, main components, design parameters and efficiency of PTCs are discussed. Advantages and disadvantages of DSG are commented upon. The particular DSG parabolic-trough solar thermal power plant considered in this work, the DISS facility, is also described.

Chap. 4 - Two-Phase Flow. Basic relations useful for modeling homogeneous two-phase flows in one-dimensional horizontal evaporators are given in this chapter. The International Association for the Properties of Water and Steam (IAPWS)-Industrial Formulation 1997 (IF97) standard and the *Modelica Media* library, which implements this standard in Modelica, are also introduced. The original contribution in this chapter is a proposed method for calculating the time derivative of the mean void fraction.

Chap. 5 - Modeling the DISS facility. Each component of the DISS facility is modeled using the EOO modeling paradigm and the Modelica language. The evaporator is presented as the keystone in the DISS model, and several two-phase flow models are described, finally selecting a homogeneous model. Two discretization schemes are studied in order to model the evaporator, the FVM and the moving boundary method, which are described in detail in Chaps. 6 and 7, respectively. The original contribution in this chapter is the updating of each DISS model component to the latest advances in Modelica. Additionally, new energy losses models are implemented.

- **Chap. 6 - Finite Volume Method.** The FVM, together with the staggered grid scheme and the Upwind Differencing Scheme (UDS), are introduced in order to model and discretize the evaporator. Two Modelica libraries, *ThermoFluid* and *Modelica Fluid*, are presented. The *ThermoFluid* library is extensively used, improving it and overcoming any problems inherent in this library, e.g. chattering. The *Modelica Fluid* library was adopted at the end of 2011. The original contributions in this chapter are: a study of chattering in finite volume homogeneous two-phase flow models and the development of approaches that can be employed to avoid chattering in dynamic simulations, namely the *Mean Densities* and *Heuristic* approaches.

- **Chap. 7 - Moving Boundary Models.** The moving boundary method is also studied in the evaporator. The basic aspects of Moving Boundary Models (MBMs) are introduced in this chapter, and a brief state-of-the-art study is also presented, which reveals that none of the current models considers dynamic switching between all possible flow configurations in evaporators and condensers. The original contributions in this chapter are mathematical MBMs for one-dimensional homogeneous two-phase flow horizontal evaporators and condensers considering dynamic switching between all possible flow configurations. Additionally, a new Modelica library is developed following an OO approach. Stability and integrity tests are also presented.

- **Chap. 8 - Simulation, Calibration and Validation.** Simulations, calibrations and validations of the DISS Modelica models, considering the FVM and MBMs, are presented in this chapter. The original contributions in this chapter are: simulation and calibration frameworks, calibration and validation results and the parallelization of the initial section of the DISS model, obtaining as a result a reduction in simulation time. The assumption of linear enthalpy distribution in each Control Volume (CV) in MBMs is justified. Finally, a comparison of different DISS models against experimental data in terms of accuracy and performance is presented.

- **Chap. 9 - Contributions, Conclusions and Future Work.** In the last chapter, contributions are presented, conclusions are drawn and future work tasks are proposed.

This book includes a theoretical background for potential readers not specialising in M&S (Chap. 2), concentrating solar technologies (Chap. 3) and two-phase flow (Chap. 4). Readers familiar with these topics can skip these chapters without losing continuity. Except for § 2.12, where the models developed and simulations performed in this work are classified, § 3.6 where the DISS test facility is described and § 4.4 where the method proposed to calculate the time derivative of the mean void fraction is introduced.

1.4 Scientific Contributions

During the development of this work, several scientific contributions have been published or presented. They are organized in two sections. § 1.4.1 lists all the contributions included in this book, and § 1.4.2 lists other contributions directly related to it.

1.4.1 Scientific Contributions Included in this Book

In this section the scientific contributions included in this book are enumerated and organized into categories.

Publication in Journals

1. Bonilla, J., Yebra, L. J., Dormido, S. and Zarza, E. Parabolic-trough solar thermal power plant simulation scheme, multi-objective genetic algorithm calibration and validation. *Solar Energy*, vol. 86(1), pages 531–540, 2012. ISSN 0038-092X. JCR - Impact factor: 2.475 (2011). Q2 - *Energy & fuels.*

2. Bonilla, J., Yebra, L. J. and Dormido, S. Chattering in dynamic mathematical two-phase flow models. *Applied Mathematical Modelling*, vol. 36(5), pages 2067–2081, 2012. ISSN 0307-904X. JCR - Impact factor: 1.579 (2011). Q1 - *Engineering, multidisciplinary*, Q1 - *Mathematics, interdisciplinary applications*, Q2 - *Mechanics.*

3. Bonilla, J., Yebra, L. J. and Dormido, S. A heuristic method to minimise the chattering problem in dynamic mathematical two-phase flow models. *Mathematical and Computer Modelling*, vol. 54(5-6), pages 1549–1560, 2011. ISSN 0895-7177. JCR – Impact factor: 1.346. Q1 - *Computer science, software engineering*, Q1 - *Mathematics, applied*, Q2 - *Computer science, interdisciplinary applications.*

4. Bonilla, J., Yebra, L. J. and Dormido, S. Mean Densities in Dynamic Mathematical Two-phase Flow Models. *CMES - Computer Modeling in Engineering & Science*, vol. 67(1), pages 13–37, 2010. ISSN 1526-1492.

Conference Proceedings

5. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Library of Switching Moving Boundary Models for Two-phase Flow Evaporators and Condensers. In *Proceedings of the 9th International Modelica Conference*, oral communication, Munich, Germany, 2012.

6. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Modeling of Switching Moving Boundary Models for Two-phase Flow Evaporators. In *Proceedings of the 7th Vienna International Conference on Mathematical Modelling (MATHMOD)*, oral communication, Vienna, Austria, 2012.

7. Bonilla, J., Yebra, L. J. and Dormido, S. Exploiting OpenMP in the Initial Section of Modelica Models (Work in Progress). In *Proceedings of the 4th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools (EOOLT)*, oral communication, ETH Zurich, 2011. ISSN 1650-3686.

8. Bonilla, J., Yebra, L. J., Zarza, E. and Dormido, S. Chattering in Dynamic Mathematical Two-Phase Flow Models. In *Proceedings of the European Control Conference 2009*, oral communication, Budapest, Hungary.

Speeches

9. Bonilla, J. Modeling and Simulation of Solar Thermal Power Plants (in Spanish). In *XXXI Jornadas de Automática 2010, Modeling and Simulation Group*, Invited oral communication, September 8 - 10, 2010, Jaén, Spain.

Ph.D. Colloquiums

10. Bonilla, J., Yebra, L. J., Zarza, E., Dormido, S. Chattering problem in dynamic mathematical two-phase flow models.. In 5th *Sollab Doctoral Colloquium on Solar Concentrating Technologies*, Oral communication and poster, June 23 - 26, 2009, Cologne, Germany.

11. Bonilla, J., Yebra, L. J., Zarza, E., Dormido, S. Chattering problem in dynamic mathematical two-phase flow models.. In 4th *Sollab Doctoral Colloquium on Solar Concentrating Technologies*, Oral communication, September 10 - 12, 2008, Tabernas (Almería), Spain.

1.4.2 Scientific Contributions Related to this Work

Other contributions directly related to this work are listed in this section. These contributions are organized according to topics and categories.

M&S of Parabolic-trough Solar Thermal Power Plants

Publication in Journals

12. Yebra, L. J., Berenguel, M., Bonilla, J., Roca, L., Dormido, S. and Zarza, E. Object-oriented modelling and simulation of ACUREX solar thermal power plant. *Mathematical and Computer Modelling of Dynamical Systems*, vol. 16(3), pages 211–224, 2010. ISSN 1387-3954. JCR - Impact factor: 0.452, Q4 - *Computer science, interdisciplinary applications.* Q4 - *Mathematics, applied.*

Conference Proceedings

13. Yebra, L. J., Berenguel, M., Bonilla, J., Roca, L., Dormido, S. and Zarza, E. Object oriented modelling and simulation of ACUREX solar thermal power plant. In *Proceedings of the 6th Vienna International*

Conference on Mathematical Modelling (MATHMOD), oral communication, pages 2025–2033. 2009.

M&S of Concentrating Solar Thermal Power Plants

Conference Proceedings

14. Bonilla, J., Roca, L., Yebra, L. J. and Dormido, S. Real-Time Simulation of CESA-I Central Receiver Solar Thermal Power Plant. In *Proceedings of the 7th International Modelica Conference*, oral communication, pages 345–353. Linköping University Electronic Press, Como, Italy, 2009.

15. Bonilla, J., Roca, L., González, J. and Yebra, L. J. Modelling and real-time simulation of heliostat fields in central receiver plants. In *Proceedings of the 6th Vienna International Conference on Mathematical Modelling (MATHMOD)*, poster, pages 2576–2579, 2009.

M&S of Dynamical Systems

Meetings

16. de la Calle, A., Roca, L., Bonilla, J., Yebra, L. J. Modeling and simulation of a submerged evaporator (in Spanish). In *XXXII Jornadas de Automática*, poster, Seville, Spain, 2011.

1.5 Research Projects

The development of this work could not have been performed without reference to the frameworks developed in the following research projects.

1. Hybridization of Renewable Energy in a Power Plant (Hibridación de Tecnologías Renovables en una Planta de Generación de Energía) (HIBIOSOLEO) - IPT-440000-2010-004. Spanish INNPACTO project, 01/01/2011 - 30/06/2013, Principal researcher: Dr. Luis J. Yebra Muñoz.

2. Modeling and Control Strategy Development for a Hybrid Solar Desalination Plant (Estratégias de Modelado y Control para una Desaladora Solar Híbrida) - DPI2010-21589-C05-02. Spanish National Plan Project 2010, 01/01/2011 - 31/12/2013, Principal researcher: Dr. Luis J. Yebra Muñoz.

3. Solar Facilities for the European Research Area (SFERA) European project, Seventh Framework Programme (FP7), 01/06/2009 - 01/06/2012, Collaborative project: CIEMAT, DLR, CNRS, PSI, ETH, WEIZMANN, ENEA, DIN, UPS, AUNERGY, CEA and INESCD-ID.

II

Theoretical Background

2

Modeling & Simulation

The research rat of the future allows experimentation without manipulation of the real world. This is the cutting edge of modeling technology.

John Spencer

Abstract: This chapter reviews the basic concepts and terminology relating to M&S. It also justifies the development of complex models and the use of computer simulations. Mathematical models and computer simulations are classified into categories relating to different criteria. The evolution of mathematical M&S is reviewed, and special attention is given to the Modelica language and the DASSL numerical integrator.

2.1 System

When talking about M&S, the first and the most difficult concept to be defined is the system concept, for which there is no precise definition (Cellier, 1991). However, this section tries to define or at least explain the concept of a system. The word 'system' comes from the Latin *systēma*, *whole compounded of several parts or members, system*, literary *composition*. A system is a set of interacting or interdependent components forming an integrated whole (IEEE, 2013).

A system, in modeling terms, is a set of elements and relations whose properties need to be studied. A system can contain other subsystems that can be systems by themselves, depending on the scope of the study to be performed. According to the previous definition, a system can be natural, artificial or a mix of both. Systems can have:

- *Inputs*: Environmental variables which influence the system. They are or are not controllable. When they are not controllable, they are called disturbance inputs or disturbances.

- *Outputs*: System variables which may influence the environment. These are or are not observable. When they cannot be measured, they are called internal variables. Internal state variables make up the smallest subset of system variables which can represent the entire state of the system at any time.

- *Variables acting as inputs and outputs*: An a-causal form of behavior is considered when relations between the variables do not have a causal direction; this is the case when relations are described by equations.

2.2 Experiment

Another important concept in modeling terms is the experiment concept, because the concept of a model is defined by a system and an experiment.

An experiment is the process of obtaining information from a system by modifying its inputs. The experimentation process consists in providing a set of inputs and observing the system's reaction by measuring the outputs. Describing the system's reaction is not easy when disturbances and internal states are involved in the reaction process.

2.3 Model

A system model is simply anything where an experiment can be performed in order to obtain information. From the previous simple definition some important conclusions can be inferred (Fritzson, 2004):

- A model is always related to a system and an experiment.

- A model can be used to obtain information without performing an experiment in the real system itself.

- The statement *the model is invalid* only makes sense when the system and an experiment are considered.

- A model can be valid for one experiment but invalid for a different one.

2.4 Simulation

The simulation definition is straightforward – a simulation is an experiment performed on a model (Fritzson, 2004). As mentioned in § 2.3, one of the main advantages of having a model is the ability to perform experiments in the model itself, which is called 'simulation'.

When simulating a model it is important to note that two definitions must be available: one to define the model and another one to define the experiment.

2.5 Motivation

Now we have introduced the system, experiment, model and simulation concepts, the ongoing question is why a model must be built, an experiment must be defined and a simulation must be performed. Simulation is a powerful tool with a great number of advantages.

- Models can be used to obtain a better understanding of the system.
- Models can be design prototypes, which allow for experimentation with systems which might not exist.
- Models can be easily manipulated, parameters can be changed and how these changes affect the system can be observed.
- Experiments in the real system can be too expensive to be performed.
- Experiments in the real system can be dangerous for people or resources.
- Experiments in the real system can be too fast or to slow to be properly studied.
- Disturbances can be neglected to focus on a particular reaction without second-order effects.

2.6 Classification of Mathematical Models

From this point on, only mathematical models will be considered. However, there are other kinds of models besides mathematical models, i.e. mental models (statements), verbal models (expressed in word) and physical models (small physical systems) (Fritzson, 2004).

Mathematical models are represented by equations, functions, computer programs, etc. In general, they describe systems by using mathematical concepts. Mathematical models can be classified by different criteria. The following subsections briefly describe the most common categories in which mathematical models can be classified.

2.6.1 Black-Box, White-Box and Grey-Box Models

Mathematical models can be classified according to the design process performed to obtain a model.

- *Black-box model* – also known as a parametric model, which is adapted to measured data obtained from experiments performed in a real system. No knowledge about the system is reflected in the model.

- *White-box model* – also called a physical model. This model is the most detailed type of model wherein physical processes are described at as low a level as possible, with no approximations or bulk parameters.

- *Grey-box model* – also called a parametrized physical model. In common with white-box models, a grey-box model provides a physical representation, but some of the physics are approximated, considering parameters or simplifying relations in the model.

2.6.2 Linear and Nonlinear Models

Mathematical models can be linear, when all the equations describing the model are linear, and nonlinear otherwise. Nonlinear problems tend to be more difficult to study and solve. A common practice used to solve nonlinear problems is linearization, although a linearized model is only accurate in a narrow range of operation.

2.6.3 Dynamic and Static Models

Depending on the consideration of time as a variable in the mathematical formulation, a model can be a dynamic when time is considered, or a static otherwise.

Static models are also called steady-state models. They do not take into account transient states. An equilibrium situation, where outputs do not change if inputs are held constant, is considered.

Dynamic models can be dependent on their previous state, i.e. if a time derivative term is present in the mathematical formulation. This time derivative is commonly numerically integrated by numerical solvers in M&S tools, where previous values are normally used.

2.6.4 Continuous, Discrete and Hybrid Models

Mathematical models can also be classified according to when the model evolves. It is important to note that all these kinds of models are dynamic, because all of them evolve over time.

- *Continuous-time model.* Model variables evolve continuously over time.
- *Discrete-event model.* Model variables may change their value at discrete points in time, called events.
- *Hybrid Model.* Includes a combination of interacting continuous-time and discrete-event dynamics. It is also called a 'combined model' (Cellier, 1979).

2.6.5 Variable-structure and Static-structure Models

In a variable-structure model the number/type of equations or variables may change. On the other hand, a static-structure model implies that the number/type of equations and variables remain the same. Variable-structure models are not generally supported by M&S tools. There are some modeling languages and tools which support variable-structure models, but none of the existing variable-structure M&S tools supports the handling of higher-index systems (Zimmer, 2010). The notion of a differential index is introduced in § 2.9.2.

2.6.6 Deterministic and Probabilistic Models

In deterministic mathematical models, state variables are uniquely determined by inputs, parameters and the previous state of the model. On the other hand, probabilistic or stochastic models have a random component, where state variables are not uniquely defined by certain values but by a probability distribution thereof.

2.6.7 Quantitative and Qualitative Models

Mathematical models can also be classified by the kind of information they provide. In quantitative models, variable values are represented numerically according to a measurable scale, whereas qualitative models provide a classification in a finite set of values. The information provided by qualitative models is, by definition, discrete and rough – they are discrete-event models and provide rough approximations, e.g. a qualitative model could provide values such as bad, good, outstanding, etc.

2.7 Modeling

Modeling is the *art* of translating and simplifying problems into tractable formulations which provide insights, answers or useful information. Modelers must be skillful and have a fair amount of knowledge in order to express the aspects of reality that are of interest. In the engineering domain, the appropriate modeling formalism is mathematics. Mathematical modeling is an arduous task, and real-world problems should be expressed in a simplified and tractable mathematical formulation reflecting the principal dynamics. Modeling languages should make it easy to represent the modeler's ideas, whereas modeling paradigms and tools should facilitate the implementation process and code reuse. This section inspects some common modeling paradigms.

2.7.1 Modeling Paradigms

Several modeling paradigms have been adopted by M&S tools over the years, and each one has its own way of representing the system. Some of the most commonly extended modeling paradigms are summarized as follows. An outstanding modeling formalisms review is presented in Åström et al. (1998).

2.7.1.1 Block-diagram and Equation-based Modeling

- *Block-diagram modeling.* It is inherited from the analog simulations, where each block represents a basic operation (addition, subtraction, multiplication, etc). Basic operations can be combined to describe an explicit equation. The union between blocks is commonly done graphically drawing lines between blocks. This paradigm can be also described as a set of known and unknown quantities, each block uses known quantities to compute the unknowns. This approach is commonly used to design control systems (Tiller, 2004).

- *Equation-based modeling.* This modeling paradigm is also known as physical modeling (Åström et al., 1998), a-causal modeling or first principle modeling (Tiller, 2004). A typical procedure for this kind of modeling consists in cutting the system into subsystems and communicating with them through interfaces. This paradigm considers a-causal behavior, meaning that relations between variables do not have a causal direction, and there is no explicit specification of system inputs and outputs. This is the case when relations are described by equations. When modeling using this paradigm, the system of equations does not have to be manually manipulated to obtain explicit equations depending on the experiment, because the inputs and outputs are not explicitly defined. This contributes considerably to model maintenance, validation, correctness and reusability (Cellier, 1996).

2.7.1.2 Object-Oriented and Procedural Modeling

Object-oriented and procedural modeling paradigms are related to programming paradigms.

- *Procedural modeling*. Procedural modeling is based on the concept of the procedure call. Procedures, also known as methods, routines or functions, contain a series of computational instructions. Therefore, explicit equations or parts of the equations which are called several times can be generalized (using arguments) in a procedure, which provides basic code reuse.

- *Object-oriented modeling*. This paradigm considers a set of elements (objects) interacting between themselves. This definition is similar to the system definition previously introduced in § 2.1, which means that each object can be modeled independently from each other, thus defining an interface and an implementation. Each object has its own implementation and interacts via its interface. This paradigm has a fair amount of features: encapsulation, abstraction, polymorphism, inheritance, etc., all of which contribute to model maintenance, reusability and decoupling (Cellier et al., 1996). If the equation-based modeling paradigm (cf. § 2.7.1.1) is also considered, an EOO modeling paradigm is defined. The EOO modeling paradigm improves objects reusability even more, because objects can be used for a wider range of models and experiments.

2.7.1.3 Process-Oriented Modeling

This modeling paradigm represents logistic systems in a natural way, in terms of entities, processes and resources. It focuses on how entities flow through different processes in the system (Law and Kelton, 1991). In order to process entities, resources might be necessary. Resources are acquired and released by processes. An excellent review of process-oriented modeling, particularly for Modelica, is given in Sanz (2010).

2.8 Classification of Simulations

The following subsections provide two different kinds of classification for mathematical model simulations.

2.8.1 Static and Dynamic Simulations

Depending on the consideration of time as a variable in the mathematical formulation, simulations can be dynamic when time is considered or static otherwise.

A static simulation, in its more general sense, refers to a simulation where time is not involved; it is also referred to as stationary simulation. However, static simulations are also referred to as Monte Carlo simulations. These methods are used for iteratively evaluating a model using sets of random numbers as inputs. The Monte Carlo simulation is a method for analyzing uncertainty propagation, where the goal is to determine how random variation, lack of knowledge or errors affect the sensitivity, performance or reliability of the model (Rubinstein and Kroese, 2008).

On the other hand, dynamic simulations evaluate how models evolve over time.

2.8.2 Continuous, Discrete and Hybrid Simulations

This section is analogous to § 2.6.4, and it has been included for completeness and to emphasize that for discrete and hybrid simulations, the numerical solver must treat simulation events, which is commented on in § 2.9.2. These kinds of simulations are for dynamic models, hence they are dynamic simulations because time is explicitly considered.

2.9 Simulation Techniques

Simulation is as important as modeling, as for a model to be useful it must be simulated. A considerable part of the modeling effort is spent on obtaining models which have good numerical properties in order to be simulated.

However, one question nowadays is becoming more and more confusing: *Where does modeling end and simulation begin?* (Cellier, 1991). As the author states, years ago the answer was clear: modeling was done manually by the user and simulation was done automatically by the computer. Obviously this answer is not satisfactory, because it states that the simulation concept depends on the simulation tool – some M&S tools accept implicit mathematical models while others do not. A more appropriate answer could be formulated from Fig. 2.1.

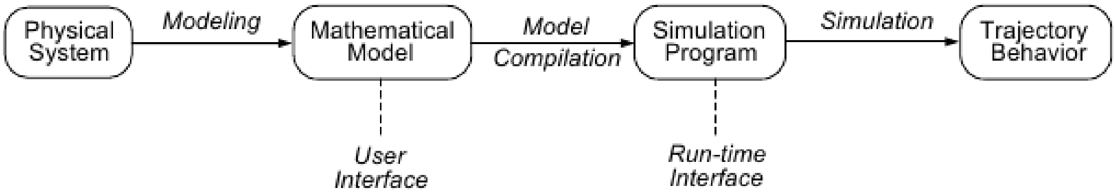


Figure 2.1: Modeling & simulation (Cellier, 1991)

The mathematical model acts as the user interface which allows the user to express knowledge about the physical system to be simulated, but it has absolutely nothing to do with considerations of how that model is going to be used by the M&S tool. Once the mathematical model has been formulated, the M&S tool can make use of that model to perform model compilations, run simulations and produce simulation results (Cellier, 1991).

2.9.1 Transforming Mathematical Models to be Solvable

This section aims at generally enumerating the necessary previous step to transform any mathematical model following the EOO modeling paradigm to a solvable form for an Ordinary Differential Equation (ODE) numerical solver. § 2.9.2 mentions the kinds of systems of equations resulting from mathematical models.

1. *Flattening.* This step consists in unwrapping the equations from the object-oriented approach; hence, at the end of the process, a set of flat equations is obtained.

2. *Horizontal sorting.* This consists in the assignment of causality, determining which equation is used to calculate each model variable.

3. *Vertical sorting.* It consists in sorting the equations in such a way that the dependable variables of each equation are calculated before the equation is calculated.

Horizontal and vertical sorting processes are not trivial tasks, because two major problems can arise: algebraic loops and structural singularities (Cellier and Elmqvist, 1993). The Tarjan algorithm (Tarjan, 1972) consists of a graphical technique to simultaneously sort systems of equations, both horizontally and vertically.

The Tarjan algorithm can be used to identify algebraic loops. When this algorithm becomes stuck it is because there is an algebraic loop, so a tearing variable must be selected in order to break the algebraic loop. Algebraic loops can be solved analytically or numerically. When loop equations are non-linear or linear but with a considerable number of equations, the Newton iteration method on the tearing variable may be used. If the loop equations are linear and the number of equations is small, it can be more efficiently solved by using matrix techniques or formulae manipulation (Elmqvist and Otter, 1994), e.g. the relaxation algorithm (Otter et al., 1996) or the tearing algorithm (Elmqvist and Otter, 1994). Heuristic rules are used to select the appropriate method (Cellier, 1991).

Some structural singularities can be removed by the Pantelides algorithm (Pantelides, 1988), which is a symbolic index-reduction algorithm. It may be necessary to apply the Pantelides algorithm multiple times in order to reduce the index of the system (Cellier and Elmqvist, 1993). The notion of a differential index is introduced in § 2.9.2.

2.9.2 Solving Mathematical Models

This section mentions the kinds of equation systems resulting from mathematical models. Commonly, numerical solvers are required to simulate them;

exceptionally, an analytical solution can be obtained for a certain group of models.

- *Ordinary differential equations (ODEs).* These are continuous-time models. An ODE system is a special Differential Algebraic Equation (DAE) system without algebraic equations. Numerical solution methods employed to solve ODEs are often called integration methods, since the explicit ODE system can be solved by integration. The most common methods are Runge-Kutta methods, multistep methods and stiff methods (Chapra and Canale, 1998). A special mention goes to LSODE1, LSODE2, LSODAR (Hindmarsh, 1983). LSODE1 is a variable order multistep method that uses Adams Bashforth Moulton (ABM) formulas, making it suitable for non-stiff problems. LSODE2 is a multistep Backward Differentiation Formula (BDF) method and is a good choice for stiff problems. LSODAR is a combination of LSODE1 and LSODE2, and switches between both ODE solvers if stiffness is detected. LSODAR also includes a root solver to handle discrete events for hybrid ODE systems.

- *Differential algebraic equations (DAEs).* This is also the case for some continuous-time models. A DAE system is a special hybrid DAE system without discrete or conditional equations. The main difference between solving an ODE system and a DAE system is that a DAE system may require differentiation. The differential index of a DAE system is the minimum number of times that certain equations need to be differentiated to reduce the DAE system to an explicit ODE system. A remarkable DAE solver is the Differential Algebraic System SoLver (DASSL) (Petzold, 1983).

- *Hybrid Differential algebraic equations (hybrid DAEs).* It is the case for some continuous-time discrete-event mixed models. A numerical hybrid DAE solver with support for event handling is required to solve these kinds of systems. Different DAE solvers from the DASSL family can be used to solve hybrid DAE systems, i.e. DASSRT or DASSPK, more details of which are given in § 2.9.2.1.

Partial Differential Equations (PDEs). These can also be considered as a category, although they are not as supported and extended in M&S tools as the previously mentioned categories (Saldamli et al., 2002). When the modeler is faced with a system expressed by PDEs, a method to discretize and transform it to a DAE system is applied. The classical discretization methods are: the Finite Difference Method (FDM) (Gerald and Wheatley, 1994), FVM (Patankar, 1980) and Finite Element Method (FEM) (Gerald and Wheatley, 1994), which are detailed in § 5.3. MBMs (Adams et al.,

1965) also take the discretization of PDE systems into account, and this is explained in Chap. 7.

2.9.2.1 The DAE Solvers of the DASSL Family

DASSL (Petzold, 1983) stands for Differential/Algebraic System Solver and was developed by Linda Petzold in 1982. It is a variable-step variable-order multistep method for the numerical solution of DAE systems. It implements the BDFs of orders one through five. At each integration step, the Newton iteration method is used to solve the resulting nonlinear algebraic system. DASSRT is a root-finding version of DASSL. DASPK (Brown et al., 1994) was designed for solving large-scale DAEs systems.

The DASPK solver is included in the latest version of Dymola (Dassault Systemes, 2012), and although it is called DASSL in the Dymola tool, from now on both terms will be used indistinctly. DASPK is the default numerical solver in Dymola for several reasons. First, it supports for stiff systems, which most complex models are, and second, is the production code, which involves: step-size and order control, start-up problem, discontinuity handling, etc. (Cellier and Kofman, 2006). Although DASPK can handle DAE systems (index 1), Dymola transforms models to explicit ODE systems (index 0), reducing the index as mentioned in § 2.9.1. This is primarily done in order to facilitate the setting of the initial conditions by the modeler in the Dymola tool, because setting initial conditions for DAE systems is more complex than for ODE systems. Therefore, the transformation to an explicit ODE system ensures that a consistent and complete set of initial conditions is available to properly initialize the simulation (Cellier and Kofman, 2006).

Sensitive analysis capability was implemented in DASSL and DASPK, yielding in DASSLSO and DASPKSO (Maly and Petzold, 1996). Data-parallel (DASPKF90) and message-passing (DASPKMP) versions of DASPK were also developed (Maier et al., 1995). For further details consult (Brenan et al., 1996).

2.10 Evolution of Mathematical Modeling & Simulation

The first mathematical models were simulated by building small physical systems, known as analog simulations. The general idea was to model a real system by ODEs and to build a physical system which obeys such equations (Åström et al., 1998). The mechanical differential analyzer developed at Massachusetts Institute of Technology (MIT) was considered the first general tool to simulate dynamic systems (Bush, 1931).

After that, it was proven in Ragazzini et al. (1947) that simulations could be performed using electronic circuits instead of mechanical devices, which also facilitated experiment set up and the measurement of variables represented by electric tensions. This encouraged the use of analog computers – and hence emerging companies that produced electronic simulators. Jackson (1960) gives an excellent review of analog computing. However, there were issues with this simulation approach which led to error-prone modeling. As equations must be transformed and represented using basic operations, e.g. integrations, additions and multiplications, it was necessary that multiple interconnections between the model elements, used to represent multivariable functions and mathematical manipulations, had to be performed manually in order to represent the model, e.g. the removal of algebraic loops (Åström et al., 1998).

Soon after the appearance of digital computers, it was determined whether they could be used to perform simulations (Åström et al., 1998); for instance, Selfridge (1955) demonstrated a differential analyzer emulated in a digital computer. This was an intense period, because previous work could be reused by analog computer diagrams, the block-diagram paradigm. It therefore seemed that changing the technology was easier than changing the paradigm (Cellier, 1991).

MIMIC (Oregon State University, 1969), developed in 1964, was one of the first equation-oriented languages for M&S. The MIMIC language was totally flat and did not allow for any kind of organization (Zimmer, 2010). In 1967, the Continuous System Simulation Language (CSSL) standard (Strauss, 1967) was developed by the Simulation Council inc. (SCi) in order to unify the concepts and language structures of simulation programs available up to that point. An exceptional review of the CSSL standard is given in Rimvall and Cellier (1986). The CSSL standard was implemented by a great number of software products, the most remarkable example of which was Advanced Continuous Simulation Language (ACSL) , developed by Mitchell and Gauthier (1976). Different approaches to CSSL were also taken, e.g. the program SIMNON, developed by Lund University in 1972 (Elmqvist, 1975).

The first numerical methods for solving DAEs appeared in 1970. One of the first publications was made by Gear (1971). This was an active research period for ODE numerical solvers, e.g. Runge-Kutta methods (Gerald and Wheatley, 1994), and DAE numerical solvers, e.g. DASSL (Petzold, 1983).

However, even considering digital computers with programming languages and numerical solvers substituting for physical devices, modeling was still an arduous and error-prone task.

Prototype graphical environments were designed in 1970, but they were not commonly used in modeling until the advent of modern PCs. In 1980, the matrix environment Matrix$_X$ (Shah et al., 1985) and Matlab (The MathWorks Inc., 2012) were introduced, and M&S tools were later introduced in both environments, namely SystemBuild (1984) and Simulink (Grace, 1991), respectively. These tools supported the graphical block-diagram methodology (cf. § 2.7.1.1) inherited from analog computers. This methodology provides advantages when regarding modularity and hierarchy, but reusability is still limited because the causality of the models is fixed. This means that the modeler has to manually perform manipulations for equation systems.

Two of the first contributions related to EOO modeling were created by Elmqvist (1978) and Runge (1977). The first author was concerned about symbolic formulae manipulation in order to efficiently execute simulation code, so he implemented the first version of the Dynamic Modeling Laboratory (Dymola) language. The second author aimed to solve implicit DAE models directly.

However, Dymola still had two major drawbacks, namely algebraic loops and structural singularities. After a period of active research with some remarkable contributions, e.g. trying to solve these two problems in a generic way (Cellier and Elmqvist, 1993), symbolic tearing methods (Elmqvist and Otter, 1994) and heuristic algorithms used to select tearing variables, finally the tool was prepared in 1997 to automatically reduce any implicit DAE system to an explicit ODE system. The Dymola tool (not the Dymola language) was first presented in 1982.

In terms of software for aiding the design of numerical algorithms, Matlab became the industry standard. Special mention should go to the advances in tools for computational algebra and symbolic manipulation, e.g. Mathematica (Wolfram, 2010) and Maple (MapleSoft, 2012a), which made great advances in facilitating the design of new algorithms – even errors present in previous numerical algorithms were detected using computational algebra tools (Gander and Gruntz, 1999).

At the end of the 1990s, the situation was similar to that in the sixties, whereby there were considerable numbers of modeling languages. In 1996, a new association was established with the aim of unifying the knowledge and experiences of previous modeling languages, the Modelica Association. The first Modelica language specification was released in 1997. The development of the language included several characteristics from previous languages, e.g. ASCEND (Piela et al., 1991), ALLAN (Jeandel A. et al., 1997), Dymola (Elmqvist, 1978), Neutral Model Format (NMF) (Sahlin et al., 1996), ObjectMath (Fritzson et al., 1995), Omola (Andersson, 1990), SIDOPS+ (Breunese and Broenink, 1997), Smile (Kloas et al., 1995) and gPROMS (Barton and Pantelides, 1994), the latter of which was the commercial version of ABACUSS II (Clabaugh, 2001), which is especially important because this language generated a set of processing methods for DAEs, e.g. the Pantelides algorithm (Pantelides, 1988) .

Presently, there are many languages that support the EOO methodology, and some of the more remarkable are the following: Modelica (Modelica Association, 2010), gPROMS (Barton and Pantelides, 1994), EcoSimPro (EA International, 2011), Verilog-AMS (Frey and O'Riordan, 2000), VHDL-

AMS (IEEE, 1997) and 20-Sim (Weustink et al., 1998). Regarding academic approaches, the following are worth mentioning: HYBRSIM (Mosterman, 2002), ABACUSS II, χ (Chi) (Van Beek and Rooda, 2000) and SOL (Zimmer, 2010).

Lately, a number of academic modeling language initiatives have sought to tackle the problem of simulating variable-structure mathematical models, e.g. Mosilab (Bastian et al., 2010) , HYBRSIM, χ (Chi), Hydra (Giorgidze and Nilsson, 2009) and SOL. Another recent approach consists in using scripts to handle switching during simulation (Mehlhase, 2011). Following this approach, Matlab is used to switch between Modelica models in Krüger et al. (2012), while in Mehlhase (2012), a python tool performs the switching function.

2.11 The Modelica Language

Modelica is a non-proprietary EOO modeling language. It was developed and is maintained by the Modelica Association, a non-profit, non-governmental international association composed of organizations and individual members from Europe, USA and Canada, with the aim of developing and promoting the Modelica modeling language for the modeling and simulation of physical and technical systems.

The Modelica Association provides newsletters, free educational materials, training courses, consultant services, job offers and student work. They are also involved in the organization of the International Modelica Conference, where users, library developers, tool vendors and language designers can share their knowledge and learn about the latest scientific and industrial progress related to Modelica. Modelica-related technology is an active research field, exemplified by the fact that Europe invested 54 million euros during the 2007-2012 period to foster further improvements in the EUROSYSLIB (EuroSysLib, 2010), MODELISAR (Modelisar, 2011) and OPENPROD (OpenProd, 2012) Information Technology for European Advancement (ITEA)-2 (ITEA-2, 2012) projects.

2.11.1 Main Features

The main feature in the Modelica language is support for the object-oriented and equation-based paradigms, from which can be derived other important aspects which define the language.

- *Equations and Algorithms.* Models can be described by a-causal equations, where the tool is in charge of performing symbolic manipulations, or by algorithms with fixed causality.
- *The Initialization Problem.* Modelica provides constructions for tackling the initialization problem, e.g. *initial values* which can be fixed or

guessed values, and *initial equations* in order to define equations which are only valid during the initialization process.

- *Selection of State Variables.* State variables can be selected by the user or automatically selected by the M&S tool.
- *Events.* Modelica supports time and state events, so the tool is then in charge of defining the resulting hybrid DAE . This operation involves root finding to locate the time instant and the reinitialization of the simulation (Cellier, 1991).
- *Abstraction.* This feature is supported by means of partial classes, models, etc. These elements are not directly instantiable but define general knowledge, hence abstracting and sharing that knowledge for specialized classes or models in the hierarchy.
- *Encapsulation.* As in conventional programming languages, each class or model can define public, protected or private elements, hence defining an interface with the rest of the elements and hiding the implementation.
- *Multiple inheritance.* Modelica supports multiple inheritance, which facilitates the inclusion of different information from interfaces or partial classes.
- *Polymorphism.* This feature allows objects with the same interface to have their own functionality and the selection of the appropriate one as required.
- *Advanced Parametrization.* Modelica includes a powerful feature, an advanced system of parametrization, which allows for changes to models (polymorphism) or even the model *class*, besides conventional model parametrization involving only values of basic data types.
- *Connectors.* Interaction between different models is defined by physical ports, which are known as ‘connectors’ in Modelica. Connecting two connectors implies different relations between the port variables, depending on whether or not they have flow or stream prefixes. The sum of all corresponding flow variables in a connector is set to zero; if there is no prefix, the variables are matched. Stream variables were introduced for designing thermo-fluid components. For further information, consult Franke et al. (2009a).
- *Annotations.* They help modelers to define additional information about models or libraries, e.g. graphical information such as icons and component diagrams, documentation, version, etc. Some annotations, however, are dependent on the M&S tool.
- *External functions.* The Modelica language supports the use of code written in other languages: Java, Fortran and C. This feature facilitates code reuse, even across different languages.

2.11.2 Modelica Standard Library

The most important Modelica library is the free Modelica Standard Library (MSL) (Modelica Association, 2010) maintained by the Modelica Association. The latest version available is the 3.2 build 5. This library provides models and interfaces for different engineering domains, and the main subpackages in the MSL are summarized in the following list. Besides the MSL, there are other free and commercial libraries available, a complete list of which can be found in Modelica Association (2012a).

- *Blocks.* Continuous, discrete, math, logical, nonlinear, routing, source and table blocks.
- *Constants.* Mathematical and physical constants.
- *Electrical.* Electric and electronic components.
- *Fluid.* One-dimensional fluid components.
- *Icons.* Icons for components, models and packages.
- *Magnetic.FluxTube.* Devices based on magnetic flux tubes concepts.
- *Math.* Scalar and vectorized mathematical functions and operators.
- *Mechanics.* Rotational, translational and multi-body mechanical components.
- *Media.* Medium models for fluids, about 1,250 forms of media.
- *SIunits.* SI-units and Non-SI-units type definitions, as well as conversion functions.
- *StateGraph.* Hierarchical state machines.
- *Thermal.* Heat transfer and fluid heat flow components.
- *Utilities.* Files, system commands, stream and string utility functions.
- *ModelicaServices.* Functions and models to be used in the MSL which require tool-specific implementation.

2.11.3 Modelica Tools

A complete list of available Modelica tools can be obtained from Modelica Association (2012b), including M&S environments, teaching materials, Modelica editors, a Modelica test suite and other useful tools. Some of the most remarkable commercial M&S environments for Modelica are: Dymola (Dassault Systemes, 2012) , Vertex (Delthatheta, 2012b), Converge (Delthatheta, 2012a), MOSILAB (Bastian et al., 2010), SimulationX (ITI, 2012), LMS Imagine.Lab AMESim (LMS, 2012), MapleSim (MapleSoft, 2012b), MathModelica (MathCore Engineering AB, 2012), Modelon Toolbox for Matlab (Modelon AB, 2012) and Wolfram SystemModeler (Wolfram, 2012). Free M&S tools include: JModelica.org (JModelica.org, 2012), Modelicac (Digiteo, 2012), OpenModelica (Pelab, 2012) and SimForge (SimForge, 2012).

The main goal in this work is the development of models intended to be used as design tools for ACSs. Modelica allows for the design of ACSs, although Matlab/Simulink is most commonly used for this purpose. Therefore, one desirable feature would be integration between both environments. The Dymola tool provides a Simulink block called *Dymola Block* (Dassault Systemes, 2012) which allows one to import any Modelica model into Simulink. The *Dymola Block* is used for calibration purposes in this work (cf. § 8.5), but this is a tool-dependent solution. An independent tool solution is being developed in the ITEA-2 project MODELISAR (Modelisar, 2011), called Functional Mockup Interface (FMI). The FMI (Modelisar, 2012b) defines an open interface which can be implemented by an executable, called a Functional Mockup Unit (FMU). FMI functions are used by M&S tools in order to create instances of FMUs (which are models themselves) and to run these models accordingly (Otter et al., 2011).

One of the objectives of the development and usage of tool-independent modeling languages is to ease the model exchange between M&S environments (Blochwitz et al., 2010). Several tools offer proprietary model interfaces, e.g. S-Function in Simulink. Currently, no tool-independent standard for model exchange is available, besides FMI. The same holds for the situation in the field of co-simulation. Presently, some modeling and M&S tools support FMI, e.g. Dymola, Simulink, NI Labview, etc., directly through the tool or through free or commercial toolboxes or plugins. For a complete list of M&S tools which support FMI, consult Modelisar (2012a). Some of these tools support FMI partially by means of including only a few features, such as importing or exporting exchange models or co-simulation, while others plan to include new features soon. Additionally, a FMU Standard Development Kit (SDK) is available, which allows for integrating FMI support in any tool. The latest stable specification version of FMI is v1.0 for model exchange (Modelisar, 2010b) and co-simulation (Modelisar, 2010a).

2.12 Summary and Conclusions

This chapter has briefly described the main concepts of M&S. Classifications of mathematical models and simulations and different modeling paradigms have also been introduced. The evolution of M&S has been presented, together with details of the Modelica language and the DASSL solver. Further information can be found in Cellier (1991) for mathematical modeling, Cellier and Kofman (2006) for simulation and Fritzson (2004); Tiller (2004) for the Modelica language. Tab. 2.1 enumerates the modeling language, the M&S tool and the numerical solver used in this work, together with the categories of the developed models and performed simulations.

Modelica was selected as the modeling language in this work because it is an EOO modeling language used widely in academia and industry, and it is strongly supported in Europe through investment and research projects. The Dymola tool has been used because it supports the latest Modelica specification, the MSL, the DASSL solver, FMI (for future developments) and provides mechanisms to import models in Simulink, which is commonly used for the design of ACSs. DASSL has been used due to its good performance, its support for stiff systems and its production code for the initialization and handling of discontinuities.

<table>
<tr><td rowspan="2">Modeling language</td><td>Language</td><td>Paradigms</td></tr>
<tr><td>Modelica
2.2.1 / 3.2 (§ 2.11)</td><td>Equation-based (§ 2.7.1.1)
Object-oriented (§ 2.7.1.2)</td></tr>
<tr><td rowspan="2">Developed models</td><td colspan="2">Categories</td></tr>
<tr><td colspan="2">Parametrized Physical (Grey-box) (§ 2.6.1)
Nonlinear (§ 2.6.2), Dynamic (§ 2.6.3)
Hybrid (§ 2.6.4), Static-structure (§ 2.6.5)
Deterministic (§ 2.6.6), Quantitative (§ 2.6.7)</td></tr>
<tr><td rowspan="2">M&S tool</td><td colspan="2">Name</td></tr>
<tr><td colspan="2">Dymola 6.0b / 2013 (§ 2.11.3)</td></tr>
<tr><td rowspan="2">Numerical solver</td><td>Name</td><td>Type</td></tr>
<tr><td>DASSL (§ 2.9.2.1)</td><td>Hybrid DAE (§ 2.9.2)</td></tr>
<tr><td rowspan="2">Performed simulations</td><td colspan="2">Categories</td></tr>
<tr><td colspan="2">Dynamic (§ 2.8.1), Hybrid (§ 2.8.2)</td></tr>
</table>

Table 2.1: Modeling language, M&S tool, numerical solver, categories of models developed and simulations performed

3

Solar Thermal Power Plants based on Parabolic-Trough Collectors and Direct Steam Generation

> *We are like tenant farmers chopping down the fence around our house for fuel when we should be using Nature's inexhaustible sources of energy — Sun, wind and tide... I'd put my money on the Sun and solar energy. What a source of power! I hope we don't have to wait until oil and coal run out before we tackle that.*
>
> Thomas Edison, *in conversation with Henry Ford and Harvey Firestone*, 1931

Abstract: This chapter intends to give a brief review about concentrating solar thermal technology and its purpose. Special mention is given to PTCs, wherein their applications, main components, working principles, design parameters, HTFs, heat losses, efficiency and energy balance are presented. DSG technology is also introduced, and different DSG operating modes in parabolic-trough solar thermal power plants are presented and discussed. Finally, an experimental DSG parabolic-trough solar thermal power plant, the DISS test facility, is described. The concepts previously introduced for PTCs are then particularized for the DISS solar thermal power plant.

3.1 Concentrating Solar Thermal Power

About 93% of solar energy may be theoretically converted to mechanical work (Winter et al., 2003), because solar energy has a high exergetic value originating from nuclear processes occurring on the Sun's surface, which is approximately equivalent to a black-body of 5777 K. Conversion from solar heat to mechanical work, according to thermodynamics and Planck's equation, is limited to the Carnot efficiency. Therefore, in order to maximize the conversion rate, energy should be transferred to a thermal fluid.

The irradiance available for terrestrial use, and for supplying to a thermal fluid, is only slightly higher than 1 kW/m^2, although the source, at its origin, is high in temperature and energy with a heat flux of 63 MW/m^2 (Romero-Alvarez and Zarza, 2007). This decrease in heat flux is a consequence of the dilution of the Sun's power along its journey to the Earth's surface, and extraterrestrial irradiance is modified as it enters the atmosphere because of absorption and multiple dispersions. For this reason, it is essential for solar thermal power plants to use optical concentration devices that allow for a high solar flux transfer and minimal heat loss. The solar radiation that passes through the atmosphere directly to Earth's surface is called DSR, while radiation that scatters out from the direct beam is called 'diffuse solar radiation'. Only DSR can be concentrated.

The design of concentrating solar thermal power plants should consider a large and highly efficient optical (high reflectivity) surface which concentrates solar radiation onto a receiver with a small aperture area, high absorptance and transmittance, and low reflectance and emittance. Ideally, the receiver should not have convective or conductive losses. Heat should be transferred to a thermal fluid which, through the solar thermal power plant, should acquire a high enough temperature to feed a heat engine or a turbine in order to produce electricity (Romero-Alvarez and Zarza, 2007). From different concentrating solar thermal elements, several solar power plants are defined.

Transients and fluctuations in irradiance can be mitigated by using oversized mirror fields and by loading excess energy into thermal or chemical storage . Hybrid plants, running conventional fossil power plants in series or in parallel, are possible. Heat storage systems and hybrid plants make concentrating solar power systems highly flexible for integrating with existing conventional power plants, biomass or geothermal power plants. Hybridization can also be used in power booster and fuel saver modes with natural gas combined cycles and coal-fired Rankine plants, which may accelerate the deployment of projects due to reduced project risk and improved economics (Romero-Alvarez and Zarza, 2007).

3.2 Motivation

Solar concentration generates high quality energy, because higher temperatures, and hence greater capacity for generating mechanical work, can be obtained. According to the second law of thermodynamics, the higher the temperature, the better the efficiency of a heat engine (a cold reservoir at a constant temperature). With solar concentration, the absorber aperture area can be reduced, thus minimizing heat losses and therefore reducing investment costs. The most remarkable advantages of concentrating solar power are summarized in the following list.

- Dispatchability on demand is one important difference between concentrating solar plants and other renewable energy technologies, e.g. wind or photovoltaics (Romero-Alvarez and Zarza, 2007). Thermal energy storage allows for the dispatching of a predefined energy load, even under disturbances (not for seasonal differences).
- Concentrating solar plants are the most economical form of solar technology for electricity production, according to an independent study promoted by the World Bank (Enermodal Engineering Limited, 1999).
- Solar energy does not contaminate and it is independent of political or economic interruptions to supply.
- Proven capabilities of plants in operation in the United States, Europe, Northern Africa, Middle East and other countries (Fernández-García et al., 2010).
- Modular and thus suitable for large power plants.
- They can be rapidly deployed using existing infrastructures.
- Proven potential for further cost reduction (Romero-Alvarez and Zarza, 2007).
- Concentrating solar power plants do not produce CO_2, therefore they are suitable for maintaining standards of living, without compromising environmental issues.
- The energy payback time is less than one year (Romero-Alvarez and Zarza, 2007).
- Most solar field materials and structures can be recycled and used again for new plants.

3.3 Concentrating Solar Thermal Power Plant Technologies

At this point in time, there are at least four concentrating solar technologies in pilot plants: PTCs, LFs, CRSs and DEs, all of which are represented in Fig. 3.1.

The parabola is the simplest and most common geometrical concentrator used in concentrating solar systems. Reflected radiation is concentrated onto a focal point located on the optical axis. Diffuse solar radiation cannot be

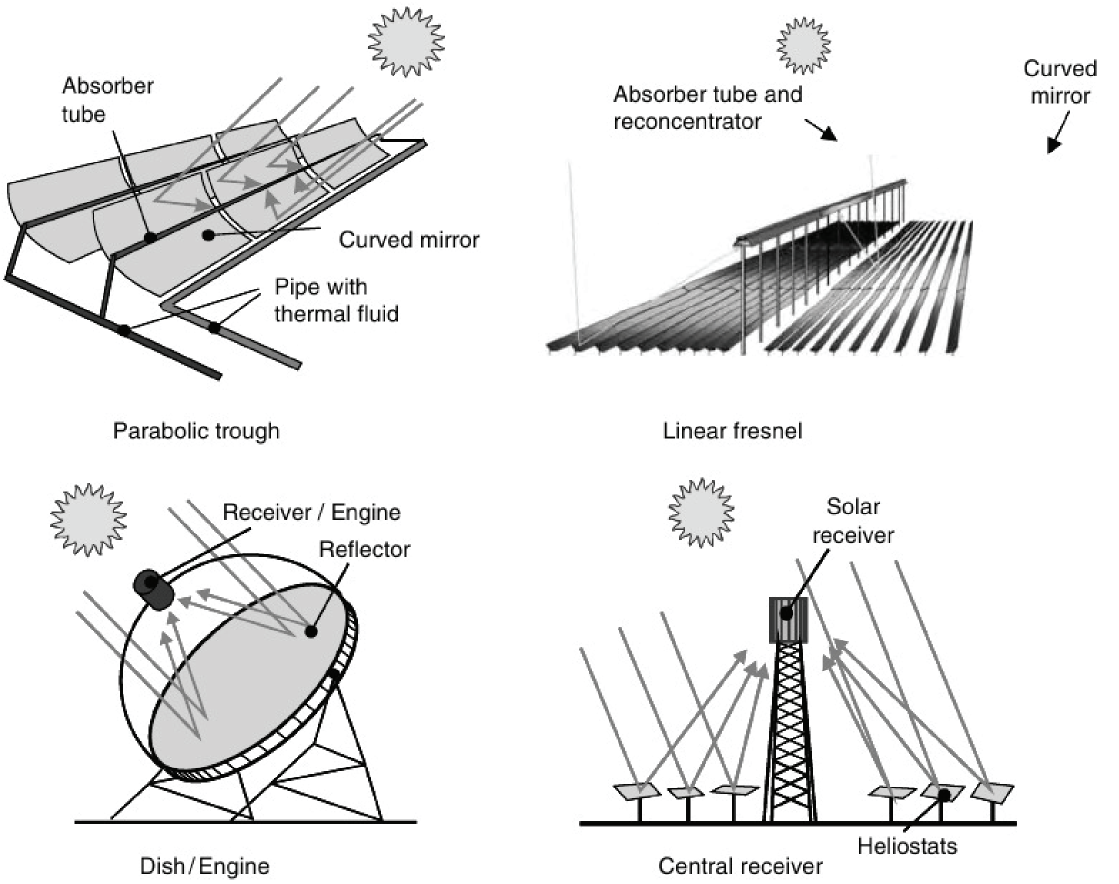


Figure 3.1: Types of concentrating solar thermal power plants (Romero-Alvarez and Zarza, 2007)

concentrated on the absorber – only the Sun's rays perpendicular to the reflecting surface can be concentrated, so only DSR can be used. Mechanical devices are therefore required to track the Sun.

PTCs and LFs concentrate the solar radiation flux by 30 to 80 times, heating a thermal fluid up to 393 °C, which is suitable for a Rankine steam turbine/generator cycle (Zarza, 2000). They concentrate incoming radiation onto a focal line using a one-axis tracking reflector (2D concentrators).

In CRSs, incoming solar radiation is concentrated by two-axis tracking heliostats which form a large discretized paraboloid – this is 3D concentrating technology with concentration factors ranging from 200 to 1000 (Romero-Alvarez and Zarza, 2007). They are suitable for advanced thermodynamic cycles. Saturated and superheated steam, molten salts, atmospheric or pressurized air are the most common thermal fluids for these kinds of concentrating solar systems.

DEs are autonomous, small and modular systems which produce electricity by means of a Stirling engine or Brayton mini-turbine located at the focal point, so we class these as 3D concentrators. Their concentration factors range between 1000 and 4000 (Romero-Alvarez and Zarza, 2007).

3.4 Parabolic-Trough Collectors

PTCs are by far the most mature forms of concentrating solar technology, as demonstrated commercially (Fernández-García et al., 2010). PTCs are suitable for working in the mid-temperature interval, 150 ℃ - 400 ℃ (Price et al., 2002).

PTCs are linear (2D) solar concentrators which convert DSR into thermal energy and then heat a HTF. Fig. 3.2 illustrates the parabolic-trough working principle and its main components.

There are two PTCs designed specially for large concentrating solar thermal power plants: the LS-3 and EuroTrough (ET-100) PTCs (Fernández-García et al., 2010).

3.4.1 Applications

The potential market for PTCs can be seen clearly from energy consumption data. Considering Spain, 50% of the total energy is consumed by industry. From that 50%, the mid-temperature range accounts for about 35% (Romero-Alvarez and Zarza, 2007), where PTCs are suitable.

PTC applications include: electricity production, IPH by means of Heat Exchangers (HEs)(unfired boilers), Flash Boilers (FBs) or DSG, domestic hot water and space heating, air-conditioning, refrigeration, irrigation water pumping , desalination and solar chemistry: detoxification. For further details consult Fernández-García et al. (2010).

3.4.2 Main Components

The main components of a PTC are as follows:

- *Parabolic-Trough CoNcentrator (PTCN).* A parabola-shaped mirror which concentrates solar radiation onto an absorber tube.

- *Absorber tube.* Located in the parabola's focal line through which the HTF is pumped, thus acquiring thermal energy.

- *Heat transfer fluid.* The element which stores and transports thermal energy obtained from concentrated solar radiation.

- *Solar tracking system.* Optical concentration requires the PTC to follow the apparent daily movement of the Sun by means of rotating along its tracking axis.

- *Metallic structure and interconnections.* Hold and allow the concentrator to track the Sun's trajectory.

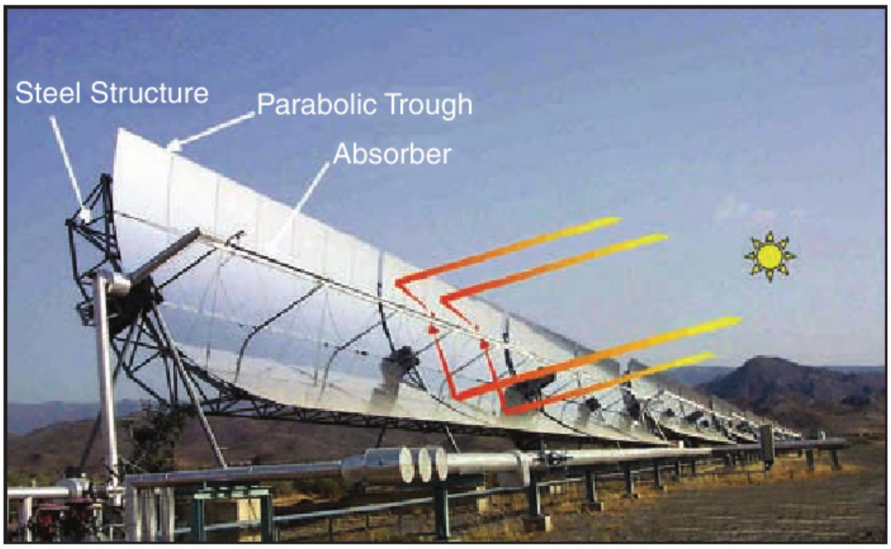


Figure 3.2: PTC main components and working principles (Zarza, 2000)

3.4.2.1 Parabolic-Trough Concentrator (PTCN)

PTCNs concentrate solar radiation onto an absorber tube, and they have a high specular reflectance in order to reflect as much radiation as possible. Different materials can be used for PTCNs, but most commonly back-silvered glass is used because it is more durable than other options such as polished aluminium and metalized acrylic. Furthermore, reflectance is higher for silvered glass PTCNs (93%) than for polished aluminium PTCNs (87%) (Romero-Alvarez and Zarza, 2007). Commonly, silvered glass PTCNs use low-iron glass to improve transmissivity, which can be also used in the glass cover for the absorber tube.

3.4.2.2 Absorber Tube

The PTC absorber tube is commonly composed of an inner steel absorber pipe and an outer glass cover to minimize heat loss caused by convection. The steel absorber pipe has a selective coating with high absorptivity and low emissivity. The selective coating also reduces radiative heat losses. An anti-reflexivity treatment is applied to the glass cover in order to maximize its transmissivity. Between both elements, the steel absorber pipe and the glass cover, there is a vacuum in order to minimize thermal losses caused by conduction and convection. However, when the working temperature is lower than 250 ℃, an absorber tube with no vacuum can be used because thermal losses are not so relevant in these cases (Romero-Alvarez and Zarza, 2007). The Rayleigh number is used to calculate the optimum space between the absorber tube and the glass cover in order to minimize convective heat loss in no-vacuum absorber tubes (Ratzel and Simpson, 1979).

Certain constraints in the manufacturing process limit the maximum length of a single tube to up to 6 m. The maximum total length for single tubes welded together in a series is approximately of 150 m (Romero-Alvarez and Zarza, 2007). Flexible metal differential expansion joints or bellows are used to attach the glass cover to the steel absorber pipe in order to compensate for the different thermal expansion rates of glass and steel (cf. Fig. 3.3). Getters are placed over the steel absorber pipe in order to maintain the vacuum, thus absorbing gas molecules from the HTF that flows through the steel absorber pipe wall (Zarza, 2000). There are two major manufacturers of PTC vacuum absorber tubes, namely Schott and Solel (Fernández-García et al., 2010).

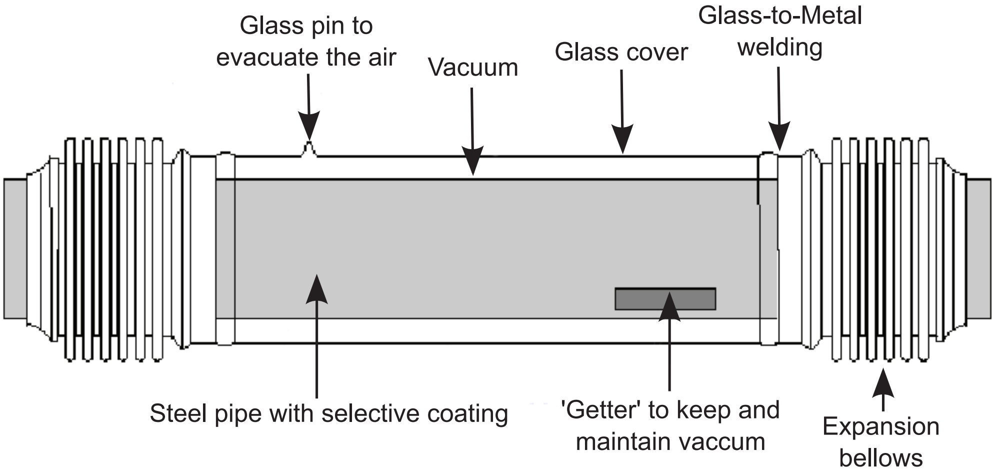

Glass pin to evacuate the air
Vacuum
Glass cover
Glass-to-Metal welding
Steel pipe with selective coating
'Getter' to keep and maintain vaccum
Expansion bellows

Figure 3.3: Absorber tube (Zarza, 2000)

3.4.2.3 Heat Transfer Fluid

The HTF is an element which stores and transports thermal energy obtained from concentrated solar radiation.

The most common HTFs in parabolic-trough solar thermal power plants for temperatures between 200 °C and 400 °C are thermal oils (Zarza, 2000). The most important factors for choosing a thermal oil are economic and the maximum oil temperature, because above certain temperatures oil cracking and rapid degradation occur. VP1, which is the most widely used oil for temperatures up to 395 °C (Romero-Alvarez and Zarza, 2007), is an eutectic mixture of 73.5% diphenyl oxide and 26.5% diphenyl. Its solidification temperature is 12 °C, so an auxiliary heating system is required. It must be pressurized by an inert gas (nitrogen, argon, etc.) because its boiling temperature is 257 °C at atmospheric pressure (Eastman, 2013). For lower temperatures, there are other thermal oils, e.g. Therminol 55 and 59 up to 290 °C and 315 °C, respectively (Eastman, 2013). For slightly higher temperatures, there are other options, but they are too expensive for solar thermal power plants.

When the temperature is lower than 200 °C there is no problem in using pressurized liquid water as a HTF (Romero-Alvarez and Zarza, 2007). A mixture of water and ethylene glycol can be also used. Using water at higher temperatures implies that the absorber tube is subjected to high pressure; this requires stronger joints and pipes, increasing considerably the price of the entire system. Nevertheless, this very promising technology, which is called DSG, increases overall system efficiency while reducing investment costs, as explained in § 3.5.

3.4.2.4 Solar Tracking System

PTCs concentrate incoming radiation onto a focal line using a one-axis tracking. Drive units are required in order to follow the Sun's apparent movement. For small PTCs, the drive unit is composed of an electric motor and a gearbox, whereas for large PTCs, powerful hydraulic drive units are required. The drive unit is commonly placed in the central pylon (cf. Fig. 3.4) and is controlled by a local control unit. There are two categories of local control units (Romero-Alvarez and Zarza, 2007).

- *Sun sensors* detect the Sun's position using photocells such as shadow bands and flux line trackers. Shadow bands have two photo-sensors and a separating shadow wall. When the PTC is correctly tracking the shadow wall, both photo-sensors are equally shaded and their electric output signals are the same; otherwise, a voltage difference indicates that the PTC has to be moved. Flux line trackers are located in the absorber tube, and two sensors are placed on both sides of the absorber tube to detect the concentrated flux. When both sensors measure the same value, the PTC is tracking correctly; if not, the PTC should be moved.

- *Solar position algorithms* are mathematical algorithms used to accurately determine the Sun's elevation and azimuth as functions of local time and geographic position. The angular position of the PTC's rotation axis is computed by considering the plant's orientation and inclination. Electronic devices are used to measure and establish angular position, such as angular encoders or magnetic coded tapes.

The rotation axes of PTCs are commonly orientated in either north-south or east-west orientations. North-south orientated energy production is highly affected by the seasons, being three or even four times greater in summer than in winter, depending on the latitude. However, north-south orientation provides more energy than east-west orientation. Changes in energy production due to the season of the year are lower in the east-west orientation. On the other hand, Incidence Angle (θ) (IA) variations are greater with this orientation (Zarza, 2000).

There have been attempts to design a two-axis Sun-tracking system for PTCs, e.g. the Helioman 3/32 PTC (Fernández-García et al., 2010). Its evaluation concluded that they are less cost-effective. Even though a two-axis tracking system allows an IA equal to 0°[a], the passive piping length and thermal losses are higher than in one-axis tracking systems. Moreover, they require more complex mechanical systems and more maintenance.

3.4.2.5 Metallic Structure and Interconnections

The steel structure must support mechanical rigidity to torsion, for which there are two different approaches – one based on a steel torque box of trusses and beams, and another one based on two V-trusses held together by end-plates (cf. Fig. 3.4) (Romero-Alvarez and Zarza, 2007).

The interconnection between PTCs in a series is performed by flexible joints, in order to allow for the independent rotation of the PTCs when tracking the Sun's trajectory. The linear thermal expansion of the absorber tubes must be also taken into consideration. There are two main flexible joints: flexible hoses and ball joints (Zarza, 2000).

There are additionally two kinds of flexible hoses, for temperatures lower and higher than 300 °C (Romero-Alvarez and Zarza, 2007). For high temperatures, the flexible hose is not so flexible, and due to its high friction coefficient it causes significant pressure drops. Flexible hoses are also likely to suffer fatigue failures.

The main advantages of ball joints are: their low pressure drops (equivalent to a 90° elbow), their two degrees of freedom and that they are provided with an inner graphite seal which reduces friction and avoids leaks. Ball joints also have lower maintenance costs and are more reliable than flexible hoses.

3.4.3 Design Parameters

The most important design parameters of PTCs are: the concentration ratio (C), the acceptance angle (β) and the rim angle (ϕ), cf. Fig. 3.5.

3.4.3.1 Concentration Ratio

The ratio between the PTC's aperture and the absorber tube area is the concentration ratio (C) (cf. Fig. 3.5). The concentration ratio is defined by Eq. 3.1 (Zarza, 2000), where l_{ap} is the PTC aperture, l the PTC length and d_o the absorber tube's outer diameter. A typical concentration ratio value is 20, although the theoretical maximum is 70 (Fernández-García et al., 2010).

$$C = \frac{l_{ap} l}{\pi d_o l} = \frac{l_{ap}}{\pi d_o}. \tag{3.1}$$

[a] There is always an error related to the minimum angular velocity of the motor

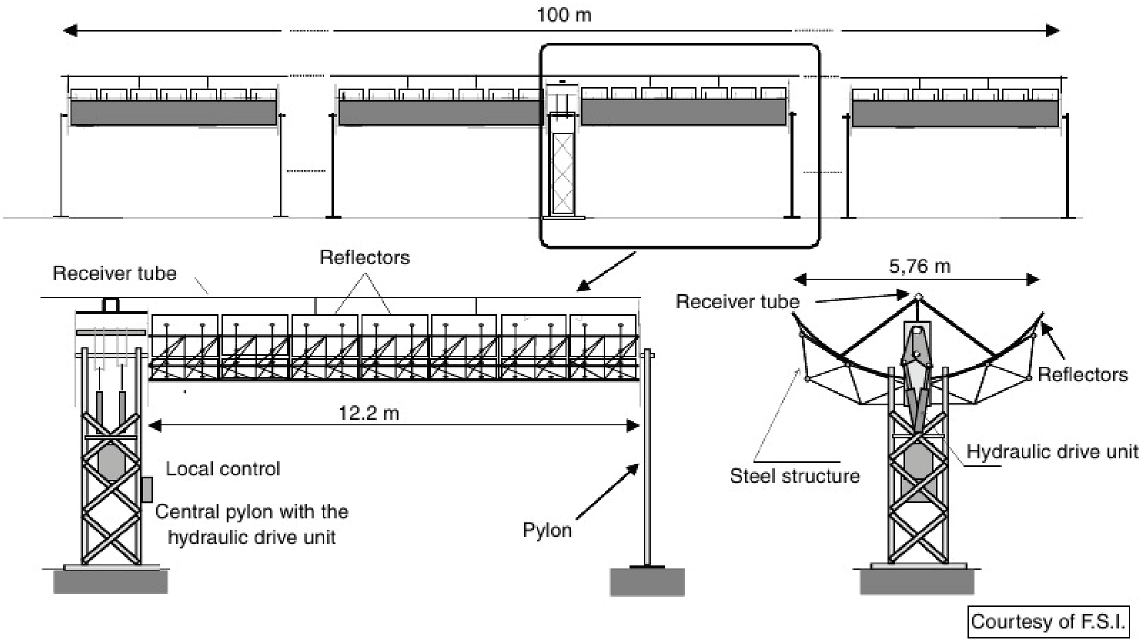


Figure 3.4: LS-3 PTC structure (Romero-Alvarez and Zarza, 2007)

3.4.3.2 Acceptance Angle

The acceptance angle (β) is the maximum angle formed by two of the Sun's rays in a plane transversal to the PTC in order to intercept both rays by the absorber tube after being reflected in the PTCN. A small acceptance angle implies a more precise and expensive tracking system. Small acceptance angles are associated with high concentration ratios.

From the Earth, the Sun is seen as a sphere with an average solid angle of 0.53°; therefore, this is the minimum acceptance angle for a PTC without losing a proportion of DSR. Common acceptance angles are between 1° and 2°, while higher values lead to small concentration ratios (Zarza, 2000).

3.4.3.3 Rim angle

The rim angle (ϕ) is the angle formed by half the PTC length (cf. Fig. 3.5). This value can be calculated by Eq. 3.2 (Zarza, 2000). Typical rim angle values are between 70° and 110° (Romero-Alvarez and Zarza, 2007). Lower angles reduce the aperture's surface, whereas higher angles increase the PTC's surface but do not increase aperture width.

$$\tan\frac{\phi}{2} = \frac{l_{ap}}{4f}. \tag{3.2}$$

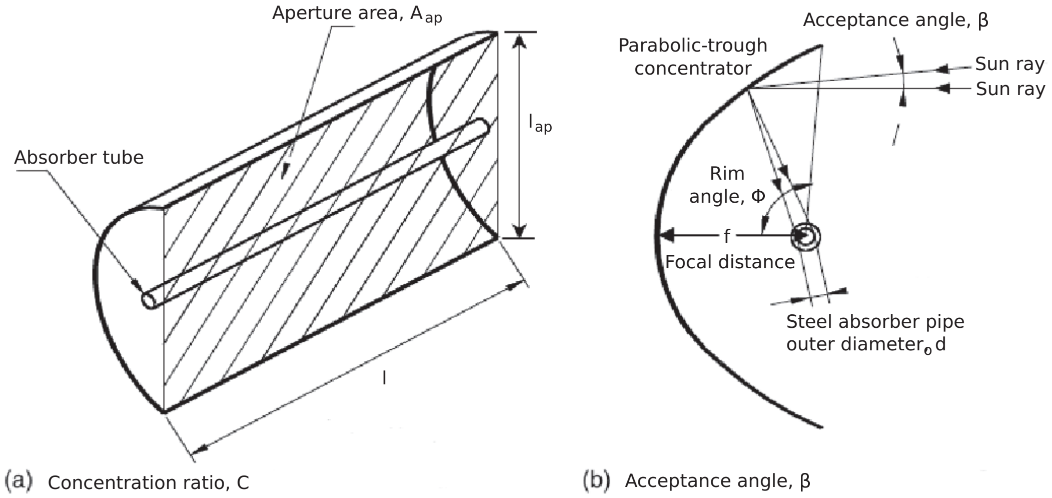


Figure 3.5: Design parameters of a PTC (Romero-Alvarez and Zarza, 2007)

3.4.4 Energy Losses and Efficiency

Once all the elements of a PTC have been presented and its design parameters introduced, the next concern is its efficiency or, in other terms, energy losses. Energy losses in a PTC can be classified as thermal, optical or geometrical losses in order of importance.

3.4.4.1 Optical Losses

Fig. 3.6 shows the parameters of a PTC which influence optical losses; losses in the concentrator: reflectivity (r) and the intercept factor (γ_i), and in the absorber tube: transmissivity (τ) and absorptivity (α_a).

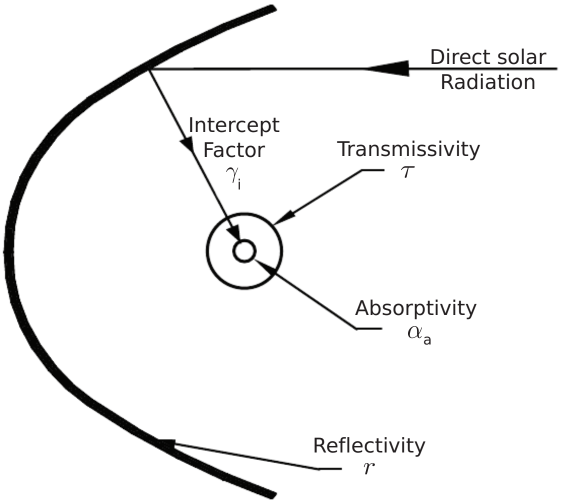


Figure 3.6: Optical losses in a PTC (Zarza, 2000)

Parabolic-Trough Concentrator Optical Losses. The parameters to take into account for optical losses in a PTCN are as follows.

- *Reflectivity (r)* is the proportion of incident radiation reflected by the PTCN.

- *Intercept factor (γ_i)* is the fraction of reflected radiation which reaches the absorber tube.

Absorber Tube Optical Losses. In the absorber tube, the following parameters also affect optical losses, cf. Fig. 3.6.

- *Transmissivity (τ)* is the proportion of incident radiation that passes through the glass cover.

- *Absorptivity (α_a)* is the proportion of radiation absorbed by the steel absorber pipe.

Multiplying all the previous optical losses yields the peak optical efficiency $(\eta_{opt,0^\circ})$, Eq. 3.3 which considers an 0° IA. As will be shown in § 3.4.4.3, there are also optical losses associated with the IA.

$$\eta_{opt,0^\circ} = r \cdot \gamma_i \cdot \tau \cdot \alpha_a. \tag{3.3}$$

3.4.4.2 Thermal Losses

Thermal losses in a PTCN can be neglected. They mainly appear in the glass cover and in the steel absorber pipe, but they are much more important in the latter (Zarza, 2000). The following list summarizes thermal losses in a PTC (cf. Fig. 3.7).

- *Steel absorber pipe to ambient,* radiative thermal losses.

- *Steel absorber pipe to glass cover,* convective and conductive thermal losses.

- *Glass cover to ambient,* radiative, convective and conductive thermal losses.

Overall PTC thermal losses to ambient $(\dot{Q}_{ptc\to amb})$ can be simplified and calculated by Eq. 3.4 (Zarza, 2000), using an absorber tube loss coefficient $(U_{loss,abr})$ which considers all the previously mentioned thermal losses, where T_{abr} and T_{amb} are the absorber tube and ambient temperatures, respectively. Alternatively, the PTCN loss coefficient $(U_{loss,ptcn})$ can be used. The relation

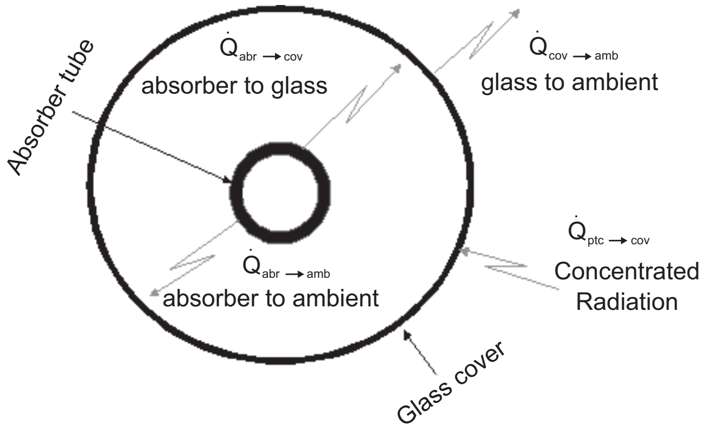


Figure 3.7: Thermal losses in the absorber tube (Zarza, 2000)

between both loss coefficients is given by Eq. 3.5 (Zarza, 2000), where C can be calculated by Eq. 3.1.

$$\dot{Q}_{ptc \to amb} = U_{loss,abr} \pi d_o l (T_{abr} - T_{amb}), \tag{3.4}$$

$$U_{loss,ptcn} = \frac{U_{loss,abr}}{C}. \tag{3.5}$$

The thermal loss coefficient is usually expressed as a second-order polynomial expression, as shown in Eq. 3.6 (Romero-Alvarez and Zarza, 2007). The coefficients a_1, a_2 and a_3 are calculated by experimentation, performing several tests where the PTC is operated at different temperatures. However, it is difficult to fit experimental data considering the second-order polynomial with unique values for the coefficients, in which case several coefficient values can be used in different temperature ranges.

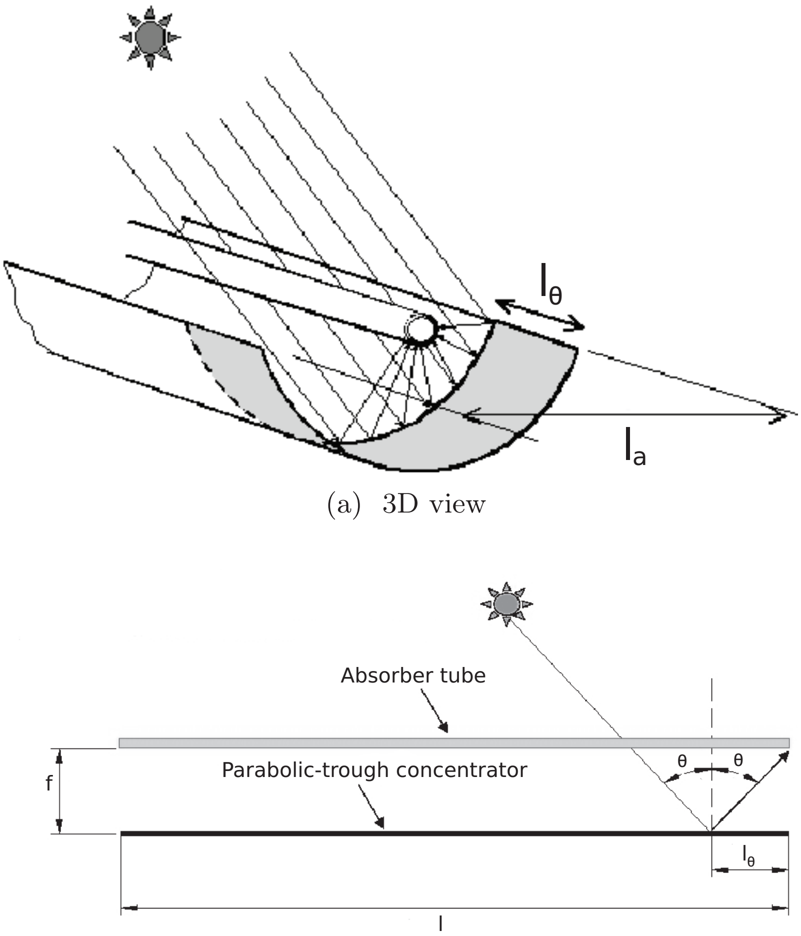
$$U_{loss,abr} = a_1 + a_2(T_{abr} - T_{amb}) + a_3(T_{abr} - T_{amb})^2. \tag{3.6}$$

3.4.4.3 Geometrical Losses

Since PTCs can only track the Sun's trajectory in one axis, there is also a geometrical loss associated with the IA (θ). The IA is defined as the angle between the solar vector and the vector perpendicular to the PTC aperture. Fig. 3.8 shows the loss of reflexive area in a PTC which depends on the IA. Eq. 3.7 (Zarza, 2000) estimates the end-loss area of a PTC (A_θ), which reduces the PTC's aperture. The PTC end loss area is calculated as a function of the IA, the mean distance between the concentrator and the absorber tube ($\bar{f}$), calculated by Eq. 3.8, the PTC width (l_{ap}) and the focal length of the parabola (f) (Zarza, 2000).

$$A_\theta = l_{ap} \cdot l_\theta = l_{ap} \cdot \bar{f} \cdot tan(\theta), \tag{3.7}$$

$$\bar{f} = f + \frac{f \cdot l_{ap}^2}{48 \cdot f^2}. \tag{3.8}$$



(a) 3D view

(b) Aerial view

Figure 3.8: PTCN geometrical end losses (Zarza, 2000)

The IA also affects the PTC's optical parameters (reflectivity, absorptivity, intercept factor and transmissivity) because they are not isotropic (Romero-Alvarez and Zarza, 2007). Therefore, in order to quantify this effect on useful aperture area and on optical efficiency, an Incidence Angle Modifier $K(\theta)$ (IAM) is considered. The IAM, which estimates these optical and geometrical losses, is usually given by a polynomial expression and depends on the type of PTC, its position, its orientation and the Sun's position (Romero-Alvarez and Zarza, 2007). The geometrical efficiency of a PTC is then given by Eq. 3.9 (also including some optical losses). Particular expressions for $K(\theta)$ are given in § 3.7.3.

$$\eta_{geo} = K(\theta). \tag{3.9}$$

If there are multiple rows in the plant, shadows between adjacent PTCs rows must be studied and considered (Stuetzle, 2002). Additionally, shading caused by the absorber tube in the PTCN and by the absorber tube arms and bellows (Zarza and Sánchez, 2002; Stuetzle, 2002) might be also considered.

3.4.4.4 Efficiency

Once heat losses in a PTC have been estimated, the expression for PTC efficiency is given by Eq. 3.10. Where $\eta_{opt,0^\circ}$ is given by Eq. 3.3 and η_{geo} by Eq. 3.9. The F_e term in Eq. 3.10 is the soiling factor, which considers the progressive soiling of mirrors and glass tubes after washing. Soiling in a PTC means that reflectivity and transmissivity are lower than the nominal values considered, so the optical efficiency is lowered. Common values for F_e are around 0.97 (Zarza, 2000).

$$\eta_{ptc} = \eta_{opt,0^\circ} \cdot \eta_{the} \cdot \eta_{geo} \cdot F_e. \tag{3.10}$$

PTC thermal losses are not usually computed as a coefficient (η_{the}); rather, they are classed as a loss coefficient, as shown previously in § 3.4.4.2. However, now we know the remaining terms, PTC thermal efficiency can be expressed as Eq. 3.11.

$$\eta_{the} = 1 - \frac{\dot{Q}_{ptc\rightarrow amb}}{\dot{Q}_{sun\rightarrow ptc}\eta_{opt,0^\circ}\eta_{geo}F_e} \tag{3.11}$$

The term $\dot{Q}_{sun\rightarrow ptc}$ is the Sun's heat flux incident on the PTC's aperture, and it is given by Eq. 3.12 (Zarza, 2000), where A_{ap} is the PTC aperture area, G_b is Direct Solar Irradiance (DSI) and θ is IA. The term $G_b \cdot \cos(\theta)$ is known as the Direct Normal Irradiance (DNI).

$$\dot{Q}_{sun\rightarrow ptc} = A_{ap} \cdot G_b \cdot \cos(\theta)\,. \tag{3.12}$$

3.4.5 Energy Balance

Considering global PTC efficiency (η_{ptc}), the Sun's heat flux delivered to the absorber tube ($\dot{Q}_{ptc\rightarrow abr}$) is calculated by Eq. 3.13.

$$\dot{Q}_{ptc\rightarrow abr} = \dot{Q}_{sun\rightarrow ptc} \cdot \eta_{ptc}. \tag{3.13}$$

Substituting Eq. 3.10 in Eq. 3.13 yields Eq. 3.14.

$$\dot{Q}_{ptc\rightarrow abr} = \dot{Q}_{sun\rightarrow ptc} \cdot \eta_{opt,0^\circ} \cdot \eta_{the} \cdot \eta_{geo} \cdot F_e. \tag{3.14}$$

Substituting Eqs. 3.3, 3.11 and 3.9 ($\eta_{opt,0^\circ}$,η_{the} and η_{geo} terms) in Eq. 3.14 yields Eq. 3.15.

$$\dot{Q}_{ptc\rightarrow abr} = \dot{Q}_{sun\rightarrow ptc} \cdot r \cdot \gamma_i \cdot \tau \cdot \alpha_a \cdot K(\theta) \cdot F_e - \dot{Q}_{ptc\rightarrow amb}. \tag{3.15}$$

Finally, substituting Eqs. 3.4 and 3.12 ($\dot{Q}_{ptc\rightarrow abr}$ and $\dot{Q}_{sun\rightarrow ptc}$ terms) in Eq. 3.15 yields Eq. 3.16.

$$\begin{aligned}\dot{Q}_{ptc\rightarrow abr} = {} & A_{ap} \cdot G_b \cdot \cos(\theta) \cdot r \cdot \gamma_i \cdot \tau \cdot \alpha_a \cdot K(\theta) \cdot F_e \\ & - U_{loss,abr} \pi d_o l (T_{abr} - T_{amb}).\end{aligned} \tag{3.16}$$

3.5 Direct Steam Generation

Commonly, PTC solar thermal power plants use thermal oils as HTFs. This technology has been improved continuously since its first commercial implementation. However, this design can be optimized by changing the HTF, which results in directly producing superheated steam in the PTC field. This technology, known as DSG, increases overall system efficiency while reducing investment costs. Tab. 3.1 shows the main advantages and disadvantages of this kind of technology.

DSG presents some important challenges due to phase changes that affect the HTF. The HTF circulates as a sub-cooled liquid, superheated steam or a two-phase flow water-steam mixture (vapor and liquid simultaneously in the same volume) through the PTC field. The existence of a two-phase flow involves some uncertainties, e.g. stability and controllability of the process and the gradient temperatures in the pipes. All of these practical aspects have been studied in detail in Zarza (2000) and Valenzuela et al. (2004).

More details about two-phase flows are given in Chap. 4, but in order to introduce the problem, a quick glimpse is given in this section. Typical two-phase flow patterns in horizontal pipes are: bubbly, interment, stratified and annular (Romero-Alvarez and Zarza, 2007). In bubbly, intermittent and annular flows, the pipe wall is well-wetted; therefore, a high temperature gradient between the top and the bottom of the pipe is avoided when the pipe is heated from one side. However, when the two-phase flow is in the stratified region (liquid in the bottom and steam in the top) and the pipe is heated from one side (cf. Fig. 3.9), there is a steep temperature gradi-

Advantages	Disadvantages
Elimination of thermal oil	High-pressure valves and pipes
No danger of fire	High pressure in joint elements
Suppression of contaminants	High temperature in joint elements
Simplification of the plant	Temperature gradients in pipes
Improvements in performance	Process stability
Reduction in plant size	More complex control systems
Reduction in parasitic load	Development of storage systems

Table 3.1: Advantages and disadvantages of DSG

ent between the top and bottom which causes thermal stress and bending that may break the pipe. However, this water stratification problem can be avoided, as experimental results proved in Zarza (2000). Considering the single phase, temperature gradients cannot appear because the flow fills the pipe completely, thus making it well-wetted (Zarza, 2000).

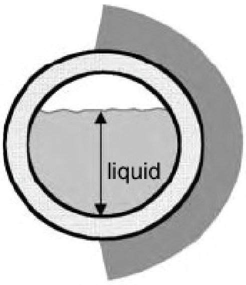


Figure 3.9: Two-phase stratification with the pipe heated from the right side (Romero-Alvarez and Zarza, 2007)

The DISS project (Zarza, 2000) demonstrated the feasibility of the DSG process in PTC solar thermal power plants. Temperature gradients in the pipe could be kept to safe limits for a wide range of mass fluxes, while inclining the absorber tubes reduced the stratified region. However, the performed tests showed that inclination is not required to guarantee safe limits with respect to pipe thermal stress (Zarza et al., 2004).

3.5.1 Operating Modes

DSG PTC solar thermal power plants can work in three different basic operating modes, namely: once-through, recirculation and injection modes (Zarza, 2000). Fig. 3.10 shows the different DSG operating modes, together with their advantages and disadvantages.

- *Once-through mode.* In this mode feed-water is preheated, evaporated, and converted into superheated steam as it circulates from the inlet to the outlet of the PTC field. A water injector is placed in front of the last PTC to control the outlet steam temperature. This mode is the simplest and its main issue is the controllability of the outlet superheated steam.

- *Injection mode.* In addition to the once-through mode, water is injected at several points along the PTC row. The main drawback is its complexity and high operating costs (Zarza et al., 2002).

- *Recirculation mode.* In this mode a water-steam separator is placed at the end of the evaporation section of the PTC row. In this separator, excess water is recirculated to the field inlet and mixed with pre-heated

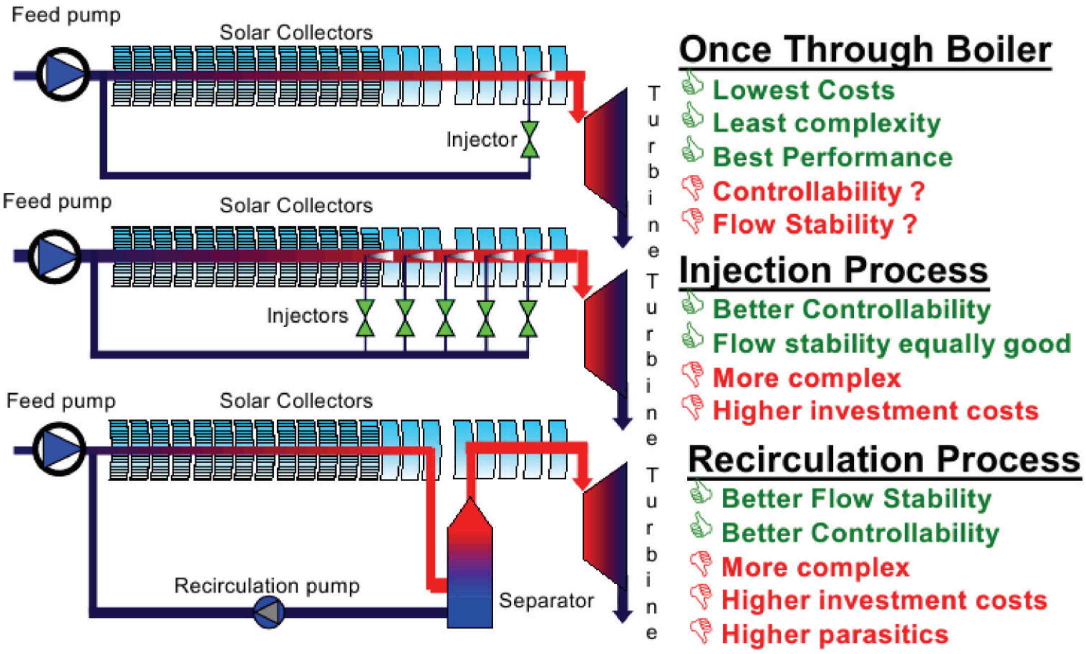


Figure 3.10: The three basic DSG processes (Zarza et al., 2002)

water. This guarantees good wetting of the absorber tube. In the separator, the remaining steam is used to feed the superheating section. This operating scheme is highly controllable (Valenzuela et al., 2006) but also increases parasitic load due to the water-steam separator. The recirculation mode can have central or decentral field separators (Zarza et al., 2006).

In the DISS project, several control schemes were developed and tested for the once-through and the recirculation modes (Valenzuela et al., 2005, 2006). The number of tests performed in the injection mode was small due to problems with special devices. However, enough information was available to evaluate and compare the operating modes. From the experimental results, the recirculation mode was shown as the most financially and technically acceptable option for the application of DSG PTC technology to commercial solar thermal power plants (Zarza et al., 2004).

3.6 DISS Test Facility

The first DSG parabolic-trough solar thermal power plant was designed, built and set up at PSA between 1996 and 1998. This was the first phase of the DISS project (Zarza, 2000) and it was devoted to investigating the feasibility of DSG in a PTCs under real solar conditions.

The DISS test facility is a parabolic-trough solar thermal power plant owned by CIEMAT - PSA (cf. Fig. 3.11).

The aim of the DISS facility is to develop a new generation of solar power plants using PTCs to produce high pressure steam in absorber tubes, thus eliminating the oil commonly used as a heat transfer medium between the solar field and the conventional power block.

The DISS facility is appropriate, not only for the study and development of control schemes, but also for the study and optimization of the operating procedures or new elements that must be implemented in DSG solar fields. Tab. 3.2 summarizes the most relevant features of the DISS test facility.

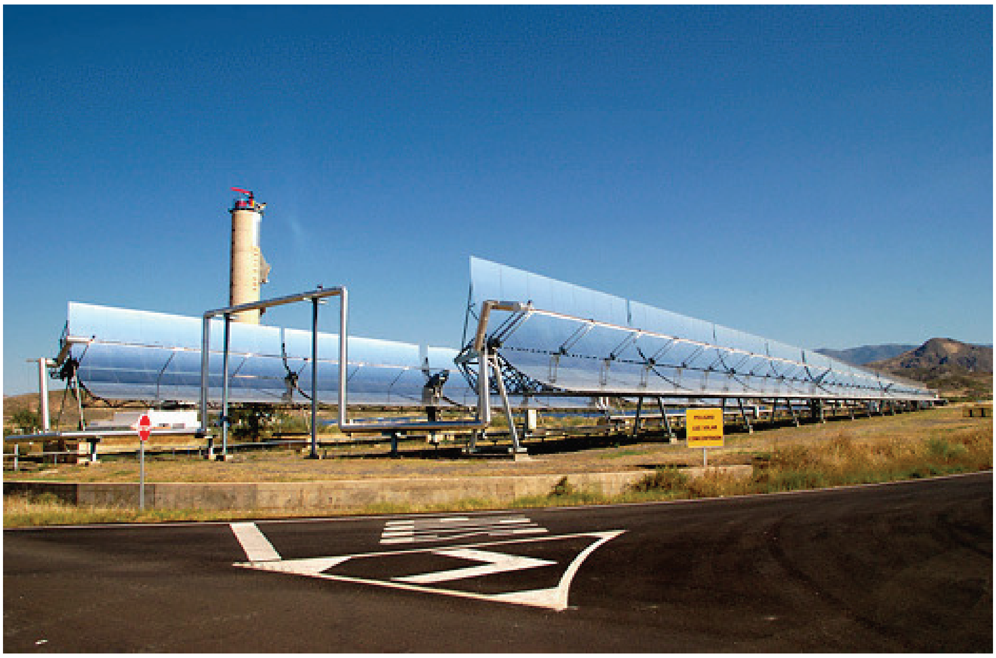


Figure 3.11: General view of the DISS test facility

The DISS test facility accumulated more than 3500 h of operation from 1999 to 2001 (Zarza and Sánchez, 2002). The resulting experimental data was used to evaluate and compare the three DSG operating modes: recirculation, injection and once-through.

The DISS test facility initially had 11 modified LS-3 PTCs connected in series, with 2700 m^2 of PTC aperture and a length of 550 m. The DISS facility operates in a closed loop, where superheated steam is delivered to the power block (cf. Fig. 3.12), which then condensates it accordingly. The resulting liquid water is used to feed the loop again. The water-steam separator (cf. Fig. 3.12) connects the end of the evaporator section to the beginning of the steam superheating section when the plant is operating in the recirculation mode.

Two additional PTCs were built at the beginning of the field in 2003, increasing its length and its aperture area to 750 m and 3822 m^2, respectively

Module width	5.76 m	
PTC width	11.98 m	
PTC length	25 m (2 modules)	
	50 m (4 modules)	
Number of PTCs	11	
Inclination	0° (all PTCs)	
	2°, 4° (1^{st} - 9^{th} PTCs)	
	6°, 8° (9^{th} PTC)	
Orientation	North-South	
Abr. pipe inner diameter	50 mm	
Abr. pipe outer diameter	70 mm	
Max. Outl. pressure	10 MPa	
Max. Outl. mass flow rate	1 kg/s	
Max. Outl. temperature	673 K	
Max. pressure injectors	1.4 MPa	
Nominal operating modes	Inl.: 4 MPa/210 °C	Outl.: 3 MPa/300 °C
	Inl.: 6.8 MPa/270 °C	Outl.: 6 MPa/350 °C
	Inl.: 10.8 MPa/300 °C	Outl.: 10 MPa/375 °C

Table 3.2: Main features of the DISS solar thermal power plant (no preheating PTCs)

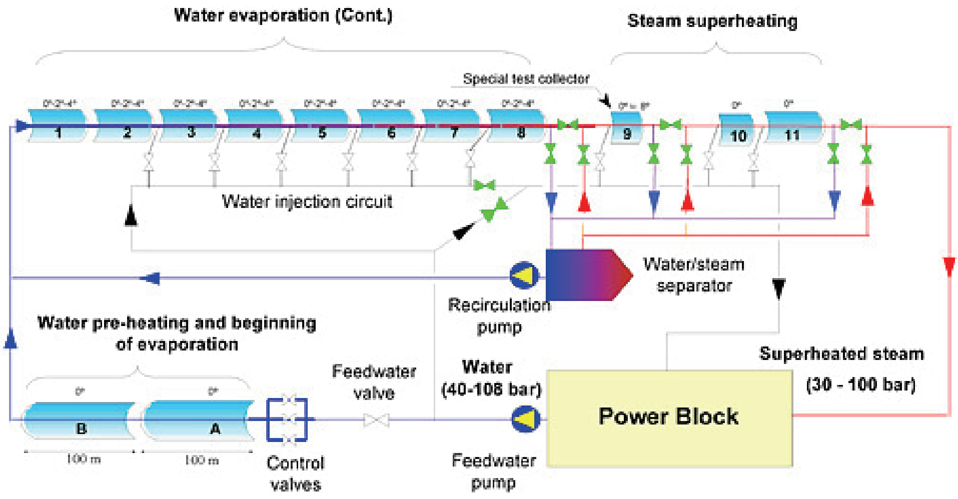


Figure 3.12: Diagram of the DISS loop (Zarza, 2000)

(in the framework of the INtegration of the DIrect steam generation Technology for Electricity Production (INDITEP) project (Zarza et al., 2006)).

In 2007, the DISS facility was connected to the prototype heat storage system designed and developed by the DIrect STORage (DISTOR) project (Steinmann and Tamme, 2008). In the framework of the DUrchlaufKonzept

- Entwicklung und erprobung (DUKE) project (2011-2014) (Feldhoff, 2012), the DISS facility is going to be updated. Some of the more relevant changes are: PTC row extended to 1000 m, new injection instrumentation, ball joints, absorber tubes, superheated injectors, relocation of inlet valves, temperature sensors along the end of the evaporation and around the cross sections, and a new power block injector.

3.6.1 DISS Test Facility Instrumentation

The DISS test facility is well instrumented with several kinds of sensors. Fig. 3.14 shows a summary of the instrumentation layout for the initial 11 PTCs, where the DISS field is duplicated for the sake of clarity. A more detailed view of the particular sensors in each PTC is given by Fig. 3.13, which shows instrumentation in the 9th PTC. Some sensors in the 9th PTC are listed in Tab. 3.3.

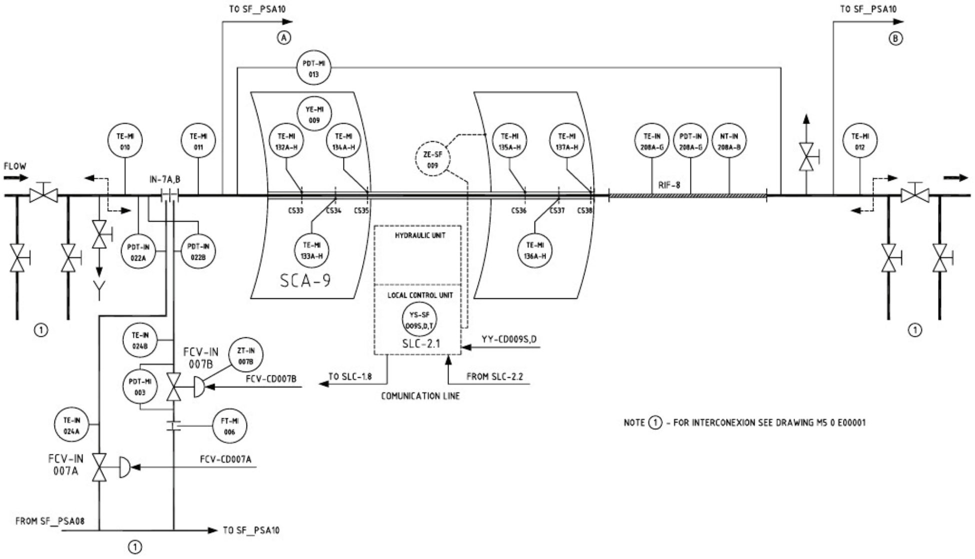


Figure 3.13: Instrumentation layout in the 9^{th} PTC (Zarza, 2000)

Sensor tag	Description
TE-MI-010/011	Inlet temperature before/after injector 7
TE-MI-012	Outlet temperature
TE-IN-024A/B	Inlet temperature, injector 7-A/7-B
PDT-IN-022A/B	Pressure difference, injector 7-A/7-B
PDT-MI-013	Pressure difference
FT-MI-006	Inlet mass flow rate, injector 7-B

Table 3.3: List of some sensors in the 9^{th} PTC

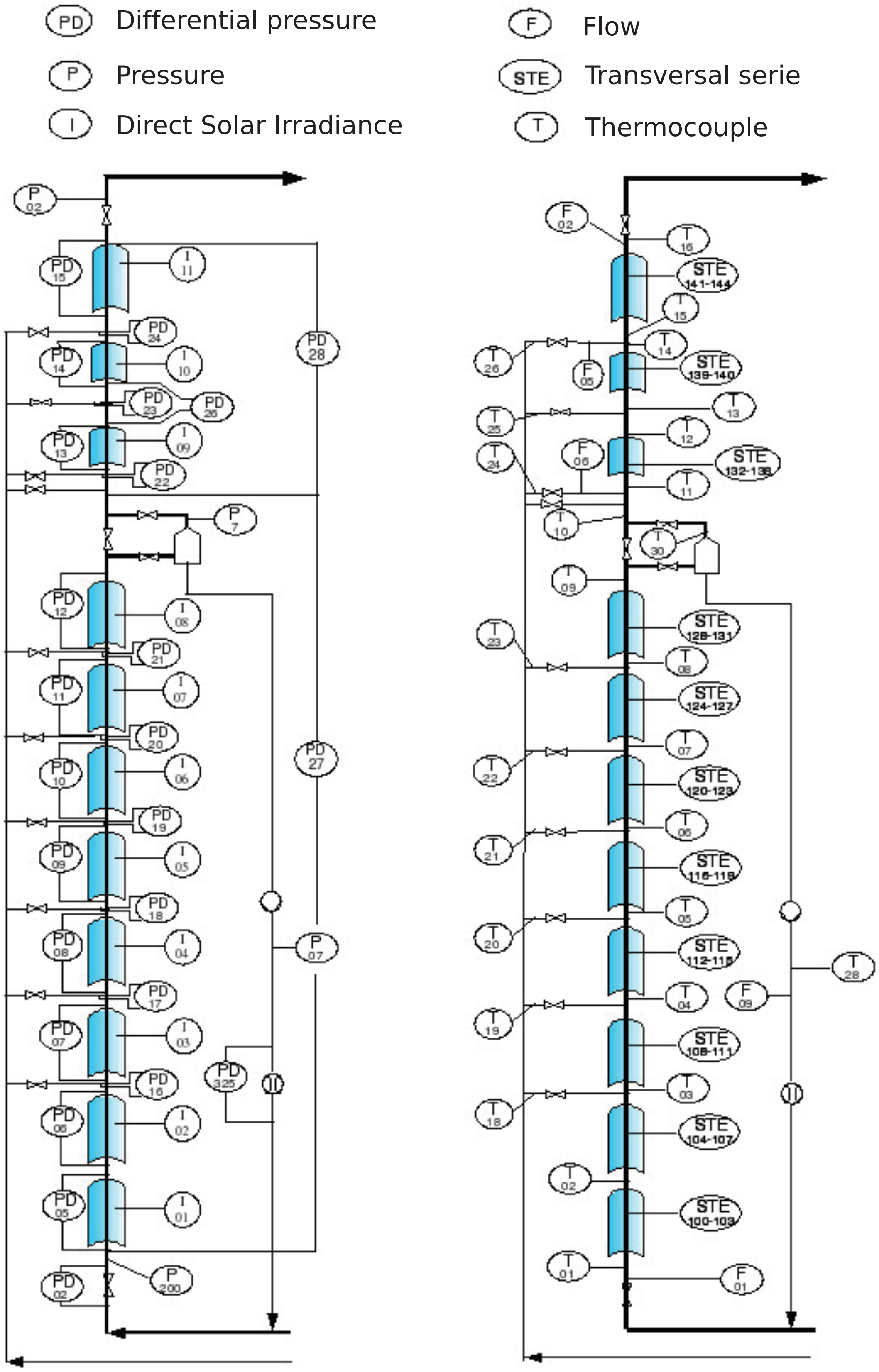


Figure 3.14: Instrumentation layout in the DISS field

3.7 DISS Parabolic-Trough Collectors

This section, which gives specific details about the modified LS-3 PTC used in the DISS test facility, describes the main components, design parameters and heat losses. Figs. 3.2 and 3.15 shows a DISS PTC with four modules.

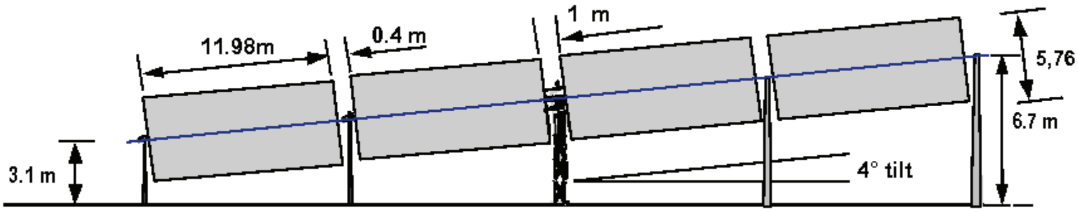


Figure 3.15: PTC with four modules (Zarza, 2000)

3.7.1 Main Components

Each of the main components previously introduced for PTCs in § 3.4.2 is particularized in the following subsections for the modified LS-3 DISS PTCs.

3.7.1.1 Parabolic-Trough Concentrator (PTCN)

DISS PTCNs are made of glass, 3.5 mm thick and coated with a thin silver film. Nine of the PTCNs are composed of four reflective parabolic trough modules (cf. Fig. 3.15), while two collectors (nos. 9 and 10) have only two modules. Each module is 11.98 m long by 5.76 m wide.

3.7.1.2 Absorber Tube

The DISS absorber tube is composed of an A335 grade P22 (10CrMo910) alloy steel pipe and a glass cover (cf. Fig. 3.3). The steel absorber pipe has a selective coating with high absorptivity ($\geq 93\%$) and low emissivity ($< 30\%$). An anti-reflexivity treatment is applied to the glass cover in order to maximize its transmissivity ($\geq 96\%$). Between both elements is a vacuum (10^{-4} Pa), which is employed to minimize thermal losses caused by conduction and convection. There are also getters, which are used to maintain this vacuum. The inner/outer steel absorber pipe diameters are 50/70 mm, while the inner/outer glass cover diameters are 115/120 mm.

3.7.1.3 Heat Transfer Fluid

The DISS facility is a DSG parabolic-trough solar thermal power plant. The HTF used in the DISS test facility is two-phase-flow water-steam fluid, which circulates in three different states: sub-cooled liquid, a water-steam mixture and superheated steam.

3.7.1.4 Solar Tracking System

The one-axis-tracking DISS PTCs orientation is north-south. The solar tracking system uses a hydraulic drive unit. The solar vector is determined by a high-precision solar position algorithm. Angular encoders were initially used to measure and establish the angular position, but they were substituted by magnetic coded tapes in 2007.

3.7.1.5 Metallic Structure and Interconnections

The DISS PTC steel structure is based on two V-trusses held together by end-plates. The flexible joints in the DISS PTCs are ball joints.

3.7.2 Design Parameters

The design parameters are summarized in Tab. 3.4 (Fernández-García et al., 2010).

Design parameter	Nominal value
Concentration ratio (C)	26.2
Acceptance angle (β)	1.37°
Rim angle (ϕ)	80°

Table 3.4: Modified LS-3 DISS PTC design parameters

3.7.3 Energy Losses

The particularized energy losses for a modified LS-3 PTC are described in this section, while optical parameters are summarized in Tab. 3.5 (Zarza, 2000).

Thermal losses in a PTC are commonly described by a second-order polynomial, as stated in § 3.4.4.2. Tab. 3.6 gives specific values for coefficients a_1, a_2 and a_3 in Eq. 3.6. These values were obtained experimentally in Ajona (1999) for a modified LS-3 PTC.

Optical parameter	Nominal value
Reflectivity (r)	93%
Intercept factor (γ_i)	90%
Transmissivity (τ)	96%
Absorptivity (α_a)	93%

Table 3.5: Modified LS-3 DISS PTC optical parameters

T_{abr} (°C)	a_1	a_2	a_3
< 200	0.687257	0.001941	0.000026
≤ 300	1.433242	-0.00566	0.000046
> 300	2.895474	-0.0164	0.000065

Table 3.6: Coefficients a_1, a_2 and a_3 for the thermal losses model of the modified LS-3 DISS PTC (Ajona, 1999)

As previously explained in § 3.4.4.3, geometrical losses (also some optical losses) are calculated by using the IAM model ($K(\theta)$). This section provides information about IAMs for the modified LS-3 PTC proposed by two authors.

The IAM model proposed by González et al. (2001), applied by Zarza (2000), is described by Eq. 3.17, where θ is the IA in degrees.

$$\begin{aligned} K(\theta) &= 1 - 2,23073 \cdot 10^{-4} \cdot \theta - 1.1 \cdot 10^{-4}\theta^2 + 3,18596 \cdot 10^{-6} \cdot \theta^3 \\ &\quad - 4,85509 \cdot 10^{-8} \cdot \theta^4 \qquad (0° \leq \theta < 80°), \\ K(\theta) &= 0 \qquad (80° \geq \theta \leq 90°). \end{aligned} \tag{3.17}$$

Another experimental IAM model was proposed by Eickhoff (2002) and applied by Eck et al. (2003). This model considers two different IAM models for the PTCs with four ($K_4(\theta)$) and two ($K_2(\theta)$) modules, described by Eq. 3.18 and Eq. 3.19, respectively, where θ is expressed in radians.

$$K_4(\theta) = 1 - 0.00188 \cdot \theta - 1.49206 \cdot 10^{-4} \cdot \theta^2 \tag{3.18}$$

$$K_2(\theta) = 1 - 0.00362 \cdot \theta - 1.32337 \cdot 10^{-4} \cdot \theta^2 \tag{3.19}$$

3.8 Summary and Conclusions

This chapter has introduced the basic concepts on solar concentration technology, especially PTCs, whose main components, design parameters, thermal losses and efficiency, together with energy balance, have been detailed. DSG and its operating modes have been also presented. Finally, the DISS solar thermal power plant has been explained, mentioning its main features and operating modes. The goal of this chapter is to provide theoretical knowledge to deal with the dynamic modeling of parabolic-trough solar thermal power plants. Further information can be found in Romero-Alvarez and Zarza (2007) for concentrating solar power, in Zarza (2000) for DSG PTC technology and the experimental DISS solar thermal power plant, in Fernández-García et al. (2010) for a review of PTC technology and its applications and in Valenzuela et al. (2005, 2006) for ACSs of DSG parabolic-trough solar thermal power plants.

4 Two-Phase Flow

I am an old man now, and when I die and go to Heaven there are two matters on which I hope for enlightenment. One is quantum electrodynamics and the other is the turbulent motion of fluids. And about the former I am rather more optimistic.

Sir Horace Lamb, *in an address to the British Association for the Advancement of Science*, 1932.

Abstract: In this chapter, a brief review of the basics of two-phase flow fluids is given. Two-phase flow patterns in horizontal pipes are also described and discussed, while basic relations useful for modeling one-dimensional, homogeneous two-phase flows are given. Additionally, a method for calculating the time derivative of the mean void fraction for two-phase flows is proposed. The water-steam IAPWS-IF97 standard and the *Modelica Media* library, used to develop the two-phase flow models, are also presented.

4.1 Single-phase and Two-phase Flows

Equally as important as the division of single-phase flows in laminar and turbulent flows is the division of two-phase flows in its flow patterns. As previously mentioned in § 3.5, one of the main concerns was to avoid a stratified pattern in the horizontal pipes of the DISS solar thermal power plant in order to avoid high temperature gradients. It was demonstrated experimentally in Zarza (2000) that a two-phase stratified pattern can be avoided and the DISS plant safely operated. Considering the single phase, temperature gradients cannot appear in the DISS solar thermal power plant, because the flow fills the tube completely, in which case it is well-wetted (Zarza, 2000).

Besides the stratified pattern, there are other possible two-phase flow patterns which depend on pressure, mass flow rate, heat flux and channel geometry. The patterns introduced in § 4.2 are for a horizontal cylindrical pipe. One important detail is that there is no direct method to determine the flow pattern, but visual inspection or indirect methods are usually used, e.g. applying thermal, electrical or optical principles (Hewitt, 1978). This leads to subjectivity when determining the flow pattern. Furthermore, there is no general consensus among authors on all possible patterns; however, some of them are generally accepted.

4.2 Flow Patterns in Horizontal Pipes

The flow pattern is the function of liquid and steam superficial velocities in the two-phase flow mixture, cf. Fig. 4.1. The four main two-phase flow patterns in horizontal pipes are: bubbly, intermittent, annular and stratified. However, additional flow patterns, or slight variations thereof, can also be enumerated (Zarza, 2000; Jensen, 2003).

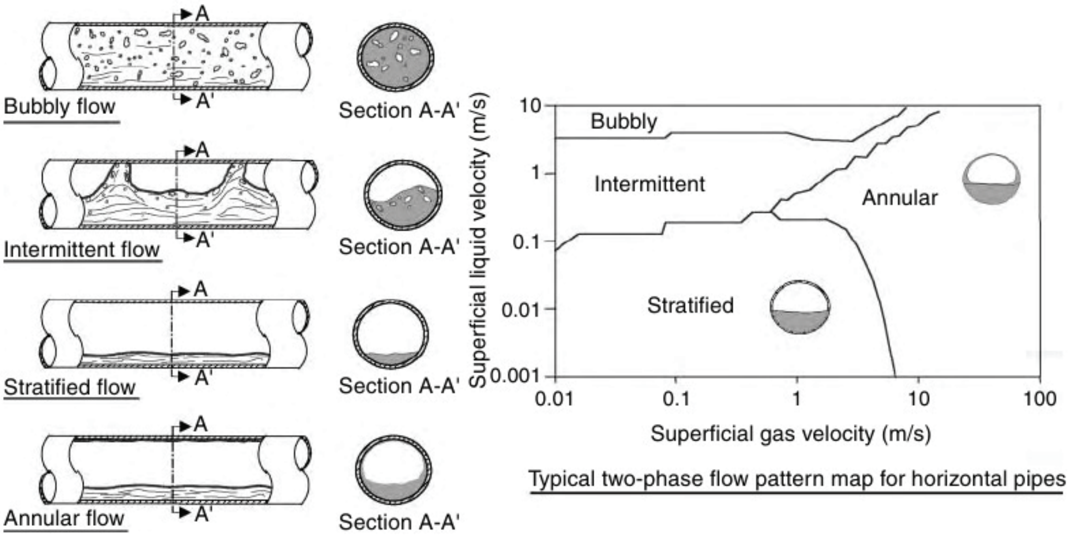


Figure 4.1: Main two-phase flow patterns in horizontal pipes (Romero-Alvarez and Zarza, 2007)

The following list briefly describes some of the most common two-phase flow patterns in horizontal pipes.

- *Bubbly flow.* If shear forces are dominant when vapor bubbles appear in the fluid, the bubbles tend to distribute homogeneously. Typically, this flow pattern is associated with high flow rates in horizontal pipes.

  - *Plug flow.* If the vapor bubbles collide, larger bubbles – called plugs – can appear. This pattern is defined by large bubbles (plugs) flowing in the upper half of the pipe.

- *Stratified flow.* At low velocities, the two phases flow separately with a clear interface border, where the vapor is at the top and the liquid is at bottom of the horizontal pipe due to gravity.

  - *Stratified-wavy flow.* If vapor velocity increases with respect to fluid velocity, e.g. for further evaporation, waves can appear in the interface between both phases.

- *Intermittent flow.* Increasing vapor velocity even more, waves become larger and wet the top of the horizontal pipe. If a cross-sectional area of the pipe is analyzed, this flow pattern can be seen as interment waves.

  - *Slug flow.* When waves touch the upper side of the pipe, slugs of liquid are formed. A slug can be pushed by the vapor at high velocity through the pipe, thus forming a liquid film.

- *Annular flow.* At higher velocities, the slugs create a liquid film around the pipe where the vapor flows in the core of the pipe. The liquid film in the upper part of the pipe is thinner due to gravity. Droplets can appear in the gas core if waves are still present. The highest convective heat transfer coefficients are associated with this flow pattern, and for this reason it is the predominant flow regime in evaporators.

  - *Mist flow.* This is is also called droplet flow, because droplets exist in the superheated vapor flow, even after the liquid film has dried out.

Fig. 4.2 shows some of the two-phase flow patterns in a horizontal pipe during evaporation. The boiling and dry-out points are also included.

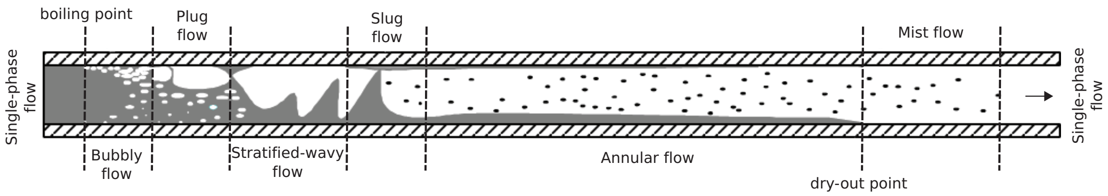


Figure 4.2: Some two-phase flow patterns in a horizontal evaporator

It is important to mention that the thermodynamic equilibrium boiling point is usually located downstream from the instantaneous boiling point (the point where the first bubbles appear), although under special flow conditions it may also be located upstream (Jensen, 2003). Furthermore, the thermodynamic equilibrium dry-out point is commonly located upstream from the instantaneous dry-out point, because liquid droplets may be still in the two-phase flow, as previously mentioned in the mist flow description.

According to Wedekind and Stoecker (1965), the instantaneous dry-out point exhibits stochastic behavior, showing some random fluctuations over an average value even under steady-state boundary conditions. The reason for these random fluctuations is related to the formation of slugs in the two-phase flow. Wedekind and Stoecker (1965) also found that the distribution of these fluctuations was not symmetric around the average liquid dry-out point and that these fluctuations could be estimated by a Rayleigh distribution. The reason for asymmetry in the distribution can be explained by changes in wall temperatures (Jensen, 2003).

4.3 Two-phase Flow Physic and Thermodynamic Properties

Physic and thermodynamic properties of two-phase flow are introduced in this section. These properties are used in this work for modeling one-dimensional two-phase flows. Fig. 4.3 shows liquid and vapor phases divided by an interface when considering a one-dimensional boiling channel. Fig. 4.3 represents a stratified pattern for the sake of clarity. However, the physic and thermodynamic properties derived in this chapter can be used independently of flow pattern. These properties depend on time (t) and on the horizontal spatial coordinate (z) (cf. Fig. 4.3). However, these dependencies have been omitted in most of the equations presented in this chapter for the sake of clarity.

Properties of a particular phase are called phasic properties, while properties of the mixture are called mixture properties. The properties introduced in Fig. 4.3 are: phasic velocities (w',w'') and phasic mass flow rates ($\dot{m}'$, $\dot{m}''$). Cross-sectional areas (A',A'') are also shown in Fig. 4.3.

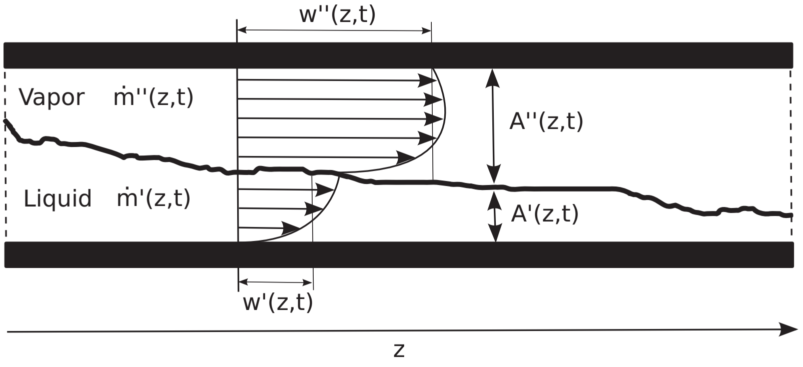


Figure 4.3: One-dimensional two-phase flow volume

The total cross-sectional area (A) is calculated by geometry, Eq. 4.1.

$$A = A' + A''. \tag{4.1}$$

Phasic mass flow rates can be calculated by Eqs. 4.3 and 4.4 and total mass flow rate by Eq. 4.2.

$$\dot{m} = \dot{m}' + \dot{m}'', \tag{4.2}$$

$$\dot{m}' = A'\rho'w', \tag{4.3}$$

$$\dot{m}'' = A''\rho''w''. \tag{4.4}$$

Total mass and masses for both phases are calculated by Eqs. 4.5, 4.6 and 4.7 considering the interval $[z, z + \Delta z]$.

$$m = m' + m'', \tag{4.5}$$

$$m' = V'\rho' = \Delta z A'\rho', \tag{4.6}$$

$$m'' = V''\rho'' = \Delta z A''\rho''. \tag{4.7}$$

4.3.1 Phasic Velocities

The velocity of the vapor (w'') is usually greater than the velocity of the liquid (w') due to the low viscosity and density of the vapor, although the velocity in the interface is the same for both phases. The slip ratio (S) is the ratio between the vapor and the liquid velocities, as stated in Eq. 4.8.

$$S = \frac{w''}{w'}. \tag{4.8}$$

When homogeneous flow is considered, the flow is in thermodynamic equilibrium and the phasic velocities are the same ($w = w' = w''$), in which case $S = 1$. However, when the flow is inhomogeneous, if both phases flow in the same direction, $S > 1$; otherwise, $S < 1$.

4.3.2 Local, Volumetric Mean and Area Mean Void Fractions

This section briefly describes the concepts of the local, volumetric mean and area mean void fractions introduced in Wedekind et al. (1978).

Local void fraction (γ_l). If it is considered a particular spatial point in the two-phase flow volume, this point is part of the vapor or of the liquid. This point, in a discrete value of time, takes the value 1, if it belongs to the vapor, or 0, if it belongs to the liquid. This definition for local void fraction $\gamma_l(x, y, z, t)$, was introduced in Wedekind et al. (1978).

Volumetric mean void fraction (γ_v). If three-dimensional volume is considered, then the volumetric void fraction represents the instantaneous mean value of all the local void fractions in volume V, as expressed in Eq. 4.9, where x, y and z are the spatial coordinates.

$$\gamma_v = \gamma_v(t) = \frac{1}{V}\int_V \gamma_l(x, y, z, t)dV. \tag{4.9}$$

Area mean void fraction (γ_A). Considering three-dimensional volume, the area mean void fraction represents the instantaneous mean value of all the local void fractions in area A, as expressed by Eq. 4.10.

$$\gamma_A = \gamma_A(z, t) = \frac{1}{A}\int_A \gamma_l(x, y, z, t)dA. \tag{4.10}$$

The area mean void fraction can be also expressed by Eq. 4.11, which is obtained by substituting the integral with $A''(z, t)$ in Eq. 4.10, due to the definition of local void fraction. In this book, the term ‘void fraction’ will be used indistinctly from area mean void fraction ($\gamma_A = \gamma$), because mean void fraction refers to volumetric mean void fraction ($\bar{\gamma} = \gamma_v$), since a one-dimensional two-phase flow is considered. This also applies to the remaining *mean* thermodynamic properties.

$$\gamma = \gamma_A = \frac{A''}{A}. \tag{4.11}$$

Isolating A'' in Eq. 4.11 produces Eq. 4.12. Eq. 4.13 is derived, after substituting Eq. 4.11 in Eq. 4.1.

$$A'' = \gamma A. \tag{4.12}$$

$$A' = (1 - \gamma)A, \tag{4.13}$$

Eq. 4.13 can be substituted in Eqs. 4.3 and 4.6 and Eq. 4.12 in Eqs. 4.4 and 4.7 to avoid the dependence of the phasic cross-sectional areas (A', A'').

4.3.3 Static and Vapor Qualities

This section introduces the concepts of static and vapor qualities.

Static quality (x). This is defined as the ratio between the mass of the vapor and the total mass. Applying Eqs. 4.5, 4.6, 4.7, 4.13 and 4.12 yields Eq. 4.14.

$$x = \frac{m''}{m} = \frac{A''\rho''}{A'\rho' + A''\rho''} = \frac{\gamma\rho''}{\gamma\rho'' + (1-\gamma)\rho'}. \tag{4.14}$$

Vapor quality ($\dot{x}$). This is also known as flow quality. It is the ratio between the mass flow rate of the vapor and the total mass flow rate, using the same derivation process that the one used for static quality; applying Eqs. 4.2, 4.3, 4.4, 4.11 and 4.8 yields Eq. 4.15. If a homogeneous two-phase flow is considered ($S = 1$), then $x = \dot{x}$.

$$\dot{x} = \frac{\dot{m}''}{\dot{m}} = \frac{\gamma\rho''}{\gamma\rho'' + \dfrac{1}{S}(1-\gamma)\rho'}. \tag{4.15}$$

4.3.3.1 Mixture Properties as a Function of Static Quality

Considering any intensive, also called specific, thermodynamic property (ψ) of a two-phase flow, any mixture property by definition can then be expressed by Eq. 4.16.

$$m\psi = m'\psi' + m''\psi''. \tag{4.16}$$

Dividing Eq. 4.16 by the total mass (m) yields Eq. 4.17.

$$\psi = \frac{m'}{m}\psi' + \frac{m''}{m}\psi''. \tag{4.17}$$

Substituting the first part of Eq. 4.14 in Eq. 4.17 yields Eq. 4.18.

$$\psi = (1-x)\psi' + x\psi''. \tag{4.18}$$

Mixture-intensive properties, e.g. specific enthalpy (h), specific internal energy (u), specific volume (ν) or velocity (w), can be calculated using Eq. 4.18. Considering the case of specific enthalpy (h) yields Eq. 4.19.

$$h = (1-x)h' + xh''. \tag{4.19}$$

Isolating x from Eq. 4.19 yields Eq. 4.20.

$$x = \frac{h-h'}{h''-h'}. \tag{4.20}$$

Eq. 4.20 is useful for calculating static quality in a two-phase flow as a function of specific enthalpy (h) and pressure (p). Vapor quality can also be calculated from Eq. 4.20 when homogeneous two-phase flow is considered, because $x = \dot{x}$.

Pressure and specific enthalpy are commonly the choice of state variables when modeling two-phase thermal-hydraulics for two reasons. First, according to the state postulate (Çengel and Boles, 2010) and considering the most simple system, only two independent intensive variables are needed to fully specify the entire state of the system. Second, temperature cannot be a state variable when considering two-phase flow, because it depends on pressure.

4.3.4 Mixture Density

Eqs. 4.21 and 4.22 are the phasic mass flow rates, obtained by Eqs. 4.3 and 4.4 substituting Eqs. 4.13 and 4.12, respectively.

$$\dot{m}' = A'\rho'w' = (1-\gamma)A\rho'w', \tag{4.21}$$

$$\dot{m}'' = A''\rho''w'' = \gamma A\rho''w''. \tag{4.22}$$

Substituting Eqs. 4.21 and 4.22 in the total mass flow rate expression, Eq. 4.2, yields Eq. 4.23.

$$\rho w = (1-\gamma)\rho'w' + \gamma\rho''w''. \tag{4.23}$$

Considering a homogeneous two-phase flow ($w = w' = w''$) yields Eq. 4.24, which is the mixture density for homogeneous two-phase flow.

$$\rho = (1-\gamma)\rho' + \gamma\rho''. \tag{4.24}$$

4.3.5 Density and Specific Enthalpy Product of the Mixture

Considering specific enthalpy (h) as the intensive thermodynamic property in Eq. 4.16 yields Eq. 4.25.

$$mh = m'h' + m''h''. \tag{4.25}$$

Substituting mass definition, Eqs. 4.6 and 4.7 in Eq. 4.25 yields Eq. 4.26.

$$A\rho h = A'\rho'h' + A''\rho''h''. \tag{4.26}$$

Finally, substituting Eqs. 4.13 and 4.12 and dividing by A yields Eq. 4.27, which is the density and specific enthalpy product of the mixture.

$$\rho h = (1-\gamma)\rho'h' + \gamma\rho''h''. \tag{4.27}$$

4.4 Proposed Calculation Method for the Time Derivative of the Mean Void Fraction

A method which can be employed to calculate the time derivative of the mean void fraction in a one-dimensional volume, which to knowledge of the author has not been applied before, is proposed in this section. This method was applied in the development of the MBMs, which are detailed in Chap. 7.

The static quality (x) and the area mean void fraction (γ) are related by Eq. 4.14. Isolating γ from Eq. 4.14 yields Eq. 4.28.

$$\gamma = \frac{x\rho'}{x\rho'(1-x)\rho''}. \tag{4.28}$$

Static quality is also defined by Eq. 4.20. Therefore, γ in Eq. 4.28 depends on pressure and specific enthalpy (p, h).

If a two-phase flow volume is considered, pressure (p) is assumed to be uniform (but time-dependent) within the volume, and a linear distribution of specific enthalpy $[h_a, h_b]$ is also assumed in the volume (this assumption is justified in § 8.4). The mean void fraction ($\bar{\gamma}$) can be calculated by using Eq. 4.29, which corresponds to the symbolic integration of Eq. 4.28 over specific enthalpy interval $[h_a, h_b]$ and replacing x by Eq. 4.20. The symbolic integration in Eq. 4.29 can be achieved by using software which can deal with symbolic manipulation and calculation. Mathematica (Wolfram, 2010) is used in order to perform this symbolic operation.

$$\bar{\gamma} = \frac{1}{z_b - z_a}\int_{z_a}^{z_b} \gamma\, dz = \frac{1}{h_b - h_a}\int_{h_a}^{h_b} \gamma\, dh = \frac{1}{h_b - h_a}\int_{h_a}^{h_b} \frac{x\rho'}{x\rho'(1-x)\rho''}\, dh = \frac{\rho'}{(h_a - h_b)(\rho' - \rho'')^2}\left((h_a - h_b)\,\rho' + \rho''\left(h_b - h_a + \left(h' - h''\right)\ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\right)\right), \tag{4.29}$$

$$\Gamma_1 = \rho'\left(h_a - h'\right) + \rho''\left(h'' - h_a\right), \tag{4.30}$$

$$\Gamma_2 = \rho'\left(h_b - h'\right) + \rho''\left(h'' - h_b\right). \tag{4.31}$$

The time derivative of $\bar{\gamma}$ can be calculated by using Eq. 4.32, applying the chain rule (cf. Appx. A.3) and considering that $\bar{\gamma}$ depends on (p, h_a, h_b). The partial derivatives of the mean void fraction were calculated symbolically from Eq. 4.29. These partial derivatives are rather lengthy, though, so they are detailed in Appx. B, $\partial\bar{\gamma}/\partial h_a$ (Appx. B.1), $\partial\bar{\gamma}/\partial h_b$ (Appx. B.2) and $\partial\bar{\gamma}/\partial p$ (Appx. B.3).

$$\frac{d\bar{\gamma}}{dt} = \frac{\partial\bar{\gamma}}{\partial h_a}\frac{dh_a}{dt} + \frac{\partial\bar{\gamma}}{\partial h_b}\frac{dh_b}{dt} + \frac{\partial\bar{\gamma}}{\partial p}\frac{dp}{dt}. \tag{4.32}$$

4.5 Water-Steam Thermodynamic Properties Calculation

The section details how water-steam thermodynamic properties were calculated in the developed models. § 4.5.1 introduces a standard to efficiently compute thermodynamic properties, and § 4.5.2 gives information about the Modelica library which implements such a standard.

4.5.1 The IAPWS-IF97 Standard

The International Association for the Properties of Water and Steam (IAPWS) in its latest published Industrial Formulation 1997 (IF97) (Wagner et al., 2000; IAPWS, 2007), recommends an optimized formulation with short computing times and low Central Processing Unit (CPU) load. The formulation provided in this release is recommended for industrial use. The IAPWS also has a formulation intended for general and scientific use, the IAPWS-95 (IAPWS, 2009). However, the IAPWS-IF97 is optimized, being much faster than the IAPWS-95, in the order of 100 to 200 times faster (Wang et al., 2012). When designing dynamic models intended to be used for advanced ACSs, fast computation is highly recommended. For this reason, the IAPWS-IF97 has been used in this work.

The IAPWS-IF97 formulation consists of a set of regional equations, depending on the flow state. Fig. 4.4 shows the five regions into which the entire range of IAPWS-IF97 validity is divided, with basic (rectangular boxes) and backward equations. The backward equations were developed in such a way that they are numerically very consistent with the corresponding basic equations. Supplementary releases on backward equations were published in IAPWS (2001, 2004b,a, 2005); Kretzschmar et al. (2007). Revised releases of IAPWS-IF97 were presented in 2007, 2009 and 2010 involving minor corrections to typographical errors and the updating of references.

The thermodynamic regions of the IAPWS-IF97 are as follows.

- *Region 1.* Sub-cooled water.
- *Region 2.* Superheated steam.
- *Region 3.* Water saturation and two-phase flow.
- *Region 4.* Boiling and condensation.
- *Region 5.* Superheated steam, $\mathrm{p} < 50\,\mathrm{MPa}$ and $\mathrm{T} \in [1073.15, 2273.15]$ K.

The IAPWS-IF97 achieves fast computation by dividing the formulation into five sub-regions and then applying different formulas. Fast computation can be also achieved using the Tabular Taylor Series Expansion (TTSE) method (Miyagawa and Hill, 2001), which was adopted by the IAPWS as a guideline in 2003 (IAPWS, 2003).

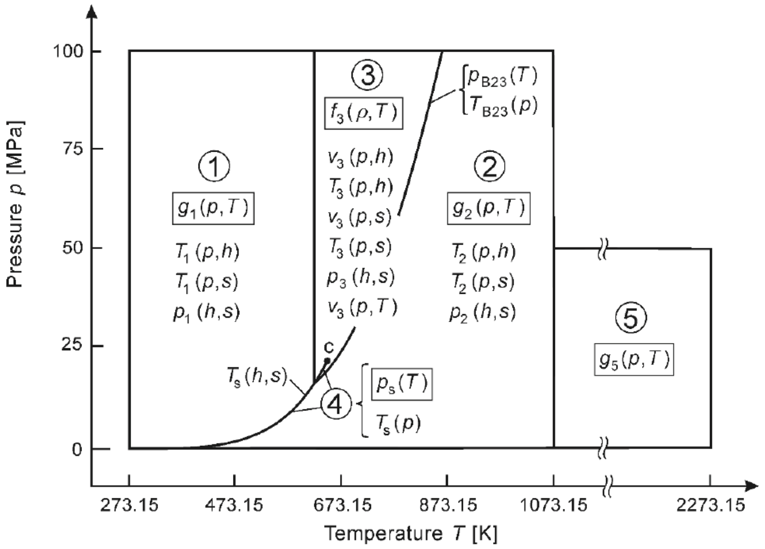


Figure 4.4: Structure and regions of IAPWS-IF97 (Wagner et al., 2000)

The table look-up method has proven to be computationally faster than the TTSE method (Wang et al., 2012). However, the table look-up method cannot represent continuous thermodynamic properties over the entire thermodynamic surface (Wang et al., 2012). On the other hand, the IAPWS-IF97 divides the formulation into regions, thus allowing fast calculation, but this can lead to inconsistencies at region boundaries. These discontinuities have no physical reality but are artifacts of the formulation method (The American Society of Mechanical Engineers, 1999). While usually small, these discontinuities can be troublesome when the solver is trying to find a solution near a region boundary. This inconvenience will be discussed in Chap. 6.

4.5.2 The Modelica Media Library

The *Modelica Media* library (Casella et al., 2006) provides media models for most common liquids and gases, about 1250 in total, including a high precision water model. It also provides a framework for the development of new fluid models, allowing for the computation of all the relevant fluid properties in efficient numerical simulations. External codes can be reused by the *External Media* library (Casella and Richter, 2008), which provides a framework for implementing *Modelica Media* compatible fluid models using external code.

The library found its origins in 2002 as part of the *ThermoFluid* library, based mainly on contributions from the Ph.D. dissertations of Tummescheit (2002), Eborn (2001) and Wagner (2000). The main author and maintainer of the library is Hubertus Tummescheit, while some of the more remarkable researchers involved in this project are: Jonas Eborn, Falko Jens Wagner, Martin Otter, Michael Tiller, Hilding Elmqvist, Hans Olsson, Sven Erik

Mattsson, Katrin Prößß, Rüdiger Franke, Francesco Casella, John Batteh, Daniel Bouskela, Andreas Idebrant, Charles Newman and Gerhart Schmitz among others.

Modelica Media, and specifically the *Modelica Media Water* package, which follows the IAPWS-IF97 standard, makes it possible to work with water-steam two-phase flow models in Modelica. *Modelica Media* was used in this work in the development of two-phase flow models.

4.6 Summary and Conclusions

This chapter introduced the main concepts of two-phase flows, the main patterns in horizontal pipes and the main physic and thermodynamic properties of two-phase flows. Some relations between thermodynamic properties for homogeneous two-phase flows were also presented. These expressions will be extremely useful, because the two-phase HTF in the DISS facility can be considered homogeneous, as will be detailed in § 5.4. Reviews for further information about two-phase flow basics can be found in Jensen (2003) and Yebra (2006). Additionally, a proposed calculation method for calculating the time derivative of the mean void fraction, when considering a linear distribution of specific enthalpy and uniform pressure in a two-phase flow volume, has been detailed. Finally, the IAPWS-IF97 standard and the *Modelica Media* library were presented.

4.7 Related Scientific Contributions

The scientific contributions related to this chapter are enumerated and briefly commented on in the following list. These contributions were previously listed in § 1.4.

5. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Library of Switching Moving Boundary Models for Two-phase Flow Evaporators and Condensers. In *Proceedings of the 9th International Modelica Conference*, oral communication, Munich, Germany, 2012.

Application of the proposed calculation method for the time derivative of the mean void fraction in general, flooded and dry evaporators and condensers.

6. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Modeling of Switching Moving Boundary Models for Two-phase Flow Evaporators. In *Proceedings of the 7th Vienna International Conference on Mathematical Modelling (MATHMOD)*, oral communication, Vienna, Austria, 2012.

Proposed calculation method for the time derivative of the mean void fraction and its application in a moving boundary model of a flooded two-phase flow evaporator.

III

Equation-Based Object-Oriented Modeling of Dynamic Thermo-Fluid Systems

5

Modeling the DISS facility

We have no idea about the 'real' nature of things... The function of modeling is to arrive at descriptions which are useful.

Richard Bandler and John Grinder, *Frogs into Princes: Neuro Linguistic Programming*, 1979

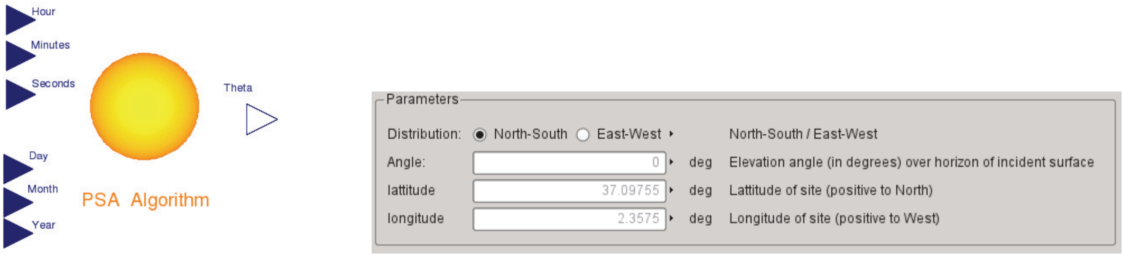
Abstract: Details about the modeling of the different components which compose the DISS test facility are given in this chapter. The only considered operating mode of the DISS test facility is the once-through operating mode, while each component of a PTC, previously introduced in Chap. 3, is detailed in modeling terms. Once all the components have been described, the complete DISS model is also presented. This dynamic model is intended to be used primarily for the development of advanced ACSs. Modelica is the modeling language chosen, which was previously described in Chap. 2.

5.1 Computing the Solar Vector

The DNI is calculated using the DSI, measured by a pyrheliometer installed in the DISS facility, multiplied by the cosine of the IA. Therefore, the IA is required and the solar vector must be calculated.

A solar position algorithm, called the *PSA algorithm*, is used to compute the solar vector (cf. § 3.4.2.4). This algorithm was presented in Blanco-Muriel et al. (2001) and it is a high-precision astronomical algorithm. Fig. 5.1a shows the Modelica component icon which computes the solar vector using the *PSA algorithm*.

The only input into the model is local time (hours, minutes, seconds, day, month and year), cf. Fig. 5.1a. The parameters of the model are geographic positions (latitude and longitude), inclination over the horizon and the distribution of solar concentrators (north-south or east-west), cf. Fig. 5.1b. The model output is the IA (θ).



Parameters

Distribution:	● North-South ○ East-West		North-South / East-West
Angle:	0	deg	Elevation angle (in degrees) over horizon of incident surface
lattitude	37.09755	deg	Lattitude of site (positive to North)
longitude	2.3575	deg	Longitude of site (positive to West)

(a) Icon (b) Parameters

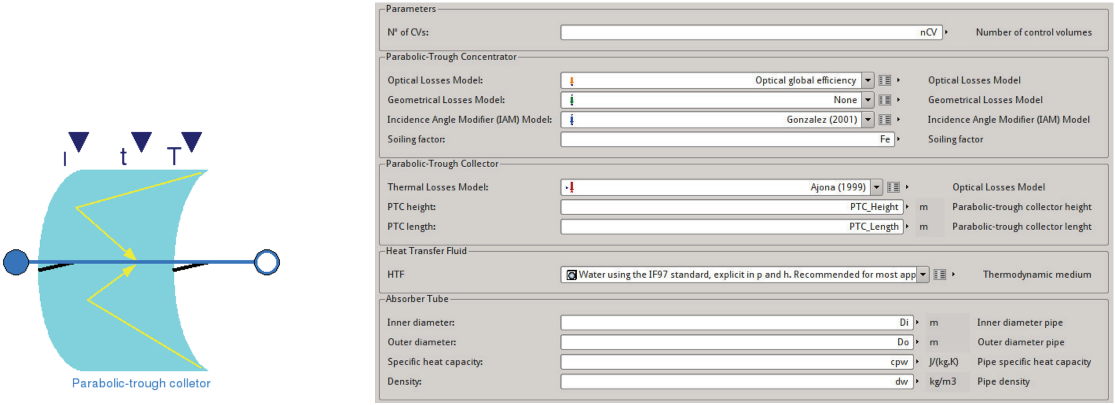
Figure 5.1: PSA algorithm Modelica component

5.2 Parabolic-Trough Collector Model

The PTC is the main component to be modeled in the DISS solar thermal power plant in terms of concentrating solar energy. PTCs are linear 2D solar concentrators which convert DSI into thermal energy, which in turn heats a HTF. PTCs were previously introduced in § 3.4.

Fig. 5.2a shows the PTC Modelica icon, whereas Fig. 5.2b shows its parameters. They are related which each one of the PTC main components. The evaporator parameters are not presented here, but they will be discussed in § 5.3, wherein the evaporator model is introduced. The PTC model is one-dimensionally discretized in the horizontal axis; the number of elements is a parameter of the model.

The PTC inputs are: the DSI (Irr), the IA (theta) and the ambient temperature (Tamb), cf. Fig. 5.2a and Fig. 5.3.



Parameters

Nº of CVs:	nCV		Number of control volumes

Parabolic-Trough Concentrator

Optical Losses Model:	Optical global efficiency		Optical Losses Model
Geometrical Losses Model:	None		Geometrical Losses Model
Incidence Angle Modifier (IAM) Model:	Gonzalez (2001)		Incidence Angle Modifier (IAM) Model
Soiling factor:	Fe		Soiling factor

Parabolic-Trough Collector

Thermal Losses Model:	Ajona (1999)		Optical Losses Model
PTC height:	PTC_Height	m	Parabolic-trough collector height
PTC length:	PTC_Length	m	Parabolic-trough collector lenght

Heat Transfer Fluid

HTF	Water using the IF97 standard, explicit in p and h. Recommended for most app		Thermodynamic medium

Absorber Tube

Inner diameter:	Di	m	Inner diameter pipe
Outer diameter:	Do	m	Outer diameter pipe
Specific heat capacity:	cpw	J/(kg.K)	Pipe specific heat capacity
Density:	dw	kg/m3	Pipe density

(a) Icon (b) Parameters

Figure 5.2: PTC Modelica component

In the component diagram of the PTC, which is shown in Fig. 5.3, the PTCN, absorber tube, evaporator and replaceable thermal losses models can be seen. Each one of the PTC's main component models are described in § 5.2.1, while energy losses models are described in § 5.2.2. The interface for the PTC model is composed of inputs and two connectors for the HTF. These connectors belong to the *Modelica Fluid* library (Casella et al., 2006) in this particular case. However, there are several versions of the PTC model which depend on the type of evaporator used. This is discussed in § 5.5.

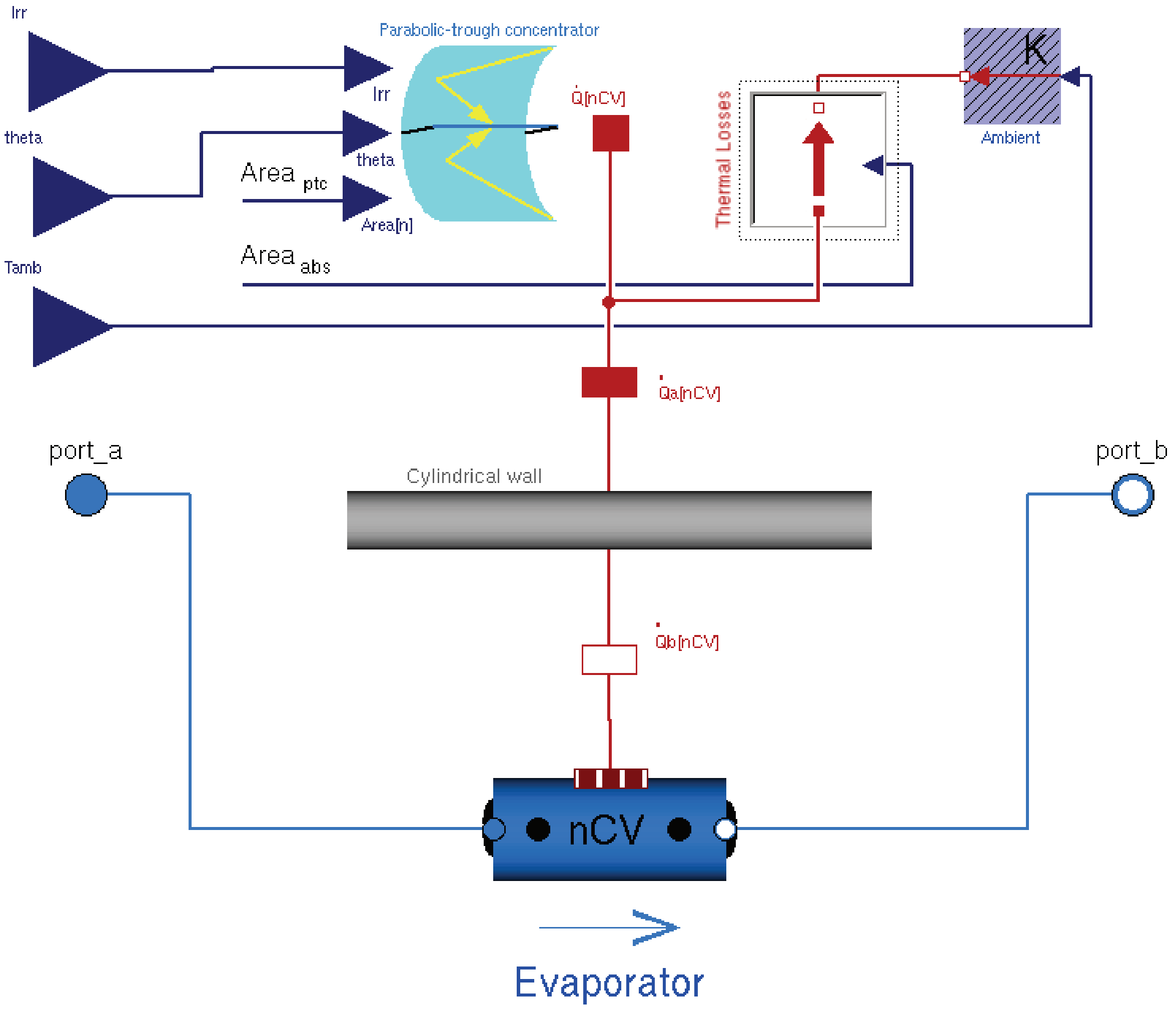


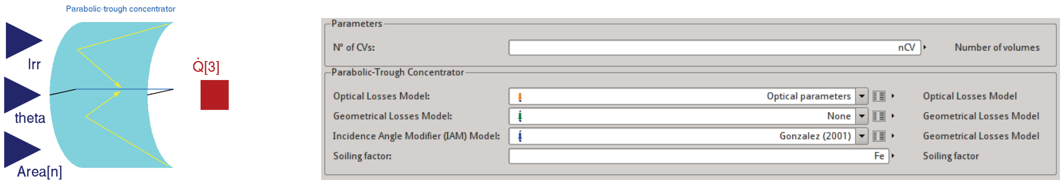
Figure 5.3: PTC component diagram

5.2.1 Main Components

This section details the main components of the PTC model. The steel structure does not need to be modeled if the goal is to model the thermal-hydraulic process in the system. The solar tracking system is supposed to follow the Sun's trajectory properly, so it has not been modeled. The evaporator model is discussed in § 5.3.

5.2.1.1 Parabolic-Trough Concentrator Model

The PTCN Modelica model icon can be seen in Fig. 5.4a, and the model inputs are: the DSI (Irr), the IA (theta) and the PTCN aperture area (area). This model considers optical and geometrical losses and the IAM (cf. Fig. 5.5). Although some optical losses are associated with the absorber tube (transmissivity and absorptivity), they have all been included in one optical losses model. The soiling factor (Fe) is included in the PTCN model too, even though soiling also affects the glass cover in the absorber tube. A one-dimensional discretization in the horizontal axis has been also considered in the PTCN model; the number of elements is a parameter of the model.



Parameters

Nº of CVs:	nCV	Number of volumes

Parabolic-Trough Concentrator

Optical Losses Model:	Optical parameters	Optical Losses Model
Geometrical Losses Model:	None	Geometrical Losses Model
Incidence Angle Modifier (IAM) Model:	Gonzalez (2001)	Geometrical Losses Model
Soiling factor:	Fe	Soiling factor

(a) Icon (b) Parameters

Figure 5.4: PTCN Modelica component

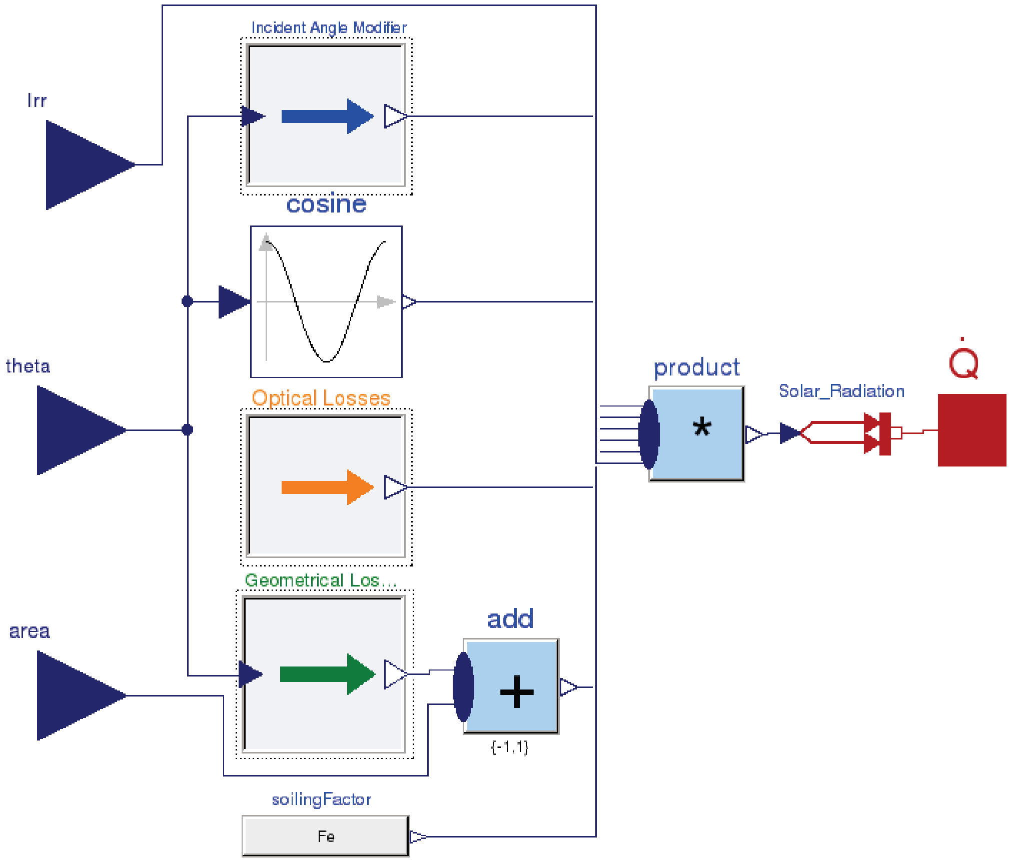


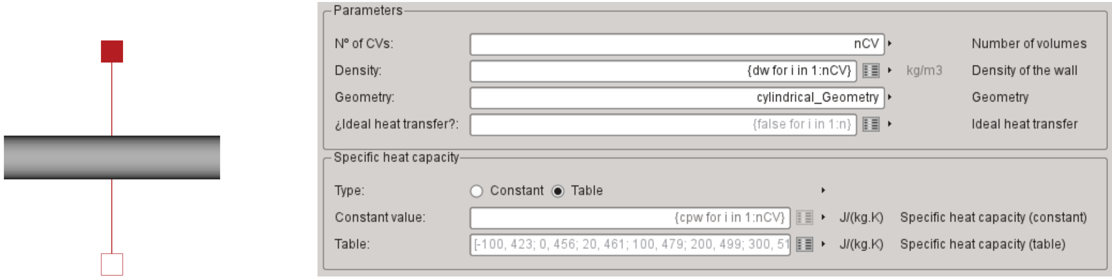
Figure 5.5: PTCN component diagram

The energy losses components are declared as *replaceable* parameters (cf. Fig. 5.4b) (cf. advanced parametrization § 2.11.1), meaning that they can be *redeclared* with different models. They are described in more detail in § 5.2.2. The IAM models usually include geometrical losses, in which case the geometrical losses model can be *redeclared* to a model (called *None*) that does not modify the concentrated solar radiation, which is the output of the PTCN model ($\dot{Q}$). This output is propagated to any interconnected model through a *HeatPort* which belongs to the *Modelica Thermal* library. The PTCN model also considers the PTCN area and the IA to calculate the DNI on the absorber tube. Therefore, the PTCN model computes Eq. 5.1 (derived from Eq. 3.14) – the only missing term is thermal efficiency (η_{the}), which is globally considered in the PTC model (cf. § 5.2).

$$\dot{Q}_{ptc\rightarrow abr} = \dot{Q}_{sun\rightarrow ptc} \cdot \eta_{opt,0} \cdot \eta_{geo} \cdot F_e = A_{ap} \cdot G_b \cdot \cos(\theta) \cdot \eta_{opt,0} \cdot \eta_{geo} \cdot F_e. \quad (5.1)$$

5.2.1.2 Absorber Tube Model

The icon and parameters of the absorber tube model are shown in Figs. 5.6a and 5.6b, respectively. Optical losses due to the glass cover (transmissivity) and the absorber steel pipe (absorptivity) are considered in the PTCN model. The model parameters are: the number of elements, density, specific heat capacity and the geometry of the pipe, which is a parametric class. Specific heat capacity can be a constant value or a variable value as a function of the temperature, and these values are stored in a table. Tab. 5.1 shows specific heat capacity as a function of temperature for an A335 grade P22 (10CrMo910) alloy steel pipe, according to the EN 10273 standard (UNI, 2008). An additional parameter can be set to *true* for any volume, if heat is ideally transferred to the HTF. The model interface is compounded of two *HeatPorts* from the *Modelica Thermal* library, which interconnect the PTCN model and ambient temperature with the evaporator model.



Parameters

Nº of CVs:	nCV		Number of volumes
Density:	{dw for i in 1:nCV}	kg/m3	Density of the wall
Geometry:	cylindrical_Geometry		Geometry
¿Ideal heat transfer?:	{false for i in 1:n}		Ideal heat transfer

Specific heat capacity

Type:	○ Constant ● Table		
Constant value:	{cpw for i in 1:nCV}	J/(kg.K)	Specific heat capacity (constant)
Table:	[-100, 423; 0, 456; 20, 461; 100, 479; 200, 499; 300, 51	J/(kg.K)	Specific heat capacity (table)

(a) Icon

(b) Parameters

Figure 5.6: Absorber tube Modelica component

T (°C)	-100	0	20	100	200	300	400	500	600
c_p (J/(kg · K))	423	456	461	479	499	517	536	558	597

Table 5.1: Specific heat capacity for A335 grade P22 (10CrMo910) alloy steel

Conduction in the steel absorber pipe has been neglected because the inner and outer temperatures are almost identical in the DISS absorber tubes (Zarza, 2000). Hence, the absorber tube model represents heat transfer from the absorber steel pipe to the HTF by convection, as stated in Eq. 5.2 (Tummescheit, 2002; Jensen, 2003).

$$A_{abr}\rho_{abr}c_{p,abr}\frac{dT_{abr}}{dt} = \dot{q}_{abr} - \dot{q}_{htf}. \tag{5.2}$$

On the other hand, when ideal heat transfer is considered, $T_{abr} = T_{htf}$.

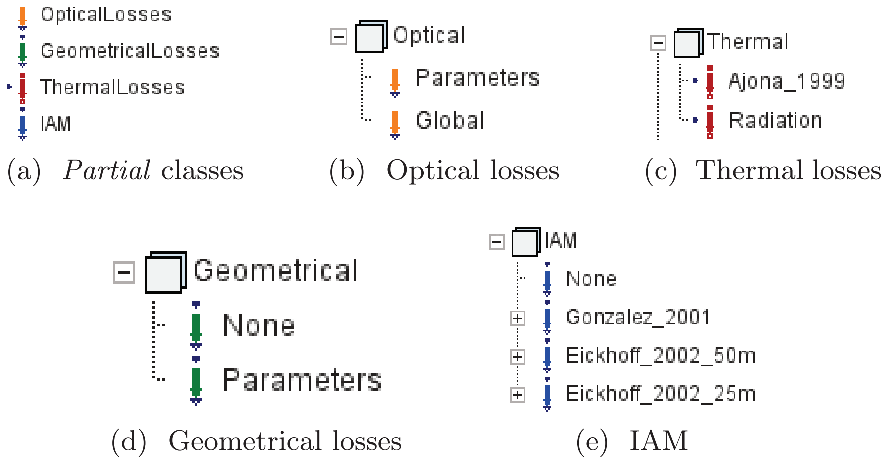
Geometry Model. A partial base class for geometry has four variables: inner and outer cross-sectional areas and the surface areas of the absorber tube. From this base class different geometries can be created and used, inheriting from it and selecting one of them as a model parameter. In Fig. 5.6b, cylindrical geometry is considered. This geometry model implements Eqs. 5.3a, 5.3b, 5.3c and 5.3d to calculate inner and outer cross-sectional areas (A_i, A_o) and surface areas (S_{a_i}, S_{a_o}), where d_i and d_o are the inner and outer diameters and l is the length.

$$\begin{gathered} A_i = \pi\frac{d_i^2}{4} \quad (a), \quad A_o = \pi\frac{d_o^2}{4} \quad (b), \\ S_{a_i} = \pi d_i l \quad (c), \quad S_{a_o} = \pi d_o l \quad (d). \end{gathered} \tag{5.3}$$

5.2.1.3 Heat Transfer Fluid Model

In the DISS facility, the HTF is a two-phase water-steam flow which circulates in three different states: sub-cooled liquid, a water-steam mixture and superheated steam.

The *Modelica Media* library (cf. § 4.5.2) was used to compute the thermodynamic properties of this HTF. The *Modelica Media* library follows the IAPWS-IF97 standard – an optimized formulation with short computing times and low CPU load (cf. § 4.5.1). Mixture thermodynamic properties were calculated using the material presented in Chap. 4. The HTF model is included in the evaporator model.



(a) *Partial* classes (b) Optical losses (c) Thermal losses

(d) Geometrical losses (e) IAM

Figure 5.7: Energy losses models

5.2.2 Energy Losses Models

As described in § 3.4.4, thermal, optical and geometrical losses are associated with a PTC. *Partial* classes have been developed for each type of energy losses in order to allow for different models (cf. Fig. 5.7a). In each category, several experimental correlations or parametrized models have been implemented. Furthermore, new energy losses models, inheriting from the partial base classes, can be implemented. Energy losses models can be selected through the Graphical User Interface (GUI), as shown in Figs. 5.2b and 5.4b.

5.2.2.1 Optical Losses Model

Two optical losses models are implemented – one based on the optical parameters described in § 3.4.4.1 (reflectivity, intercept factor, transmissivity and absorptivity), and another one based on a global optical efficiency parameter (cf. Fig. 5.7b). Sometimes, it is easier to estimate optical efficiency using just one global optical efficiency parameter rather than each optical efficiency parameter separately.

5.2.2.2 Thermal Losses Model

The experimental thermal losses model for a modified LS-3 PTC proposed in Ajona (1999) (cf. § 3.7.3) is implemented (cf. Fig. 5.7c). Additionally, a net radiation loss rate model is also included according to the Stefan-Boltzmann law and considering an emissivity parameter (ϵ), Eq. 5.4. Other models, inheriting from the partial base class, can be also implemented.

$$\dot{Q}_{abr\to amb} = \varepsilon \cdot \sigma \cdot S_{a_o} \cdot (T_{abr}^4 - T_{amb}^4). \tag{5.4}$$

5.2.2.3 Geometrical Losses Model

The geometrical losses model described in § 3.4.4.3, based on the PTC's geometrical end losses, is available for use. Additionally, *none* of the geometrical losses model can be selected if the selected IAM model includes geometrical losses, cf. Fig. 5.7d.

5.2.2.4 Incidence Angle Modifier Model

The IAM models proposed by González et al. (2001) and Eickhoff (2002) in its two variants (50 m and 25 m) and described in § 3.7.3 have been implemented, cf. Fig. 5.7e. Furthermore, the *None* IAM model can be also selected.

5.3 The Evaporator Model

The evaporator model is the keystone in the modeling of the DISS solar thermal power plant. The most generic equation used to model the motion of fluids and convective heat transfer from the absorber steel pipe to the fluid flow is a PDE, which applies to most physical phenomena. The motion of fluids is usually modeled by the conservation laws of mass, energy and momentum, establishing balances between regions or volumes, while heat transfer is analogously modeled by the energy conservation law.

As mentioned in § 2.9.2, PDE systems are not commonly supported by M&S tools. Therefore, it is necessary to apply a discretization method to transform PDE to DAE systems, in order to be solvable by M&S tools. The collection of methods employed to perform this task are known as Computational Fluid Dynamics (CFD). The classical discretization methods are: FDM, FVM and FEM (Anderson, 1995).

The FDM was the first of these methods to be developed. When given a number of discretization points defining a mesh with an unknown and an equation in each point in the most simple case, the FDM replaces the derivatives of the unknown with finite differences through the use of Taylor expansions. However, this method presents a bottleneck when handling complex geometries in multiple dimensions (Peiró and Sherwin, 2005), and it becomes difficult to use when the coefficients involved in the equations are discontinuous (Eymard et al., 2000).

The FVM and FEM are discretization methods based on the integral form of PDEs, and both methods involve discretization in a number of Finite Elements (FEs)/CVs.

The FEM uses simple piecewise functions to describe the local variations of unknowns. Piecewise approximating functions are substituted into the equations, and a residual is then defined to measure the errors. Residuals are minimized by multiplying them by a set of weighing functions and then integrating them accordingly. As a result, a set of DAEs for the unknown coefficients of the approximating functions is obtained.

The FVM applies a formal integration of equations over CVs. A variety of approximations for the terms in the integrated equation are usually applied, which transforms the integral equations into a DAE system. With the FVM, discontinuities of the equation coefficients will not be a problem if the mesh is chosen such that the discontinuities of the coefficients occur on the boundaries of the CVs. The FVM is widely used in industry and academia because it provides a robust, easy and reliable discretization method for conservation laws, and it behaves well even for difficult equations. Although in some cases it can be difficult to design schemes with enough precision, the FEM can provide more precision with high order polynomials which, in some cases, would be difficult to obtain. (Eymard et al., 2000).

Besides the classical discretization methods, the moving boundary method is a different technique which takes explicitly into account the discretization of PDE systems by means of using CVs of variable length in an evaporator.

In this book, two different approaches to discretizing an evaporator are studied: FVMs and MBMs. Both approaches are detailed in Chaps. 6 and 7, respectively.

5.4 Two-phase Flow Models

Once discretization methods have been defined to divide the evaporator into different CVs, the next concern is how to model two-phase flow in each CV. Tab. 5.2 shows a classification of discretized two-phase flow models proposed in Todreas and Kazimi (1993). In the most general case, formulating separate mass, energy and momentum balances (balance equations in Tab. 5.2) for both phases is required. It is also necessary to have mass, energy and momentum balances between both phases (interface equations in Tab. 5.2). In order to complete the model, heat transfer and frictional force equations – between each phase and the absorber tube – are also required (wall equations in Tab. 5.2). All of these considerations lead to the most complex model (the 6-Eqs. model in Tab. 5.2). However, with some assumptions, the resulting model can be drastically simplified.

The Homogeneous Equilibrium Model (HEM) reduces the number of required equations through a number of constraints. Mixture balance equations are considered in this model, so the two-phase is treated as a single phase with average thermodynamic properties, which is why it is is called ‘homogeneous’. The thermodynamic equilibrium between phases is considered in HEMs, hence the temperatures in each phase correspond with the saturation temperatures (T', T''). Additionally, phasic velocities are the same $(w' = w'')$ because the two-phase flow is homogeneous. Finally, the equations for heat transfer and pressure loss due to friction are also required.

Model	**3-Eqs.**		**4-Eqs.**			**5-Eqs.**			**6-Eqs.**
	HEM	**InHEM**	**A**	**B**	**C**	**A**	**B**	**C**	
Balance equations									
Mass	1	1	2	1	1	2	2	1	2
Energy	1	1	1	2	1	2	1	2	2
Mom.	1	1	1	1	2	1	2	2	2
Total	*3*	*3*	*4*	*4*	*4*	*5*	*5*	*5*	*6*
Phasic constraints									
T	2[a]	2[a]	1[b]	1[b]	2[a]	0	1[b]	1[b]	0
w	1[c]	1[d]	1[d]	1[d]	0	1[d]	0	0	0
Total	*3*	*3*	*2*	*2*	*2*	*1*	*1*	*1*	*0*
Wall equations									
$\mathbf{F_f}$	1	1	1	1	2	1	2	2	2
$\mathbf{\dot{Q}_w}$	1	1	1	2	1	2	1	2	2
Total	*2*	*2*	*2*	*3*	*3*	*3*	*3*	*4*	*4*
Interface equations									
Mass	0	0	1	1	1	1	1	1	1
Energy	0	0	0	1	0	1	0	1	1
Mom.	0	0	0	0	1	0	1	1	1
Total	*0*	*0*	*1*	*2*	*2*	*2*	*2*	*3*	*3*

Table 5.2: Classification of discretized two-phase flow models (Todreas and Kazimi, 1993)

The only difference between the HEM and the InHomogeneous Equilibrium Model (InHEM) is that the latter does not consider a homogeneous flow. Therefore, a ratio between the phasic velocities is required, which is known as a slip ratio (cf. § 4.3.1). Different slip ratio correlations (S) are available in the literature (Whalley, 1987; Collier and Thome, 1996; Ghiaasiaan, 2007).

The 4-Equation A model includes two separate mass balances. Therefore, there is non-thermodynamic equilibrium in one of the phases, reducing the number of phasic temperature constraints; the mass balance equation between phases is then required.

The same situation occurs with the 4-Equation B model but with two separate energy balances. Hence, two separate pipe wall heat transfer equations are required ($\dot{Q}_w$), as well as mass and energy balance equations between phases.

The 4-Equation C model considers two separate momentum balances

[a] $T'(p)$ and $T''(p)$.

[b] $T'(p)$ or $T''(p)$.

[c] $w' = w''$.

[d] $s = w''/w'$.

and, consequently, no slip correlation is required (S). Moreover, two equations for pressure loss due to friction (F_f) must be considered, while mass and momentum balance equations between phases are also required. The same reasoning can be applied to the 5-Equation models, which also consider non-thermodynamic equilibrium models.

According to Fowler (1997), some 6-Equation models are ill-posed. This problem can be overcome by using proper averaging within each phase (Jensen, 2003). Todreas and Kazimi (1993) mentioned that 6-Equation models can be extended to 9-Equation models, including balance equations for bubbles, liquid, vapor and the liquid droplets, but this approach is not commonly used due to its complexity.

In this work, the main goal is to develop models as tools for the design of advanced ACSs; therefore, fast models are required. Furthermore, when considering relatively high HTF pressures, the slip ratio is close to 1 (Jensen and Tummescheit, 2002), which is the case for the DISS facility, where nominal outlet working pressures are 3 MPa, 6 MPa and 10 MPa (cf. Tab. 3.2). Due to the aforementioned reasons, all the models considered in this book are HEMs.

5.5 The DISS Model

The first version of the DISS Modelica Model was developed by Yebra (2006) in his Ph.D. dissertation. This dynamic model is intended to be an essential tool for the design of advanced ACSs in the DISS solar thermal power plant. Since the first version of the model, some new models have been created and adapted to new interfaces, and advanced features of the Modelica language have been included, e.g. classes parametrization, more parameters, and the model has been adapted to Modelica 3.2. The new version of the DISS Modelica model using the *Modelica Fluid* library can be seen in Fig. 5.8.

The eleven PTC components which originally composed the DISS solar field can be seen in Fig. 5.8. The two additional PTCs, built at the beginning of the field in 2003 (cf. § 3.6), are not considered in the model, while interconnections between the PTCs have been neglected for two reasons, in order to simplify the model.

Firstly, energy losses are low because the pipes are well insulated and the maximum temperature difference at the interconnection is always lower than 1.5 °C[e] (Zarza, 2000). Secondly, pressure loss in the DISS field is lower than 10 bar (1 MPa) (cf. Tab. 3.2) and 1 bar in each PTC, including the interconnection. In the worst case, interconnection pressure loss is equivalent to 11 m of PTC, according to the detailed steady-state pressure loss model developed in Zarza (2000), i.e less than 0.18 bar. Furthermore, the detailed

[e] The accuracy of the DISS temperature sensors is ±1.25 °C (cf. § 8.5.3).

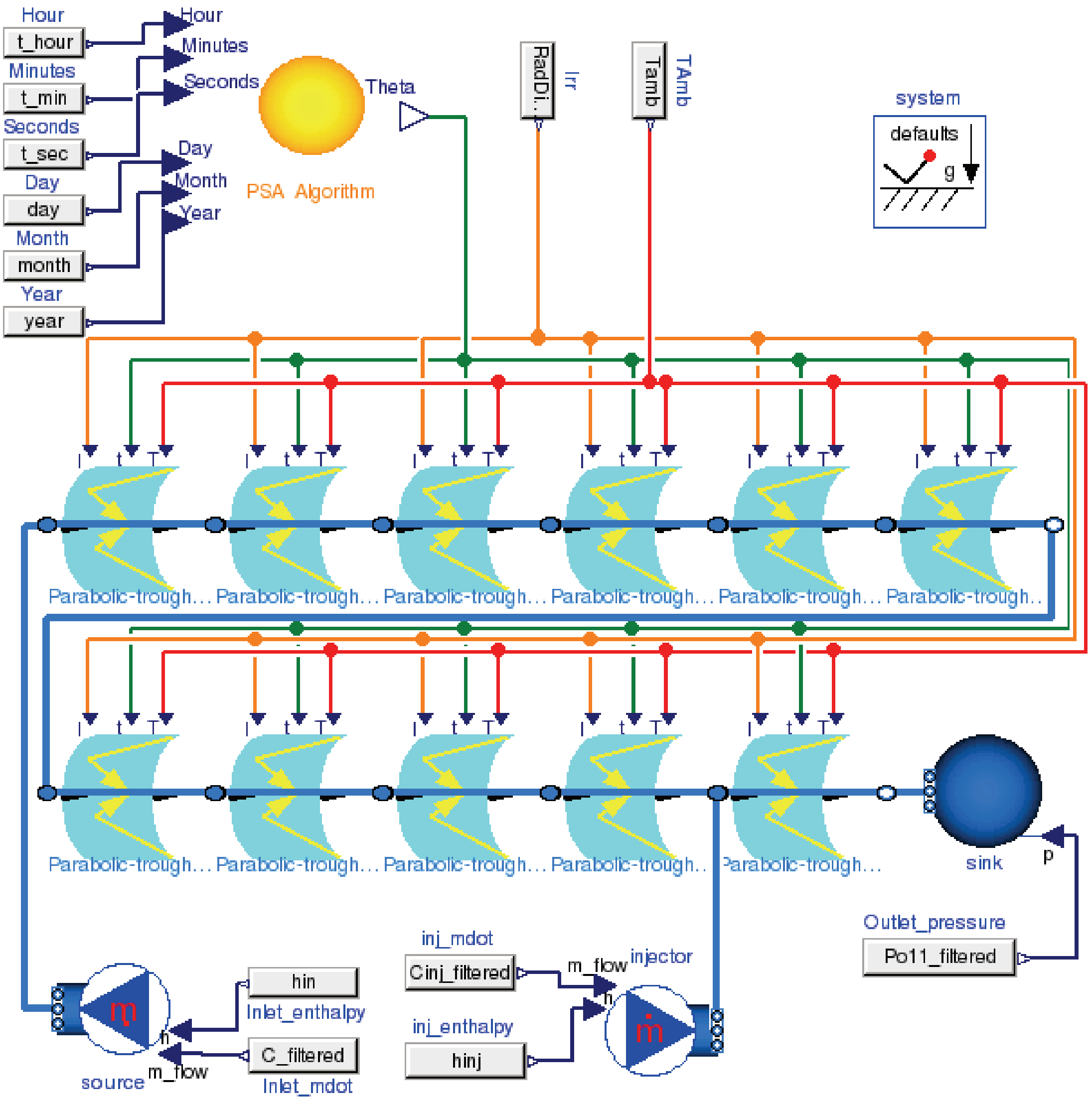


Figure 5.8: DISS Modelica model

pressure loss model with interconnections implies discontinuities in pressure loss, which could be troublesome during simulation.

The DISS model inputs are: the DSI (Irr), the IA obtained from the PSA algorithm component and ambient temperature (TAmb). The PSA algorithm component requires local time as an input. The thermo-hydraulic system inputs are: the inlet mass flow rate (Inlet_mdot), inlet specific enthalpy (Inlet_enthalpy), outlet pressure (Outlet_pressure) and the inlet mass flow rate (inj_mdot), as well as inlet specific enthalpy (inj_enthalpy) for the 11th PTC injector.

With the aim of calibrating and validating the model, a simulator scheme for the DISS model was defined. The DISS simulator allows for using experimental data from the DISS facility as an input (input data files) and compares the simulation results against experimental data. This simulator scheme also facilitates setting initial conditions and retrieving experimental data from the Data Acquisition System (DAS) to be stored in the input data files. The DISS simulator is described in detail in Chap. 8.

The main elements of a PTC have been described previously. However, there is still a missing component that only has been mentioned briefly in § 5.3, the evaporator model. This model is responsible for the evaporation of fluid through the boiling channel.

The first version of the DISS Modelica Model applies the FVM to discretize the boiling channel, dividing the evaporator into CVs. In each CV, the state variables and thermodynamic properties of the HTF are calculated. However, one of the major problems that faced the author in Yebra (2006) was *chattering*.

Chap. 6 presents the FVM, demonstrating particular implementations by means of conservation equations and Modelica libraries. These libraries define different versions of evaporators – and hence PTCs. It also introduces the chattering problem and a number of approaches to avoiding it. Chap. 7 presents a different discretization method to model the evaporator, which considers a low number of CVs (one for each flow region) but with variable lengths (the moving boundary method) instead of considering a high number of CVs with fixed lengths (the finite volume method). MBMs explicitly apply discretization, avoiding phase changes within the CVs but finding them in the boundaries, which avoids discontinuities in certain HTF thermodynamic properties present in finite volume models, as will be shown in Chap. 6. Furthermore, although MBMs are low-order models, they can describe the dynamic behavior of evaporators and condensers accurately (Bendapudi et al., 2008).

5.6 Summary and Conclusions

This chapter has described the developed Modelica DISS model and its main components in terms of inputs, outputs and parameters. PTC and PTCN components, as well as thermal losses models, have been also detailed. The evaporator model has been discussed, with an emphasis on the need for an applied CFD in order to select a discretization method for transforming the PDE system in a numerically solvable – with the present M&S tool – DAE system. The two discretization approaches selected, the FVM and MBMs, will be discussed in Chaps. 6 and 7, respectively.

The classification of discretized two-phase flow models has been reviewed – the HEM seems to be suitable because high pressure in the DISS facility makes the slip ratio close to 1. The simplified two-phase flow HEM also contributes to performing fast dynamic simulations, which are advisable for the design of advanced ACSs.

5.7 Related Scientific Contributions

The scientific contributions related to this chapter are enumerated and briefly commented on in the following list. These contributions were previously listed in § 1.4.

9. Bonilla, J. Modeling and Simulation of Solar Thermal Power Plants (in Spanish). In *XXXI Jornadas de Automática 2010, Modeling and Simulation Group*, Invited oral communication, September 8 - 10, 2010, Jaén, Spain.
EOO modeling of the DISS solar thermal power plant.

12. Yebra, L. J., Berenguel, M., Bonilla, J., Roca, L., Dormido, S. and Zarza, E. Object-oriented modeling and simulation of ACUREX solar thermal power plant. *Mathematical and Computer Modeling of Dynamical Systems*, vol. 16(3), pages 211–224, 2010. ISSN 1387-3954.
EOO modeling of parabolic-trough collectors.

13. Yebra, L. J., Berenguel, M., Bonilla, J., Roca, L., Dormido, S. and Zarza, E. Object-oriented modeling and simulation of ACUREX solar thermal power plant. In *Proceedings of the 6th Vienna International Conference on Mathematical Modeling (MATHMOD)*, oral communication, pages 2025–2033. 2009.
EOO modeling of parabolic-trough collectors.

14. Bonilla, J., Roca, L., Yebra, L. J. and Dormido, S. Real-Time Simulation of CESA-I Central Receiver Solar Thermal Power Plant. In *Proceedings of the 7th International Modelica Conference*, oral communication, pages 345–353. Linköping University Electronic Press, Como, Italy, 2009.
EOO modeling of solar concentrating systems.

15. Bonilla, J., Roca, L., González, J. and Yebra, L. J. Modelling and real-time simulation of heliostat fields in central receiver plants. In *Proceedings of the 6th Vienna International Conference on Mathematical Modelling (MATHMOD)*, poster, pages 2576–2579, 2009.
EOO modeling of solar concentrating systems.

16. de la Calle, A., Roca, L., Bonilla, J., Yebra, L. J. Modeling and simulation of a submerged evaporator (in Spanish). In *XXXII Jornadas de Automática*, poster, Seville, Spain, 2011.
EOO modeling of dynamical systems.

IV

Computational Fluid Dynamics

6
Finite Volume Method

Divide et impera (Divide and conquer).

Julius Caesar, *creator and ruler of the Roman Empire*, 100 - 44 B.C.

Abstract: This chapter introduces the mass, energy and momentum conservation laws for a one-dimensional horizontal evaporator. The FVM is introduced and applied to discretize such balance equations. Details about particular Modelica implementations of finite volume models are also given. The chattering problem which arises in finite volume models is discussed, and details are provided regarding solutions to this practical problem and the particular approaches implemented to tackle it accordingly.

6.1 Basics of Finite Volume Methods

As previously stated in § 5.3, PDE systems can be transformed to solvable systems for present M&S tools applying CFD. The FVM is one of these discretization schemes.

In a horizontal evaporator, which was previously introduced in § 5.3, the concern is the motion of the fluid, which is usually modeled by the conservation laws of mass, energy and momentum, establishing balances between regions or volumes. The FVM has good properties with regards to maintaining these conserved quantities (Skoglund et al., 2006).

Considering the case of a quasi one-dimensional horizontal evaporator, its volume can be divided horizontally in CVs, as shown in Fig. 6.1. Each CV is denoted by V_j, with n being the number of the CVs, hence $j \in [1, n]$. The horizontal spatial coordinate is z.

Specific enthalpy (h) and pressure (p) are commonly the choice of state variables when modeling thermo-hydraulic systems (cf. § 4.3.3.1), and both

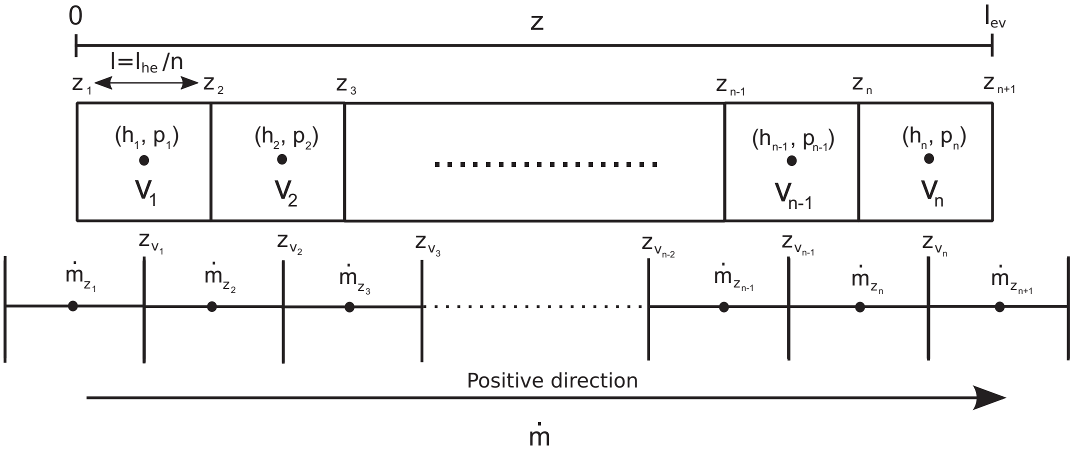


Figure 6.1: The finite volume method and the staggered grid scheme applied to a one-dimensional horizontal evaporator

state variables are represented in each CV by (h_j, p_j). Two discretization grids are shown in Fig. 6.1. The one at the top denotes the discretization grid for the conservation laws of mass and energy, whereas the one at the bottom is the discretization grid applied to the momentum balance equation. Therefore, fluxes (mass flow rates) are calculated between mass and energy in the CVs. This layout scheme is known as a 'staggered grid scheme' (Harlow and Welch, 1965).

On a staggered grid, the scalar variables (pressure, density, specific enthalpy, etc.) are calculated at CV centers, whereas velocity or momentum variables are located in the CV boundaries. The main disadvantage of staggered storage, which is used mainly on structured grids, is that different variables are stored in different places, and this makes it more difficult to handle the storage of the variables. The staggered grid scheme claims to provide better convergence properties through the better approximation of the pressure gradient (Jensen, 2003). The motivation for using it is mainly for numerical reasons.

In a collocated grid arrangement (Rhie and Chow, 1983) all variables are stored in the same positions. The collocated grid approach might cause unreal pressure fields (checkerboard and zig-zag), which arise when SIMPLE-type pressure-velocity coupling is used (Meier et al., 1999).

Fig. 6.1 shows the FVM with a staggered grid scheme. To clarify nomenclature, when a certain variable ψ is calculated in a CV center, e.g. in a V_j center (z_{v_j}), it is denoted by ψ_j. On the other hand, if this variable is calculated at a CV boundary, it is denoted by ψ_{z_j} or $\psi_{z_{j+1}}$ for the left (z_j) or right (z_{j+1}) boundaries, respectively. In order to calculate fluxes at the boundaries, additional boundary values are also required. As such, these values must be calculated or estimated at the boundaries in some way. Two common approaches are applied to calculations when considering basic discretization schemes: the Central Differencing Scheme (CDS) and the UDS (Courant et al., 1952). Basic discretization schemes also include the Hybrid Differencing Scheme (HDS) (Spalding, 1972) and the Power-Law Differencing Scheme (PLDS) (Patankar, 1980). Besides basic discretization schemes, there are also High Resolution Schemes (HRSs) (Waterson, 1994; Waterson and Deconinck, 1995).

In the CDS, also known as linear interpolation, a particular thermodynamic property is calculated at the *momentum* CV center. For example, in order to compute $\dot{m}_{z_j}$, it might be necessary to calculate ψ_{z_j}, which in this case would be an average value of its values in the center of V_{j-1} and V_j, cf. Fig. 6.1. On the other hand, the UDS, also known as First-Order Upwind (FOU), uses an adaptive approach to more properly numerically simulate the direction of information propagation in a flow field. In the example previously mentioned, if the flow direction is positive (cf. Fig. 6.1), the value of ψ_{z_j} would be its value in the V_{j-1} center (ψ_{j-1}); otherwise, it would be its value in the V_j center (ψ_j), thus adapting its value to the flow direction.

6.2 One-dimensional Governing Equations

Considering the evaporator as a one-dimensional volume, the time-dependent PDEs of the conservation laws – mass, energy and momentum – can be expressed by Eqs. 6.1, 6.2 and 6.3, respectively (Anderson, 1995; Thomas, 1999). Where P denotes power, $\dot{Q}$ denotes the heat flow rate exchanged with its surroundings, while F_{fg} denotes forces exerted by friction and gravity. The summation terms denote that all possible contributions are included.

$$\frac{\partial(\rho A)}{\partial t} + \frac{\partial(\rho A w)}{\partial z} = 0, \tag{6.1}$$

$$\frac{\partial(\rho u A)}{\partial t} + \frac{\partial\left(\rho w\left(u + \frac{p}{\rho}\right)A\right)}{\partial z} = \sum_c \frac{\partial P_c}{\partial z} + \sum_d \frac{\partial \dot{Q}_d}{\partial z}, \tag{6.2}$$

$$\frac{\partial(\rho w A)}{\partial t} + \frac{\partial(\rho w^2 A)}{\partial z} = -A\frac{\partial p}{\partial z} - \frac{\partial F_{fg}}{\partial z}. \tag{6.3}$$

6.3 Discretization of the Governing Equations

Integrating V_j into the horizontal spatial coordinate (z) using the FVM and the staggered grid scheme (cf. Fig. 6.1), and assuming a constant cross-sectional area (A), the mass, energy and momentum equations, Eqs. 6.1, 6.2

and 6.3, transform into Eqs. 6.4, 6.5 and 6.6.

$$\int_{z_j}^{z_{j+1}} \frac{\partial(\rho A)}{\partial t} = \rho_{z_j} w_{z_j} A - \rho_{z_{j+1}} w_{z_{j+1}} A, \tag{6.4}$$

$$\int_{z_j}^{z_{j+1}} \frac{\partial(\rho u A)}{\partial t} + \rho_{z_{j+1}} w_{z_{j+1}} A \left(u_{z_{j+1}} + \frac{p_{z_{j+1}}}{\rho_{z_{j+1}}} \right) - \rho_{z_j} w_{z_j} A \left(u_{z_j} + \frac{p_{z_j}}{\rho_{z_j}} \right) = \\ = \sum_c P_{c,j} + \sum_d \dot{Q}_{d,j}, \tag{6.5}$$

$$\int_{z_{v_{j-1}}}^{z_{v_j}} \frac{\partial(\rho w A)}{\partial t} + \rho_j w_j^2 A - \rho_{j-1} w_{j-1}^2 A = A(p_{j-1} - p_j) - F_{fg_{z_j}}. \tag{6.6}$$

Applying the Leibniz integral rule (Woods, 1926) (cf. Appx. A.1) to the first term in Eqs. 6.4, 6.5 and 6.6, with a fixed CVs ($dz_j/dt = 0$) and, consequently, a constant CV length (l), produces Eqs. 6.7, 6.8 and 6.9.

$$\frac{d(\rho_j A l)}{dt} = \rho_{z_j} w_{z_j} A - \rho_{z_{j+1}} w_{z_{j+1}} A, \tag{6.7}$$

$$\frac{d(\rho_j u_j A l)}{dt} + \rho_{z_{j+1}} w_{z_{j+1}} A \left(u_{z_{j+1}} + \frac{p_{z_{j+1}}}{\rho_{z_{j+1}}} \right) - \rho_{z_j} w_{z_j} A \left(u_{z_j} + \frac{p_{z_j}}{\rho_{z_j}} \right) = \\ = \sum_c P_{c,j} + \sum_d \dot{Q}_{d,j}, \tag{6.8}$$

$$\frac{d(\rho_{z_j} w_{z_j} A)}{dt} l + \rho_j w_j^2 A - \rho_{j-1} w_{j-1}^2 A = A(p_{j-1} - p_j) - F_{fg_{z_j}}. \tag{6.9}$$

Eqs. 6.7, 6.8 and 6.9 hold for each CV, except for the boundary CVs, i.e. V_1 and V_n, where particular equations may be required, depending on the particular implementation. This will be discussed in § 6.4. It is then assumed in this section that $j \in [2, n-1]$, where $n \geq 3$.

6.3.1 Mass Balance Equation

Considering, $m_j = \rho_j A l$ and $\dot{m}_{z_j} = \rho_{z_j} w_{z_j} A$, mass balance Eq. 6.7 can be expressed as Eq. 6.10:

$$\frac{dm_j}{dt} = \dot{m}_{z_j} - \dot{m}_{z_{j+1}}. \tag{6.10}$$

6.3.2 Energy Balance Equation

Considering internal energy $U_j = \rho_j u_j A l$, mass flow rate $\dot{m}_{z_j} = \rho_{z_j} w_{z_j} A$ and specific enthalpy $h_{z_j} = u_{z_j} + p_{z_j}/\rho_{z_j}$, Eq. 6.8 can be expressed as Eq. 6.11

$$\frac{dU_j}{dt} = \dot{m}_{z_j} h_{z_j} - \dot{m}_{z_{j+1}} h_{z_{j+1}} + \sum_c P_{c,j} + \sum_d \dot{Q}_{d,j}, \tag{6.11}$$

finally, defining the enthalpy flow rate as $\dot{H}_{z_j} = \dot{m}_{z_j} h_{z_j}$ (Elmqvist et al., 2003) yields Eq. 6.12:

$$\frac{dU_j}{dt} = \dot{H}_{z_j} - \dot{H}_{z_{j+1}} + \sum_c P_{c,j} + \sum_d \dot{Q}_{d,j}. \tag{6.12}$$

6.3.3 Momentum Balance Equation

Considering mass flow rate $\dot{m}_{z_j} = \rho_{z_j} w_{z_j} A$, Eq. 6.9 transforms into Eq. 6.13:

$$\frac{d\dot{m}_{z_j}}{dt} l = \rho_{j-1} w_{j-1}^2 A - \rho_j w_j^2 A + A(p_{j-1} - p_j) - F_{fg_{z_j}}. \tag{6.13}$$

6.4 Modelica Libraries

Instead of developing and implementing a new one-dimensional finite volume evaporator Modelica model from scratch, a state-of-the-art study was performed in order to identify possible models and libraries which could be used instead. This section provides details about two free Modelica libraries which focus on the design of thermo-hydraulic systems: the *ThermoFluid* library and the *Modelica Fluid* library. Another recent free Modelica library devoted to thermo-hydraulic systems is the *ThermoSysPro* library (El Hefni and Bouskela, 2012). It is also worth mentioning the *ThermoPower* library (Casella and Leva, 2003, 2006), which is currently being updated to support Modelica 3.

6.4.1 ThermoFluid Library

The *ThermoFluid* library is based mainly on contributions from the Ph.D. dissertations of Tummescheit (2002), Eborn (2001) and Wagner (2000). This library provides a framework and basic components for modeling thermo-hydraulic and process systems in Modelica, and it focuses on models of compressible, homogeneous one-phase and two-phase flows . Most of the models are discretized in one dimension or in lumped parameter approximations. The *Thermofluid* library was initially used in the first version of the DISS Modelica model (Yebra, 2006).

6.4.1.1 Governing Equations of the ThermoFluid Finite Volume Evaporator Model

This section considers the particular implementation of the *ThermoFluid* library for a finite volume evaporator model. The inlet and outlet specific enthalpies' (h_a, h_b), pressures' (p_a, p_b) and mass flow rates' $(\dot{m}_a, \dot{m}_b)$ locations are detailed in Fig. 6.2. *ThermoFluid* implementation is based on Eqs. 6.10, 6.12 and 6.13.

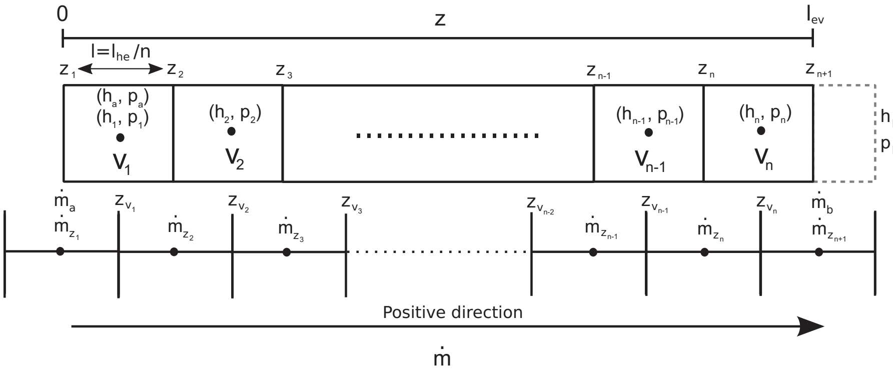


Figure 6.2: Finite volume method and staggered grid scheme applied to a one-dimensional horizontal evaporator in the *ThermoFluid* library

Mass Balance Equation. The mass balance equation is defined by Eq. 6.10, but it has been repeated here for completeness.

$$\frac{dm_j}{dt} = \dot{m}_{z_j} - \dot{m}_{z_{j+1}}, \qquad \forall j \in [1, n]. \tag{6.14}$$

Energy Balance Equation. The energy balance equation in the *ThermoFluid* library is Eq. 6.15, where pressure-volume work (P_p) and dissipative work (W_{diss}), together with heat transfer $(\dot{Q}_{ht})$, are considered. However, other possible contributions such as heat from reactions and kinetic and potential energy terms are neglected in the *ThermoFluid* library.

$$\frac{dU_j}{dt} = \dot{H}_{z_j} - \dot{H}_{z_{j+1}} + \dot{Q}_{ht_j} + P_{p_j} - W_{diss_j}, \tag{6.15}$$

Pressure-volume work is $P_{p_j} = -p_j dV_j/dt$ and dissipative work is zero by default, although the modeler can modify this as necessary. These two considerations lead to Eq. 6.16:

$$\frac{dU_j}{dt} = \dot{H}_{z_j} - \dot{H}_{z_{j+1}} + \dot{Q}_{ht_j} - p_j \frac{dV_j}{dt}, \qquad \forall j \in [2, n]. \tag{6.16}$$

The *ThermoFluid* library implements ideal heat transfer between the HTF and the pipe wall, although other heat transfer models, inheriting from a partial base class, can be implemented by the modeler.

Momentum Balance Equation. Considering mass flow rate $\dot{m}_j = \rho_j w_j A$ in the first two terms of the Right-Hand Side (RHS) of Eq. 6.13, friction force $F_{f_{z_j}} = (1/2)\rho_{z_j} w_{z_j} |w_{z_j}| f_{f_{z_j}} S_c l$ (Elmqvist et al., 2003) and gravitational force $F_{g_j} = A\rho_{z_j} g \Delta y_{z_j}$ yield Eq. 6.17.

$$\frac{d\dot{m}_{z_j}}{dt} l = \frac{\dot{m}_{j-1}^2}{A\rho_{j-1}} - \frac{\dot{m}_j^2}{A\rho_j} + A(p_{j-1} - p_j) - \frac{1}{2}\rho_{z_j} w_{z_j} |w_{z_j}| f_{f_{z_j}} S_c l - A\rho_{z_j} g \Delta y_{z_j}. \tag{6.17}$$

Considering the hydraulic diameter $D_h = 4A/S_c$, the moody friction factor $f_{m_j} = 4f_{f_j}$ and vertical displacement $\Delta y = l \sin(\alpha_i)$, where α_i is the inclination angle over the horizontal plane, Eq. 6.17 transforms in Eq. 6.18.

$$\frac{d\dot{m}_{z_j}}{dt} l = \frac{\dot{m}_{j-1}^2}{A\rho_{j-1}} - \frac{\dot{m}_j^2}{A\rho_j} + A(p_{j-1} - p_j) - \frac{1}{2}\frac{1}{\rho_{z_j} A D_h} \dot{m}_{z_j}^2 f_{m_{z_j}} l - A\rho_{z_j} g l \sin(\alpha_i). \tag{6.18}$$

Dividing Eq. 6.18 by A, considering a static momentum balance where the mass flow rate time derivative $d\dot{m}_{z_j}/dt$ is neglected, and also neglecting momentum fluxes $\dot{m}_j^2/(A\rho_j)$, yields Eq. 6.19.

$$0 = (p_{j-1} - p_j) - \frac{1}{2}\frac{1}{\rho_{z_j} A^2 D_h} \dot{m}_{z_j}^2 f_{m_{z_j}} l - \rho_{z_j} g l \sin(\alpha_i) \qquad \forall j \in [2, n]. \tag{6.19}$$

An additional momentum balance equation is required for the $n+1$ momentum CV (cf. Fig. 6.2).

$$0 = (p_n - p_b) - \frac{1}{2}\frac{1}{\rho_{z_{n+1}} A^2 D_h} \dot{m}_{z_{n+1}}^2 f_{m_{z_{n+1}}} l - \rho_{z_{n+1}} g l \sin(\alpha_i). \tag{6.20}$$

Although the *ThermoFluid* library has different pressure loss models: the simple pressure model, the simple distributed pressure model, the pressure loss model driven by a total friction factor, the pressure loss model driven by a Moody friction factor and Blasius' pressure loss correlation (Tummescheit, 2002), a static momentum balance (Eqs. 6.19 and 6.20) was considered in the first version of the DISS Modelica model. This simplification was performed because the main dynamics of interest were thermal dynamics (Yebra, 2006).

Additional Information. If the left and right boundary scalar and flux variables are $p_a, h_a, \dot{m}_a$ and $p_b, h_b, \dot{m}_b$, respectively, the following relations are considered in the *ThermoFluid* implementation.

$$\begin{aligned} &p_a = p_1, \quad h_a = h_1, \quad \dot{m}_a = \dot{m}_{z_1}, \quad \dot{H}_a = \dot{H}_{z_1}, \\ &\dot{m}_b = \dot{m}_{z_{n+1}}, \quad \dot{H}_b = \dot{H}_{z_{n+1}}. \end{aligned} \tag{6.21}$$

As previously mentioned in § 6.1, intensive properties are calculated in the V_j center. In order to approximate the value of $\dot{H}_{z_j}$ for the energy balance equation, a UDS is assumed, and it is defined by Eq. 6.22. The special case for $\dot{H}_{z_{n+1}}$ is given by Eq. 6.23. The expression for ρ_{z_j} is also given by Eq. 6.24.

$$\dot{H}_{z_j} = \begin{cases} \dot{m}_{z_j} \cdot h_{j-1} & \dot{m}_{z_j} \geq 0 \\ \dot{m}_{z_j} \cdot h_j & \dot{m}_{z_j} < 0 \end{cases} \qquad \forall j \in [2, n], \tag{6.22}$$

$$\dot{H}_{z_{n+1}} = \begin{cases} \dot{m}_{z_{n+1}} \cdot h_n & \dot{m}_{z_{n+1}} \geq 0 \\ \dot{m}_{z_{n+1}} \cdot h_b & \dot{m}_{z_{n+1}} < 0, \end{cases} \tag{6.23}$$

$$\rho_{z_j} = \rho_{j-1}, \qquad \forall j \in [2, n]. \tag{6.24}$$

6.4.2 Modelica Fluid library

The development of the free *Modelica Fluid* library was a collaborative effort which started in 2002, initially organized by Martin Otter and since 2004 organized by Francesco Casella. Seventeen people contributed to the development of the *Modelica Fluid* library over six years. Some of the most remarkable researchers involved in this project were: Rüdiger Franke, Franceso Casella, Michael Sielemann, Katrin Prölß, Martin Otter, Michael Wetter, Jonas Eborn, Hilding Elmqvist, Manuel Gräber, Carsten Heinrich, Kilian Link, Christoph Richte and Hubertus Tummescheit among others.

The main goal of the *Modelica Fluid* library was to standardize the modeling of thermo-fluid systems and to become a part of the MSL as a framework to simplify the development of thermo-fluid systems, facilitating model exchanges and the sharing of knowledge. A first version of this library, called *Modelica_ Fluid*, was presented in 2006 (Casella et al., 2006), following which its redesign continued through the addition of even features to the Modelica language, e.g. the stream connector (Franke et al., 2009a). This version of the library received its present name, the *Modelica Fluid* library (Franke et al., 2009b).

The *Modelica Fluid* library provides one-dimensional thermo-fluid flows. Among its features, stream connectors, rigorous steady-state and dynamic implementations of mass, energy and momentum balances, decoupled pressures loss, heat transfer correlations and friction models are worth mentioning. All components are compatible with media from the *Modelica Media* library, which means that incompressible or compressible media, single or multiple substance media with one or more phases, as long as the flow is homogeneous, might be used.

The *Modelica Fluid* library was adopted in the development of the models in this work at the end of 2011. Although the final version of this library was released in mid-2009, it was necessary to acquire new versions of commercial tools (in this case Dymola) because free Modelica M&S tools did not support the complete flattening and simulation of models in the *Modelica Media* and *Modelica Fluid* libraries at the time of writing this book. For this reason, the *ThermoFluid* library was used extensively, which allowed for improving it and overcoming inherent problems, as will be detailed in § 6.6.

6.4.2.1 Governing Equations of the Modelica Fluid Finite Volume Evaporator Model

This section presents the particular mass, energy and momentum equations implemented in the *Modelica Fluid* library. This library is powerful and considers aspects such as multiple substance media, trace substances and multiple identical parallel evaporators, which are beyond the scope of this book. Hence, these aspects will be omitted.

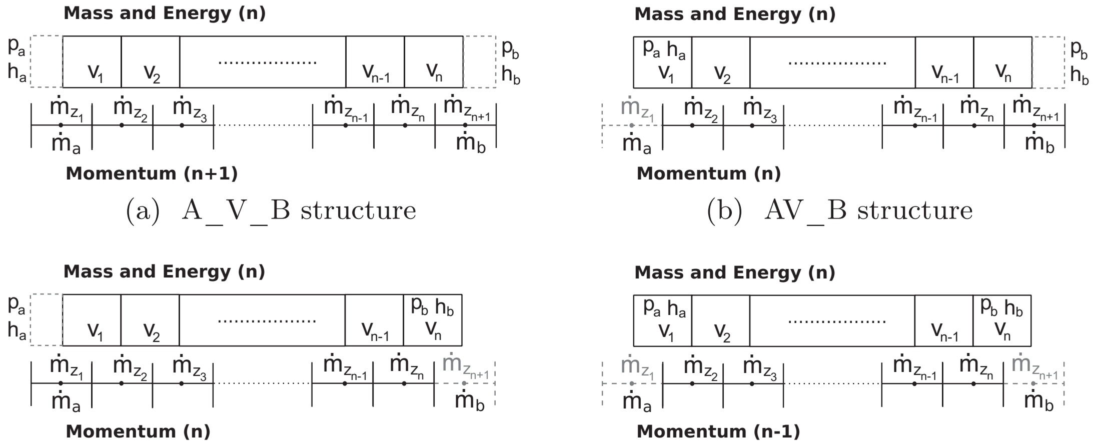
It is important to note that the evaporator model in the *Modelica Fluid* library is highly customizable, and it even allows a modeler to change its structure. Fig. 6.3 shows the four different configurations of discretization grids in the *Modelica Fluid* library when applying the FVM and the staggered grid scheme. In the different mass and energy discretization grids, the inlet and outlet specific enthalpies and pressures are located accordingly. In the momentum discretization grid, the missing CVs are denoted by gray dashed lines, cf. Fig. 6.3. The different configurations are briefly explained in the following paragraphs (Franke et al., 2009a).

- *A_ V_ B*. Symmetric configuration with the n CVs in the mass and energy grid and the $n+1$ CVs in the momentum grid. This configuration results in high index DAE systems because both ports, the inlet and outlet ports, expose their thermodynamic states.

- *AV_ B*. Asymmetric configuration with the n CVs in both grids. The last mass and energy grid CV is placed between the $n-1^{th}$ and n^{th} momentum CVs, which causes as potential state variable p_a.

- *A_ VB*. Asymmetric configuration with the n CVs in both grids. The first mass and energy grid CV is placed between the 1^{st} and 2^{nd} momentum CVs, which causes as potential state variable p_b.

- *AV_ VB*. Symmetric configuration with the n CVs in the mass and energy grid and the $n-1$ CVs in the momentum grid. Good start values are required for p_a and p_b in order to solve large nonlinear systems, because both pressures are algebraic variables.



(a) A_V_B structure

(b) AV_B structure

(c) A_VB structure

(d) AV_VB structure

Figure 6.3: Discretization grids for mass, energy and momentum balances in the evaporator model of the *Modelica Fluid* library

Mass Balance Equation. The dynamic mass balance equation is Eq. 6.10, repeated here for completeness:

$$\frac{dm_j}{dt} = \dot{m}_{z_j} - \dot{m}_{z_{j+1}}, \qquad \forall j \in [1, n], \tag{6.25}$$

the *Modelica Fluid* library also includes a steady-state mass balance, Eq. 6.26.

$$\dot{m}_{z_j} = \dot{m}_{z_{j+1}}, \qquad \forall j \in [1, n], \tag{6.26}$$

additionally, the following equations are considered for the boundaries:

$$\dot{m}_a = \dot{m}_{z_1}, \qquad \dot{m}_b = \dot{m}_{z_{n+1}}. \tag{6.27}$$

Energy Balance Equation. The *Modelica Fluid* library considers heat transfer and pressure loss in the energy balance equation (Eq. 6.12), which produces Eq. 6.28 (Franke et al., 2009b).

$$\frac{dU_j}{dt} = \dot{H}_{z_j} - \dot{H}_{z_{j+1}} + \dot{Q}_{ht_j} + w_j A \Delta p_j, \qquad \forall j \in [2, n-1]. \tag{6.28}$$

Implementing the energy balance equation depends on the grid structure. There are also particular equations for the first and last CVs. In the steady-state version, the dU_j/dt term is substituted by zero. The *Modelica Fluid* library implements ideal heat transfer, heat transfer driven by a heat transfer coefficient and an empirical Heat Transfer Correlation (HTC) (Verein Deutscher Ingenieure, 1997). Additionally, other heat transfer models, inheriting from a partial base class, can be implemented by the modeler.

The same scenario occurs with pressure loss, no friction, laminar, quadratic turbulent, laminar and quadratic turbulent, and a detailed model can be selected (Franke et al., 2009b). Other pressure loss models can be also implemented.

Momentum Balance Equation. Eq. 6.29 is the same equation as Eq. 6.13, although it has been included here for completeness. The F_{fg} term takes into account friction and gravity. *Modelica Fluid* implements several models for pressure loss caused by wall friction. Furthermore, the modeler can implement additional models inheriting from a partial base class.

$$\frac{d\dot{m}_{z_j}}{dt} l = \rho_j w_j^2 A - \rho_{j-1} w_{j-1}^2 A + A(p_{j-1} - p_j) - F_{fg_{z_j}}. \tag{6.29}$$

Additional Information. If the left and right boundary specific enthalpies are h_a and h_b, respectively, the following relations are considered in the *Modelica Fluid* implementation of the one-dimensional evaporator model.

$$h_a = h_1, \quad h_b = h_n. \tag{6.30}$$

If the grid structure is AV_VB or AV_B, Eq. 6.31 is also considered:

$$p_a = p_1. \tag{6.31}$$

On the other hand, if the grid structure is AV_VB or A_VB, Eq. 6.32 is included:

$$p_b = p_n. \tag{6.32}$$

The UDS for calculating the density terms (ρ_{z_j}) and the energy equation ($\dot{H}$ terms) is taken into account by Eqs. 6.33 and 6.34. The special cases for ρ_{z_1}, $\dot{H}_{z_1}$ and $\dot{H}_{z_{n+1}}$ are described by Eqs. 6.35, 6.36 and 6.37, respectively. In Eqs. 6.36 and 6.37, the Modelica *inStream* operator applied to a *stream* variable provides the value between brackets for the inlet flow (Franke et al., 2009a).

$$\rho_{z_j} = \begin{cases} \rho_{j-1} & \dot{m}_{z_j} \geq 0 \\ \rho_j & \dot{m}_{z_j} < 0 \end{cases} \quad \forall j \in [2, n], \tag{6.33}$$

$$\dot{H}_{z_j} = \begin{cases} \dot{m}_{z_j} \cdot h_{j-1} & \dot{m}_{z_j} \geq 0 \\ \dot{m}_{z_j} \cdot h_j & \dot{m}_{z_j} < 0 \end{cases} \quad \forall j \in [2, n], \tag{6.34}$$

$$\rho_{z_1} = \begin{cases} \rho_a & \dot{m}_{z_1} \geq 0 \\ \rho_1 & \dot{m}_{z_1} < 0 \end{cases}, \tag{6.35}$$

$$\dot{H}_{z_1} = \begin{cases} \dot{m}_{z_1} \cdot inStream(h_a) & \dot{m}_{z_1} \geq 0 \\ \dot{m}_{z_1} \cdot h_1 & \dot{m}_{z_1} < 0 \end{cases}, \tag{6.36}$$

$$\dot{H}_{z_{n+1}} = \begin{cases} \dot{m}_{z_{n+1}} \cdot inStream(h_b) & \dot{m}_{z_{n+1}} \geq 0 \\ \dot{m}_{z_{n+1}} \cdot h_n & \dot{m}_{z_{n+1}} < 0 \end{cases}. \tag{6.37}$$

6.5 Heat Transfer and Friction Factor Correlations

Several convective heat transfer correlations between the absorber tube and the HTF, together with Friction Factor Correlations (FFCs), have been implemented for the *ThemoFluid* library.

6.5.1 Heat Transfer Correlations

The implemented convective HTCs for evaporation, considering one and two phases, are summarized in Tab. 6.1. Additionally, ideal heat transfer and heat transfer driven by a constant heat transfer coefficient are also available.

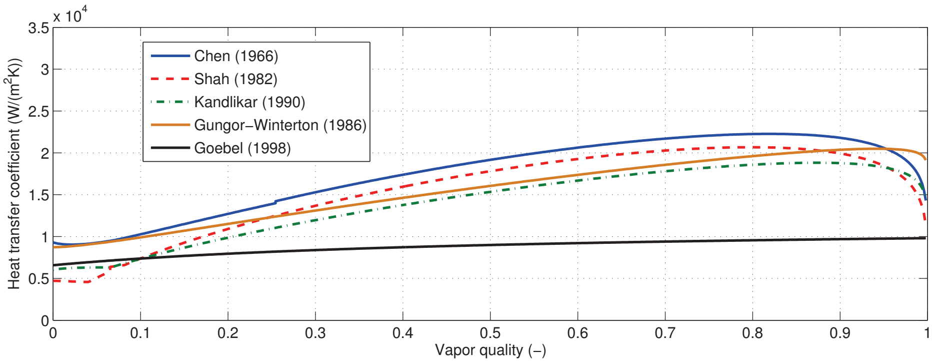


Figure 6.4: Comparison of two-phase flow HTCs ($p = 3\,\text{MPa}$, $\dot{Q}_{ht} = 5827\,\text{kw}$, $\dot{m} = 0.6\,\text{kg/s}$, $\epsilon = 3 \cdot 10^{-5}\,\text{m}$)

These HTCs have been validated against an independent implementation developed in Zarza (2000). The convective HTC can be selected through the GUI, while a test case for the two-phase flow HTCs is shown in Fig. 6.4.

Heat transfer correlation	Fluid phase
Dittus and Boelter (1930)	One-phase
Chen (1966)	Two-phase
Petukhov (1970)	One-phase
Gnielinski (1976)	One-phase
Shah (1982)	Two-phase
Gungor and Winterton (1986)	Two-phase
Kandlikar (1990)	Two-phase
Goebel (1998)	Two-phase

Table 6.1: Some convective HTCs for evaporation

6.5.2 Friction Factor Correlations

Some FFCs for smooth and rough pipes, always considering turbulent flow, have been implemented. They are summarized in Tab. 6.2. Additionally, a constant friction factor can be also used.

Friction factor correlation	Kind of pipe
Karman-Prandtl (1930) (Schlichting and Gersten, 2000)	Rough
Colebrook (1939)	Any
Chen (1979)	Any
Explicit simplified Chen (1979)	Any
Denn (1980)	Smooth
Haaland (1983)	Any

Table 6.2: Some FFCs for turbulent flow

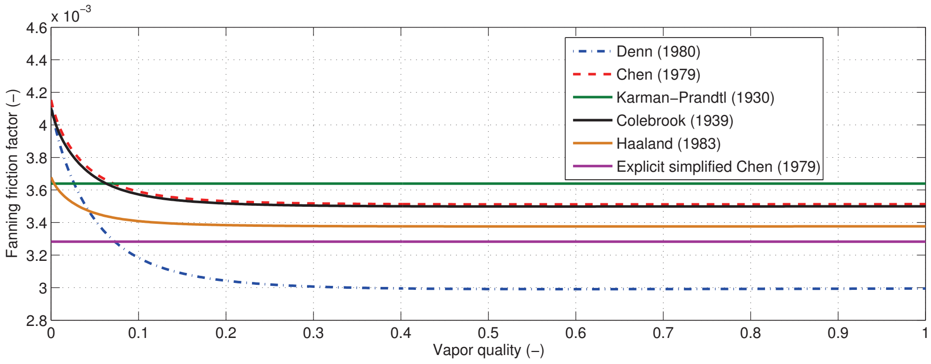


Figure 6.5: Comparison of FFCs ($p = 3\,\mathrm{MPa}$, $\dot{m} = 0.6\,\mathrm{kg/s}$, $\epsilon = 3 \cdot 10^{-5}\,\mathrm{m}$, $L = 1\,\mathrm{m}$, $d = 0.05\,\mathrm{m}$)

The implemented FFCs have been validated against an independent implementation developed in Zarza (2000). They can also be selected through the GUI. Different test cases have been developed in order to validate the implementation, and a test case for the FFCs is shown in Fig. 6.5.

6.6 The Chattering Problem

Chattering, also called 'zenoness', in its general physical sense is defined as an infinite number of events occurring in a finite time period. One of the most remarkable examples is impacting systems, in which an interesting property is the possibility of chattering, for instance a ball bouncing to rest on a horizontal surface (Budd and Dux, 1994).

In M&S terms, the numerical chattering problem, which is a type of cycle limit, may occur when discontinuities in the variables are present during simulation (Jensen, 2003). Discontinuities, although properly modeled by conditional sentences such as if-sentences, generate events. Let us consider a simple example. An if-sentence evaluates a variable and has two different RHSs. The if-sentence is evaluated after an event, and each time it is evaluated it leads the evaluated variable to the other RHSs (the discontinuity) and hence to the if-sentence. This is repeated many times, thus producing a great amount of events (Tummescheit, 2002). It might transpire that after some events the variable is driven out of the discontinuity. However, in this case chattering reduces the computational performance, leading to slow simulations and limiting the applicability of the model. Under certain circumstances, simulations do not progress and instead become unfeasible – in which case the simulations are not completed. The chattering problem must be tackled in order to effectively take advantage of dynamic simulations.

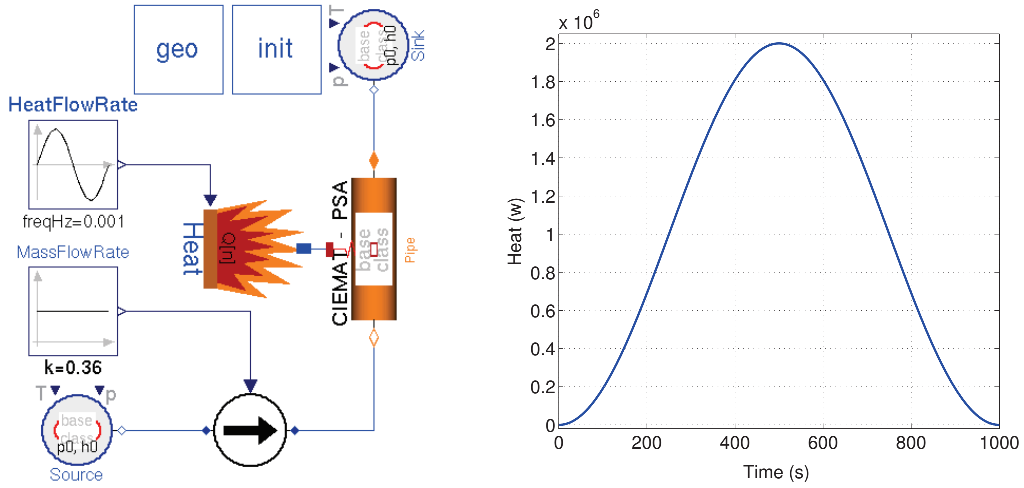
Chattering is a serious problem in simulations using discretized models involving phase changes. The discretization level and numerical integration accuracy influence the occurrence of chattering, but not in a systematic way (Jensen, 2003). None of the current M&S tools treats the chattering problem in a satisfying way, as they neither clarify nor diagnose the occurrence of this problem (Mosterman, 1999). Some remedies consist in smoothing the discontinuities or setting hysteresis intervals (Lee and Utkin, 2007); the reformulation of discretized models seems to be the only way to avoid chattering.

Chattering, although it has rarely been studied, is a well-known problem (Utkin and Lee, 2006). In discretized two-phase flow models, studies have been performed in Tummescheit (2002); Jensen (2003); Casella (2006); Yebra (2006).

6.6.1 Boiler Pipe Model

A simple boiler pipe model, using the *ThermoFluid* library, is presented in this section as an example to focus attention on the chattering problem. For reference, the *Modelica Fluid* library also exhibits chattering in more complex systems, although less commonly (but still appearing), which are mentioned in § 8.2.2.

The Modelica component diagram, which is shown in Fig. 6.6a, has a sinusoidal heat source equally distributed through a pipe discretized into ten CVs. The heat source, in a 1,000-second simulation, can be seen in Fig. 6.6b. The inlet water flow is constant (0.36 l/s) and is pumped through the heated pipe. As the heat increases, the boiling barrier moves from the end of the pipe toward the beginning, and vice versa.



(a) Modelica component diagram (b) Sinusoidal heat source

Figure 6.6: Boiler pipe model

Modelica 2.2.1 (Modelica Association, 2007) and Dymola 6.0b (Dynasim AB, 2006) were used to implement this model, while DASSL (cf. § 2.9.2.1) was the numerical solver used in this simulation and the absolute and relative tolerances were set to 10^{-4}. The simulation statistics are shown in Tab. 6.3.

CPU-time for integration (s)	16.6
Number of state events	$9,281$
Minimum integration stepsize	$5.45 \cdot 10^{-7}$
Maximum integration stepsize	23.2
Maximum integration order	5

Table 6.3: Boiler pipe model simulation statistics

Time instants which trigger events can be identified by analyzing the simulation results, as shown in Tab. 6.4. Each of these time instants, where large numbers of events are triggered, are where chattering appears in the simulation. It can be seen in Tab. 6.4 that chattering appears in phase transitions (in different CVs) that correspond to the saturation curves.

Time (s)	CV	Phase transition
632.96	5	Two-phase flow→Superheated steam
833.08	4	Two-phase flow→Sub-cooled liquid
865.58	5	Two-phase flow→Sub-cooled liquid
890.12	6	Two-phase flow→Sub-cooled liquid
910.97	7	Two-phase flow→Sub-cooled liquid
932.07	8	Two-phase flow→Sub-cooled liquid
958.73	9	Two-phase flow→Sub-cooled liquid

Table 6.4: Chattering time instants in the boiler pipe model

6.6.2 Source of the Chattering Problem

Phase boundaries require special attention, as even continuously differentiable state variable trajectories – pressure and specific enthalpy – lead to small discontinuous density changes with discontinuous partial derivatives.

Considering a simplified model for a CV, where the volume is constant, kinetic and potential energies are neglected. This model is represented by Eq. 6.38 (Yebra, 2006), where it was demonstrated that in order to satisfy the continuity conditions for numerical integration, density (ρ) and its partial derivatives ($\frac{\partial\rho}{\partial h}, \frac{\partial\rho}{\partial p}$) must be continuous.

$$\begin{pmatrix} \dfrac{dp}{dt} \\ \\ \dfrac{dh}{dt} \end{pmatrix} = \frac{c^2}{\rho V} \begin{pmatrix} \rho + h\dfrac{\partial\rho}{\partial h} & -\dfrac{\partial\rho}{\partial h} \\ \\ 1 - h\dfrac{\partial\rho}{\partial p} & \dfrac{\partial\rho}{\partial p} \end{pmatrix} \begin{pmatrix} \dfrac{dm}{dt} \\ \\ \dfrac{dU}{dt} \end{pmatrix}. \tag{6.38}$$

However, the IAPWS-IF97 divides the formulation into regions and thus allows fast calculation, but this can lead to inconsistencies at the region boundaries because there are small discontinuities between regions. These discontinuities have no physical reality but are artifacts of the formulation method (The American Society of Mechanical Engineers, 1999). Furthermore, while usually small, these discontinuities can be troublesome when the solver is trying to find a solution near a region boundary.

These discontinuities can be seen in Fig. 6.8 and Fig. 6.9, which show $\frac{\partial\rho}{\partial h}$ and $\frac{\partial\rho}{\partial p}$ around the saturation curves in two (at the top) and three (at the bottom) dimensions$^{(a)}$. These figures were obtained from the *Modelica Media* library, which follows the IAPWS-IF97 standard. Although discontinuities also appear in density, the discontinuity magnitude is so small in comparison with the density magnitude that it cannot be seen easily in the density surface; cf. Fig. 6.10.

[a] Discontinuous peaks are caused by the sampling of thermodynamic properties. Discontinuities exist throughout all the saturation curves.

Therefore, changes to fluid phases must be implemented using crossing functions$^{(b)}$ (Cellier, 1991), because the IAPWS-IF97 divides the formulation in regions where discontinuities arise in thermodynamic properties. This step is required for robustness and efficiency, but it may also lead to chattering (Tummescheit, 2002), as will be detailed.

Considering the simplified CV model, when the phase boundaries are crossed, only the RHS of Eq. 6.38 changes discontinuously, as the value of the states before and after the event remain the same, they are C^0 continuous (Tummescheit, 2002). This is a consequence of having two different RHSs in Eq. 6.38. Each RHS under certain circumstances sets a vector field with a gradient that drives the solution towards the discontinuity (the saturation curves).

Fig. 6.7 shows a sketch of this behavior, where two different gradients are shown in each phase, the one-phase and two-phase (discontinuous line arrows). State variables (p, h) (solved for each simulation step) are driven toward the discontinuity, the phase change curve, in this case the specific enthalpy of the saturated liquid curve (h').

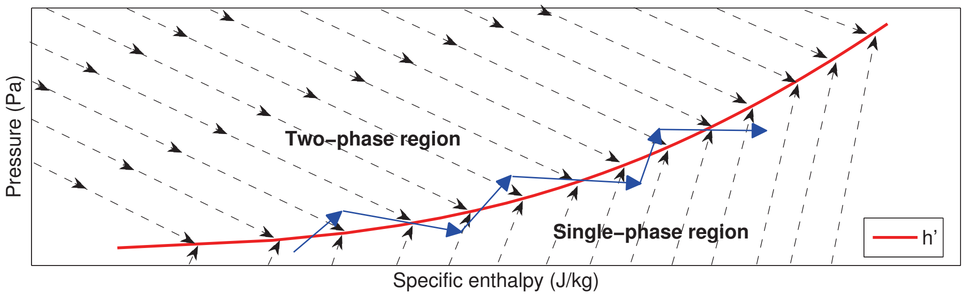


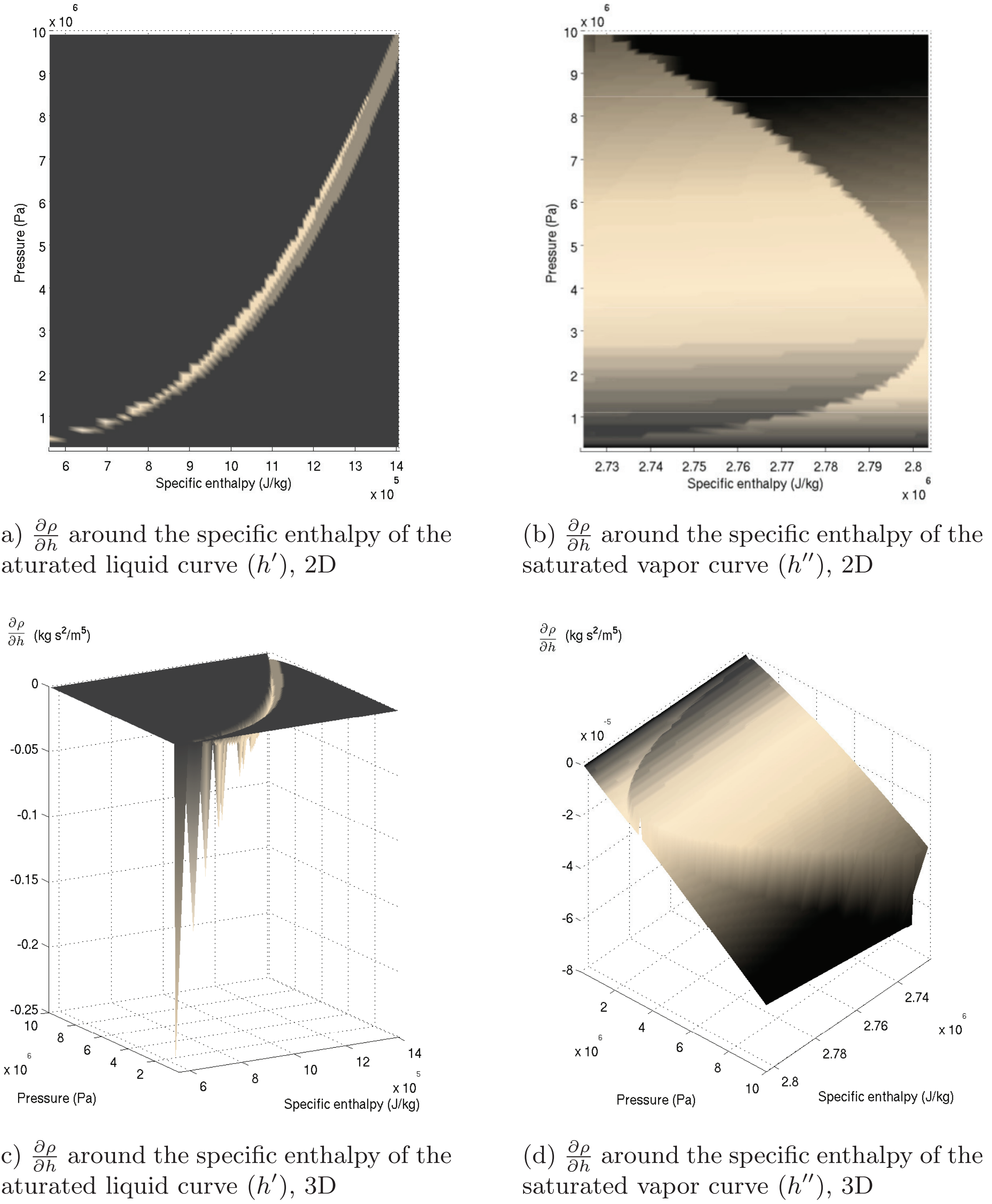
Figure 6.7: Chattering sketch

The trajectory of the state variables is represented by continuous line arrows which describe a path with many phase changes due to confrontation between both vector fields. This confrontation does not always happen, as chattering does not always appear when the state variables cross the phase boundaries, which occurs because the mass flow rate also affects it.

Density influences several terms in momentum balance, such as the mass flow rate changing with a phase change. Mass flow rate changes influence the RHS of Eq. 6.38, which in turn affects state variable (p,h) calculation and once again influences density calculation (Tummescheit, 2002).

Phase change discontinuities are managed by the solver using crossing functions which restart the integration process. This process is computationally expensive, and it is repeated until one of the vector fields sets a gradient which drives the solution outside the discontinuous region; this process increases simulation time considerably.

[b] Crossing functions are generated automatically by the Modelica tool from the conditional sentences.



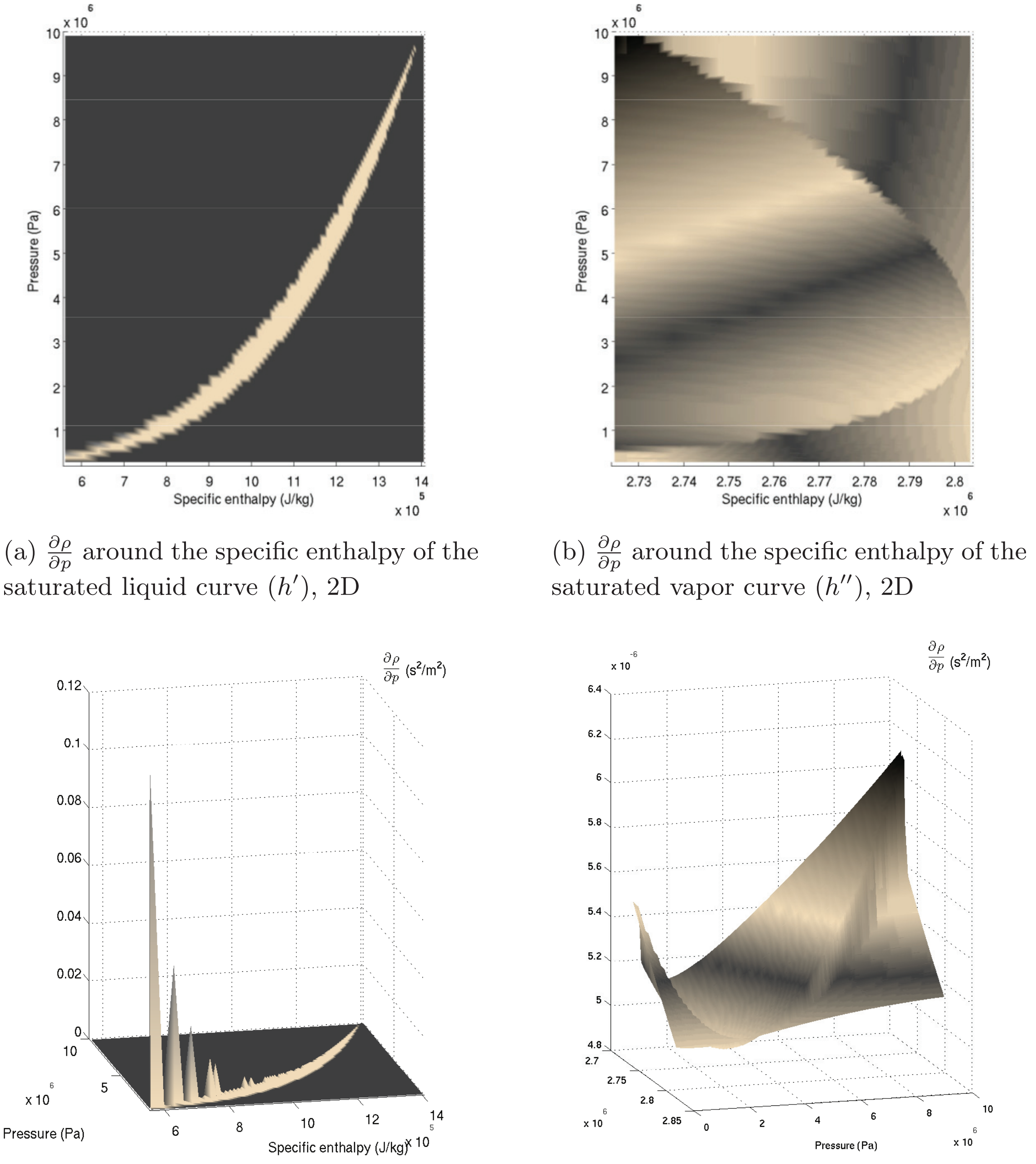
(a) $\frac{\partial\rho}{\partial h}$ around the specific enthalpy of the saturated liquid curve (h'), 2D

(b) $\frac{\partial\rho}{\partial h}$ around the specific enthalpy of the saturated vapor curve (h''), 2D

(c) $\frac{\partial\rho}{\partial h}$ around the specific enthalpy of the saturated liquid curve (h'), 3D

(d) $\frac{\partial\rho}{\partial h}$ around the specific enthalpy of the saturated vapor curve (h''), 3D

Figure 6.8: $\frac{\partial\rho}{\partial h}$ around the saturation curves



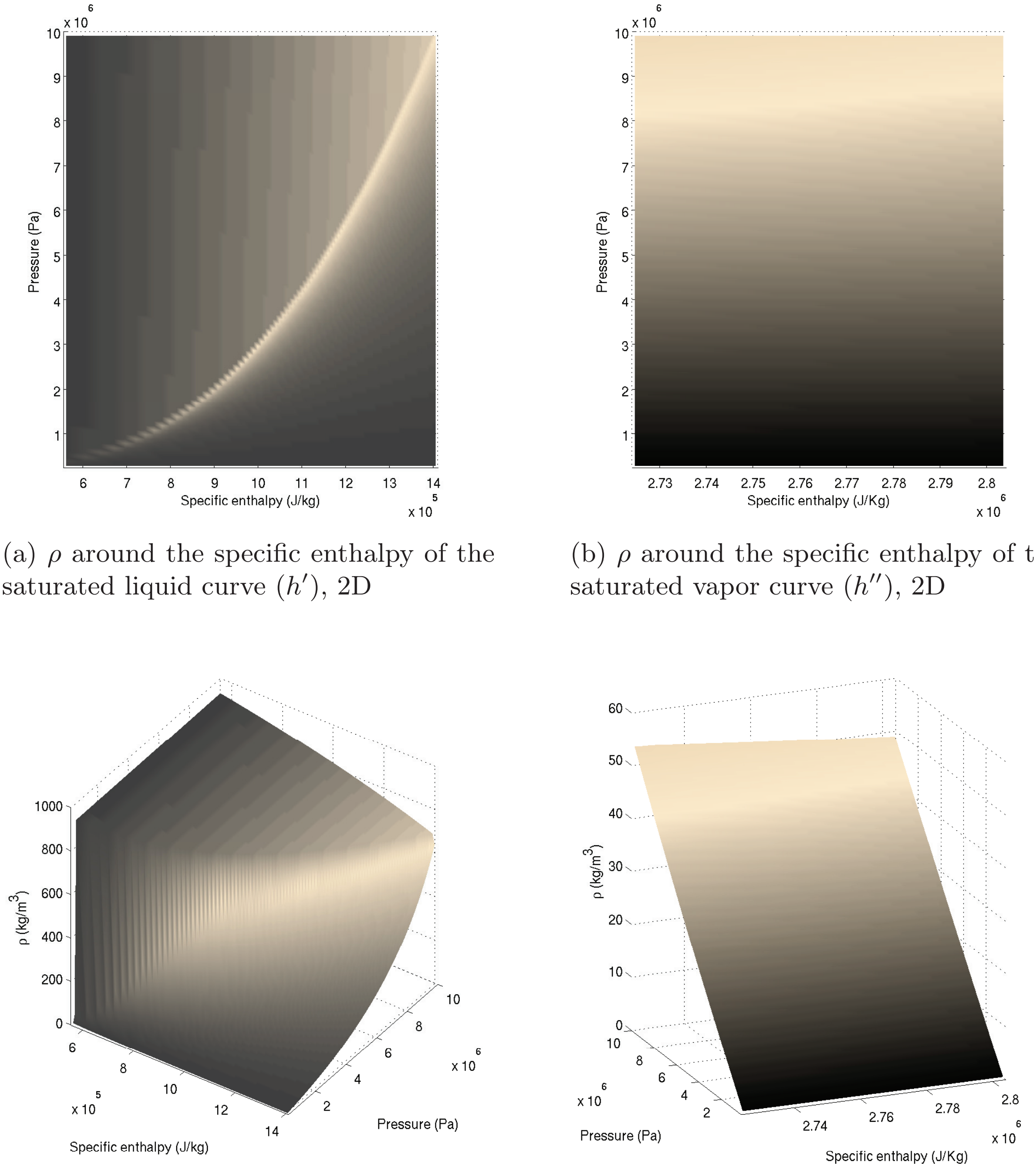
(a) $\frac{\partial\rho}{\partial p}$ around the specific enthalpy of the saturated liquid curve (h'), 2D

(b) $\frac{\partial\rho}{\partial p}$ around the specific enthalpy of the saturated vapor curve (h''), 2D

(c) $\frac{\partial\rho}{\partial p}$ around the specific enthalpy of the saturated liquid curve (h'), 3D

(d) $\frac{\partial\rho}{\partial p}$ around the specific enthalpy of the saturated vapor curve (h''), 3D

Figure 6.9: $\frac{\partial\rho}{\partial p}$ around the saturation curves



(a) ρ around the specific enthalpy of the saturated liquid curve (h'), 2D

(b) ρ around the specific enthalpy of the saturated vapor curve (h''), 2D

(c) ρ around the specific enthalpy of the saturated liquid curve (h'), 3D

(d) ρ around the specific enthalpy of the saturated vapor curve (h''), 3D

Figure 6.10: Density (ρ) around the saturation curves

Chattering is not limited to discontinuities in thermodynamic properties, as other model variables with discontinuities which influence the state variables might cause chattering (Jensen, 2003), e.g. heat transfer coefficients, friction factors, etc.

6.6.3 Chattering in the Boiler Pipe Model

Chattering time instants in the boiler pipe model, described in § 6.6.1, were identified in Tab. 6.4. In these time instants, chattering effects can be seen clearly during simulation. For instance, focusing on the 9^{th} CV, Fig. 6.11 shows the transitions between the single and two-phase regions (1 for single-phase and 2 for two-phase), while Fig. 6.12 shows $\frac{\partial \rho}{\partial h}$.

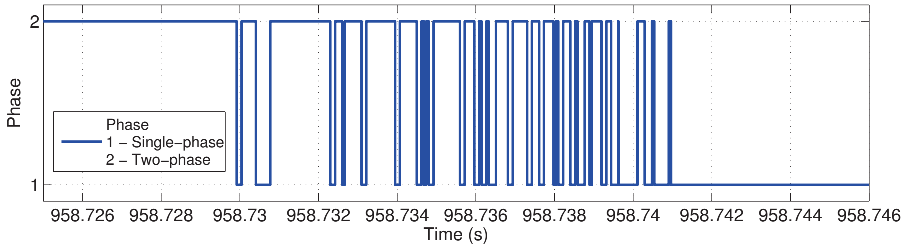


Figure 6.11: Boiler pipe model: phase changes, 9^{th} CV, chattering instant

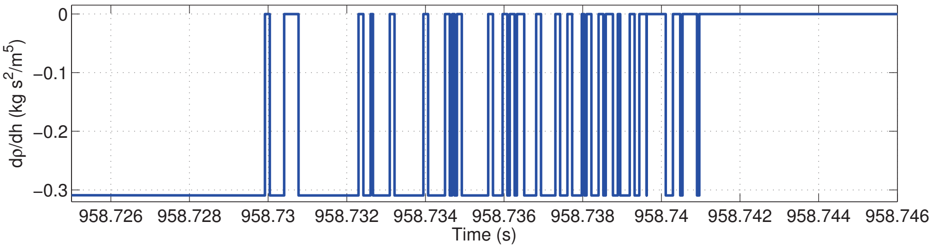
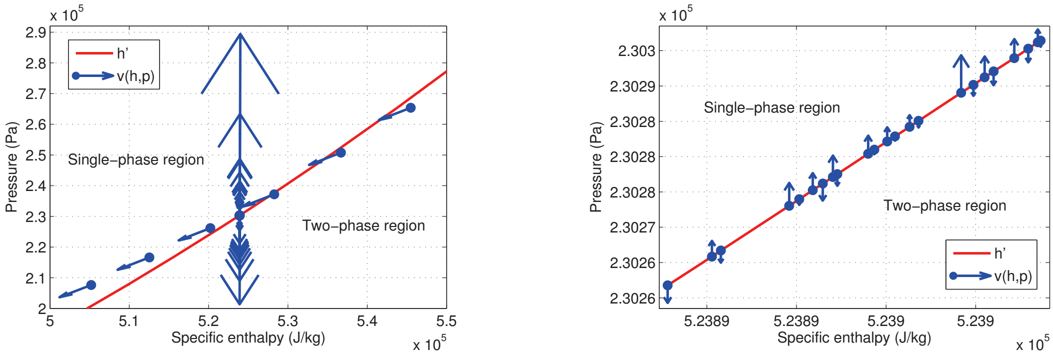


Figure 6.12: Boiler pipe model: $\frac{\partial \rho}{\partial h}$, 9^{th} CV, chattering instant

The trajectory of state variables cannot be represented easily because state variables are close to the saturation curve with respect to the distance between the solutions in each integration step. Fig. 6.13a shows state variables (dots) in each integration step and the vector field (arrow) when crossing the specific enthalpy of saturated liquid from the two-phase flow region to the sub-cooled liquid region in the 9^{th} CV. Before and after crossing the saturation curve, the solution is driven correctly by the vector field. However, chattering occurs during phase transition, cf. Fig. 6.13b which shows a close-up of Fig. 6.13a where the solution is driven continuously to cross the saturation curve.



(a) Crossing h'

(b) Zoom

Figure 6.13: Boiler pipe model: vector field in the 9^{th} CV, chattering instant

6.6.4 Solutions to the Chattering Problem

Now that the chattering problem has been analyzed, this section enumerates solutions which can be implemented to avoid or minimize the phenomenon in dynamic simulations. There are three ways to achieve this end. The first two options, which are mentioned in Tummescheit (2002), are based on the idea of avoiding thermodynamic properties with discontinuities. The third solution minimizes chattering by detecting it and then trying to stop it. These three different ways of tackling chattering are explored in this book.

- *Modifying thermodynamic properties to be continuous.* If thermodynamic properties are always continuous during the simulation, there is no need for conditional sentences, in which case there are no simulation events. Consequently, chattering does not occur.

  One such approach is based on the idea of constructing continuous surfaces for thermodynamic properties with discontinuities, using cubic splines. This is discussed in Appx. C. This attempt was unsuccessful, probably due to the kind of sampling grid chosen. Nevertheless, it helps to understand the chattering problem and is worth mentioning.

  Relevant contributions improve the computation speed of thermodynamic properties based on spline surfaces. Additionally, these approaches also tackle the problem of discontinuous thermodynamic properties by means of using cubic splines. In 2007, the IAPWS established a group to develop extremely fast and accurate property algorithms for water and steam based on a double-grid spline scheme. As a result, a thermodynamic surface of $T(p,h)$ for region 2 (cf. § 4.5.1) was achieved, obtaining a reduction in computational time, high accuracy and numerical consistency (Kunick and Kretzschmar, 2008). To the knowledge of the author, no new publications have been released by this IAPWS group. Recently, a new fluid property calculation approach based on bicubic spline interpolation was developed by Gräber et al. (2012), allowing for fast and smooth model evaluation; this paper points out that the approach taken in Kunick and Kretzschmar (2008) has two mayor drawbacks. First, it is based on quadratic splines, which yield continuity for only the first derivative. Second, the interpolation grid runs across discontinuities in the saturation curves, leading to inaccurate data. This can be avoided by leaving the saturation curves out of the splines and transforming the grid coordinates to a rectangular shape (consult Appx. C for more details). Another recent contribution is Wang et al. (2012), where spline functions for $g(p, T)$, $\rho(p, T)$, $s(p, T)$ and $h(p, T)$ are based on IAPWS-95 thermodynamic properties. This approach uses a double-grid table-lookup biquadratic spline scheme, and coordinate system transformation is performed to avoid discontinuities in the saturation curves. Spline functions achieve faster computational speed and more accuracy than those from IAPWS-IF97, if compared to the IAPWS- 95.

Another approach of this kind is based on the idea of calculating a mean density value in each CV, taking into account explicit discontinuities due to phase changes. This approach was presented in Casella (2006) and applied in the *ThermoPower* library, which uses the FVM to discretize mass and energy equations, as well as a lumped model for the momentum equation. In this book, the same idea has been applied in the *Mean Densities* approach, which considers the FVM and the staggered grid scheme. This is explained in detail in § 6.7.1.

- *Avoiding numerically integrated thermodynamic properties with discontinuities.* No discontinuities in the numerical integration means no state event, and hence no chattering.

  MBMs treat this statement explicitly in discretization, thus avoiding phase changes within a CV but in the boundaries thereof, therefore avoiding the numerical integration of discontinuities in thermodynamic properties which may be present in the finite volume models. MBMs are discussed in Chap. 7.

- *Detecting chattering and taking actions to stop it.* The major drawback of chattering is that it slows down a simulation, making it unfeasible under certain circumstances. If chattering could be detected and actions taken to lead the simulation out of chattering, this certainly would lead to faster simulation, even though the chattering problem is not avoided but controlled. A heuristic approach has been developed based on this idea. This *Heuristic* approach is described in detail in § 6.7.2.

6.7 Approaches to the Chattering Problem

This section describes two approaches to the chattering problem: *Mean Densities* and the *Heuristic approach*. The approach based on cubic splines is described in Appx. C, and MBMs are discussed in Chap. 7.

6.7.1 Mean Densities

As previously mentioned in § 6.6.4, one solution to avoiding the chattering problem is to modify thermodynamic properties so that they become continuous. The thermodynamic properties that produce chattering are density and its partial derivative, which were commented on in § 6.6.2. The *Mean Densities* approach aims to modify these thermodynamic properties so that they become continuous.

The original idea of calculating mean densities was proposed in Casella (2006). The approach is based on the separated integration of the mean density in each flow region. In Casella (2006), equations for the conservation laws – mass, energy and momentum – were developed together with the mean density calculation. The FVM was used to calculate the mass and energy equations. On the other hand, the momentum equation was calculated for the whole HE by using a lumped model. The *ThermoPower* library implements this approach.

The *Mean Densities* approach considered here is based on the same idea, but it has been applied to a FVM by considering the staggered grid scheme described in § 6.4.1. The *Mean Densities* approach has been implemented in the *ThermoFluid* library. Additionally, specific enthalpies at the boundaries are calculated from the specific enthalpies at the CV centers by considering a linear distribution (cf. Fig. 6.14); the partial derivatives have been calculated symbolically.

Therefore, *Mean Densities* change the way density is calculated in the mass and energy grid. In order to compute fluxes when the staggered grid is applied, density values from the mass and energy grid boundaries are required (cf. Fig. 6.14). This calculation is done by the UDS, as previously described in § 6.4.1.1, Eq. 6.24.

6.7.1.1 Assumptions

The first assumption is to consider a uniform pressure in order to calculate the density in each CV, even though pressure is calculated in each CV by the momentum balance equation. This certainly leads to a small error when computing density, but when modeling thermal-hydraulic systems, pressure loss in the evaporator is much lower than in other elements, e.g. valves or turbines (Casella, 2006). Uniform pressure for the whole evaporator (cf. Fig. 6.14) is the mean pressure in the evaporator, which is shown by Eq. 6.39.

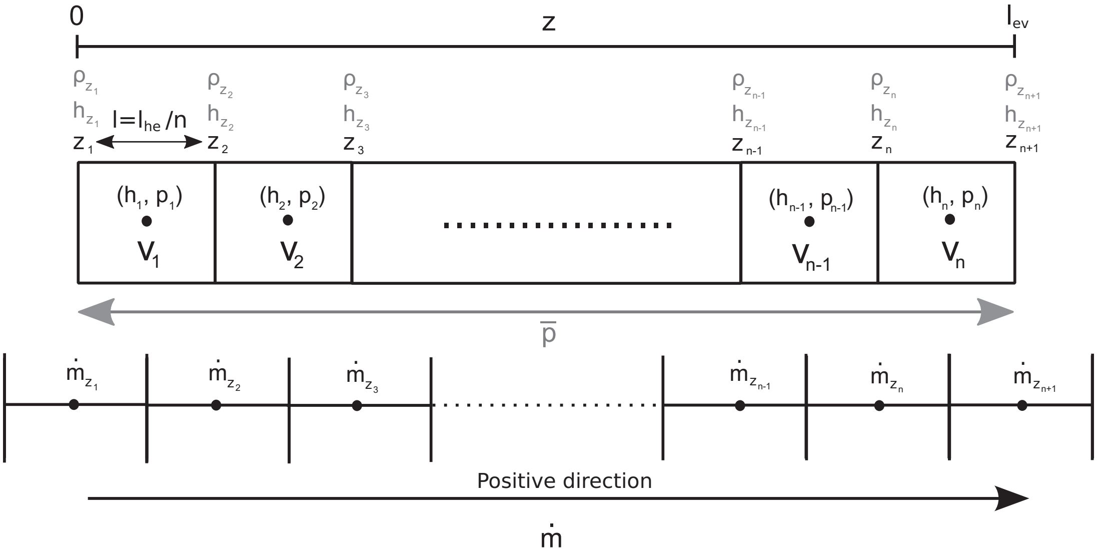


Figure 6.14: Finite volume method and staggered grid scheme applied to a one-dimensional horizontal evaporator in the *Mean Densities* approach

$$\bar{p} = \frac{\sum_{j=1}^{n} p_j}{n}. \tag{6.39}$$

The *Mean Densities* approach calculates the mean density in each CV by integration, so specific enthalpies at the boundaries $(h_{z_j}, h_{z_{j+1}})$ are required, cf. Fig. 6.14. Density at the CV boundaries is calculated from the specific enthalpy in that boundary and the mean pressure, as shown in Eq. 6.40 for V_j in its left boundary, cf. Fig. 6.14.

$$\rho_{z_j} \approx \rho(\bar{p}, h_{z_j}). \tag{6.40}$$

Assuming a linear specific enthalpy distribution between CVs, specific enthalpies at the boundaries can be calculated by Eq. 6.41. The particular cases for the left and right evaporator boundaries are given by Eqs. 6.42 and 6.43, which consider the specific enthalpies at the two first/last CVs.

$$h_{z_j} = h_j + \frac{1}{2}(h_j - h_{j-1}), \quad j \in [2, n], \tag{6.41}$$

$$h_{z_1} = h_1 - \frac{1}{2}(h_2 - h_1), \tag{6.42}$$

$$h_{z_{n+1}} = h_n + \frac{3}{2}(h_n - h_{n-1}). \tag{6.43}$$

CV lengths (l) are all the same and remain constant (cf. Fig. 6.14). For V_j, the length is $l = z_{j+1} - z_j$.

6.7.1.2 Mean Densities Regions

The calculation of density and its partial derivatives in each CV depends on the fluid phase. For this reason, several regions have been defined by considering the fluid phase in the left and right CV boundaries. Tab. 6.5 assigns numbers to each region as a function of the fluid phase at the boundaries of V_j, cf. Fig. 6.14.

z_{j+1} z_j	SC liquid	TP flow	SH steam
SC liquid	1	3	6
TP flow	8	2	4
SH steam	9	7	5

Table 6.5: *Mean Densities* regions

Tab. 6.6 defines mathematically the *Mean Densities* regions when V_j is considered.

Region	Region range	
1	$h_{z_j} < h' \wedge h_{z_{j+1}} < h'$	(6.44)
2	$h_{z_j} \geq h' \wedge h_{z_j} \leq h'' \wedge h_{z_{j+1}} \geq h' \wedge h_{z_{j+1}} \leq h''$	(6.45)
3	$h_{z_j} < h' \wedge h_{z_{j+1}} \geq h' \wedge h_{z_{j+1}} \leq h''$	(6.46)
4	$h_{z_j} \geq h' \wedge h_{z_j} \leq h'' \wedge h_{z_{j+1}} > h''$	(6.47)
5	$h_{z_j} > h'' \wedge h_{z_{j+1}} > h''$	(6.48)
6	$h_{z_j} < h' \wedge h_{z_{j+1}} > h''$	(6.49)
7	$h_{z_j} > h'' \wedge h_{z_{j+1}} \geq h' \wedge h_{z_{j+1}} \leq h''$	(6.50)
8	$h_{z_j} \geq h' \wedge h_{z_j} \leq h'' \wedge h_{z_{j+1}} < h'$	(6.51)
9	$h_{z_j} > h'' \wedge h_{z_{j+1}} < h'$	(6.52)

Table 6.6: Intervals of *Mean Densities* regions

6.7.1.3 Mean Density Calculation

The mean density calculation in each CV is divided into regions 1-9 (Eqs. 6.44 - 6.52). The calculation process is explained in this section, considering V_j, where ρ_{z_j} and $\rho_{z_{j+1}}$ are calculated according to Eq. 6.40 and h_j and h_{j+1} are calculated according to Eqs. 6.41, 6.42 and 6.43. The saturation enthalpies h' and h'' are calculated by applying the mean pressure $\bar{p}$ (cf. Eq. 6.39) and using a library of thermodynamic properties for the particular fluid under consideration (e.g. *Modelica Media* library). The mean density calculation procedure is analogous to the procedure described in Casella (2006).

Region 1 and 5: One-phase flow (Sub-cooled liquid/Sub-cooled liquid or Superheated steam/Superheated steam). Considering mean pressure ($\bar{p}$), density is then continuous in these regions. Hence, for small variations of specific enthalpy (h), density distribution can be assumed as linear (Casella, 2006), as shown in Eq. 6.53 for V_j.

$$\bar{\rho}_j = \frac{1}{l}\int_{z_j}^{z_{j+1}} \rho\, dz = \frac{\rho_{z_j} + \rho_{z_{j+1}}}{2}. \tag{6.53}$$

Assuming a linear enthalpy distribution, Eq. 6.53 can be expressed as Eq. 6.54.

$$\bar{\rho}_j = \frac{1}{l}\int_{z_j}^{z_{j+1}} \rho\, dz = \frac{1}{h_{z_{j+1}} - h_{z_j}}\int_{h_{z_j}}^{h_{z_{j+1}}} \rho\, dh = \frac{\rho_{z_j} + \rho_{z_{j+1}}}{2}. \tag{6.54}$$

Considering the last two expressions in Eq. 6.54 and isolating the integral, Eq. 6.55 is obtained. Eq. 6.55 will be useful for the density calculation in other *Mean Densities* regions.

$$\int_{h_{z_j}}^{h_{z_{j+1}}} \rho\, dh = \frac{\rho_{z_j} + \rho_{z_{j+1}}}{2}(h_{z_{j+1}} - h_{z_j}). \tag{6.55}$$

Region 2: Two-phase flow / Two-phase flow. Density is also continuous in this region, but its value can change considerably across a small distance if pressure is low and if the specific enthalpy is just over the h' curve (Casella, 2006). Static quality (x) is calculated with Eq. 4.20, which is repeated here by Eq. 6.56. The specific volume (ν) is calculated by using Eq. 6.57, which is obtained from Eq. 4.18.

$$x = \frac{h - h'}{h'' - h'}, \tag{6.56}$$

$$\nu = x\nu'' + (1 - x)\nu'. \tag{6.57}$$

Replacing Eq. 6.56 in Eq. 6.57 yields Eq. 6.58.

$$\nu = x\nu'' + (1 - x)\nu' = \frac{1}{h'' - h'}[h(\nu'' - \nu') + h''\nu' - h'\nu'']. \tag{6.58}$$

Calculating for V_j, $\nu_{z_{j+1}} - \nu_{z_j}$ from Eq. 6.58 yields Eq. 6.59.

$$\nu_{x_{j+1}} - \nu_{z_j} = (h_{z_{j+1}} - h_{z_j})\frac{\nu'' - \nu'}{h'' - h'}. \tag{6.59}$$

Since h is assumed to be linear with respect to z, whereby ν is linear with respect to h (cf. Eq. 6.59), and considering $\nu = 1/\rho$, the mean density in this region is defined by Eq. 6.60.

$$\begin{aligned}\bar{\rho}_j &= \frac{1}{l}\int_{z_j}^{z_{j+1}} \rho\, dz = \frac{1}{h_{z_{j+1}} - h_{z_j}} \int_{h_{z_j}}^{h_{z_{j+1}}} \rho\, dh = \frac{1}{\nu_{z_{j+1}} - \nu_{z_j}} \int_{\nu_{z_j}}^{\nu_{z_{j+1}}} \frac{1}{\nu}\, d\nu = \\ &= \frac{1}{h_{z_{j+1}} - h_{z_j}} \frac{h'' - h'}{\nu'' - \nu'} \int_{\nu_{z_j}}^{\nu_{z_{j+1}}} \frac{1}{\nu}\, d\nu = \frac{1}{h_{z_{j+1}} - h_{z_j}} \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\nu_{z_{j+1}}}{\nu_{z_j}}\right) = \\ &= \frac{1}{h_{z_{j+1}} - h_{z_j}} \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho_{z_j}}{\rho_{z_{j+1}}}\right).\end{aligned} \tag{6.60}$$

Note that Eq. 6.60 becomes singular when $h_{z_j} = h_{z_{j+1}}$. Hence, it is necessary to check if the difference between both terms in the previous equation is smaller than a certain threshold (ξ), Eq. 6.61.

$$\bar{\rho}_j = \frac{1}{h_{z_{j+1}} - h_{z_j}} \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho_{z_j}}{\rho_{z_{j+1}}}\right), \quad |h_{z_{j+1}} - h_{z_j}| \geq \xi,$$

$$\bar{\rho}_j = \rho_j, \quad |h_{z_{j+1}} - h_{z_j}| < \xi. \tag{6.61}$$

Considering the second and last expressions in Eq. 6.60 and isolating the integral, Eq. 6.62 is obtained. Eq. 6.62 will be useful for density calculations in other *Mean Densities* regions.

$$\int_{h_{z_j}}^{h_{z_{j+1}}} \rho\, dh = \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho_{z_j}}{\rho_{z_{j+1}}}\right). \tag{6.62}$$

Region 3: Sub-cooled liquid / Two-phase flow. In this region, density has a discontinuous derivative due to the phase change in the saturation curve h', as described in § 6.6.2. The mean density in this region can be calculated assuming a linear distribution of the specific enthalpy and splitting the integral into two parts, a sub-cooled liquid and a two-phase flow, using Eq. 6.55 and Eq. 6.62 to calculate both integrals and thus obtaining Eq. 6.63.

$$\begin{aligned}\bar{\rho}_j &= \frac{1}{l}\int\limits_{z_j}^{z_{j+1}} \rho\, dx = \frac{1}{h_{z_{j+1}} - h_{z_j}} \int\limits_{h_{z_j}}^{h_{z_{j+1}}} \rho\, dh = \\ &= \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int\limits_{h_{z_j}}^{h'} \rho\, dh + \int\limits_{h'}^{h_{z_{j+1}}} \rho\, dh\right] = \\ &= \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\frac{\rho_{z_j} + \rho'}{2}(h' - h_{z_j}) + \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho'}{\rho_{z_{j+1}}}\right)\right].\end{aligned} \tag{6.63}$$

The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.46. Density in the rest of the regions is calculated in the same way, assuming a linear distribution of the specific enthalpy and splitting the integral into two or three parts when phase changes occur, and then calculating each integral from Eqs. 6.55 and 6.62 for one-phase flow or two-phase flow, respectively. The two first steps necessary to obtain Eq. 6.63 are the same in the remaining regions. They have been omitted for the sake of brevity.

Region 4: Two-phase flow / Superheated steam. Splitting the integral into two parts, two-phase flow and superheated steam, and then using Eq. 6.62 and Eq. 6.55 yields Eq. 6.64. The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.47.

$$\begin{aligned}\bar{\rho}_j =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int\limits_{h_{z_j}}^{h''} \rho\, dh + \int\limits_{h''}^{h_{z_{j+1}}} \rho\, dh\right] = \\ =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho_{z_j}}{\rho''}\right) + \frac{\rho'' + \rho_{z_{j+1}}}{2}(h_{z_{j+1}} - h'')\right].\end{aligned} \tag{6.64}$$

Region 6: Sub-cooled liquid / Superheated Steam. Splitting the integral into three parts – sub-cooled liquid, two-phase flow and superheated steam – using Eq. 6.55, Eq. 6.62 and Eq. 6.55 again yields Eq. 6.65. The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.49.

$$\begin{aligned}\bar{\rho}_j =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int\limits_{h_{z_j}}^{h'} \rho\, dh + \int\limits_{h'}^{h''} \rho\, dh + \int\limits_{h''}^{h_{z_{j+1}}} \rho\, dh\right] = \frac{1}{h_{z_{j+1}} - h_{z_j}} \cdot \\ &\cdot \left[\frac{\rho_{z_j} + \rho'}{2}(h' - h_{z_j}) + \frac{h'' - h'}{\nu'' - \nu'} \ln\left(\frac{\rho'}{\rho''}\right) + \frac{\rho'' + \rho_{z_{j+1}}}{2}(h_{z_{j+1}} - h'')\right].\end{aligned} \tag{6.65}$$

Region 7: Superheated Steam / Two-phase flow. Splitting the integral into two parts, superheated steam and two-phase flow, using Eq. 6.55 and Eq. 6.62 yields Eq. 6.66. The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.50.

$$\begin{aligned}\bar{\rho}_j =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int_{h_{z_j}}^{h''} \rho \, dh + \int_{h''}^{h_{z_{j+1}}} \rho \, dh \right] = \\ =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\frac{\rho_{z_j} + \rho''}{2} (h'' - h_{z_j}) + \frac{h'' - h'}{\nu'' - \nu'} \ln \left(\frac{\rho''}{\rho_{z_{j+1}}} \right) \right] .\end{aligned} \tag{6.66}$$

Region 8: Two-phase flow / Sub-cooled liquid. Splitting the integral into two parts, two-phase flow and sub-cooled liquid, using Eq. 6.62 and Eq. 6.55 yields Eq. 6.67. The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.51.

$$\begin{aligned}\bar{\rho}_j =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int_{h_{z_j}}^{h'} \rho \, dh + \int_{h'}^{h_{z_{j+1}}} \rho \, dh \right] = \\ =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\frac{h'' - h'}{\nu'' - \nu'} \ln \left(\frac{\rho_{z_j}}{\rho'} \right) + \frac{\rho' + \rho_{z_{j+1}}}{2} (h_{z_{j+1}} - h') \right] .\end{aligned} \tag{6.67}$$

Region 9: Superheated Steam / Sub-cooled liquid. Splitting the integral into three parts – superheated steam, two-phase flow and sub-cooled liquid – using Eq. 6.55, Eq. 6.62 and Eq. 6.55 again yields Eq. 6.68. The $h_{z_{j+1}} = h_{z_j}$ case is not possible due to Eq. 6.52.

$$\begin{aligned}\bar{\rho}_j =& \frac{1}{h_{z_{j+1}} - h_{z_j}} \left[\int_{h_{z_j}}^{h''} \rho \, dh + \int_{h''}^{h'} \rho \, dh + \int_{h'}^{h_{z_{j+1}}} \rho \, dh \right] = \frac{1}{h_{z_{j+1}} - h_{z_j}} \cdot \\ & \cdot \left[\frac{\rho_{z_j} + \rho''}{2} (h'' - h_{z_j}) + \frac{h'' - h'}{\nu'' - \nu'} \ln \left(\frac{\rho''}{\rho'} \right) + \frac{\rho' + \rho_{z_{j+1}}}{2} (h_{z_{j+1}} - h') \right] .\end{aligned} \tag{6.68}$$

6.7.1.4 Specific Enthalpy Partial Derivative of Mean Densities

This partial derivative can be calculated independently of the *Mean Densities* region, as shown in Eq. 6.69 (Casella, 2006).

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial h} =& \frac{\partial}{\partial h}\left(\frac{1}{l}\int\limits_{z_j}^{z_{j+1}} \rho\, dz\right) = \frac{1}{l}\int\limits_{z_j}^{z_{j+1}} \frac{\partial}{\partial h}\rho\, dz = \frac{1}{l}\int\limits_{z_j}^{z_{j+1}} \frac{\partial}{\partial z}\frac{dz}{dh}\rho\, dz = \\
=& \frac{1}{l}\frac{l}{h_{z_{j+1}} - h_{z_j}} \int\limits_{z_{j+1}}^{z_j} \frac{\partial \rho}{\partial z} dz = \frac{\rho_{z_{j+1}} - \rho_{z_j}}{h_{z_{j+1}} - h_{z_j}}.
\end{aligned}
\tag{6.69}
$$

Note that Eq. 6.69 becomes singular when $h_{x_{j-1}} = h_{x_j}$. Hence, it is necessary to check if the difference between both terms is smaller than a certain threshold (ξ), Eq. 6.70.

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial h} &= \frac{\rho_{z_{j+1}} - \rho_{z_j}}{h_{z_{j+1}} - h_{z_j}}, \quad |h_{z_{j+1}} - h_{z_j}| \geq \xi, \\
\frac{\partial \bar{\rho}_j}{\partial h} &= \frac{\partial \rho_j}{\partial h}\,, \quad |h_{z_{j+1}} - h_{z_j}| < \xi.
\end{aligned}
\tag{6.70}
$$

6.7.1.5 Pressure Partial Derivative of Mean Densities

The pressure partial derivative of *Mean Densities* can be calculated symbolically by considering the mean density equation in each region. The resulting expressions are rather lengthy, however, so they have been omitted here but are detailed in Appx. D.

6.7.2 Heuristic Approach

This section deals with the chattering problem using a *Heuristic* approach in an attempt to minimize the effects caused by this practical problem in simulation, decrease state events and therefore decrease simulation time.

A chattering detector, which tries to identify chattering time instants in simulation, is detailed in § 6.7.2.2. The general idea is to impose lower bounds on the amount of time that elapses between successive discrete transitions (cf. § 6.7.2.1). Zeno hybrid automatons are hybrid systems that can exhibit infinitely discrete transitions across a finite time interval (chattering or zenoness). In Johansson et al. (1999) there are different methods for solving the zenoness in hybrid automatons. These methods are regularization, averaging and Filippov solutions (Filippov and Arscott, 1988). The method presented in this book is slightly different, though. First of all, it does not change the system model, as it acts just like a supervisor. Additionally, the action taken by the chattering detector is a heuristic rule.

6.7.2.1 Temporal and Spatial Restrictions

Two different kinds of restrictions have been considered for limiting phase changes, namely temporal and spatial. Time or a certain variable change in its magnitude, with regard to the number of phase changes, is used as intervals.

Temporal Restriction

Temporal restriction detects a certain number of phase changes in a time interval. Both parameters, the number of phase changes and the time interval, can be configured. If the number of phase changes is higher than a certain value in the time interval selected, it is considered that there is chattering in the simulation.

Spatial Restriction

In this case, instead of using a time interval, a space interval is used by taking the magnitude change of a certain simulation variable, which must be chosen carefully in order to get good results. Furthermore, the selected variable must change throughout the whole simulation, otherwise it cannot be used for this purpose.

6.7.2.2 Chattering Detector

A chattering detector has been developed for managing temporal and spatial restrictions. Its commitments are to detect chattering time instants by using a temporal or a spatial restriction, and then to take an action which leads the state variables away from the attraction of the phase-change curves.

The action taken by the *Heuristic* approach is to increase or decrease by a certain magnitude one of the state variables, following the tendency of this state variable, in an attempt to escape from saturation curve attraction. When a chattering instant is detected, the simulation is stopped (by an event), following which the state variable is increased or decreased by a certain quantity and the simulation is resumed. If the state variables have been removed from the attraction, the simulation can carry on until it ends or until a different chattering time instant is detected. If the change in magnitude of the state variable keeps driving the solution toward the phase-change curves, this procedure is repeated until the state variables are away from the attraction field. Tab. 6.7 shows the different approaches considered and the default parameter values.

6.8 Simulation of the Boiler Pipe Model applying Mean Densities and the Heuristic Approach

This section shows the simulations of the boiler pipe model, introduced in § 6.6.1, but applying *Mean Densities* (cf. § 6.7.1) and the *Heuristic* approach (cf. § 6.7.2).

6.8.1 Mean Densities

The *Mean Densities* approach was implemented in the pipe model of the *ThermoFluid* library; the simulation described in § 6.6.1 was performed applying the *Mean Densities* approach. Fig. 6.15 shows the active *Mean Densities* regions[c] (cf. § 6.7.1.2) in the 9^{th} CV.

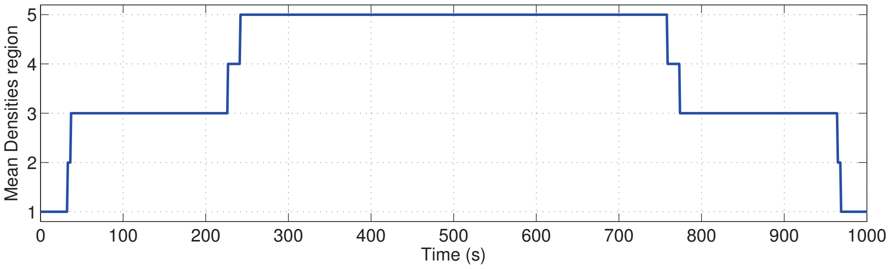


Figure 6.15: *Mean Densities* regions in the 9^{th} CV during simulation

6.8.2 Heuristic Approach

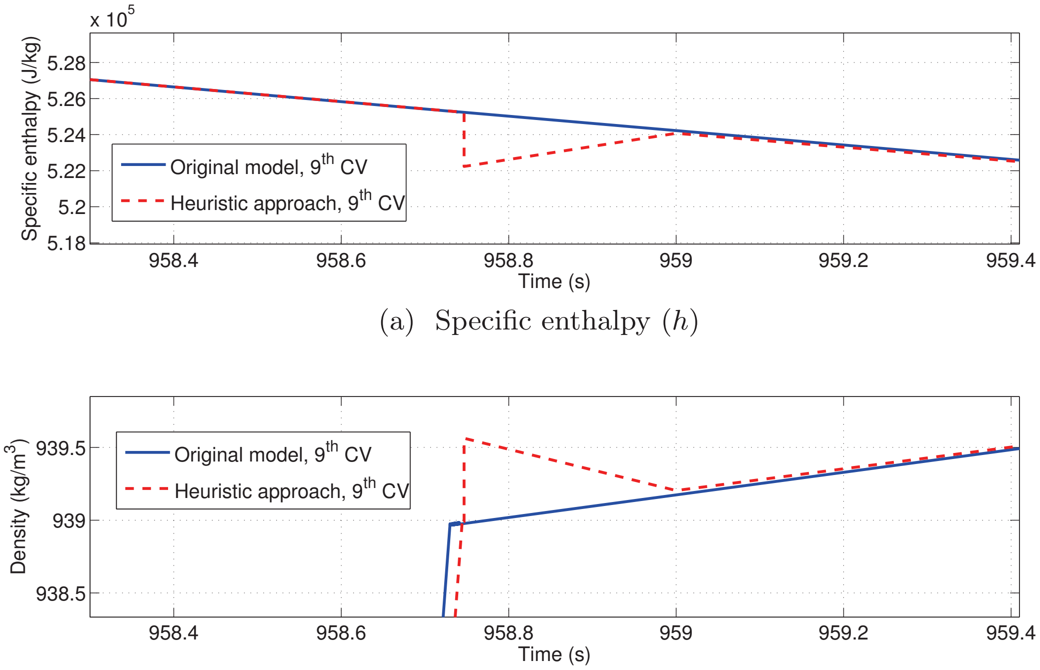
The *Heuristic* approach was also implemented in the pipe model of the *ThermoFluid* library. The simulation was performed by applying the *Heuristic* approach. Tab. 6.7 shows the parameter values considered in the simulation and previously described in § 6.7.2.2.

The action of the *Heuristic* approach, using a temporal restriction, over the specific enthalpy in the 9^{th} CV when chattering is detected can be seen in Fig. 6.16a, while its repercussions for density are shown in Fig. 6.16b. The solid line corresponds to the original model and the dashed line corresponds to the *Heuristic* approach.

Parameter	Temporal Restriction	Spatial Restriction
Variable	Simulation time (t)	Specific enthalpy (h)
Number of phase changes	4	4
Variable interval	0.1 s	1.000 J/kg
Changed state variable	Specific enthalpy (h)	Specific enthalpy (h)
Change in magnitude	±3.000 J/kg	±3.000 J/kg

Table 6.7: *Heuristic* approach default parameter values

[c] The *Mean Densities* region numbers 2 and 3 are interchanged with respect to those described in § 6.7.1.2 for the sake of clarity in Fig. 6.15.



(a) Specific enthalpy (h)

(b) Density (ρ)

Figure 6.16: Comparison of specific enthalpy and density between the original model and the *Heuristic* approach with temporal restriction in the 9^{th} CV at chattering time instant

6.8.3 Simulation Results and Statistics

Tab. 6.8 summarizes the simulation statistics, comparing the original model with the *Mean Densities* and *Heuristic* approaches. The last one, considering both temporal and spatial restrictions, is defined in Tab. 6.7.

Parameter	Original model	Mean Densities	Heuristic approach	
			Temporal	Spatial
CPU-time Int. (s)	16.6	1.02	2.59	2.61
State events	9, 281	0	63	62
Min. Int. stepsize	$5.45 \cdot 10^{-7}$	$8.9 \cdot 10^{-6}$	$5.21 \cdot 10^{-7}$	$5.25 \cdot 10^{-7}$
Max. Int. stepsize	23.2	32.7	23.2	23.2
Max. Int. order	5	4	5	5

Table 6.8: Boiler pipe model simulation statistics considering *Mean Densities* and the *Heuristic* approach

Considering the *Mean Densities* approach, Tab. 6.8 shows that the state events have been completely wiped out, from 9, 281 to 0. This happens because thermodynamic properties are continuous in the *Mean Densities* approach. For this reason, conditional sentences at phase changes are not necessary. The simulation time has been reduced considerably from 16.6 to 1.02 s, which is a reduction of 93.8% in simulation time.

Fig. 6.17 highlights that there is no chattering when the *Mean Densities* approach is used. Compare Fig. 6.17 with Fig. 6.13, where repeated phase changes are shown. On the contrary, Fig. 6.17 shows that there is no confrontation between vector fields, because the thermodynamic properties are continuous and smooth. Both transitions, from sub-cooled liquid to two-phase flow and vice versa, are represented in Fig. 6.17.

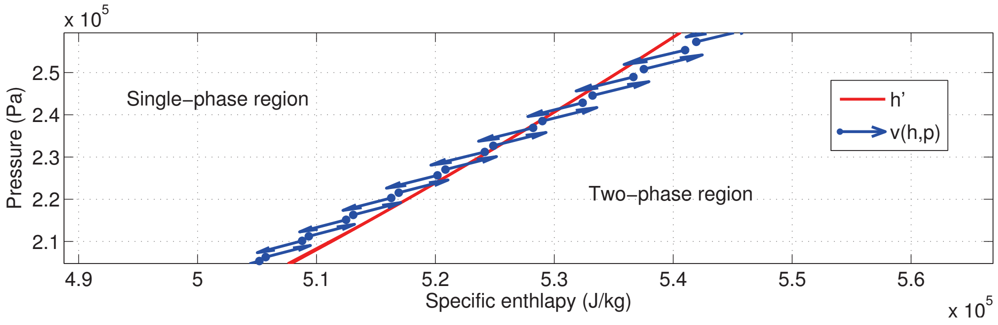


Figure 6.17: *Mean Densities* approach: vector field in the 9^{th} CV

The Percentage Relative Error (ϵ_r) (PRE) is measured according to Eq. 6.71, where μ_o is the variable in the original model and μ_n is the variable in the approach considered. The maximum PRE committed by the *Mean Densities* approach in the outlet specific enthalpy is only 0.33%, while the maximum PRE committed in density in a steady-state is higher. The maximum error is up to 17% in this case, while the error committed might be slightly higher at phase changes. This is understandable because the *Mean Densities* approach defines smooth models for thermodynamic properties.

$$\epsilon_r(\mu_n) = 100 \cdot \frac{|\mu_n - \mu_o|}{\mu_o} \tag{6.71}$$

If the *Heuristic* approach is considered, simulation time decreases considerably from 16.6 to 2.59 and 2.61 (cf. Tab. 6.8), obtaining a reduction of 84.3% in simulation time. This reduction is due to a decrease of phase changes which produce state events. State events have decreased from 9,281 to 63 and 62, and the errors committed by the *Heuristic* approach are low, with 0.57% being the maximum PRE in specific enthalpy and 1.9% in density.

Fig. 6.18 shows the vector field when the *Heuristic* approach is used. It presents a confrontation between the vector fields when crossing the saturation curve, but the numbers of phase changes have reduced in comparison with Fig. 6.13. This reduction is due to the action of the chattering detector and the restrictions imposed in the developed *Heuristic* approach.

The most left-placed vertical vector, in Fig. 6.18, represents the action taken by the chattering detector, which frees the state variables from the attraction of the vector field. The other four vectors represent the previous four phase changes which indicate to the chattering detector that there is chattering in the simulation.

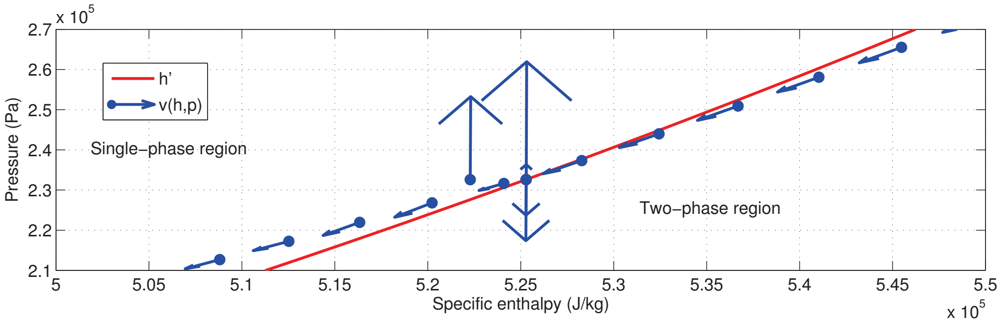


Figure 6.18: *Heuristic* approach: vector field, 9^{th} CV, chattering instant

6.9 Mean Densities vs. the Heuristic Approach

In this section, the *Mean Densities* and *Heuristic* approaches are compared. This comparison involves mean values considered in particular examples. Estimating errors committed and simulation time is highly dependent on the models being simulated. However, the results can be generalized for further models – not the numerical values, but the main conclusions inferred from them.

Tab. 6.9 shows a comparison between the *Heuristic* and *Mean Densities* approaches. The percentages in Tab. 6.9 represent the percentage increment (+) or decrement (-) in time with respect to the original model. When there is no chattering in the simulation, both approaches require more computational time because they both involve extra code.

As can be seen in Tab. 6.9, the *Heuristic* approach is the fastest option when there is no chattering (+50%) and is the one with the lowest error rates, 0.6% and 2% in (h,ρ). When there is chattering, the reduction in simulation time is considerable (-87%). On the other hand, it is a *Heuristic* approach, so chattering is detected by a heuristic rule.

Method	*Heuristic* approach	*Mean Densities*
NO chattering execution time	+50%	+70%
Chattering execution time	-87%	-93%
Maximum PRE in h, $\epsilon_r(h)$	0.6%	2%
Maximum PRE in ρ, $\epsilon_r(\rho)$	2%	17%
Chattering	Yes, detected	No

Table 6.9: Comparison: the *Heuristic* and *Mean Densities* approaches

The *Mean Densities* approach is fastest when there is chattering in the original model (-93%). However, when there is no chattering, it is slow

(+70%) in comparison to the *Heuristic* approach. Moreover, the errors committed, especially in density, are high under certain circumstances (17%). This approach is based on the separated integration of the mean density in each region and does not use a heuristic rule. Chattering caused by thermodynamic properties cannot occur with this approach, because there are no state events at phase changes.

6.10 Discontinuities in the Heat Transfer Coefficient

As previously mentioned in § 6.6.2, any variable with discontinuities during simulation which affects state variables might cause chattering. This is the case when considering different HTCs for each fluid phase. In this case, the convective heat transfer coefficient between the absorber tube and the HTF changes discontinuously when the fluid phase changes in a CV.

One remedy is to estimate constant heat transfer coefficients in each fluid phase and continuously join these constant values, for instance as a function of vapor quality. This approach was developed in Yebra (2006). In this work, constant heat transfer coefficients have been joined by considering cubic splines, which smooths the transition between coefficients and avoids further numerical issues, e.g. solution convergence.

A possible solution, and proposed as future work, would be to apply multidimensional interpolation methods to construct smooth surfaces for HTCs, because HTCs depend on several variables, e.g. pressure, specific enthalpy, heat flow rate and mass flow rate.

It is important to note that the *Heuristic* approach can deal with chattering at phase changes caused by discontinuities in the heat transfer coefficient, because this approach is not limited to discontinuities in thermodynamic properties, unlike the *Mean Densities* approach.

6.11 Summary and Conclusions

In this chapter the FVM was introduced. Mass, energy and momentum conservation laws for a one-dimensional horizontal evaporator were discretized according to the FVM and the staggered grid scheme. Two free Modelica libraries which implement one-dimensional discretized evaporators were also introduced: the *ThermoFluid* library and the *Modelica Fluid* library. HTCs and FFCs, implemented in the *ThermoFluid* library, were also mentioned. The chattering problem, which arises in discretized two-phase flow models, was described and demonstrated through simulation examples. Solutions for solving the chattering problem were discussed, and particular approaches implemented to tackle the chattering problem were also detailed in this chapter, namely the *Mean Densities* and *Heuristic* approaches. Simulation results and a comparison of both approaches were finally presented.

Although the *Heuristic* approach is based on a heuristic rule to detect chattering, it seems to be a good solution due to its high accuracy and good performance. Furthermore, the *Heuristic* approach is not limited to discontinuities caused by thermodynamic properties, as it can also deal with chattering caused by other discontinuous variables.

6.12 Related Scientific Contributions

The scientific contributions related to this chapter are enumerated and briefly commented on in the following list. These contributions were previously listed in § 1.4.

2. BONILLA, J., YEBRA, L. and DORMIDO, S. Chattering in dynamic mathematical two-phase flow models. *Applied Mathematical Modelling*, vol. 36(5), pages 2067–2081, 2012. ISSN 0307-904X.
Chattering problem: source of the problem, approaches to tackle it.

3. BONILLA, J., YEBRA, L. J. and DORMIDO, S. A heuristic method to minimize the chattering problem in dynamic mathematical two-phase flow models. *Mathematical and Computer Modelling*, vol. 54(5-6), pages 1549–1560, 2011. ISSN 0895-7177.
Heuristic approach.

4. BONILLA, J., YEBRA, L. J. and DORMIDO, S. Mean Densities in Dynamic Mathematical Two-phase Flow Models. *CMES - Computer modeling in Engineering & Science*, vol. 67(1), pages 13–37, 2010. ISSN 1526-1492.
Mean Densities approach.

8. BONILLA, J., YEBRA, L. J., ZARZA, E. and DORMIDO, S. Chattering in Dynamic Mathematical Two-Phase Flow Models. In *Proceedings of the European Control Conference 2009*, oral communication, Budapest, Hungary.
Preliminary work about the chattering problem.

10. BONILLA, J., YEBRA, L. J., ZARZA, E., DORMIDO, S. Chattering problem in dynamic mathematical two-phase flow models.. In 5^{th} *Sollab Doctoral Colloquium on Solar Concentrating Technologies*, Oral communication and poster, June 23 - 26, 2009, Cologne, Germany.
Presentation of the chattering problem and approaches to tackle it.

11. BONILLA, J., YEBRA, L. J., ZARZA, E., DORMIDO, S. Chattering problem in dynamic mathematical two-phase flow models.. In 4^{th} *Sollab Doctoral Colloquium on Solar Concentrating Technologies*, Oral communication, September 10 - 12, 2008, Tabernas (Almería), Spain.
Presentation of the chattering problem.

7
Moving Boundary Models

In some extent, moving boundary models are a compromise between lumped parameter models and distributed parameter models.

H. Qiao, R. Radermacher and V. Aute, *A Review for Numerical Simulation of Vapor Compression Systems, International Refrigeration and Air Conditioning Conference, 2010*

Abstract: The moving boundary method is an appealing approach for the design of advanced ACS. MBMs are low-order models but they can describe the dynamic behavior of HEs with accuracy. Furthermore, MBMs avoid phase changes within CV but detect them in the boundaries of the CV. Therefore, they avoid the numerical integration of discontinuities in thermodynamic properties which may be present in finite volume models. This chapter presents the basics of MBMs, a brief state-of-the-art study, the derivation of new MBMs and the development of a Modelica library which includes MBMs for two-phase flow evaporators and condensers, called the *MBMs* library, together with stability and integrity simulation tests.

7.1 Basics of Moving Boundary Models

Two of the most common CFD approaches used in fluid dynamics modeling and their counterparts in heat exchange modeling are the finite-volume distributed-parameter method (Patankar, 1980) and the moving-boundary lumped-parameter method (Adams et al., 1965). Dynamic modeling is always a challenging task, in which the trade-off between accuracy and speed must be evaluated.

MBMs, also known as models with time-varying phase boundaries, are low-order and faster models than their finite volume counterparts. Additionally, they can describe the dynamic behavior of evaporators and condensers accurately (Bendapudi et al., 2008). In the context of real-time simulation, dynamic system optimization and the design of an advanced ACS (e.g. model-based control), where fast computation is required, the moving boundary method seems to be appropriate.

The moving boundary method divides the evaporator into different regions, also called CVs, depending on the fluid phase. In each CV, lumped thermodynamic properties are averaged in some way; the barrier is not fixed and it may move between adjacent CVs. The main idea is to dynamically track the lengths of the different regions (Jensen, 2003).

Tab. 7.1 shows the different flow configurations in evaporators and condensers when considering two-phase flows. The three basic flow states are: sub-cooled liquid (SC), two-phase flow (TP) and superheated vapor (SH), which are represented by Fig. 7.1a, Fig. 7.1b and Fig. 7.1c, respectively. Considering these three basic flow states, compounded configurations can be created. In Tab. 7.1, general, flooded and dry evaporator and condenser configurations are enumerated. Their corresponding representations are shown in Fig. 7.2.

Ab.	Description	Flow states in evaporator	Fig.	Flow states in condenser	Fig.
SC	Sub-cooled liquid	SC	7.1a	SC	7.1a
TP	Two-phase flow	TP	7.1b	TP	7.1b
SH	Superheated vapor	SH	7.1c	SH	7.1c
G	General	SC - TP - SH	7.2a	SH - TP - SC	7.2b
F	Flooded	SC - TP	7.2c	TP - SC	7.2d
D	Dry	TP - SH	7.2e	SH - TP	7.2f

Table 7.1: Configurations in MBMs

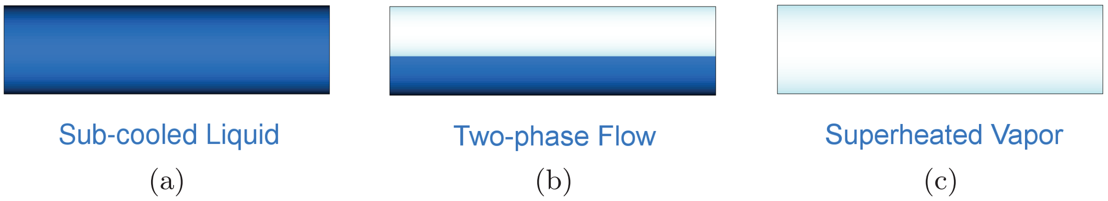
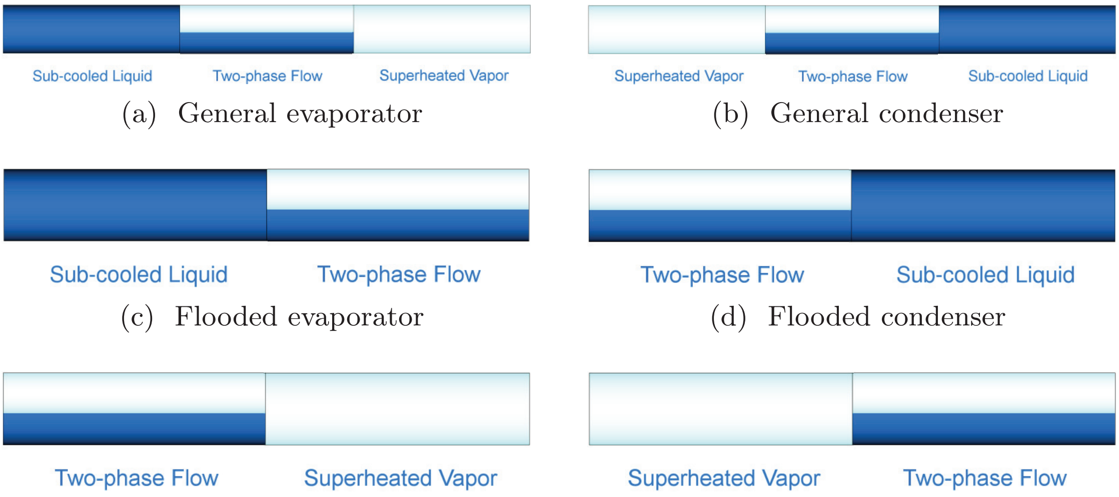


Figure 7.1: Basic flow states



(a) General evaporator

(b) General condenser

(c) Flooded evaporator

(d) Flooded condenser

(e) Dry evaporator

(f) Dry condenser

Figure 7.2: Evaporators and condensers

7.2 State-of-the-Art

Tab. 7.2 summarizes some of the most relevant MBMs for two-phase flows and shows if models for evaporators and condensers were developed, which configurations are supported (cf. Tab. 7.1, Figs. 7.1 and 7.2), if the models support switching between the different configurations and if they consider the mean void fraction ($\bar{\gamma}$) as time variant ($\bar{\gamma}(t)$).

Outstanding MBM reviews are presented in Mckinley and Alleyne (2008) and Mancini (2011). Dynamic switching between different configurations is an important aspect when developing MBMs, because switching between two different configurations implies that during the simulation the model will represent both configurations. Two different configurations can be represented by different DAEs systems, so the model must, in some way, switch between them. The main problem that the modeler faces is that different configurations can have different numbers/types of equations or variables, thus leading to a variable-structure model (cf. § 2.6.5). Furthermore, the transition must be done continuously in order to avoid numerical problems. The switching details will be discussed in § 7.10.

When inlet and outlet qualities are constant in HEs, the mean void fraction ($\bar{\gamma}$) can be considered time invariant (Wedekind et al., 1978) (Beck and Wedekind, 1981); otherwise, its calculation might describe more accurately the total amount of vapor at the output of the evaporator (Åström and Bell, 2000). Some authors neglect the time derivative of the mean void fraction ($d\bar{\gamma}/dt$) because its time dependence is related to dynamic modes that are faster than the dominant system dynamics, and also because the change in mean void fraction tends to be small during transients (Rasmussen, 2006).

Moving Boundary Model	Evaporator	Condenser	Swit.	$\bar{\gamma}(t)$
Adams et al. (1965)	G	-	No	No
Ray and Bowman (1976)	G	-	No	No
Dhar and Soedel (1979)	TP	D,F,SH	Yes	No
Grald and MacArthur (1992)	D	-	-	No
He et al. (1998)	D,TP	G	Yes	No
Willatzen et al. (1998) Pettit et al. (1998)	G,D,F,TP	-	Yes	No
Jensen and Tummescheit (2002)	G,D,F	-	No	No
Leducq et al. (2003)	D	G	-	No
Yebra et al. (2005)	G	-	-	-
Rasmussen (2006)	D	G	-	No
Zhang and Zhang (2006)	D,TP	-	Yes	Yes
Kumar et al. (2008)	D	F	-	No
Mckinley and Alleyne (2008)	-	G,D	Yes	Yes
Bendapudi et al. (2008)	D,TP	G,D,SH	Yes	Yes
Eldredge et al. (2008)	TP	D	-	Yes
Li and Alleyne (2010)	D,TP	G,D,F,TP,SH	Yes	Yes
Gräber et al. (2010)	G,D,F	G,D,F	Yes	Yes
Mancini (2011) Cecchinato and Mancini (2012)	D,TP	-	Yes	Yes
Zapata et al. (2013)	G,F,SC	-	Yes	Yes

Table 7.2: Comparison of MBMs for two-phase flow

Adams et al. (1965) pioneered the MBMs, and Ray and Bowman (1976) developed a nonlinear model based on the work of Adams et al. (1965). Extensions of this work for solar applications are presented in Ray (1980, 1981). In Dhar and Soedel (1979), the MBM consists of a liquid zone and a vapor zone which are not in thermal equilibrium. The liquid and vapor phases then exchange heat. Although this model includes switching, it potentially introduces discontinuities and numerical instability (Mancini, 2011). The MBM developed in He et al. (1998) originally did not include switching between different configurations. However, the switching criteria for this MBM were introduced in Cheng and Asada (2006). Willatzen et al. (1998) introduced auxiliary equations to ensure relatively smooth state variables using pseudo-state tracking on inactive state variables. It was widely extended to assume a uniform pressure along the HE and hence did not model the conservation of momentum because pressure loss in evaporators and condensers is usually much lower than in other elements, e.g. valves or turbines. The MBMs developed in Adams et al. (1965); Ray and Bowman (1976); Yebra et al. (2005) are the only models in Tab. 7.2 which support pressure loss in the HE. The MBM in Yebra et al. (2005) is based on Jensen and Tummescheit (2002) and additionally considers the momentum conservation equation discretized by the FVM, the staggered grid scheme and the UDS. Zhang and Zhang (2006) introduced the time-dependent mean void fraction, showing smoother curves and improved robustness during long transients processes such as start-up and shutdown. A novel switching criterion based on the void fraction was introduced in Mckinley and Alleyne (2008), in which pseudo-state variables to represent inactive states, previously introduced in Willatzen et al. (1998) and Pettit et al. (1998), were used. The consistency of results was demonstrated through mass and energy conservation integral equations for several scenarios. In Bendapudi et al. (2008), the switching approach was based on the initialization of newly created dynamic states and claimed that smooth transitions are ensured without introducing large energy imbalances at the transition (Mancini, 2011). The pseudoquality variable used to predict the formation of SC liquid/SH vapor regions at the condenser/evaporator was introduced in Eldredge et al. (2008). Experimental validation is presented in this work. Li and Alleyne (2010) extended the condenser model previously developed in Mckinley and Alleyne (2008), allowing for a total of five configurations and developing a switching evaporator model. Experimental validation was presented by considering two test cases. Gräber et al. (2010) derived their MBMs in an elegant way from first principles, and the authors also proposed a validation procedure based on infrared thermography, which allows for the high resolution measurement of wall temperatures in terms of spatial and time coordinates. Mancini (2011) also used pseudo-state variables, and a new switching criterion and choice of state variables were presented to ensure mass and energy conservation when simulating low mean void fraction start-ups, even with high numerical integration step-sizes. Zapata et al. (2013) introduced an explicit calculation of evaporator outlet mass flow rate based on average and outlet pressures.

None of the presented MBMs supports dynamic switching between all possible flow configurations (G, D, F, SC, TP, SH) in evaporators and condensers. To the knowledge of the author, only the MBMs in Jensen and Tummescheit (2002), Zhang and Zhang (2006) and Gräber et al. (2010) were developed using EOO modeling languages, and more precisely Modelica. Certainly, the equation-based paradigm was applied when developing the models, although the OO paradigm was not completely applied in these cases.

The MBMs presented in this book support all possible flow configurations in evaporators and condensers, and they also support switching between all possible flow configurations and consider a time-variant mean void fraction. The equation-based paradigm is applied because the causality is not fixed; a clear example is that the same basic models are used for evaporators and condensers. Furthermore, the OO paradigm is also applied to its maximum extent, because each basic model (SC, TP, SH) in a HE model is a reusable model itself. This allows the modeler to use the same basic models, and hence the same equations, independent of the type of HE.

7.3 Assumptions

In order to develop a low-order model but to reflect the principal dynamics, a number of assumptions must be considered.

- Horizontal orientation and a one-dimensional case.
- Cylindrical geometry with a constant cross-sectional area.
- Uniform but time-dependent pressure along the evaporator.
- HEM (cf. § 5.4). Hence, thermodynamic equilibrium between phases.
- Constant heat flow rate per unit area in each CV and linear specific enthalpy distribution in each CV. The last is justified in § 8.4.
- Average properties in each CV.
- Negligible gravitational forces, changes in kinetic energy and viscous stresses.
- Heat conduction and radiation is negligible in the fluid.
- Heat conduction is also negligible in the pipe wall.

7.4 Goals

The following list summarizes the main goals in the development of MBMs.

- General models for the three basic regions: SC liquid, TP flow and SH vapor regions for evaporators and condensers must be developed based on first principles.
- Development of compounded models from the basic three models to create general, flooded and dry evaporators/condensers.
- Development of the models applying the EOO paradigm.
- Development of a Modelica library containing the MBMs.
- Switching mechanisms between all possible flow configurations.
- Consider the mean void fraction as time variant and its time derivative in the mathematical formulation.
- Represent in some way regions/CVs in inactive states.

7.5 One-dimensional Governing Equations

The most straightforward way to derive model equations is taken from the time-dependent equations for conservation laws. Considering the assumptions presented in § 7.3, the differential formulation for the conservation of mass and energy in a fluid are represented by Eqs. 7.1 and 7.2, respectively (Anderson, 1995; Thomas, 1999), whereas Eq. 7.3 (Tummescheit, 2002; Jensen, 2003) defines the conservation of energy in the pipe wall. The equations for the conservation of mass and energy in the fluid are the same that were presented in § 6.2 (Eqs. 6.1 and 6.2) but are particularized by considering the assumptions in § 7.3. These equations have been repeated here for readability. A simplified equation of the conservation of energy in the pipe wall was also previously introduced in § 5.2.1.2 (Eq. 5.2). However, the general equation is Eq. 7.3, which will be clarified in § 7.9.

$$\frac{\partial(\rho A)}{\partial t} + \frac{\partial(\rho A w)}{\partial z} = 0, \tag{7.1}$$

$$\frac{\partial(\rho u A)}{\partial t} + \frac{\partial\left(\rho w\left(u + \frac{p}{\rho}\right)A\right)}{\partial z} = \dot{q}_{htf}, \tag{7.2}$$

$$A_w \rho_w c_{p,w} \frac{\partial T_w}{\partial t} = \dot{q}_w - \dot{q}_{htf}. \tag{7.3}$$

Considering, the mass flow rate, $\dot{m} = \rho w A$, and the specific enthalpy definition, $h = u + p/\rho$, Eqs. 7.1 and 7.2 yield Eqs. 7.4 and 7.5

$$\frac{\partial(\rho A)}{\partial t} + \frac{\partial \dot{m}}{\partial z} = 0, \tag{7.4}$$

$$\frac{\partial(\rho u A)}{\partial t} + \frac{\partial \dot{m} h}{\partial z} = \dot{q}_{htf}, \tag{7.5}$$

7.6 Definition of a One-dimensional Control Volume

The moving boundary method is based on the division of the HE in different CVs, depending on the fluid phase: SC liquid, TP flow and SH vapor CVs. Fig. 7.3 represents a CV, in which the lumped thermodynamic properties are average in some way and uniform but time-dependent in each CV $(\bar{h}, \bar{T}, \bar{\rho})$. Pressure (p) is not denoted by a mean value, because there is only one pressure for the whole evaporator. It is assumed that pressure is uniform but time-dependent along the HE. The cross-sectional areas (A, A_w) are constant. For the sake of readability and simplicity, the subscripts associated with the particular CV variables under consideration are omitted and only specified when needed, e.g. if the SC liquid CV is considered, thermodynamic properties should be $(\bar{h}_{sc}, \bar{T}_{sc}, \bar{\rho}_{sc})$.

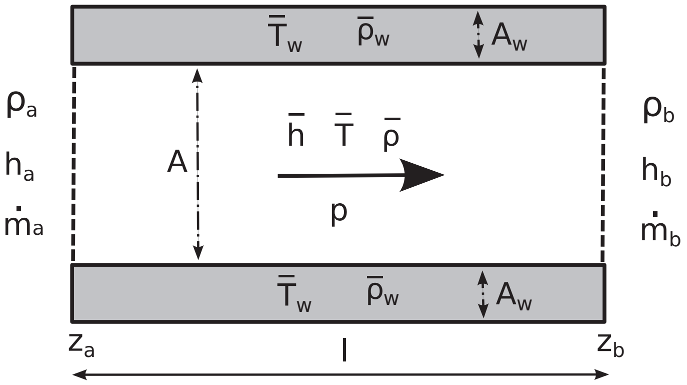


Figure 7.3: Control volume

Each CV has three interfaces or boundaries. One leads with the pipe wall, where thermodynamic properties are also considered in their mean values $(\bar{T}_w, \bar{\rho}_w)$, while the other two interfaces lead with adjacent CVs or the inlet or outlet boundaries of the HE. Flow direction is defined by the arrow (cf. Fig. 7.3). Therefore, inlet flow thermodynamic properties correspond to the a subscript variables $(\rho_a, h_a, \dot{m}_a)$ and outlet flow thermodynamic properties are defined by the b subscript variables $(\rho_b, h_b, \dot{m}_b)$. The barrier between two adjacent CVs is not fixed and it may move.

7.6.1 General One-dimensional Balance Equations in a CV

This derivation process is analogous to that performed in (Jensen, 2003). Considering the one-dimensional CV introduced in § 7.6, integrating the Eqs. 7.4, 7.5 and 7.3 over the $[z_a, z_b]$ interval and substituting u from the enthalpy definition, $u = h - p/\rho$, Eqs. 7.4, 7.5 and 7.3 yield Eqs. 7.6, 7.7 and 7.8. Note that the cross-sectional area (A) is constant.

$$A \int_{z_a}^{z_b} \frac{\partial \rho}{\partial t}\, dz = \dot{m}_a - \dot{m}_b, \tag{7.6}$$

$$A \int_{z_a}^{z_b} \frac{\partial (\rho h)}{\partial t}\, dz - A \int_{z_a}^{z_b} \frac{\partial p}{\partial t}\, dz = \dot{m}_a h_a - \dot{m}_b h_b + \dot{q}_{htf}(z_b - z_a). \tag{7.7}$$

$$A_w \rho_w c_{p,w} \int_{z_a}^{z_b} \frac{\partial T_w}{\partial t}\, dz = (\dot{q}_w - \dot{q}_{htf})(z_b - z_a). \tag{7.8}$$

Applying the Leibniz integral rule (Woods, 1926) (cf. Appx. A.1) in the first terms of the left-hand side of Eqs. 7.6, 7.7 and 7.8 yields Eqs. 7.9, 7.10 and 7.11. Note that the pressure (p) is uniform but time-dependent along the HE ($\partial p/\partial t = dp/dt$ in Eq. 7.7). Hence, its value does not change in the $[z_a, z_b]$ interval. The length in any CV is defined by $l = z_b - z_a$, according to Fig. 7.3

$$A\frac{d}{dt}\int_{z_a}^{z_b} \rho\, dz + \rho_a A\frac{dz_a}{dt} - \rho_b A\frac{dz_b}{dt} = \dot{m}_a - \dot{m}_b, \tag{7.9}$$

$$\begin{aligned} &A\frac{d}{dt}\int_{z_a}^{z_b} \rho h\, dz - Al\frac{dp}{dt} + A\rho_a h_a\frac{dz_a}{dt} - A\rho_b h_b\frac{dz_b}{dt} \\ &= \dot{m}_a h_a - \dot{m}_b h_b + \dot{q}_{htf} l. \end{aligned} \tag{7.10}$$

$$A_w \rho_w c_{p,w}\left(\frac{d}{dt}\int_{z_a}^{z_b} T_w\, dz + T_{w_a}\frac{dz_a}{dt} - T_{w_b}\frac{dz_b}{dt}\right) = (\dot{q}_w - \dot{q}_{htf})l. \tag{7.11}$$

7.7 Fluid Balance Equations

This section defines density calculation at the CV boundaries and introduces the conservation equations of mass and energy for the one-phase and two-phase regions. The derivation process is analogous to that performed in (Jensen, 2003), but does not fix the boundaries' specific enthalpies to the saturation conditions, calculates the mean void fraction by integration and does not neglect the time derivative of the mean void fraction.

Density (ρ), temperature (T), specific enthalpy (h), density and temperature at saturation conditions (h', h'', ρ', ρ'', T', T'') and the partial derivatives of specific enthalpy and density at saturation conditions ($\partial h'/\partial p$, $\partial h''/\partial p$, $\partial \rho'/\partial p$, $\partial \rho''/\partial p$) can be calculated using a library of thermodynamic properties for the particular fluid under consideration.

7.7.1 Density Calculation at CV Boundaries

The inlet and outlet densities of a CV can be calculated by pressure and specific enthalpy at the appropriate boundary:

$$\rho_a = \rho(p, h_a), \tag{7.12}$$

$$\rho_b = \rho(p, h_b). \tag{7.13}$$

7.7.2 One-phase Flow Region (SC liquid or SH vapor)

The distribution of specific enthalpy in a CV is considered linear, Eq. 7.14, as defined in § 7.3.

$$\bar{h} = \frac{1}{2}(h_a + h_b), \tag{7.14}$$

therefore, the time derivative of the average specific enthalpy is:

$$\frac{d\bar{h}}{dt} = \frac{1}{2}\left(\frac{dh_a}{dt} + \frac{dh_b}{dt}\right). \tag{7.15}$$

The average density (Eq. 7.16) and average temperature (Eq. 7.17) in a CV are approximated by treating density and temperature as the functions of pressure and the average specific enthalpy in such a CV (Jensen, 2003).

$$\bar{\rho} \approx \rho(p, \bar{h}), \tag{7.16}$$

$$\bar{T} \approx T(p, \bar{h}). \tag{7.17}$$

The time derivative for average density can be calculated as follows. First, we consider the chain rule (cf. Appx. A.3) and Eq. 7.16, following which we substitute $d\bar{h}/dt$ with Eq. 7.15.

$$\frac{d\bar{\rho}}{dt} = \frac{\partial\bar{\rho}}{\partial p}\frac{dp}{dt} + \frac{\partial\bar{\rho}}{\partial h}\frac{d\bar{h}}{dt} = \frac{\partial\bar{\rho}}{\partial p}\frac{dp}{dt} + \frac{1}{2}\frac{\partial\bar{\rho}}{\partial h}\left(\frac{dh_a}{dt} + \frac{dh_b}{dt}\right). \tag{7.18}$$

7.7.2.1 Mass Balance Equation

The rate of mass change (first term in Eq. 7.9) is defined by Eq. 7.19 by applying the mean-value theorem (Piskunov, 1965) (cf. Appx. A.2) in the first step and considering l. In the second step, the product rule for derivatives (cf. Appx. A.4) is applied.

$$\frac{d}{dt}\int_{z_a}^{z_b} \rho\, dz = \frac{d}{dt}(\bar{\rho} l) = l\frac{d\bar{\rho}}{dt} + \bar{\rho}\frac{dl}{dt}. \tag{7.19}$$

Substituting Eq. 7.19 in Eq. 7.9 yields the mass balance for the one-phase regions,

$$A\left(l\frac{d\bar{\rho}}{dt} + \bar{\rho}\frac{dl}{dt}\right) + \rho_a A\frac{dz_a}{dt} - \rho_b A\frac{dz_b}{dt} = \dot{m}_a - \dot{m}_b, \tag{7.20}$$

where $\bar{\rho}$, ρ_a, ρ_b and $d\bar{\rho}/dt$ are calculated from Eqs. 7.16, 7.12, 7.13 and 7.18, respectively.

7.7.2.2 Energy Balance Equation

The rate of enthalpy change (first term in Eq. 7.10) can be calculated by employing Eq. 7.21, applying the mean-value theorem (Piskunov, 1965) (cf. Appx. A.2) in the first step and considering l. We can then approximate

$\overline{\rho h} \approx \bar{\rho}\bar{h}$ (Jensen, 2003) in the second step and apply the product rule for derivatives (cf. Appx. A.4) in the last step.

$$\frac{d}{dt}\int_{z_a}^{z_b} \rho h \, dz = \frac{d}{dt}(\overline{\rho h} l) \approx \frac{d}{dt}(\bar{\rho}\bar{h}l) = \bar{\rho}\bar{h}\frac{dl}{dt} + \bar{\rho}\frac{d\bar{h}}{dt}l + \frac{d\bar{\rho}}{dt}\bar{h}l. \tag{7.21}$$

Replacing Eq. 7.21 in Eq. 7.10 yields Eq. 7.22, the energy balance for the one-phase regions.

$$\begin{aligned} &A\left(\bar{\rho}\bar{h}\frac{dl}{dt} + \bar{\rho}\frac{d\bar{h}}{dt}l + \frac{d\bar{\rho}}{dt}\bar{h}l\right) - Al\frac{dp}{dt} + A\rho_a h_a\frac{dz_a}{dt} - A\rho_b h_b\frac{dz_b}{dt} = \\ &\dot{m}_a h_a - \dot{m}_b h_b + \dot{q}_{htf} l, \end{aligned} \tag{7.22}$$

where $\bar{\rho}$, ρ_a, ρ_b, $\bar{h}$, $d\bar{\rho}/dt$ and $d\bar{h}/dt$ are calculated from Eqs. 7.16, 7.12, 7.13, 7.14, 7.18 and 7.15, respectively.

7.7.3 Two-phase Flow Region

Mean density in the TP flow region is defined by Eq. 7.23, assuming a homogeneous two-phase flow. This equation was previously derived in § 4.3.4 (Eq. 4.24). The average temperature is calculated as the saturation temperature by considering thermodynamic equilibrium between phases, while the saturation temperature is a function of pressure in Eq. 7.24.

$$\bar{\rho} = \bar{\gamma}\rho'' + (1 - \bar{\gamma})\rho', \tag{7.23}$$

$$\bar{T} = T'(p) = T''(p). \tag{7.24}$$

The mean void fraction ($\bar{\gamma}$) and its time derivative can be calculated from Eqs. 4.29 and 4.32, which were also derived and presented in § 4.4. The densities and specific enthalpies of saturated liquid and saturated vapor can be calculated by considering pressure (Eq. 7.25a-Eq. 7.25d).

$$h' = h'(p), \tag{7.25a}$$
$$h'' = h''(p), \tag{7.25b}$$
$$\rho' = \rho(p, h'), \tag{7.25c}$$
$$\rho'' = \rho(p, h''). \tag{7.25d}$$

The distribution of specific enthalpy in the TP flow CV is considered linear.

7.7.3.1 Mass Balance Equation

The rate of mass change (first term in Eq. 7.9) is defined by Eq. 7.26. In the first step, the mean-value theorem (Piskunov, 1965) (cf. Appx. A.2) is applied and considers l. In the second step, Eq. 7.23 is used. Afterwards, the product rule for derivatives is applied (cf. Appx. A.4), following which the terms of the equation are reorganized. Finally, the chain rule (cf. Appx. A.3) is applied, considering that ρ' and ρ'' only depend on pressure.

$$\begin{aligned}&\frac{d}{dt}\int_{z_a}^{z_b}\rho\,dz=\frac{d}{dt}(\bar{\rho}l)\approx\frac{d}{dt}(l(\bar{\gamma}\rho''+(1-\bar{\gamma})\rho'))\\&=\frac{dl}{dt}(\bar{\gamma}\rho''+(1-\bar{\gamma})\rho')+l\left(\frac{d\bar{\gamma}}{dt}(\rho''-\rho')+\bar{\gamma}\frac{d\rho''}{dp}\frac{dp}{dt}+(1-\bar{\gamma})\frac{d\rho'}{dp}\frac{dp}{dt}\right).\end{aligned}\tag{7.26}$$

Replacing Eq. 7.26 in Eq. 7.9 yields the mass balance for the two-phase flow region in Eq. 7.27.

$$\begin{aligned}&A\left(\frac{dl}{dt}(\bar{\gamma}\rho''+(1-\bar{\gamma})\rho')+l\left(\frac{d\bar{\gamma}}{dt}(\rho''-\rho')+\bar{\gamma}\frac{d\rho''}{dp}\frac{dp}{dt}+(1-\bar{\gamma})\frac{d\rho'}{dp}\frac{dp}{dt}\right)\right)\\&+\rho_aA\frac{dz_a}{dt}-\rho_bA\frac{dz_b}{dt}=\dot{m}_a-\dot{m}_b.\end{aligned}\tag{7.27}$$

The ρ_a, ρ_b, ρ', ρ'', $\bar{\gamma}$ and $d\bar{\gamma}/dt$ variables are calculated from Eqs. 7.12, 7.13, 7.25c, 7.25d, 4.29 and 4.32, respectively.

7.7.3.2 Energy Balance Equation

Assuming a homogeneous two-phase flow, the ρh product can be calculated in the TP region by using Eq. 7.28. This equation was previously derived in § 4.3.5 (Eq. 4.27).

$$\bar{\rho}\bar{h}=\bar{\gamma}\rho''h''+(1-\bar{\gamma})\rho'h'.\tag{7.28}$$

The first term of Eq. 7.10 is calculated by using Eq. 7.29. The procedure is analogous to that applied to the two-phase flow mass balance equation by applying the mean-value theorem (Piskunov, 1965) (cf. Appx. A.2), considering l, approximating $\overline{\rho h}\approx\bar{\rho}\bar{h}$ (Jensen, 2003), using Eq. 7.28, applying the product rule for derivatives (cf. Appx. A.4), reorganizing terms and finally applying the chain rule (cf. Appx. A.3).

$$\begin{aligned}
&\frac{d}{dt}\int_{z_a}^{z_b}\rho h\,dz = \frac{d}{dt}(\overline{\rho h}l) \approx \frac{d}{dt}(\bar{\rho}\bar{h}l) \approx \frac{d}{dt}(l(\bar{\gamma}\rho''h'' + (1-\bar{\gamma})\rho'h')) \\
&= \frac{dl}{dt}(\bar{\gamma}\rho''h'' + (1-\bar{\gamma})\rho'h') + l\left(\frac{d\bar{\gamma}}{dt}\left(\rho''h'' - \rho'h'\right) + \bar{\gamma}\frac{d\rho''}{dp}\frac{dp}{dt}h''\right. \\
&\left.+\bar{\gamma}\rho''\frac{dh''}{dp}\frac{dp}{dt} + (1-\bar{\gamma})\frac{d\rho'}{dp}\frac{dp}{dt}h' + (1-\bar{\gamma})\rho'\frac{dh'}{dp}\frac{dp}{dt}\right).
\end{aligned} \tag{7.29}$$

Replacing Eq. 7.29 in Eq. 7.10 yields the energy balance for the two-phase region in Eq. 7.30.

$$\begin{aligned}
&A\left(\frac{dl}{dt}(\bar{\gamma}\rho''h'' + (1-\bar{\gamma})\rho'h') + l\left(\frac{d\bar{\gamma}}{dt}\left(\rho''h'' - \rho'h'\right) + \bar{\gamma}\frac{d\rho''}{dp}\frac{dp}{dt}h'' + \right.\right. \\
&\left.\left.\bar{\gamma}\rho''\frac{dh''}{dp}\frac{dp}{dt} + (1-\bar{\gamma})\frac{d\rho'}{dp}\frac{dp}{dt}h' + (1-\bar{\gamma})\rho'\frac{dh'}{dp}\frac{dp}{dt}\right)\right) - Al\frac{dp}{dt} + A\rho_a h_a\frac{dz_a}{dt} - \\
&A\rho_b h_b\frac{dz_b}{dt} = \dot{m}_a h_a - \dot{m}_b h_b + \dot{q}_{htf} l.
\end{aligned} \tag{7.30}$$

The ρ_a, ρ_b, ρ', ρ'', h', h'', $\bar{\gamma}$ and $d\bar{\gamma}/dt$ variables are calculated from Eqs. 7.12, 7.13, 7.25c, 7.25d, 7.25a, 7.25b, 4.29 and 4.32, respectively.

7.8 Heat Exchanger Models

When modeling compound models (not only one CV model), additional equations are required besides the CV governing equations. First, Eq. 7.31 connects the CVs, where j and $j+1$ denote two adjacent CVs.

$$p_{j_b} = p_{j+1_a}, \qquad h_{j_b} = h_{j+1_a}, \qquad \dot{m}_{j_b} = \dot{m}_{j+1_a}. \tag{7.31}$$

Second, equations which relate boundary CV specific enthalpies with values at saturation conditions depend on the HE and CV. They are detailed in § 7.8.1 and § 7.8.2 for evaporators and condensers, respectively.

7.8.1 Evaporators

An easy way to relate boundary CV specific enthalpies with values at saturation conditions, considering the case of a general or flooded evaporator in the SC liquid CV, is to introduce Eq. 7.32. However, there is a problem with this approach if switching MBMs are considered.

$$h_b = h'. \tag{7.32}$$

Suppose that a flooded evaporator is being modeled, where the outlet fluid is TP flow – h_b for the SC liquid CV is not a state variable because it depends on pressure, so Eq. 7.32 is valid and h_b is an algebraic variable. However, if the outlet fluid turns into SC liquid due to a change in the model inputs, Eq. 7.32 is no longer valid and h_b is a state variable. Such a model is called a *variable-structure model.*

In a variable-structure model the number/type of equations or variables can change. On the other hand, a static-structure model implies that the number of equations as well as the number of algebraic and state variables remain the same (cf. § 2.6.5).

Variable-structure models are not currently supported by most M&S tools (including Modelica tools), and although there exist some modeling languages and tools that support variable-structure models, none of the existing variable-structure M&S tools supports the handling of higher-index systems (Zimmer, 2010). For this reason, Modelica is still our preferred modeling language, but it must be taken into account that only static-structure models can be simulated.

Therefore, the number of equations must remain the same in all different configurations of our model, and h_b for the different CVs must always be a state variable. Consequently, its value cannot be fixed to any algebraic variable and neither can h_a, because it is connected to h_b from the CV to the left, except for the case in the first CV, where the h_a value can be freely established.

In order to ensure a static-structure model, if a general or flooded evaporator is considered (cf. Figs. 7.2a and 7.2c), Eq. 7.33 is required for the SC liquid CV, and the initial value for h_b must be also set to h'.

HE	SC liquid CV	SH vapor CV
Evaporator	$\frac{dh_b}{dt} = \frac{dh'}{dt} \quad (7.33)$	$\frac{dh_a}{dt} = \frac{dh''}{dt} \quad (7.34)$
Condenser	$\frac{dh_a}{dt} = \frac{dh'}{dt} \quad (7.35)$	$\frac{dh_b}{dt} = \frac{dh''}{dt} \quad (7.36)$

Table 7.3: Specific enthalpy constraints depending on the HE and CV

If a general or dry-expansion evaporator is considered (cf. Figs. 7.2a and 7.2e), Eq. 7.34 is required for the SH vapor CV, and the initial value for h_a must be also set to h''.

7.8.2 Condensers

In order to ensure a static-structure condenser model, if a general or flooded condenser is considered (cf. Figs. 7.2b and 7.2d), Eq. 7.35 is required for the SC liquid CV, and the initial value for h_a must be also set to h'. If a general or dry condenser is considered (cf. Figs. 7.2b and 7.2f), Eq. 7.36 is required for the SH vapor CV, and the initial value for h_b must be also set to h''.

7.9 Pipe Wall Energy Balance Equation

Integrating the T_w term via the mean-value theorem (Piskunov, 1965) (cf. Appx. A.2), Eq. 7.11 leads to Eq. 7.37.

$$A_w \rho_w c_{p,w} \left(\frac{d}{dt}(\bar{T}_w(z_b - z_a)) + T_{w_a}\frac{dz_a}{dt} - T_{w_b}\frac{dz_b}{dt} \right) = (\dot{q}_w - \dot{q}_{htf})(z_b - z_a). \tag{7.37}$$

Calculating the derivative of the product (cf. Appx. A.4), rearranging terms and dividing by $l = z_b - z_a$, Eq. 7.37 leads to Eq. 7.38.

$$A_w \rho_w c_{p,w} \left(\frac{d\bar{T}_w}{dt} + \frac{\bar{T}_w - T_{w_b}}{l}\frac{dz_b}{dt} + \frac{T_{w_a} - \bar{T}_w}{l}\frac{dz_a}{dt} \right) = \dot{q}_w - \dot{q}_{htf}. \tag{7.38}$$

Eq. 7.38 models the pipe wall energy balance indistinctly in each CVs. Boundary temperatures (T_{w_a}, T_{w_b}) are discussed in § 7.9.1.

7.9.1 Wall Temperatures at the Boundaries

In Chap. 5, a pipe wall model for an absorber tube was presented (cf. § 5.2.1.2). However, wall model was designed by considering a fixed length for each CV. Eq. 7.38 is the same as Eq. 5.2 if the time derivatives of the boundary lengths are zero. In the MBMs, the wall temperature at the interface of two adjacent CVs is an important factor for robust switching between different configurations.

Wall temperatures at the interfaces have been modeled by using weighted mean equations, introduced in Zhang and Zhang (2006). As this paper points out, this approach is closer to the practical situation, whereby it is continuous and smooth during switching between different configurations. The equation employed to compute temperature at the interfaces is Eq. 7.39, where j denotes a CV and n is the total number of CVs.

$$\begin{aligned} T_{w_{a_1}} &= \bar{T}_{w_1}, \\ T_{w_{a_j}} &= \bar{T}_{w_{j-1}} \frac{l_j}{l_{j-1}+l_j} + \bar{T}_{w_j} \frac{l_{j-1}}{l_{j-1}+l_j}, \qquad \forall j \in [2,n] \\ T_{w_{b_j}} &= \bar{T}_{w_j} \frac{l_{j+1}}{l_j+l_{j+1}} + \bar{T}_{w_{j+1}} \frac{l_j}{l_j+l_{j+1}}, \qquad \forall j \in [1,n-1] \\ T_{w_{b_n}} &= \bar{T}_{w_n}. \end{aligned} \tag{7.39}$$

7.10 Switching

Switching from one configuration to another implies the disappearance of an existing CV or the appearance of a new one, e.g. when switching from a general evaporator to a flooded evaporator or vice versa. This section elaborates on how such transitions are captured by the model.

When the CV is active, its governing equations correspond to the equations described in § 7.7.2 or § 7.7.3, depending on the fluid phase. However, a different set of equations is required to describe the CV in its inactive state, which is also explained in this section. Figs. 7.4 and 7.5 show the events that trigger the appearance or disappearance of each CV when considering evaporators and condensers, respectively. Note that all possible flow configurations are supported.

In Appx. E.1, the Modelica code, which manages switching between the different configurations of a moving boundary general evaporator model, is described.

7.10.1 Disappearance of a Control Volume

A CV disappears (becomes inactive) when Eq. 7.40 becomes true (cf. Figs. 7.4 and 7.5), where l_{min} denotes a threshold that specifies the minimum length of an active CV. This value cannot be zero, in order to avoid structural singularities during simulation, and it depends on the numerical integration tolerance. The default value for this parameter has been set to 10^{-4} m.

$$l < l_{min}. \tag{7.40}$$

A relay hysteresis has been considered in order to avoid convergence problems during simulation due to numerical errors, Eq. 7.41, in case the length of the CV is close to the minimum length (l_{min}). The reason why this is required is explained in § 7.10.3. The default value for the l_{tol} parameter has been set to 10^{-4} m.

$$\begin{aligned} & l < l_{min}, && \textit{active} \rightarrow \textit{inactive}, \\ & l < l_{min} + l_{tol}, && \textit{keep being inactive}. \end{aligned} \tag{7.41}$$

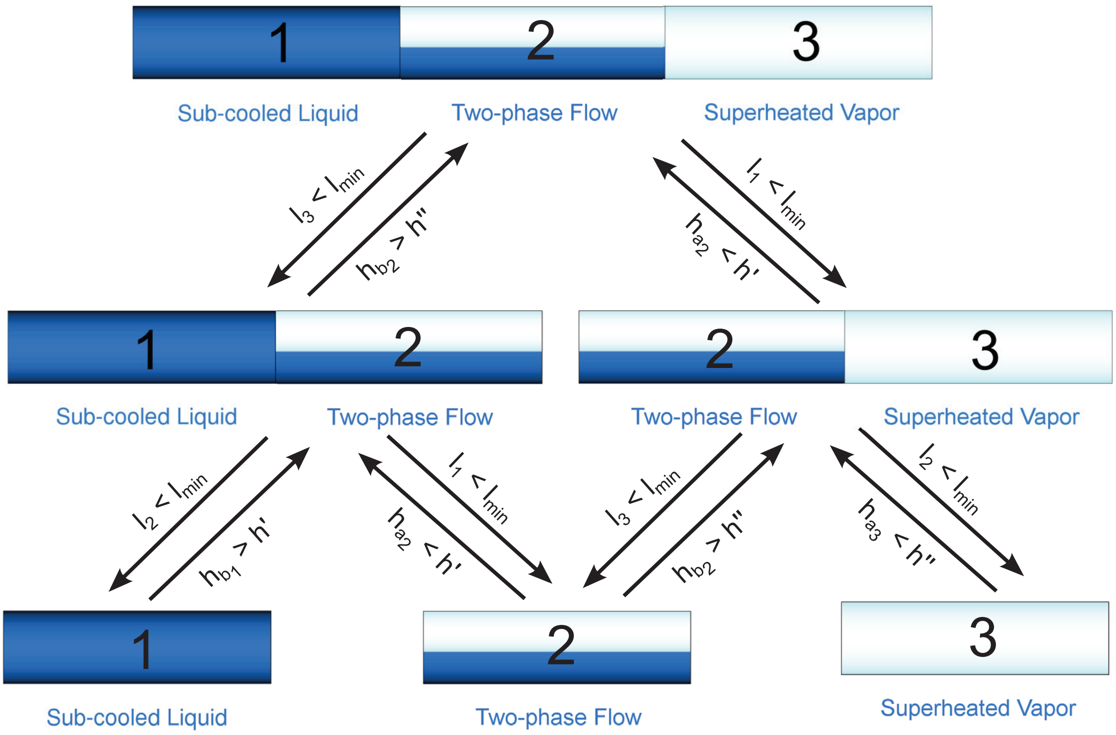


Figure 7.4: Switching in a general evaporator

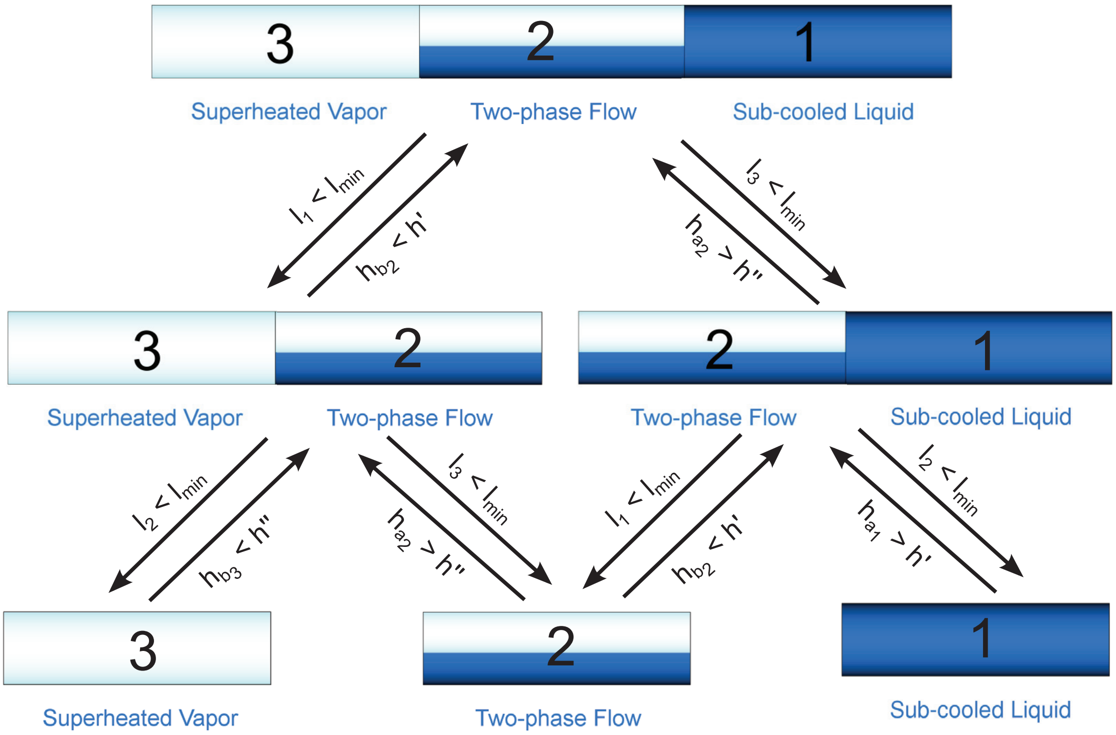


Figure 7.5: Switching in a general condenser

7.10.2 Control Volume in an Inactive State

When any CVs is inactive, the mass and energy balance equations (Eqs. 7.20 and 7.22 or Eqs. 7.27 and 7.30, depending on the CV fluid phase), are substituted by Eqs. 7.42 and 7.43, respectively. These equations guarantee that the CV is inactive and does not act on the fluid.

$$\dot{m}_a = \dot{m}_b, \tag{7.42}$$

$$\frac{dh_a}{dt} = \frac{dh_b}{dt}. \tag{7.43}$$

Furthermore, when a one-phase CV is inactive, the equation that constrains a boundary specific enthalpy must be substituted by Eq. 7.44. Tab. 7.3 shows which equation must be substituted, depending on the kind of CV and HE.

$$\frac{dl}{dt} = 0. \tag{7.44}$$

Finally, the pipe wall energy balance equation, Eq. 7.38, is substituted by Eq. 7.45, where j denotes the inactive CV and k denotes one adjacent CVs.

$$\frac{d\bar{T}_{w_j}}{dt} = \frac{d\bar{T}_{w_k}}{dt} \tag{7.45}$$

7.10.3 Appearance of a Control Volume

The event triggering the appearance of a CV depends on the particular CV and also on the kind of HE. Additionally, it is also checked to ascertain whether the CV is inactive, which is done in Eq. 7.41. For this reason, a relay hysteresis in Eq. 7.40 is required, in order to avoid convergence problems due to numerical errors in inactive CV lengths, i.e. a particular CV becomes active when it should be inactive.

Relay hysteresis has been also considered in CV appearance events in order to avoid convergence difficulties. For instance, when switching from the SC to SC-TP configurations, the event is $h_{b_1} > h'$ (cf. Fig. 7.4). In the SC configuration, $h_{b_1} = h_{a_2} \approx h_{b_2}$ (because the TP region is inactive), but in the SC-TP configurations, h_{b_2} must be greater than h_{a_2}. For this reason, the event is $h_{b_1} > h' + \xi$, where after the event $h_{b_1} = h_{a_2} = h'$ and $h_{b_2} = h' + \xi$, in order to avoid convergence problems. The default value, which for ξ is 10^{-4} J/kg, depends on the numerical integration tolerance.

Evaporator. Fig. 7.4 shows in the arrows the events that trigger the appearance of new CVs. When one of these events occurs, the particular CV state is changed from inactive to active. Hence, the governing equations are switched from the inactive equations (cf. § 7.10.2) to the mass and energy balance equations (Eqs. 7.20 and 7.22 or Eqs. 7.27 and 7.30, depending on the CV fluid phase).

Condenser. The situation is the same as for evaporators, but in this case Fig. 7.5 shows the events which trigger the appearance of new CVs.

7.11 Additional Equations

In order to complete the HE models, some additional equations are required, which are described in this section.

7.11.1 Heat Flow Rates

Thermal conductivity in the pipe wall is neglected, as previously mentioned in § 7.3. Convective heat transfer from the pipe wall to the CV ($\dot{Q}$) can be calculated by applying Newton's law of cooling (Eq. 7.46). This equation depends on the HE geometry, because the inner surface area (S_{a_i}) and length (l) of the CV are required. The geometry will be defined in § 7.11.3. Eq. 7.46 also requires the convective heat transfer coefficient (α) between the pipe wall and the fluid, which depends on the fluid phase. This value can be considered constant or obtained from an empirical correlation. HTCs are detailed in § 7.12.3.5, while $\bar{T}_w$ is calculated by Eq. 7.38 and $\bar{T}$ is computed by Eq. 7.17 or Eq. 7.24, depending on the fluid phase.

$$\dot{Q} = \dot{q}l = \alpha S_{a,i}(\bar{T}_w - \bar{T}). \tag{7.46}$$

The heat flow rate in the pipe wall ($\dot{Q}_w$) is defined by Eq. 7.47, which depends on the heat flow rate between the CV and the pipe wall, as well as between the pipe wall and the ambient, which is specific to the particular system being modeled.

$$\dot{Q}_w = \dot{q}_w l = \dot{Q} + \dot{Q}_{amb}. \tag{7.47}$$

7.11.2 Heat Exchanger Length

This equation relates the length of the HE to the length of each CV. Eq. 7.48 is the general equation where j denotes a particular CV and n the number of CVs in the HE.

$$l_{he} = \sum_{j=1}^{n} l_j. \tag{7.48}$$

7.11.3 Heat Exchanger Geometry

The cross-sectional areas for each CV and the pipe wall, the A and A_w values, are required for the mass and energy fluid balance equations and the energy pipe wall balance equation. Considering a cylindrical geometry, where the constant inner and outer diameters of the pipe wall are d_i and d_o, respectively, the cross-sectional areas for the CV and the pipe wall are defined by Eqs. 7.49 and 7.50 according to Eq. 5.3.

$$A = \frac{\pi d_i^2}{4}, \tag{7.49}$$

$$A_w = \frac{\pi (d_o - d_i)^2}{4}. \tag{7.50}$$

The inner surface area (S_{a_i}) for the cylindrical geometry is defined by Eq. 7.51, according to Eq. 5.3.

$$S_{a_i} = \pi d_i l. \tag{7.51}$$

7.12 The MBMs Modelica Library

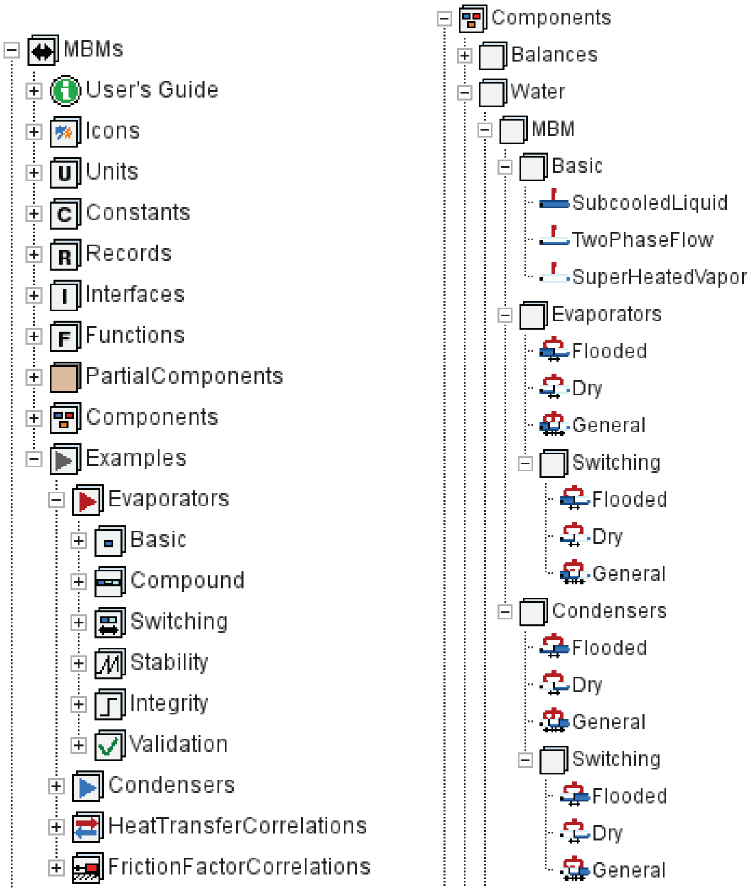
This section introduces the Modelica library that implements the mathematical models previously described, which is called *the MBMs (Moving Boundary Models) library*.

7.12.1 Library Structure

Fig. 7.6a shows the main packages that make up the *MBMs* library, each of which is briefly described in the following list. Modelica 3.2 (Modelica Association, 2010) and Dymola 2013 (Dassault Systemes, 2012) have been used to implement the *MBMs* library.

- *User Guide.* Provides documentation about the main models in the library. It also offers release notes and a to-do list.
- *Icons.* Artwork for the package as well as model icons and graphics.
- *Units.* Units not included in the MSL.
- *Constants.* Useful constants for the *MBMs* library.
- *Records.* Record definitions are included in this package, e.g. active or inactive states in a CV.
- *Interfaces.* Common interfaces for the models. There are sub-packages for flow and heat. The *Modelica Fluid* and *Modelica Thermal* ports are used in these interfaces.

- *Functions.* Common functions used across the library.
- *PartialComponents.* Base classes which add some functionality, but are not instantiable, are included in this package, e.g. *Volume class* (CV model), *MultiVolume class* (HE model), *HeatTransferCorrelation class* and *FrictionFactorCorrelation class.*
- *Components.* Fig. 7.6b shows the *Components.Water.MBM* package in an expanded view, where the basic (CV) and compound (HE) models can be seen. There are two different versions of compound models, with the only difference being whether the models support switching or not. If switching is not required when modeling a particular system, compound models which do not support switching can be used in order to lower CPU usage.
- *Examples.* Evaporators, condensers, HTCs and FFCs have been thoroughly tested through several test cases. Basic, compound and switching compound models have been tested in stability, integrity and validation tests.



(a) Packages (b) Components

Figure 7.6: The *MBMs* library

7.12.2 Interfaces

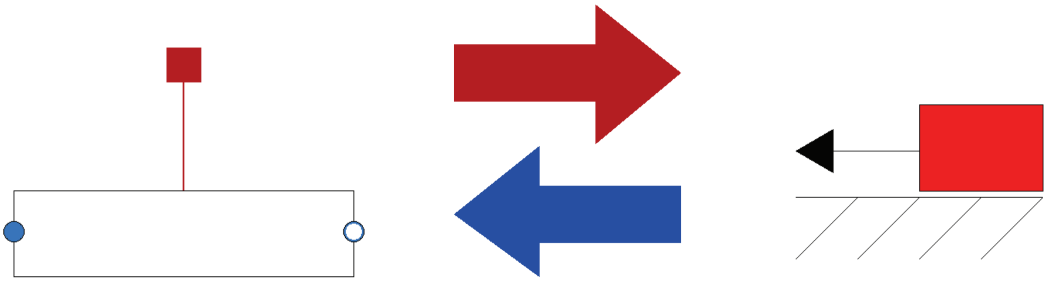
Modelica Fluid and *Modelica Thermal* ports have been used throughout, in order to define interfaces in the *MBMs* library. This guarantees that the *MBMs* library is compatible with any component from the MSL 3.2 (Modelica Association, 2010) or with third-party components that also make use of these interfaces.

7.12.3 Partial Base Classes

The most remarkable partial base classes in the *MBMs* library are: the *Volume*, the *MultipleVolume*, *Geometry*, *Pipe wall*, *HeatTransferCorrelation* and *FrictionFactorCorrelation* classes.

7.12.3.1 Volume Class

The *Volume* class defines the fluid and heat ports, the medium and some additional thermodynamic properties, which are not included in the medium, as well as the state (active or inactive) and the geometry of the CV. The fluid and heat ports belong to the *Modelica Fluid* and *Modelica Thermal* libraries. The medium can be any two-phase medium included in the *Modelica Media* library or compatible with the *Modelica Media* interface. The geometry is defined by a *Geometry* model described in § 7.12.3.3. The volume model exposes the heat flow rate ($\dot{Q}$) through the *Modelica Thermal* port; the heat flow rate is calculated in the HTC model described in § 7.12.3.5. The *Volume* class is the base class for the basic volume models, and it includes a HTC and a FFC. Fig. 7.7 shows the icons for the *Volume*, *HeatTransferCorrelation* and *FrictionFactorCorrelation* partial base classes.



(a) *Volume* icon

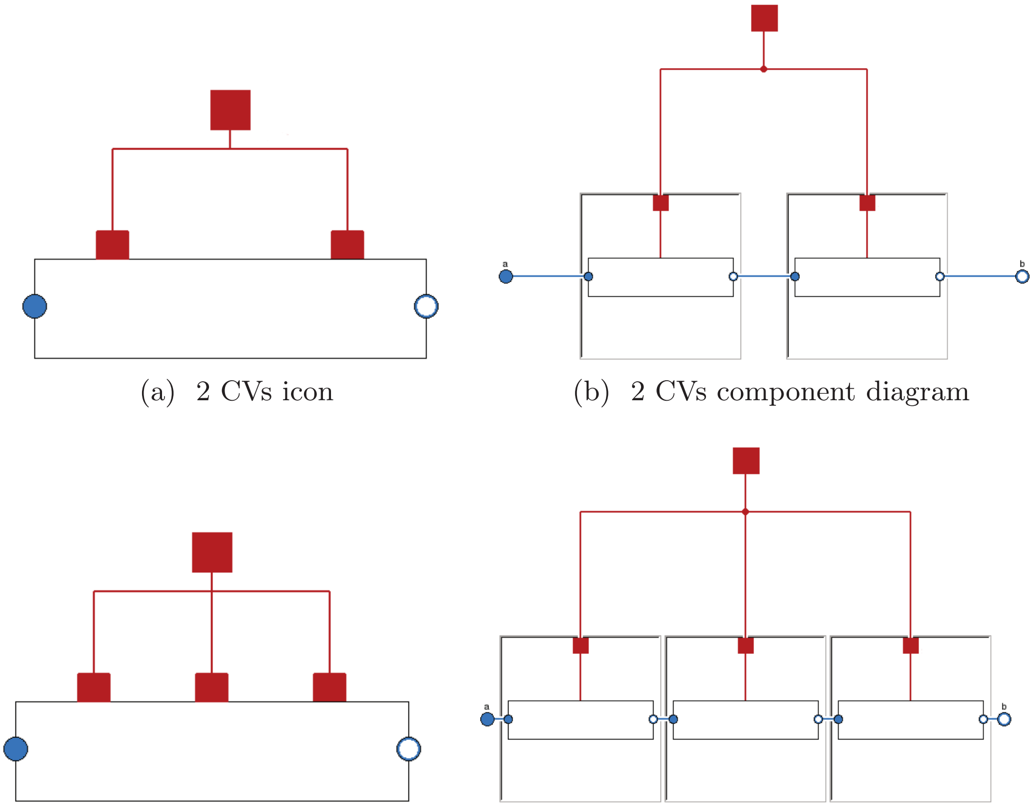
(b) Heat transfer correlation icon

(c) Friction factor correlation icon

Figure 7.7: Partial base classes

7.12.3.2 MultipleVolume Class

The *MultipleVolume* class defines two or three CVs (*Volume* classes) that can be redeclared in models that inherit from it (cf. advanced parametrization § 2.11.1). The CVs are connected through fluid connectors. The *MultipleVolume* class is the base class for all HEs. To the knowledge of the author, this is the first design and implementation of EOO switching MBMs to consider all possible flow configurations in two-phase flow evaporators and condensers with an OO design. Fig. 7.8 shows the icons and component diagrams of the two and three CVs *MultipleVolume* classes.



(a) 2 CVs icon

(b) 2 CVs component diagram

(c) 3 CVs icon

(d) 3 CVs component diagram

Figure 7.8: *MultipleVolume* partial base classes

7.12.3.3 Geometry Class

The *Geometry* class defines the geometry of the CV. This partial base class declares six variables: lengths at the beginning and at the end (z_a, z_b), inner and outer cross-sectional areas (A_i, A_o) and surface areas (S_{a_i}, S_{a_o}). The CV length and surface areas variables are shared by the HE and the pipe wall model by the *Geometry* class. In order to design the library by applying a strictly OO approach, each variable must be shared between models, but only through ports. However, for the sake of simplicity and to use only existing ports in the *Modelica Thermal* and *Modelica Fluid* libraries, this implementation was adopted.

The *MBMs* library only includes the cylindrical geometry model, although new geometries can be defined by being inherited from the *Geometry* partial base class.

7.12.3.4 Pipe Wall Class

The *Pipe Wall* class defines the pipe wall interface, which includes two *Modelica Thermal* ports – one leading to the ambient and the other one leading to the fluid. This partial class includes a partial base *Geometry* class that must be redeclared by a particular pipe wall implementation. The pipe wall class can be divided into n CVs and has two parameters: density and specific heat capacity.

7.12.3.5 Heat Transfer Correlations

There is a base class for the HTCs in the *MBMs* library. The user can inherit from it to implement new convective HTCs between the pipe wall and the fluid. A HTC can be restricted to only one particular fluid phase (one-phase or two-phase) or to only one particular process (evaporation or condensation), if required. Moreover, some convective HTCs for evaporation are already implemented in the library. These were previously described in § 6.5.1, summarized in Tab. 6.1 and a test case shown in Fig. 6.4. These HTCs have been adapted to switching, in order to avoid discontinuities and numerical problems during simulation.

7.12.3.6 Friction Factor Correlations

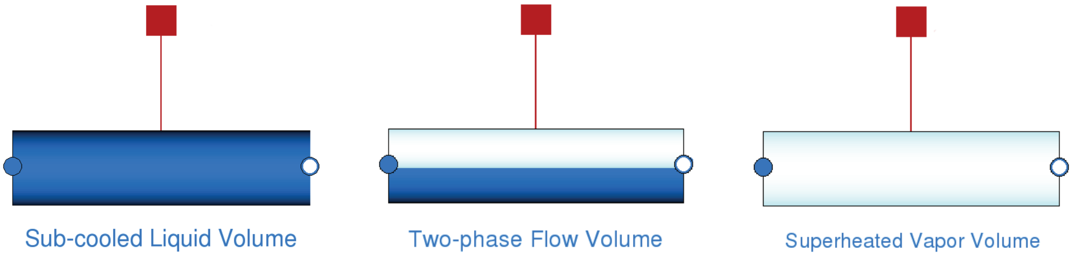
There is also a base class for FFCs, from which the user can inherit to implement new FFCs. The implemented FFCs were previously described in § 6.5.2, summarized in Tab. 6.2 and a test case shown in Fig. 6.5. These FFCs have been also adapted to switching, in order to avoid discontinuities and numerical problems during simulation. Even though pressure is assumed to be uniform in the HE, these FFCs have been included in the MBMs library because pressure loss in the HE is proposed as a future work task. Furthermore, some HTCs (Petukhov, 1970; Gnielinski, 1976) require a friction factor, which can be calculated from these FFCs.

7.12.4 Cylindrical Pipe Wall Component

The cylindrical pipe wall component includes the pipe wall model previously introduced in § 7.9, which has been adapted to support switching. The cylindrical pipe wall component redeclares the *Geometry* partial base class to define the cylindrical geometry.

7.12.5 Volume Components

Fig. 7.9 shows the icons for the SC liquid, TP flow and SH vapor models. These models, inherited from the *Volume* class, redeclare the geometry as cylindrical and add their particular mass and energy balance equations. These equations consider when the volume is active (cf. § 7.7.2 and § 7.7.3 for one-phase and two-phase regions respectively) or inactive (cf. § 7.10.2). The SC liquid and the SH vapor models inherit from an intermediate class in the hierarchy, the *OnePhaseVolume* class, because both models share the same equations.



(a) (b) (c)

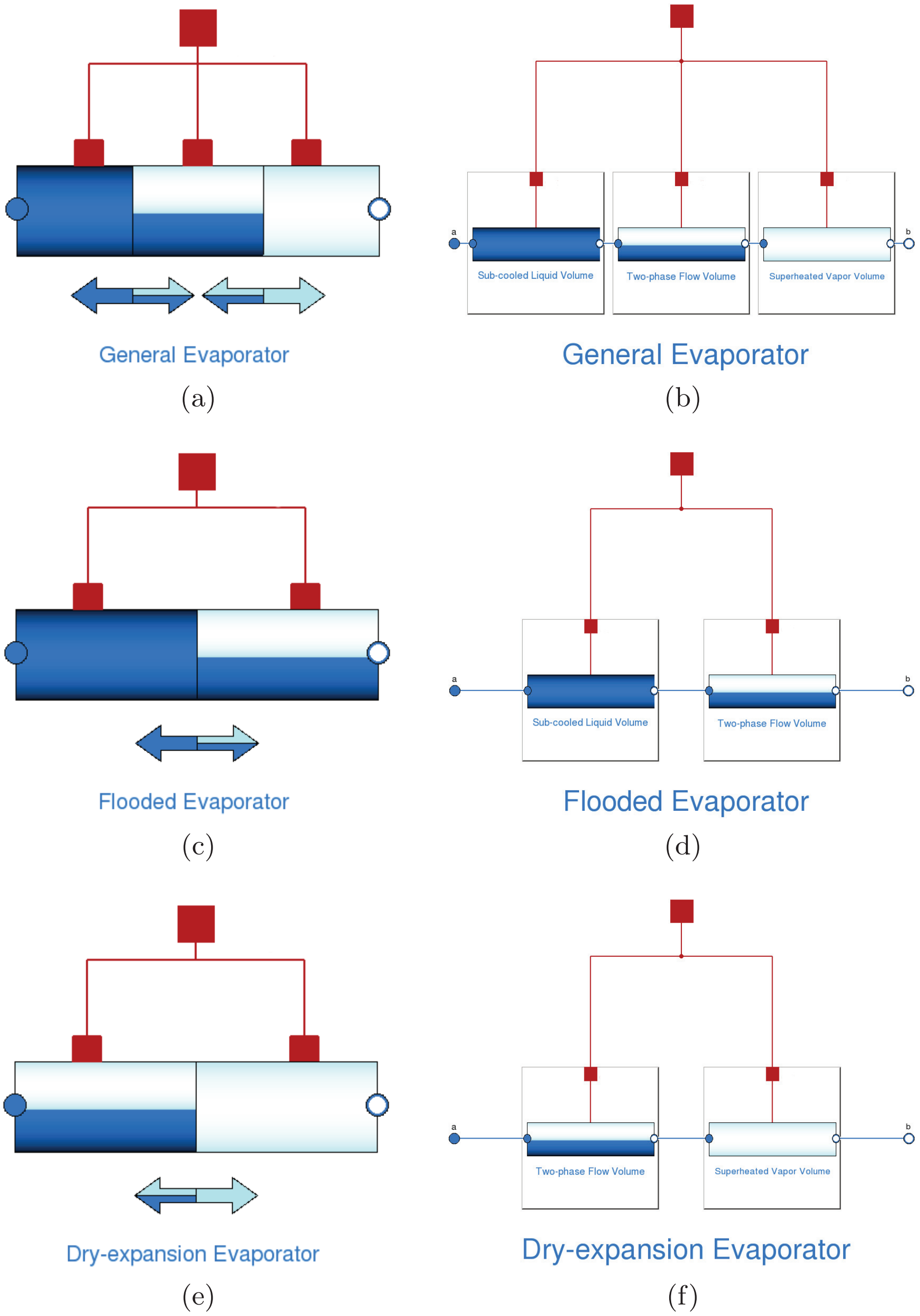
Figure 7.9: Volume components

7.12.6 Heat Exchanger Components

When redeclaring partial *Volume* classes in the *MultipleVolume* (two or three CVs) class with volume components SC liquid, TP flow and SH vapor, evaporators and condensers can be defined. HE components add the additional equations which relate inlet or outlet CV specific enthalpy to specific enthalpies at saturation conditions, according to § 7.8.1 or § 7.8.2. If switching is supported, these models must handle the events that trigger the appearance (cf. § 7.10.3) or disappearance (cf. § 7.10.1) of CVs and indicate to the CV that it must switch to the appropriate active or inactive balance equations. HE length must be taken into account also, according to Eq. 7.48.

7.12.6.1 Evaporator Components

Fig. 7.10 shows the general, flooded and dry-expansion evaporator models. The figures on the left represent the icons, whereas the figures on the right are the component diagrams, where the partial *Volume* classes from the *MultipleVolume* class have been redeclared appropriately.



(a) (b)

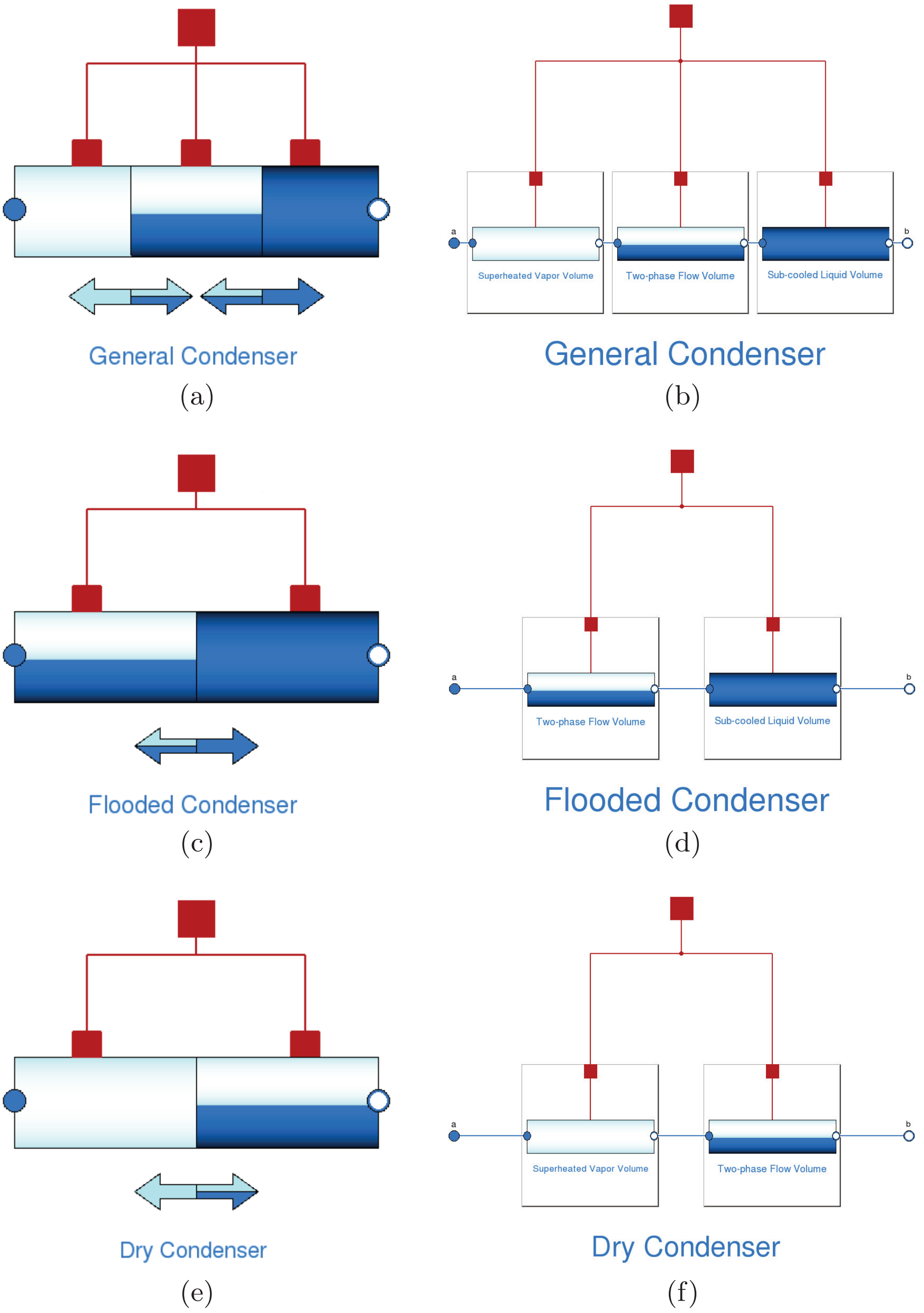
(c) (d)

(e) (f)

Figure 7.10: General, flooded and dry-expansion evaporator components

7.12.6.2 Condenser Components

For condensers, the situation is the same, apart from changing the order of the interconnected basic volumes models. Fig. 7.11 shows the general, flooded and dry condenser icons on the left, and component diagrams on the right.

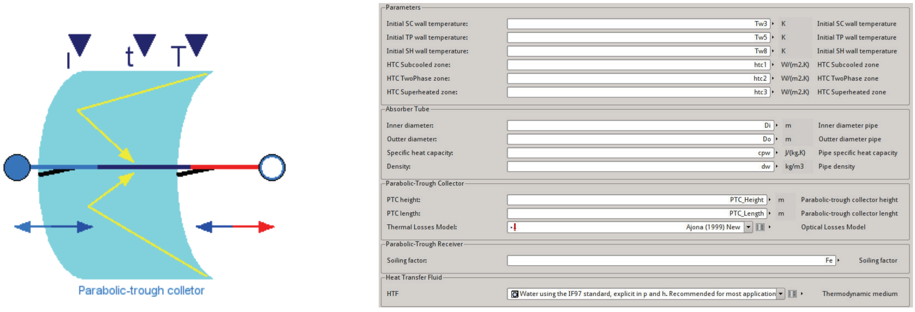


(a) (b) (c) (d) (e) (f)

Figure 7.11: General, flooded and dry condenser components

7.12.7 Parabolic-Trough Collector Model

As previously mentioned in § 5.2, there are several versions of the PTC model, one of which uses a switching moving boundary evaporator model. The other two versions use the *ThermoFluid* and *Modelica Fluid* libraries. Fig. 7.12a shows the PTC Modelica icon, whereas Fig. 7.12b shows its general parameters (additionally, there are parameters for the initialization, the HTC and the FFC). All of the parameters for the switching moving boundary evaporator model have been propagated to the PTC class. Therefore, they can be configured easily in this class.



(a) Icon

(b) General parameters

Parameters			
Initial SC wall temperature:	Tw3	K	Initial SC wall temperature
Initial TP wall temperature:	Tw5	K	Initial TP wall temperature
Initial SH wall temperature:	Tw8	K	Initial SH wall temperature
HTC Subcooled zone:	htc1	W/(m2.K)	HTC Subcooled zone
HTC TwoPhase zone:	htc2	W/(m2.K)	HTC TwoPhase zone
HTC Superheated zone:	htc3	W/(m2.K)	HTC Superheated zone
Absorber Tube			
Inner diameter:	Di	m	Inner diameter pipe
Outter diameter:	Do	m	Outter diameter pipe
Specific heat capacity:	cpw	J/(kg.K)	Pipe specific heat capacity
Density:	dw	kg/m3	Pipe density
Parabolic-Trough Collector			
PTC height:	PTC_Height	m	Parabolic-trough collector height
PTC length:	PTC_Length	m	Parabolic-trough collector lenght
Thermal Losses Model:	Ajona (1999) New		Optical Losses Model
Parabolic-Trough Receiver			
Soiling factor:	Fe		Soiling factor
Heat Transfer Fluid			
HTF	Water using the IF97 standard, explicit in p and h. Recommended for most application		Thermodynamic medium

Figure 7.12: Moving boundary PTC Modelica component

The PTC component diagram, which is shown in Fig. 7.13, is analogous to that previously described in § 5.2, but in this case the evaporator is modeled by a switching MBM.

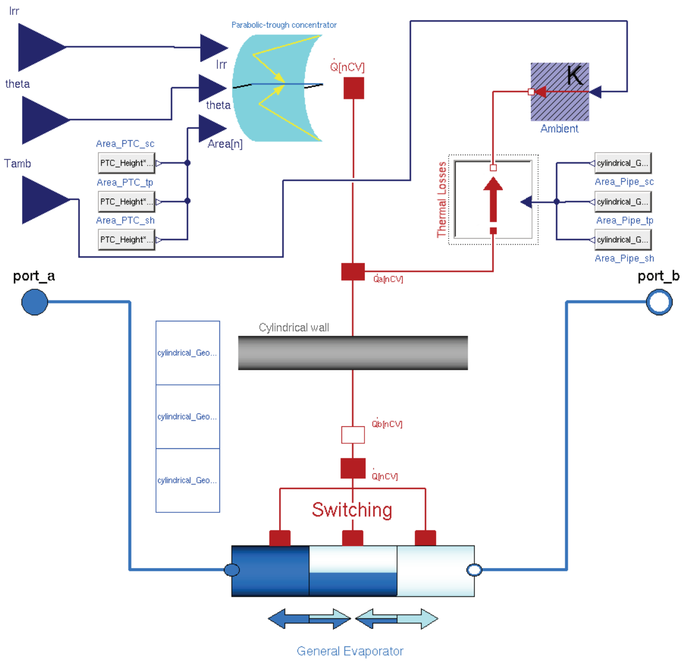


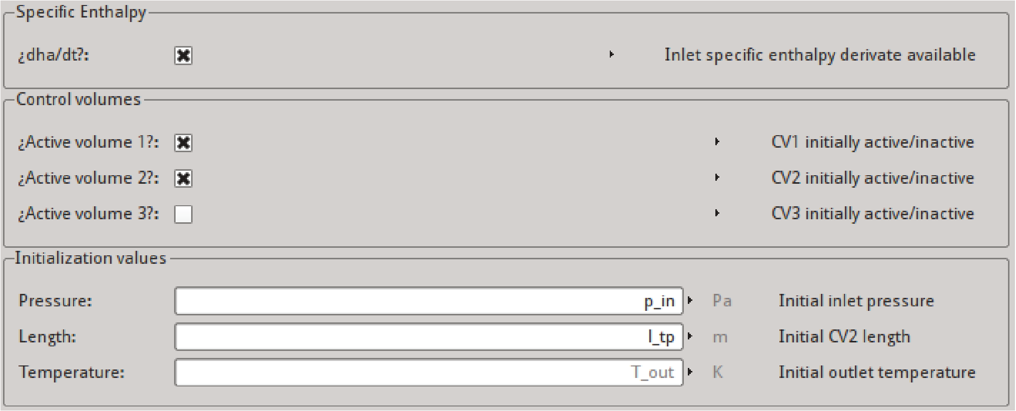
Figure 7.13: Moving boundary PTC component diagram

7.12.8 The Initialization Problem

Model initialization is always a cumbersome task, and it is especially difficult when considering initially inactive CVs in MBMs. Consequently, the initialization has been taken into account in the design of the *MBMs* library, whereby initialization options can be established through the GUI in the initialization tab for the evaporator and condenser components. Fig. 7.14 shows the initialization options for a switching general evaporator.

In the GUI, it can be specified which CVs are inactive in the initialization. The initial inlet pressure can be set as well as the initial outlet temperature (only required when the initial outlet fluid phase is one-phase flow) and initial TP flow CV length (only required when the inlet or outlet fluid is TP flow). It can be also specified whether the time derivative of inlet specific enthalpy is available. Sometimes, when considering experimental data as inputs, this thermodynamic property may not be available and cannot be estimated, in which case the inlet specific enthalpy time derivative is set to zero. Additionally, the time derivative of inlet specific enthalpy can be set to a particular variable or expression. This is currently done by code, although future work includes implementing appropriate conditional inputs in the models through the GUI, with the aim of providing the procedure to library users.

The Modelica initialization code for a switching moving boundary general evaporator, together with different initialization test cases, is described in Appx. E.2.



Specific Enthalpy

¿dha/dt?: ☒ Inlet specific enthalpy derivate available

Control volumes

¿Active volume 1?: ☒ CV1 initially active/inactive

¿Active volume 2?: ☒ CV2 initially active/inactive

¿Active volume 3?: ☐ CV3 initially active/inactive

Initialization values

Pressure: p_in Pa Initial inlet pressure

Length: l_tp m Initial CV2 length

Temperature: T_out K Initial outlet temperature

Figure 7.14: Initialization options for switching general evaporators

7.13 Simulation

This section shows the simulation of the mathematical models previously introduced and implemented in the *MBMs* library. The medium in these simulations is the two-phase flow water-steam mixture from the *Modelica Media library* (cf. § 4.5.2). The numerical solver used is DASSL (cf. § 2.9.2.1) and the absolute and relative tolerances have been set to 10^{-4}. All of the developed models have been thoroughly tested in integrity and stability tests. However, only a few are presented.

The simulations presented in this section correspond to the example model shown in Fig. 7.15, which is a PTC (cf. § 7.12.7). The model considers a source of DSI; a PTCN concentrates the DSI by considering an angle of incidence, which is zero. The concentrated solar irradiance is distributed uniformly along an absorber tube located in the geometrical focal line of the PTCN. The absorber tube is composed of a glass cover and a cylindrical steel pipe, which has a selective coating with low emissivity, and between both elements there is a high vacuum to minimize thermal losses caused by conduction and convection. Heat transfer driven by constant coefficients per fluid region is considered between the absorber tube and the HTF. The steel pipe heats the water circulating inside it and loses energy to the environment through radiation (cf. § 5.2.2.2), considering an emissivity parameter. Model inputs are the DSI, fluid inlet specific enthalpy, the fluid inlet mass flow rate, outlet pressure and ambient temperature. The input and parameter values are shown in Tab. 7.4.

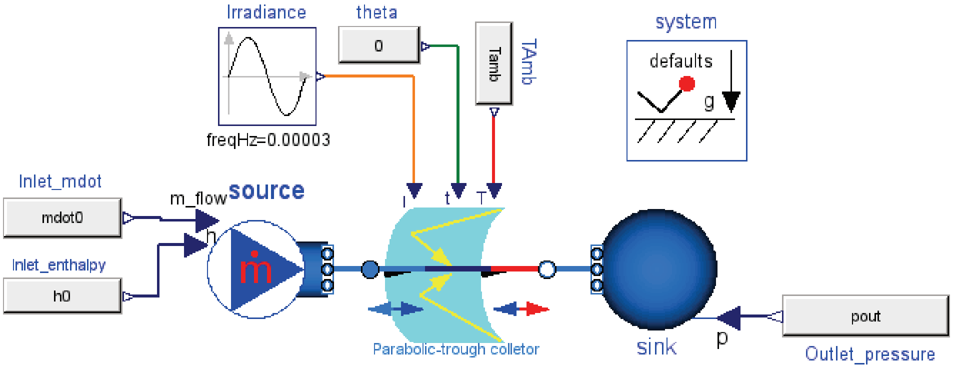


Figure 7.15: Switching moving boundary evaporator test model

7.13.1 Model Integrity

The simulation results must be verified and the governing equations of the model validated, both in steady-state and in transient predictions. To this end, the mathematical model and library implementation results were compared to those of an independently developed finite volume model and code from the *Modelica Fluid library* (cf. § 6.4.2) that belongs to the MSL 3.2. The *Modelica Fluid library* was meticulously designed and tested and is widely used in the Modelica community.

During the simulations, it was found that initialization in a steady state with prescribed values for the *Modelica Fluid* evaporator was difficult to obtain. Therefore, the initialization configuration considered in the *Modelica Fluid* models was used as the default configuration: initial guess values and dynamic balances for the mass and energy equations and steady state for the

Input	Description	Value	Units
p_{out}	Evaporator outlet pressure	6	[MPa]
$\dot{m}_{in}$	Inlet mass flow rate	0.5	[kg/s]
T_{amb}	Ambient temperature	290	[K]
h_{in}	Inlet specific enthalpy	326,515	[J/kg]
Parameter	Description	Value	Units
$l_{he} = l_{ptc}$	Evaporator length	500	[m]
d_i	Inner pipe diameter	0.05	[m]
d_o	Outer pipe diameter	0.07	[m]
l_{ap}	PTC height	2.4	[m]
α_{sc}	SC liquid heat transfer Coef.	2,000	[W/(m^2·K)]
α_{tp}	TP flow heat transfer Coef.	7,500	[W/(m^2·K)]
α_{sh}	SH vapor heat transfer Coef.	2,500	[W/(m^2·K)]
ε	Pipe emissivity	0.1	[-]
ρ_w	Pipe density	7,780	[kg/m^3]
$c_{p,w}$	Pipe specific heat capacity	550	[J/(kg·K)]

Table 7.4: Inputs and parameters of the evaporator test model

momentum equation, whereas in the *MBMs*, steady-state conditions were considered.

7.13.1.1 Model Integrity for Transient Predictions

Fig. 7.17 shows the outlet temperature for the test case shown in Fig. 7.15, where all the inputs have been held constant except the DSI. The heat flow rate over the pipe is shown in Fig. 7.16. This test case considers a switching moving boundary general evaporator model taken from the *MBMs* library (dashed blue line). A different version of the test case was developed using a finite volume model from the *Modelica Fluid library*, and simulations were performed by considering different numbers of CVs (3, 10, 20 and 50 CVs, A_V_B configuration).

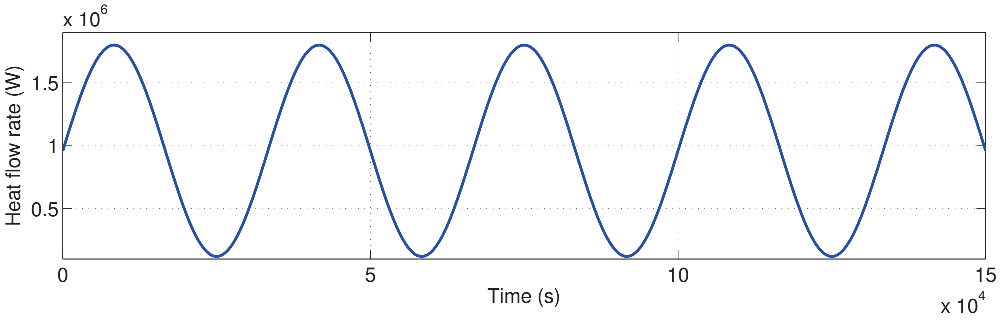


Figure 7.16: Heat flow rate over the pipe in the MBM transient integrity test

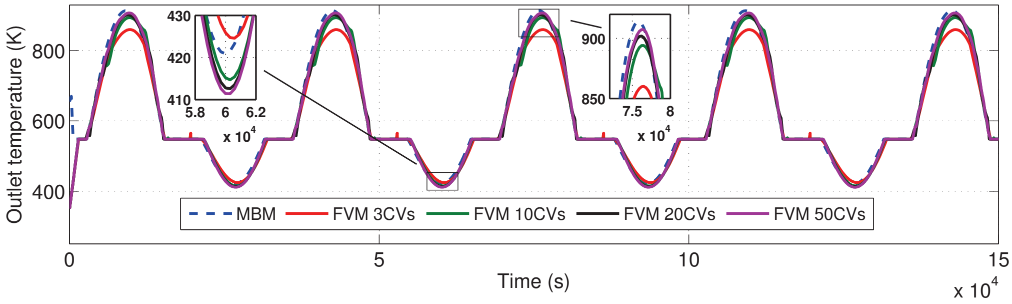


Figure 7.17: Outlet temperature in the MBM transient integrity test

Simulation results obtained with the *MBMs* library are in good agreement with those from the *Modelica Fluid library* (cf. Fig. 7.17). Considering the finite volume model with 50 CVs as a reference, Tab. 7.6 shows the mean and maximum PRE (Eq. 6.71) in outlet temperature between the finite volume model with 50 CVs and the remaining models. The MBM provides results as accurate as for a finite volume model with at least 20 CVs, but the MBM model is considerably faster (cf. Tab. 7.5). The MBM changes from SC-TP-SH / SC-TP / SC configurations forwards and backwards. Fig. 7.18 shows CV lengths in the MBM, where it can be observed which CVs are inactive during the simulation (CVs with zero length). The total length of the evaporator is 500 m.

Type of Model	MBM	FVM 3 CVs	FVM 10 CVs	FVM 20 CVs	FVM 50 CVs
CPU-time Int. (s)	0.66	0.61	4.06	12.9	70.1
State events	126	46	124	230	581

Table 7.5: Simulation statistics for the MBM transient integrity test

When the outlet temperature is constant (saturation temperature), it means that the outlet fluid phase is two-phase flow (because pressure is held constant) (cf. Fig. 7.17), while above saturation temperature, the outlet fluid phase is SH steam. Below saturation temperature, SC liquid is obtained at the HE output (cf. Fig. 7.17).

Type of Model	MBM	FVM 3 CVs	FVM 10 CVs	FVM 20 CVs	FVM 25 CVs
Mean $\epsilon_r(T)$	0.50%	1.50%	0.58%	0.53%	0.43%
Maximum $\epsilon_r(T)$	2.80%	5.36%	5.23%	4.81%	2.61%

Table 7.6: Mean and maximum PREs between the FVM 50 CVs and other configurations for the MBM transient integrity test

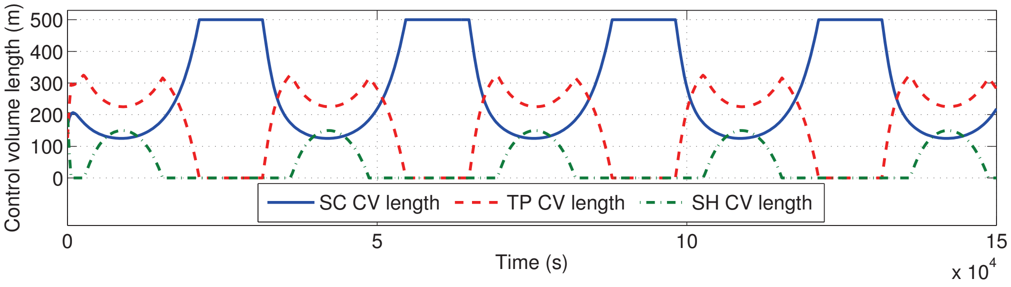


Figure 7.18: CV lengths in the MBM transient integrity test

7.13.1.2 Model Integrity for Steady-State Predictions

In this test case, the heat flow rate over the pipe is given by the values shown in Fig. 7.19. Besides this change, the integrity test model remains the same. Fig. 7.20 shows the outlet temperature for a moving boundary general evaporator model and FVMs by considering different numbers of CVs (3, 10, 20, 50 and 90 CVs, A_V_B configuration). CV lengths for the MBM are shown in Fig. 7.21. Furthermore, the simulation results obtained through the *MBMs* library are in good agreement with those from the *Modelica Fluid library* (cf. Fig. 7.20).

Considering finite volume model with 90 CVs as a reference, Tab. 7.8 shows the PRE (Eq. 6.71) in outlet temperature between the finite volume model with 90 CVs and the remaining models. The MBM provides results as accurate as a finite volume model with at least 20 CVs, but the MBM model is considerably faster (cf. Tab. 7.7).

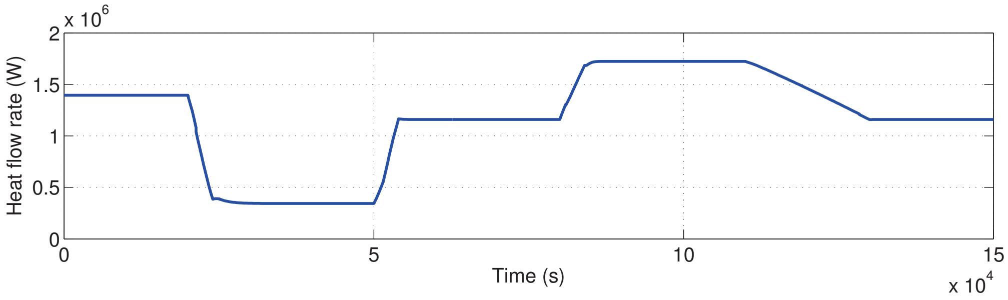


Figure 7.19: Heat flow rate over the pipe in the MBM steady-state integrity test

Type of Model	MBM	FVM 3 CVs	FVM 10 CVs	FVM 20 CVs	FVM 50 CVs	FVM 90 CVs
CPU-time Int. (s)	0.28	0.27	1.41	4.54	21.9	75.3
State events	31	14	37	85	175	370

Table 7.7: Simulation statistics for the MBM steady-state integrity test

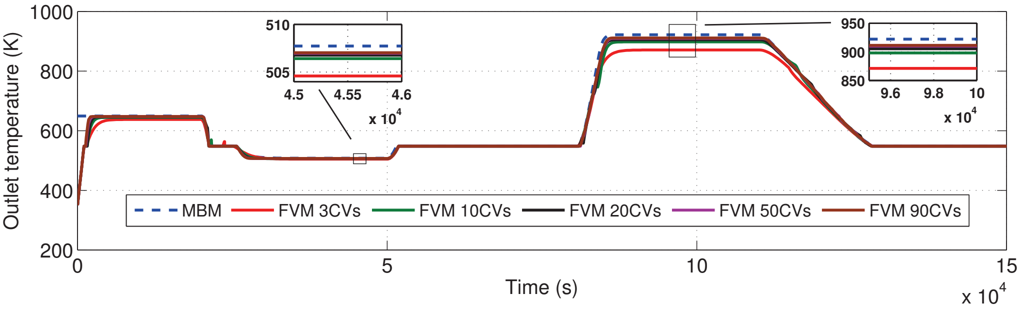


Figure 7.20: Outlet temperature in the MBM steady-state integrity test

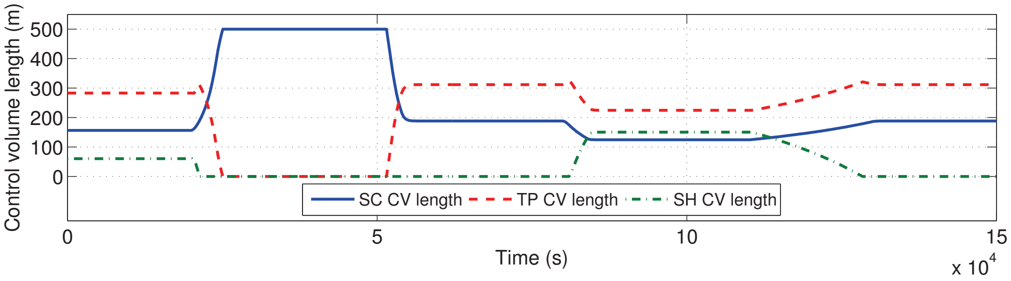


Figure 7.21: CV lengths in the MBM steady-state integrity test

Type of Model	MBM	FVM 3 CVs	FVM 10 CVs	FVM 20 CVs	FVM 25 CVs	FVM 50 CVs
Mean $\epsilon_r(T)$	0.47%	1.04%	0.54%	0.53%	0.35%	0.15%
Maximum $\epsilon_r(T)$	3.30%	6.58%	9.54%	9.11%	5.62%	2.80%

Table 7.8: Mean and maximum PREs between the FVM 90 CVs and other configurations for the MBM steady-state integrity test

The *MBM* overestimates and underestimates outlet temperature in the SH vapor and SC liquid regions, respectively, which can be observed in Fig. 7.17. These differences might be related to the fact that the viscous stress has been neglected (cf. § 7.3), among other simplifications, in the energy equation Eq. 7.2, although this must be studied further to be confirmed. It might also be related to the uniform pressure assumption in the *MBMs* library.

7.13.2 Model Stability

Model stability, especially switching stability, was checked by holding certain inputs constant during the simulation while varying others sinusoidally to force repeated switching. Variations in the heat flow rate, inlet mass flow rate, inlet specific enthalpy and outlet pressure have been tested. Fig. 7.22

shows outlet specific enthalpy (h_{out}) in a switching general evaporator when varying inlet specific enthalpy (h_{in}) sinusoidally. The two constant lines shown in Fig. 7.22 are the enthalpies of saturated liquid and saturated vapor (h', h''), which are constant because pressure is held constant. The model switches between SH / TP-SH / TP / SC-TP / SC forwards and backwards. This example shows the appearance and disappearance of CVs at the beginning and at the end of the HE. The only configuration, from all possible flow configurations, not tested in this example is SC-TP-SH, but it was previously tested in the integrity test case for transient predictions (cf. § 7.13.1.1), which can be also considered a stability test as well as an integrity test.

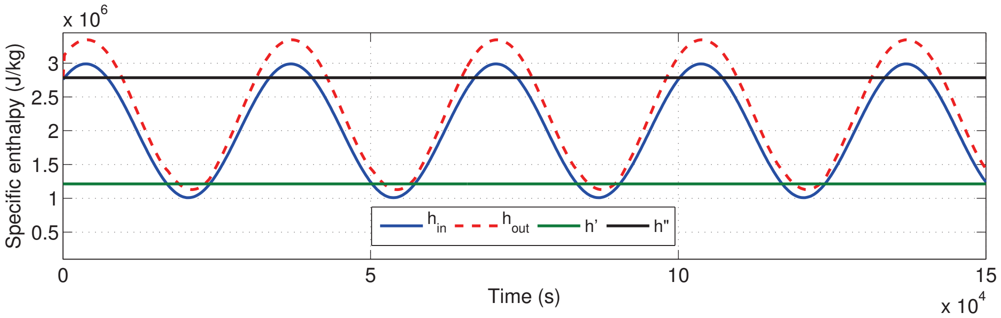


Figure 7.22: Specific enthalpies in the MBM stability test

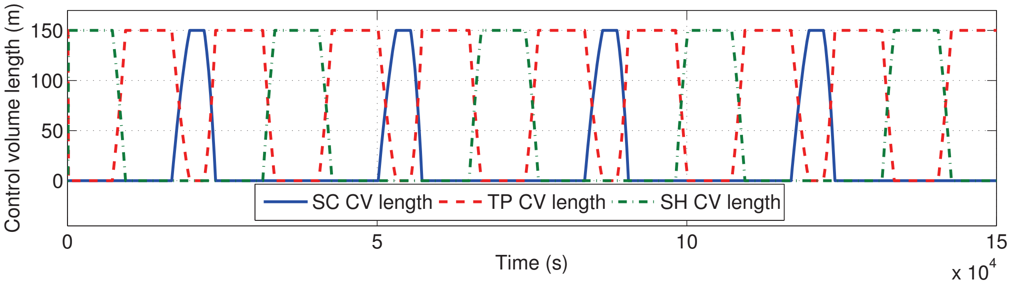


Figure 7.23: CV lengths in the MBM stability test

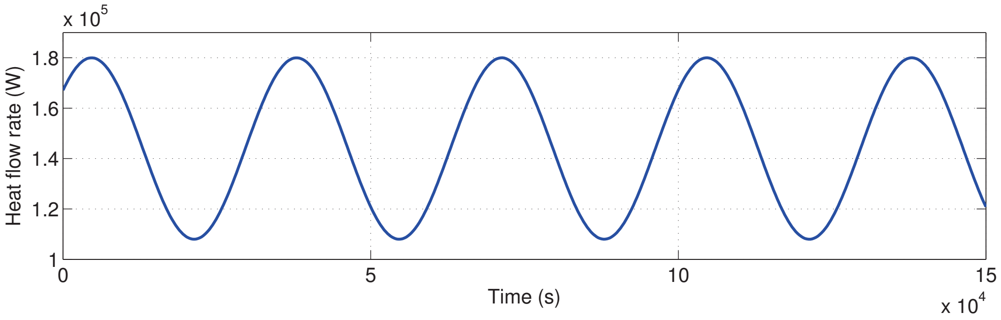


Figure 7.24: Heat flow rate over the pipe in the MBM stability test

CV lengths are shown in Fig. 7.23. The model used in the stability test is the same as the one described in § 7.13. The differences are that the HE length is 150 m, the pipe wall density 1,000 kg/m^3, the pipe wall specific heat capacity 1 J/(kg·K), the DSI is sinusoidally varied according to

Fig. 7.24 and inlet specific enthalpy is also sinusoidally varied (cf. Fig. 7.22). The simulation time for this test case is 1.26 s. Additional stability tests as well as initialization tests are presented in Appx. E.2.

7.14 Summary and Conclusions

This chapter began with an introduction to and a brief state-of-the-art study about MBMs. After that, the assumptions, goals and derivation of one-dimensional mass and energy balance equations considered in MBMs were introduced. The switching criteria were explained in order to switch between different configuration in HEs. The *MBMs* Modelica library was described thereafter; this library tackles the initialization problem in MBMs. The novelty in these models is that they support dynamic switching between all possible flow configurations in HEs. Furthermore, the implementation follows an OO approach, because basic models consider mass and energy balance equations, whereas compound models are developed by interconnecting the basic models and then adding switching support. Following this approach, all the evaporators and condensers were developed reusing the three basic models: SC liquid, TP flow and SH steam models. Finally, simulation results were presented for the switching moving general evaporator model in steady-state and transient integrity tests, which were compared with the finite volume model implemented in the *Modelica Fluid* library. Stability tests considering all possible flow configurations, and switching cases involving the appearance and disappearance of CVs, were also presented. These tests can be extended to condenser models because the governing condenser equations are the same as those used for evaporators.

7.15 Related Scientific Contributions

The scientific contributions related to this chapter are enumerated and briefly commented on in the following list. These contributions were previously listed in § 1.4.

5. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Library of Switching Moving Boundary Models for Two-phase Flow Evaporators and Condensers. In *Proceedings of the 9th International Modelica Conference*, oral communication, Munich, Germany, 2012.
The MBMs library is presented together with an integrity and stability test for a switching general moving boundary evaporator model.

6. Bonilla, J., Yebra, L. J., Dormido, S. and Cellier, F. E. Object-Oriented Modeling of Switching Moving Boundary Models for Two-phase Flow Evaporators. In *Proceedings of the 7th Vienna International Conference on Mathematical Modelling (MATHMOD)*, oral communication, Vienna, Austria, 2012. *Preliminary switching flooded moving boundary evaporator model.*

V

Dynamic Simulations

8
Simulation, Calibration and Validation

Knowing is not enough; we must apply.
Willing is not enough; we must do.
Johann Wolfgang von Goethe

Abstract: The practical aspects related to the development of this work are discussed in this chapter, which includes roughly seven main sections. The first is a DISS facility simulation framework which facilitates the set up of the simulations (§ 8.1). The different developed DISS models are described and discussed in the second section (§ 8.2). The third is intended to improve simulation performance by using OpenMP to parallelize the model code (§ 8.3). In the fourth section, the assumption of linear specific enthalpy distribution in each evaporator CV in MBMs is justified based on the balance equations (§ 8.4). The last three sections deal with calibration, validation and simulation. Calibration and validation estimate some unknown model parameters (§ 8.5 and § 8.6), while the simulation section compares different developed DISS Modelica models (§ 8.8).

8.1 DISS Facility Simulation Framework

In order to exploit DISS Modelica models, a simulation framework is required. This framework must facilitate dealing with inputs, setting parameters and initial conditions, and all of these processes should be as automated as possible in order to minimize set-up time and to focus on the simulation task. Fig. 8.1 shows a diagram of the DISS facility simulator framework. Each element is described briefly in the following paragraphs. The DISS test facility and its instrumentation were previously introduced in § 3.6.

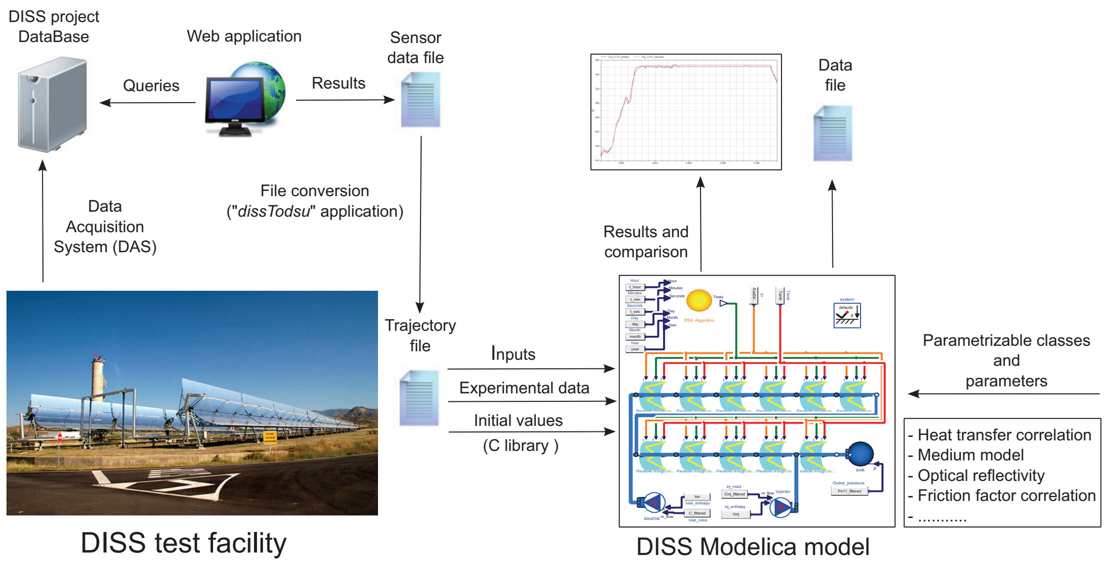
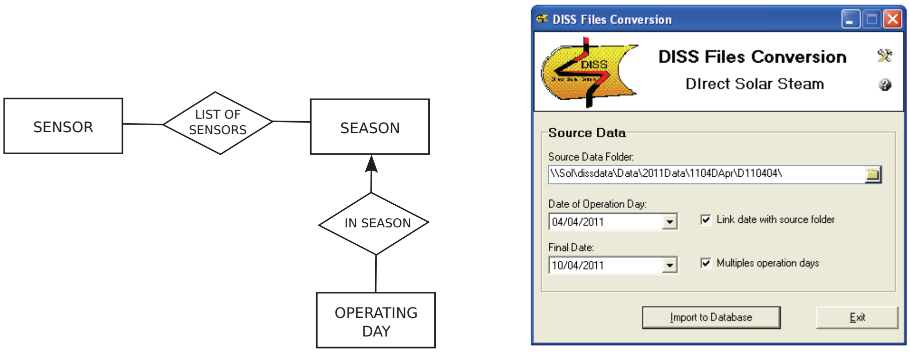


Figure 8.1: DISS test facility simulator framework

Data Acquisition System (DAS). Stores all the data taken from the DISS instrumentation (cf. § 3.6.1) in a dedicated server. Distributed ZIP-format files store all the information for each experiment. Additionally, a database was designed and implemented for the DISS project. This database has three important tables, namely sensor, season and operating day tables. Fig. 8.2a shows the DISS project database Entity-Relationship Diagram (ERD) of these three tables. The sensor table stores all the information about each sensor, including tag, description and units. The season table establishes which sensors are available for each season. The data for each experiment is stored individually in one table per day, due to the large amount of data. In order to migrate the data from the distributed ZIP-format files to the DISS project database, a computer program was developed, cf. Fig. 8.2b. This computer program performs the migration automatically, so the user need only provide the dates when the experiments were performed.



Source Data

Source Data Folder: \\Sol\dissdata\Data\2011Data\1104DApr\D110404\

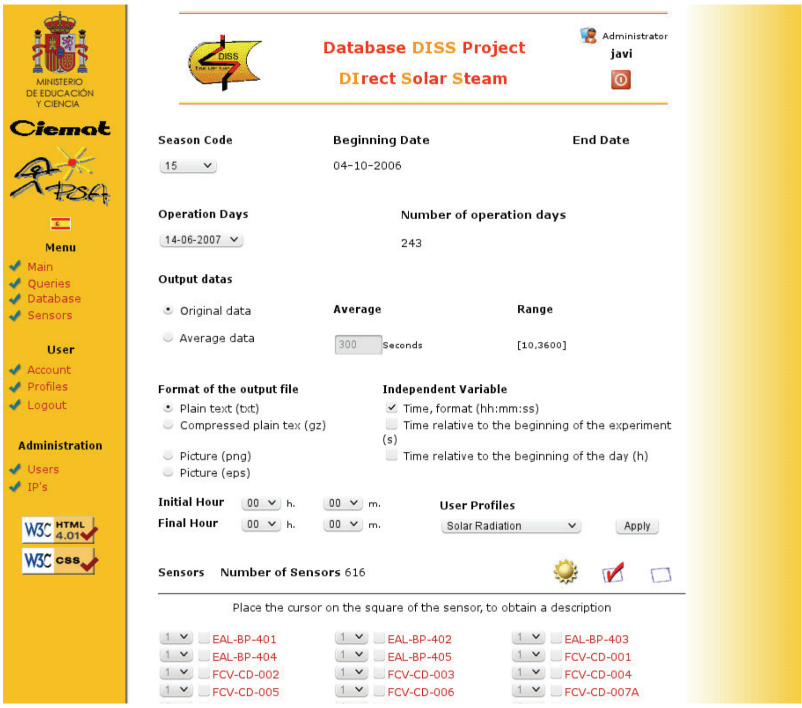
Date of Operation Day: 04/04/2011 ☑ Link date with source folder

Final Date: 10/04/2011 ☑ Multiples operation days

(a) Entity-relationship diagram (b) Data migration tool

Figure 8.2: DISS project database

Web Application. Developed in order to access experimental data easily. The user can define profiles, each of which includes a number of selected sensors. Therefore, every time the user needs to extract data it is not necessary to select every single sensor; the profile can be loaded instead. Other required input data are: the date, time and also if the data is represented in a graph or stored in a data file. The graph can be in bitmap or vector formats, whereas the data file can be stored in a plain text file in Comma-Separated Value (CSV) format or in a compressed ZIP-format file. A snapshot of the web application is shown in Fig. 8.3.



Database DISS Project

DIrect Solar Steam

Season Code: 15
Beginning Date: 04-10-2006
End Date:

Operation Days: 14-06-2007
Number of operation days: 243

Output datas
● Original data
○ Average data
Average: 300 Seconds
Range: [10,3600]

Format of the output file
● Plain text (txt)
○ Compressed plain tex (gz)
○ Picture (png)
○ Picture (eps)

Independent Variable
☑ Time, format (hh:mm:ss)
□ Time relative to the beginning of the experiment (s)
□ Time relative to the beginning of the day (h)

Initial Hour 00 h. 00 m.
Final Hour 00 h. 00 m.

User Profiles: Solar Radiation
Apply

Sensors Number of Sensors 616

Place the cursor on the square of the sensor, to obtain a description

1 □ EAL-BP-401 1 □ EAL-BP-402 1 □ EAL-BP-403
1 □ EAL-BP-404 1 □ EAL-BP-405 1 □ FCV-CD-001
1 □ FCV-CD-002 1 □ FCV-CD-003 1 □ FCV-CD-004
1 □ FCV-CD-005 1 □ FCV-CD-006 1 □ FCV-CD-007A

Figure 8.3: DISS web application

File Conversion Application. With the aim of using the obtained data files in the DISS Modelica models via the Dymola tool, a file conversion computer program was developed. This computer program converts a CSV format text file into a trajectory file, which is the required format for input files in Dymola. This conversion is performed while preserving the original tag, description and units of the sensors.

Initial values library. A Modelica library was developed in order to facilitate the process of obtaining the initial conditions for the simulation. This library automatically retrieves suitable values (e.g. inlet temperature, inlet mass flow rate and outlet pressure) from the trajectory file to solve the initial condition problem. The only required inputs are the trajectory file, the initial simulation time and the sensor tags.

The DISS project database design and initial population were performed by Pablo García in the scope of a collaborative grant CIEMAT-Universidad de ALmería (UAL). The redesign of the DISS project database, database population, migration tool and web application implementations was performed by the first author of this book in the scope of a collaborative grant CIEMAT-UAL.

8.2 DISS Modelica Models

During the development of this work, four different DISS Modelica models have been implemented, considering finite volume models introduced in Chap. 6 and MBMs described in Chap. 7. These models are explained in this section, while other possible models are discussed in § 8.2.5. Advantages and disadvantages of all these models are presented in § 8.2.6.

8.2.1 Finite Volume ThermoFluid DISS Model (FV-Th)

The first version of this model was developed in Yebra (2006). The model uses the *ThermoFluid* library (cf. § 6.4.1), which applies the FVM, the staggered grid scheme and the UDS in order to discretize the evaporator (cf. § 6.1). In the present work, the chattering problem has been tackled, and *Mean Densities* (cf. § 6.7.1) and the *Heuristic* approach (cf. § 6.7.2) have been developed in order to avoid this numerical problem (cf. § 6.6). Additionally, different HTCs and FFCs have been implemented for this model (cf. § 6.5). The finite volume *ThermoFluid* DISS model was implemented in Modelica 2.2.1 using Dymola 6.0b. Fig. 8.4 shows the component diagram of this model.

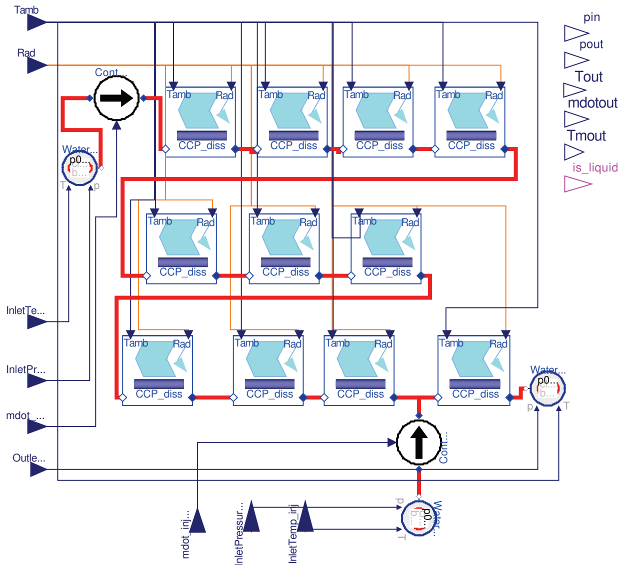


Figure 8.4: Finite volume *ThermoFluid* DISS model (FV-Th) (Yebra, 2006)

8.2.2 Finite Volume Modelica Fluid DISS Model (FV-F)

This model uses the *Modelica Fluid* library (cf. § 6.4.2), which also applies the FVM, the staggered grid scheme and the UDS in order to discretize the evaporator. This model was developed in Modelica 3.2 with Dymola 2013. Fig. 8.5 represents the component diagram of this model, which was previously introduced in § 5.5.

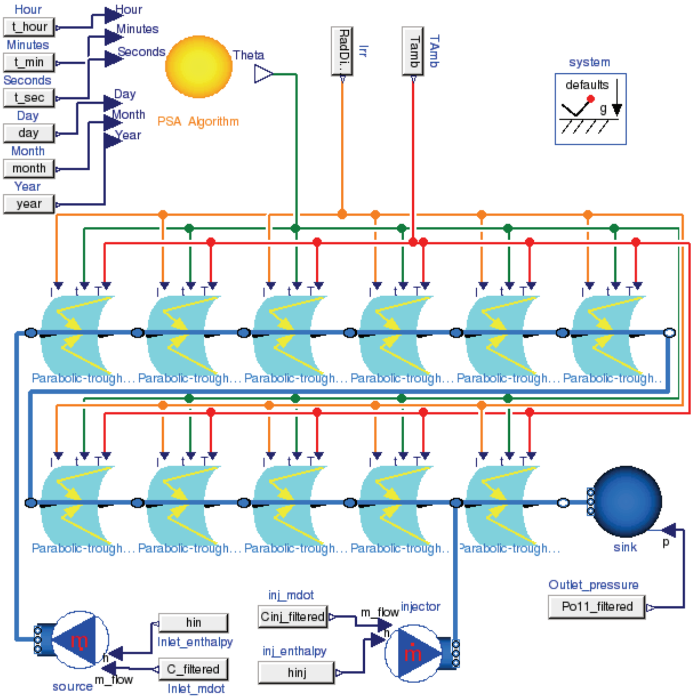


Figure 8.5: Finite volume *Modelica Fluid* DISS model (FV-F)

The *Modelica Fluid* grid structure, which produced the best results, is the A_V_B structure, cf. § 6.4.2.1. Therefore, this was adopted for all dynamic simulations. The other grid structures caused convergence problems in simulation when the number of CVs increased.

Although chattering is less common in the *Modelica Fluid* library, there are still some cases where it appears in simulation (simulations performed in § 8.7 and § 8.8). Chattering could be avoided by changing numerical integration tolerance or the number of output intervals in the Dymola tool. However, these changes were made simply by trial and error, without any particular knowledge. For this reason, this model is suitable for neither calibration purposes nor the design of an advanced ACS. Future work includes the implementation of *Mean Densities* and the *Heuristic* approach in the *Modelica Fluid* library in order to avoid the chattering problem.

An optimized version of the finite volume *Modelica Fluid* model (FV-F-Opt), depicted in Fig. 8.6, has been also implemented. In the FV-F-Opt model, the first ten PTCs from the FV-F model are considered as only one

PTC with ten times the number of CVs. The model is conceptually the same, and all the parameters can be adjusted. Additionally, the model is much faster than the FV-F model. The only drawback is that this model is not so flexible, because it does not allow the modeler to place additional injectors between PTCs. This is a problem, because supporting the injection and recirculation operating modes (cf. § 3.5.1) is proposed as a future work task.

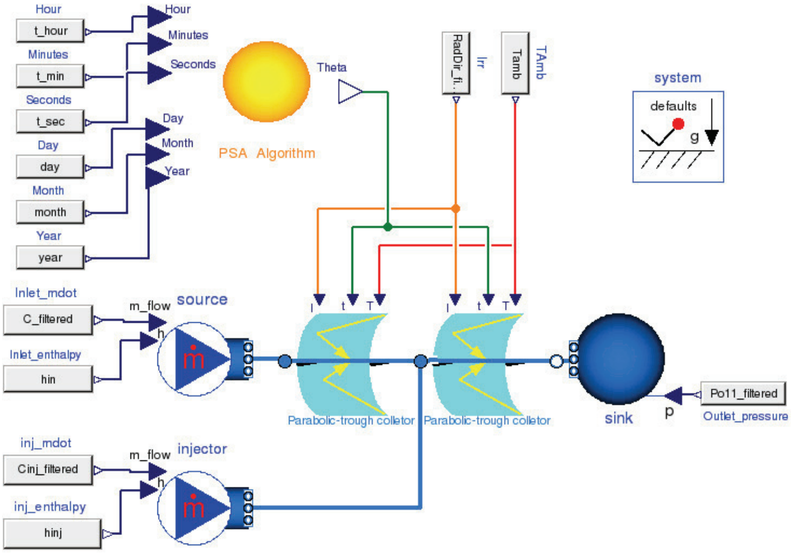


Figure 8.6: Optimized finite volume *Modelica Fluid* DISS model (FV-F-Opt)

8.2.3 Optimized Moving Boundary DISS Modelica Model (MB-Opt)

Another DISS Modelica model was developed using the *MBMs* library (cf. § 7.12) and the PTC model described in § 7.12.7. This model was developed in Modelica 3.2 using Dymola 2013. Fig. 8.7 shows the component diagram of this model. The same flexibility issues found in the optimized finite volume *Modelica Fluid* also affect this model. Additionally, optical efficiency can be only specified for each fluid region and not for each PTC.

A MBM requires as an input the time derivative of the inlet specific enthalpy (cf § 7.12.8). This value cannot be measured, although it can be estimated by considering previous temperature and pressure values, if the flow phase is not two-phase. For the *source* component (cf. Fig. 8.7), the time derivative of the inlet specific enthalpy has been calculated with the FDM (first order backward approach) by considering previous values. The time derivatives of h_{in} are calculated by the *file conversion application* (cf. § 8.1). *Freesteam* (John Pye, 2010), which is an open source C implementation of IAPWS-IF97, is used to compute specific enthalpy (h) from pressure (p) and temperature (T). In the second PTC, it is calculated in the junction component (cf. Fig. 8.7), which belongs to the *Modelica Fluid* library, by means of dynamic mass and energy fluid balances.

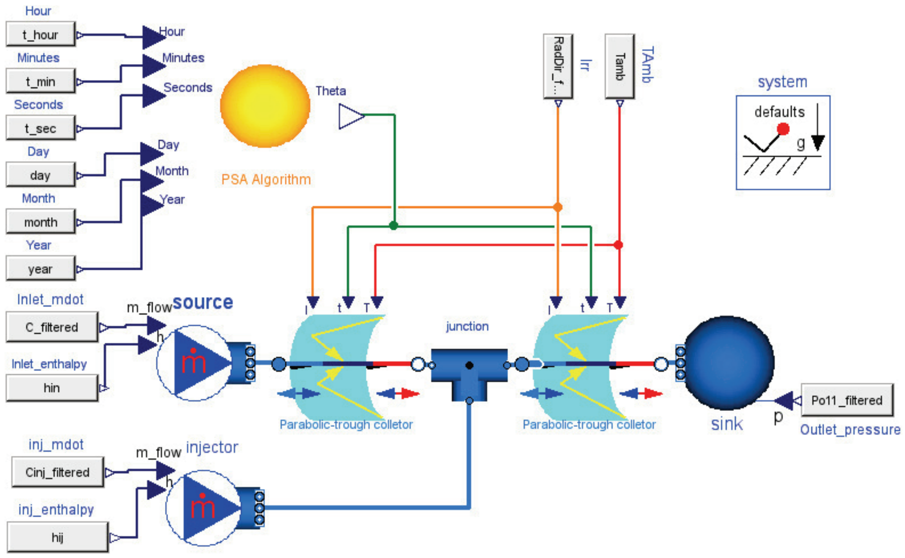


Figure 8.7: Optimized moving boundary DISS Modelica model (MB-Opt)

8.2.4 Optimized moving Boundary + Finite Volume DISS Modelica Model (MB+FV-Opt)

This is a hybrid DISS Modelica Model, where the first ten PTCs are modeled by the *MBMs* library and the last PTC by the *Modelica Fluid* library. Fig. 8.8 shows the component diagram. The same flexibility issues inherent in the two previous models also affect this model. For the first tenth PTCs, optical efficiency can be only specified for each fluid region. The inlet specific enthalpy time derivative (*source* component) is calculated as described in § 8.2.3.

8.2.5 Other Possible DISS Modelica Models

Other possible DISS Modelica models include the following.

- *Optimized Finite Volume ThermoFluid DISS Model (FV-Th-Opt)*. The FV-Th model could be also optimized as the FV-F model.

- *Moving Boundary DISS Modelica Model (MB)*. In this model, each PTC evaporator is a switching MBM (cf. § 7.12.7). This model is more complex and slower than the MB-Opt model. However, optical efficiencies can be specified in each PTC, which cannot be done in the MB-Opt model.

- *Finite Volume + Moving Boundary DISS Modelica Model (FV+MB-Opt)*. Considering from the 1st to the 10th PTCs the FVM and in the last PTC, the moving boundary method yields a different DISS model. This model is similar to the MB-FV-Opt model, but it can describe the first ten PTCs more finely and the last PTC more roughly.

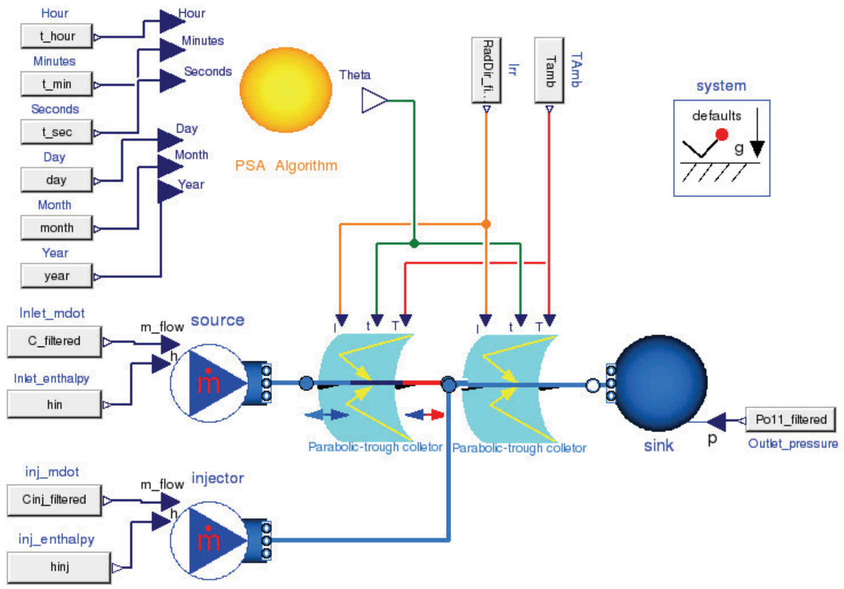


Figure 8.8: Optimized moving boundary + finite volume DISS Modelica model (MB+FV-Opt)

- *Finite volume and moving boundary DISS Modelica Model (FV-MB).* Inspecting the experimental data from the DISS facility, the PTCs where phase transitions occurred can be identified. One solution to avoiding chattering consists in using moving boundary evaporator models in the PTCs where phase transitions occur, and finite volume evaporator models in the remaining PTCs.

- *Moving boundary of finite volumes DISS Modelica Model (MBoFV).* This is another interesting idea for future research. It involves dividing the evaporator in fluid regions by applying the moving boundary method. Each fluid region is then discretized in CVs by considering the finite volume formulation. This approach has many advantages. For instance, the speed-accuracy trade-off can be configured easily, selecting appropriately the number of CVs. Additionally, it is a chattering-free approach. Chattering is caused by discontinuities in variables of discretized models due to phase changes, but this approach applies the moving boundary method. Therefore, no phase changes occur within the CVs. Analogously to the previous models, there are two versions of this model – one considers each PTC independently (MBoFV) and the other considers two PTCs (MBoFV-Opt).

8.2.6 Summary of DISS Modelica Models

Tab. 8.1 summarizes the main advantages and disadvantages of all the previously discussed DISS Modelica models. The customization term determines

Model	Advantages	Disadvantages
Finite volume models		
FV-Th § 8.2.1	Customizable Flexible	Slow Unsupported Chattering
FV-F § 8.2.2	Customizable Flexible Supported	Slow Chattering
FV-Th-Opt § 8.2.5	Customizable Speed-accuracy trade-off	Inflexible Unsupported Chattering
FV-F-Opt § 8.2.2	Customizable Speed-accuracy trade-off Supported	Inflexible Chattering
Moving boundary models		
MB § 8.2.5	Flexible Chattering free Customizable Own development	Slow
MB-Opt § 8.2.3	Fast Chattering free Own development	Inflexible Limited customization
Hybrid models		
MB+FV-Opt § 8.2.4	Partially customizable Speed-accuracy trade-off Own development & Supported	Inflexible Chattering
FV+MB-Opt § 8.2.5	Customizable Speed-accuracy trade-off Own development & Supported	Inflexible Chattering
FV-MB § 8.2.5	Flexible Customizable Chattering-free Own development & Supported	Slow
MBoFV § 8.2.5	Flexible Customizable Chattering-free Own development	Slow
MBoFV-Opt § 8.2.5	Speed-accuracy trade-off Chattering free Own development	Inflexible Limited customization

Table 8.1: Advantages and disadvantages of different DISS models

whether the model can adapt all the parameter per PTC (e.g. optical efficiency). The flexible term determines whether the model can be extended by considering the injection and recirculation operating modes. The supported/unsupported/own development terms determine whether the library used to develop the model is still maintained or has been self-developed. The slow/fast/speed-accuracy trade-off terms qualify the simulation performance in comparison to the rest of the models. It is also indicated whether chattering may occur during the simulation. If this is the case, a method to avoid chattering should be implemented.

8.3 Exploiting OpenMP in the DISS Modelica Model

Nowadays, modern EOO modeling languages continuously increase their expressiveness in order to describe and model complex systems. However, there is an important drawback in having large and complex models, as the computational effort required to simulate them is very high. Commonly, EOO models are compiled as single-threaded executable programs and do not take advantage of the newest multi-core processors which are available even on desktop computers. Considering the particular case of Modelica, the extension of the language to support multi-core processors is not easy due to the flattening of the EOO Modelica code into C code (cf. § 2.9.1).

With regard to parallelization in EOO modeling languages, it is worth mentioning: Aronsson and Fritzson (2005) and Aronsson (2006) in automatic parallelization of equation-based simulation programs, Lundvall (2008) in automatic parallelization of equation-based simulation languages using pipelining and Östlund (2009) in parallel simulation of Modelica models on the Compute Unified Device Architecture (CUDA). This section describes a straightforward parallelization method in the initial section of the resulting C code obtained from the translation of the Modelica model by the Dymola tool. It does not consider parallelization in numerical integration.

8.3.1 OpenMP Description

OpenMP (Open Multi-Processing) (Architecture Review Board, 2011) is an Application Programming Interface (API) for multi-platform (Unix and Microsoft Windows platforms) shared-memory parallel programming in C/C++ and *Fortran*. It provides a simple and flexible interface to develop parallel applications by means of a set of compiler directives, library routines and environment variables. Its applicability ranges from desktop computers to clusters and supercomputers.

OpenMP is maintained by the OpenMP Architecture Review Board (ARB). The first OpenMP API specification was released in 1997 for *Fortran* 1.0, whereas for the C/C++ languages it was released the following year, in 1998.

In 2005, both versions were unified in version 2.5. The current version, version 4, was released in November, 2012.

The key concept in OpenMP is multithreading. In this method of parallelization, the master thread forks slave threads for particular parallel tasks. The run-time environment is responsible for allocating the threads to different processors or cores. Fig. 8.9 illustrates this concept, where the master thread forks two additional threads for tasks I, three additional threads for task II and one additional thread for task III.

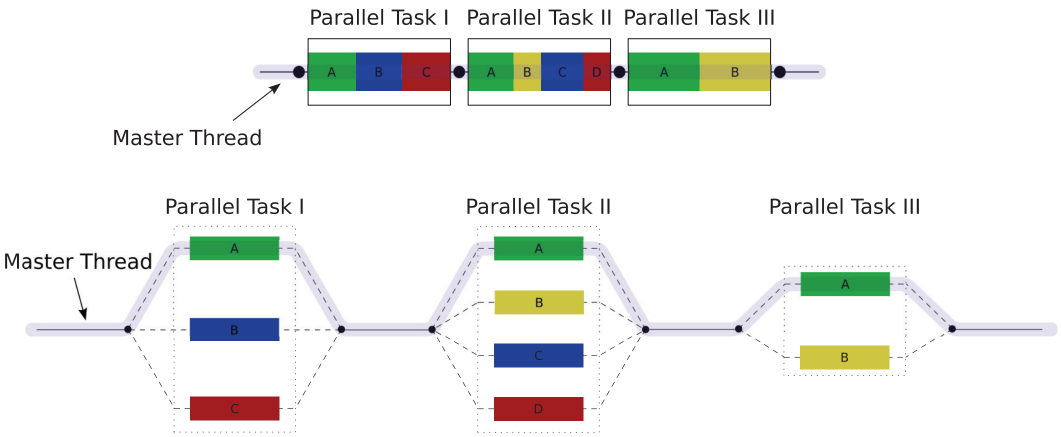


Figure 8.9: OpenMP multithreading concept

The core elements of the OpenMP API are summarized and described briefly in the following list.

- **Parallel control directives.** Control the flow execution of the program (e.g. *parallel* directive).
- **Work sharing.** These directives distribute the execution of instructions among the processors or cores (e.g. *parallel for* or *section* structures).
- **Data environment.** Defined by the scope variables *shared* and *private*.
- **Synchronization.** Controlled by the *critical*, *atomic* and *barrier* directives.
- **Run-time functions.** Set and provide run-time information (e.g. *omp_set_num_thread()*/ *omp_get_num_thread()* which sets/gets the maximum number of threads that can be allocated).

8.3.2 Initial Section Parallelization

As previously explained in section § 8.1, a Modelica library was developed to retrieve experimental data from the input trajectory files, with the aim of setting several parameters to solve the initial condition problem. Reading from a text file is a time-consuming operation, but some data must be read from this input file, e.g. initial outlet pressure, inlet temperature and inlet mass flow rate. These reading operations can be parallelized easily because they are independent from each other.

The procedure for parallelizing the initial section in the resulting C code, obtained from the translation of the Modelica model by the Dymola tool, is explained in detail in Appx. F – the process is straightforward. However, the procedure for parallelizing the initial section of Modelica models must be done manually. It might therefore be worth studying how this procedure could be performed directly in the Modelica code and not in the resulting C source code. Only constant and parameter values can be parallelized with this approach, and time-consuming operations must be involved in the code; otherwise, no performance improvement in simulation will be obtained. Consideration must also be given to the dependencies between instructions, in order to properly distribute them between the different threads.

8.3.3 OpenMP Simulation Statistics

Several tests using different operating days were performed, and mean execution times (CPU time for integration + initialization time) were calculated. The finite volume *ThermoFluid* DISS Model, FV-Th (cf. § 8.2.1) was used, together with the *Heuristic* approach (temporal restriction), in order to avoid chattering. It is important to note that parallelization is only considered in the initial section and not during numerical integration.

The simulation statistics are summarized in Tab. 8.2. Speedup (S_{up}) is defined by Eq. 8.1, where t_{seq} and t_{par} are the sequential and parallel simulation times. Although the gain in speed is not noteworthy, as just a global speedup of 1.13 was obtained, the calibration process using genetic algorithms, which will be explained in § 8.5, is time-consuming and even a modest performance improvement is worthwhile. Moreover, if it is considered that parallelization was performed only in the initial section, not in the whole simulation, the speedup obtained is considerably high, 2.80, because the mean initial section sequential and parallel execution times are 31.231 and 11.127 seconds, respectively.

$$S_{up} = \frac{t_{seq}}{t_{par}} \tag{8.1}$$

With respect to the calibration process using genetic algorithms, if 20 elements in the population and 100 iterations are considered, the calibration execution time is reduced (4.5 hours less), although it remains a time-consuming task because it was performed over three days and 4.5 hours. This calibration process was carried out on a desktop computer placed in a laboratory. This computer was different to the computer used for the rest of the simulations performed in this work.

Metric	Original Model	Parallelized Model
Simulation execution time (s)	170.727	151.415
Simulation speedup	1	1.13
Initialization section execution time (s)	31.231	11.127
Initialization section speedup	1	2.80
Calibration execution time	3 d 9 h	3 d 4.5 h

Table 8.2: Parallelization simulation statistics

8.4 Specific Enthalpy Distribution in the Evaporator

The specific enthalpy distribution in the evaporator has been studied for the FV-F-Opt model which has two finite volume evaporators (cf. § 8.2.2) and the MB-Opt model which has two switching moving boundary evaporators (cf. § 8.2.3). Several operating days have been considered in this study.

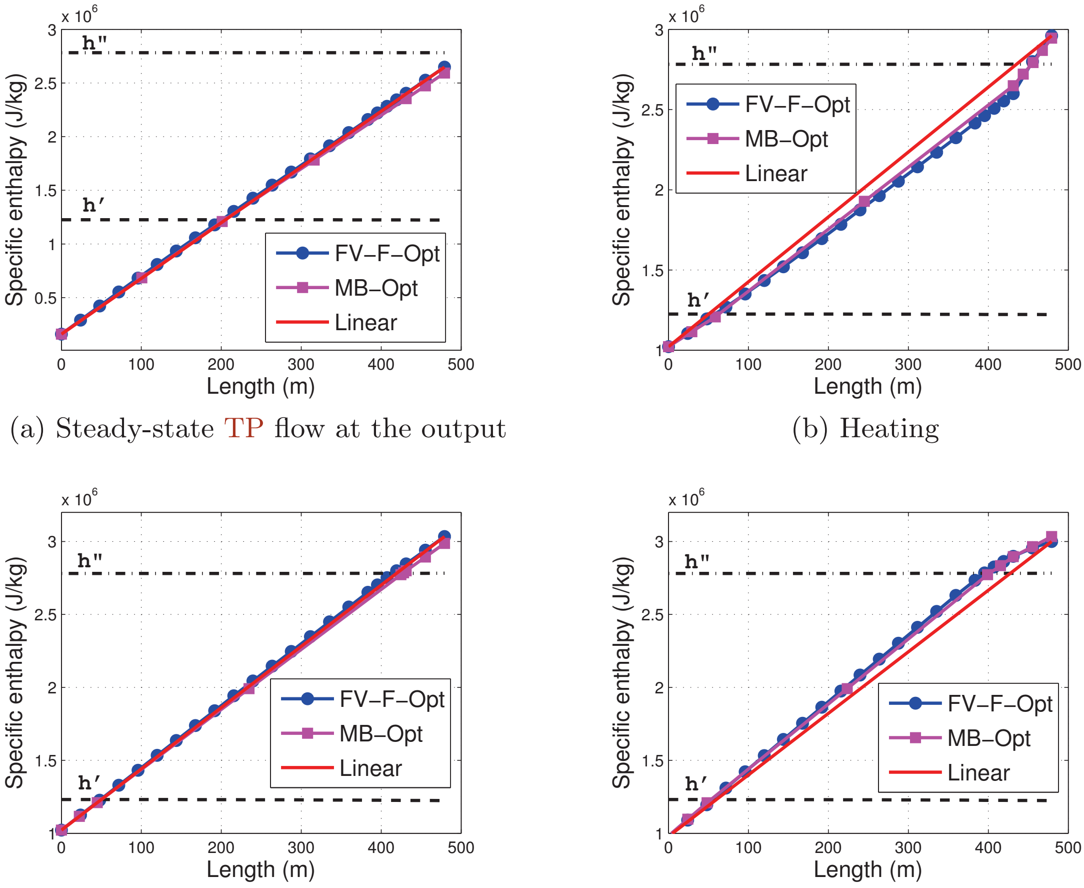
Fig. 8.10 shows the specific enthalpy distribution in the evaporator by considering the previously mentioned DISS models. The operating day corresponds to the 16^{th} April 2001 (DISS project). Fig. 8.10 shows four time instants in the simulation: steady-state two-phase flow at the output (cf. Fig. 8.10a), 11:30 (cf. Fig. 8.23), during heating (cf. Fig. 8.10b) 12:56, steady-state superheated steam at the output (cf. Fig. 8.10c) 15:33, and during cooling due to cloud disturbances (cf. Fig. 8.10d), 15:57.

The line with round markers is the specific enthalpy distribution for the FV-F-Opt model (22 CVs[(a)]), where the specific enthalpy at the inlet and outlet in each CV are represented by those round markers. Taking into account that the outlet specific enthalpy in one CV is the inlet specific enthalpy in the successive CV makes a total of 23 markers.

The plain line is a linear distribution considering the inlet and outlet specific enthalpies of the FV-F-Opt model. The line with square markers is the specific enthalpy distribution for the MB-Opt model, and each dot represents the specific enthalpy. There are three markers per CV (input, center, output), and the outlet specific enthalpy in one CV is the inlet specific enthalpy in the successive CV. Therefore, if the outlet flow is a two-phase flow, where the configurations of the two evaporators are SC-TP and TP, there are then seven markers in the figure (cf. Fig. 8.10a). If the outlet flow is superheated steam, where the configurations are SC-TP and TP-SH in this case, there are nine markers (cf. Figs. 8.10b, 8.10c and 8.10d).

The conclusions of this study are as follows.

[a] This number of CV was selected in order to clearly see the specific enthalpy values. Increasing the number of CVs produces analogous results.



(a) Steady-state TP flow at the output

(b) Heating

(c) Steady-state SH steam at the output

(d) Cooling

Figure 8.10: Specific enthalpy distribution in the evaporator considering the FV-F-Opt and MB-Opt models for the 16^{th} April 2001

- The specific enthalpy distribution in the evaporator is almost linear in steady-state using both models (cf. Figs. 8.10a and 8.10c), with the 11^{th} PTC injector being inactive (cf. Fig. 8.22e). Similar results were obtained in Ji et al. (2009).

- Switching MBMs can reproduce specific enthalpy distribution in the evaporator obtained by finite volume models in steady-state and transient predictions (cf. Fig. 8.10). This is because MBMs assume a linear specific enthalpy distribution in each CV, not in the whole evaporator. This assumption was made in § 4.4, § 7.7.2 and § 7.7.3.

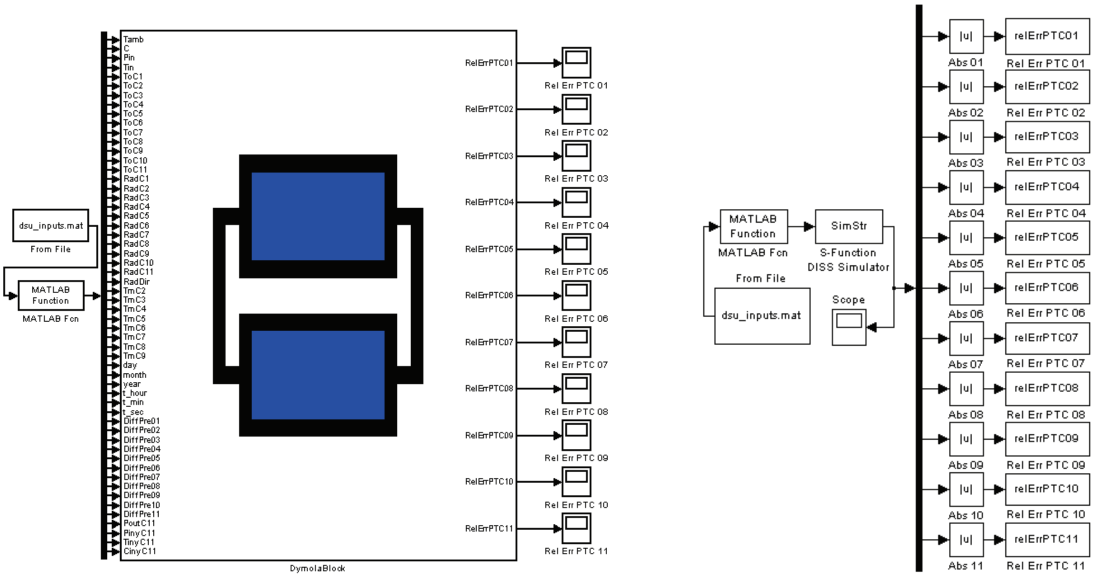
8.5 Calibration

The finite volume *ThermoFluid* DISS Model (cf. § 8.2.1) was used in the calibration process. The *Heuristic* approach (temporal restriction) was applied in order to avoid chattering, and the heat transfer coefficients were calculated using the Gnielinski (1976) and Gungor and Winterton (1986) HTCs for the one-phase and the two-phase, respectively. For the calibration process, experimental data from the DISS project (Zarza et al., 2002, 2004) was used.

Several calibration tools were analyzed and tested. The first attempt involved using the Dymola Design library from Dymola 6.0b (Dynasim AB, 2006). Unfortunately, the model was too complex and the calibration could not been performed. Secondly, the JModelica.org platform and the Optimica compiler (Åkesson et al., 2010) were considered, but unfortunately again this environment only works with Modelica 3, whereas the DISS model based on the *ThermoFluid* library is implemented in Modelica 2.2.1. However, positive feedback was obtained from the authors of this tool, and it seems to be suitable for future developments. Finally, Matlab/Simulink (The MathWorks Inc., 2012) was selected as the calibration tool. Dymola includes mechanisms to export Modelica models to Simulink in an easy and direct way, using a Simulink block (called *Dymola block*, cf. § 2.11.3) where the model, parameters and inputs can be specified. Moreover, Matlab is a widely used tool and provides toolboxes for genetic algorithms and parallel calibrations, namely the Genetic Algorithm and Parallel Computing toolboxes, respectively. The Matlab Distributed Computing Server, together with the Parallel Computing toolbox, can perform parallel calibrations. This is a future goal in order to take advantage of a 12-node cluster in the calibration process.

8.5.1 Genetic Algorithm

The multi-objective heuristic genetic algorithm function, GAMULTIOBJ, from the Matlab Genetic Algorithm Toolbox (Chipperfield et al., 1994), was used for the calibration. This function uses a controlled elitist genetic algorithm, a variant of Nondominated Sorting Genetic Algorithm (NSGA)-II (Deb, 2001) widely used in optimization. A controlled elitist genetic algorithm favors individuals that can help to increase the diversity of the population, even if they have a lower fitness value. On the other hand, an elitist genetic algorithm always favors individuals with better fitness values. It is important to maintain the diversity of the population for convergence to an optimal Pareto front. This is done by controlling elite members of the population by using two options to control this elitism – the Pareto fraction option limits the number of individuals on the Pareto front, while the distance function helps to maintain diversity on a front by favoring individuals that are relatively far away from the front. To form the next generation, the algorithm combines the current population and its offspring generated by standard bimodal crossover (Beyer and Deb, 2001) and polynomial mutation operators. Finally, the best individuals in terms of non-dominance and diversity are chosen. This new version of NSGA has a low time complexity. For more information about the multi-objective genetic algorithm consult Chipperfield et al. (1994), and for the calibration of Modelica models using genetic algorithms in Matlab/Simulink, together with a practical example,



(a) DISS *Dymola block* in Simulink (b) Calibration Model

Figure 8.11: DISS model calibration in Matlab/Simulink

consult Hongesombut et al. (2002). Fig. 8.11 shows the DISS Dymola block and the model used for the calibration process in Simulink.

8.5.2 Fitness Function

A fitness function, also called an objective function or a figure of merit, summarizes how close a particular solution is to a set of aims. In the calibration of the DISS Modelica model, the fitness function represents the sum of the differences in the outlet PTC temperatures between sampled experimental data from the temperature sensors and the simulated outlet PTC temperatures. Therefore, a lower value of the fitness function means a better solution.

Eq. 8.2 shows the PRE, $\epsilon_{r_i}(\lambda_i, j)$, considering the i^{th} PTC, the j^{th} sampled value and the particular parameter values for the i^{th} PTC, λ_i. Where $T_{s_i}(\lambda_i, j)$ is the simulated outlet temperature, $T_{r_i}(j)$ is the experimental temperature, n is the number of PTCs and m the number of samples.

$$\epsilon_{r_i}(\lambda_i, j) = 100 \cdot \frac{T_{s_i}(\lambda_i, j) - T_{r_i}(j)}{T_{r_i}(j)}, \qquad i \in [1, n], j \in [1, m]. \tag{8.2}$$

The multi-objective genetic algorithm minimizes the cumulative PRE, $\epsilon^+_{r_i}(\lambda_i)$, between the experimental and simulated outlet temperatures for each PTC, according to Eq. 8.3.

$$\epsilon^+_{r_i}(\lambda_i) = \sum_{j=1}^{m} |\epsilon_{r_i}(\lambda_i, j)|, \qquad i \in [1, n]. \tag{8.3}$$

Other metrics used in the following sections are: the mean PRE, $\bar{\epsilon}_{r_i}(\lambda_i)$, Eq. 8.4, the maximum PRE, $\epsilon^*_{r_i}(\lambda_i)$, Eq. 8.5 and a vector of PREs, $\epsilon_{r_i}(\lambda_i)$, Eq. 8.6, which is a vector containing the PREs between experimental and simulated temperatures for each sampled temperature value.

$$\bar{\epsilon}_{r_i}(\lambda_i) = \frac{\epsilon^+_{r_i}(\lambda_i)}{m}, \qquad i \in [1, n]. \tag{8.4}$$

$$\epsilon^*_{r_i}(\lambda_i) = \max_{j\in[1,m]} |\epsilon_{r_i}(\lambda_i, j)|, \qquad i \in [1, n]. \tag{8.5}$$

$$\epsilon_{r_i}(\lambda_i) = \bigcup_{j=1}^{m} \epsilon_{r_i}(\lambda_i, j), \qquad i \in [1, n]. \tag{8.6}$$

8.5.3 Globally Calibrated Parameters

One of the aims of the DISS project was to discover which of several two-phase HTCs better matched the experimental heat transfer coefficient. However, none of the tested HTCs were in good agreement with the experimental data, as all of them overestimated the experimental heat transfer coefficient. The conclusion reached in the DISS project was that the sensors did not have enough accuracy to estimate the heat transfer coefficient. The experimental heat transfer coefficient was calculated from the absorber tube and the HTF temperatures. Thermocouples (class 1) comply with International Electrotechnical Commission (IEC) 584-2:1982, which is an accuracy of ±1.25 K. It was proven in Zarza (2000) that a maximum variation of 0.6 K in temperature can cause a 20% variation in the heat transfer coefficient.

For this reason, the HTCs have not been studied in depth in this work, because using any of them does not considerably affect the outlet temperature when a low-level of discretization (or coarse grid) is considered. A coarse grid is advisable when the model is intended to be used as a tool for the design of an advanced ACS, in order to obtain a fast low-order model. Indeed, through several examples, it has been observed that using the constant heat transfer coefficients considered in Yebra (2006) against any of the HTCs previously mentioned in § 6.5.1 only caused a maximum change of less than ±2 K in the DISS model's outlet temperature. On the other hand, when a high-level of discretization (or fine grid) is considered, the heat transfer coefficient plays an important role in properly estimating HTF temperature.

The Gungor and Winterton (1986) HTC was used in the calibration because it shows one of the best agreements with experimental data (Zarza, 2000). The number of CVs considered in this calibration process were previously estimated in Yebra (2006) as two CVs per PTC, making a total of 22 CVs.

The Moody friction factor (f_m) and weights for the HTCs in each flow state – sub-cooled liquid (ω_{sc}), two-phase flow (ω_{tp}) and superheated steam (ω_{sh}) – were calibrated globally. The weights are useful for evaluating the convenience of the HTC, and values close to 1 mean a fairly good approximation. These parameters have been calibrated using a suitable set of data from several operating days. The globally calibrated parameters are shown in Tab. 8.3.

Parameter	f_m	ω_{sc}	ω_{tp}	ω_{sh}
Calibrated Value	0.02284	0.98268	0.90938	0.99412

Table 8.3: Globally calibrated parameters

8.5.4 Optical Efficiency Calibration

The DISS Modelica model is highly sensitive to the optical efficiency of the PTCs. Optical efficiency must therefore be properly estimated in order to obtain good approximations. Although there are other uncertainties in the model, e.g. thermal losses, empirical correlations have been used to model them, as described in Chap. 5.

The estimated peak optical efficiency value for a modified LS-3 PTC is 75%. However, this value can decrease considerably on a daily basis due to soiling of the PTCs (Zarza, 2000). The nominal estimated optical efficiency value must then be estimated, particularly for each experiment, in order to obtain accurate results.

The optical efficiency of a PTC (η_{opt}) is defined as a function of peak optical efficiency ($\eta_{opt,0^\circ}$) cf. Eq. 3.3, which considers a 0° IA, and as a function of the IAM. The value that must be calibrated is peak optical efficiency, and the IAM can be calculated from several empirical correlations, cf. § 3.7.3.

8.5.4.1 Calibration - 09th July 2001

For comparison purposes, the calibration data is the same as that used in Yebra (2006) and corresponds to 9th July 2001. This day was selected mainly because solar radiation disturbances (cf. Fig. 8.12a) and mass flow rate steps (cf. Fig. 8.12b) were present on this operating day.

PTC	1	2	3	4	5	6
$\eta_{opt_i,0^\circ}$	0.630456	0.628347	0.627592	0.628347	0.628519	0.623357
PTC	7	8	9	10	11	Mean
$\eta_{opt_i,0^\circ}$	0.624095	0.626673	0.626451	0.623447	0.622733	0.626355

Table 8.4: Calibrated peak optical efficiencies for 9th July 2001

Fig. 8.12 shows the inputs required for simulation which were obtained from the DISS facility database (cf. § 8.1). The DISS field inputs are: DSI, ambient temperature (Fig. 8.12a), inlet temperature, inlet mass flow rate (Fig. 8.12b) and outlet pressure (Fig. 8.12c). The 11th PTC injector inputs are: inlet temperature (Fig. 8.12d) and inlet mass flow rate (Fig. 8.12e). Additionally, Fig. 8.12 shows the DISS field and 11th PTC injector inlet pressures. The DSI, inlet mass flow rate and outlet pressure were filtered using the first-order low-pass filter.

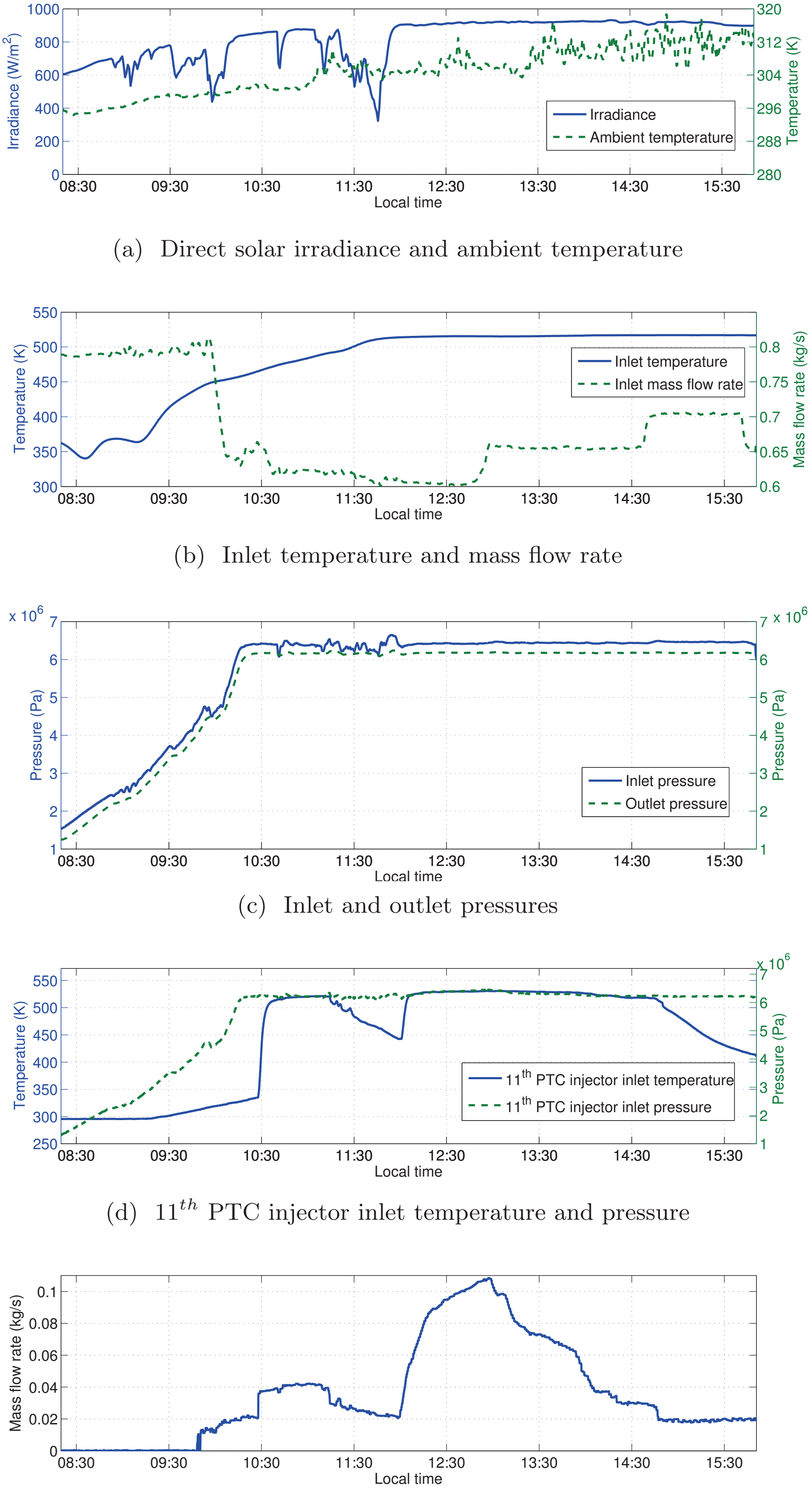
Calibration involves estimating the peak optical efficiency of each PTC ($\eta_{opt_i,0^\circ}$), which are shown in Tab. 8.4. Calibrated peak optical efficiencies are close to 63%, but these values are low in comparison to the theoretical values of 75%. Problems related to PTC tracking systems, soiling of the PTCs or unmodeled dynamics are the likely causes for this decrease, but this needs to be confirmed.

The mean and maximum PREs between experimental and simulated outlet temperatures for each PTC, according to Eqs. 8.4 and 8.5, respectively, are shown in Tab. 8.5. Figs. 8.13 and 8.14 show the experimental and simulated outlet temperatures for some PTCs, where the solid line and dashed line are the experimental and simulated outlet temperatures, respectively. It can be seen, especially in the 10th and 11th PTCs, that there is an unmodeled delay in simulated outlet temperature.

The maximum PRE for the 11th PTC has decreased, considering the previous DISS model (Yebra, 2006), from 12.7% to 10.7% (not considering the initialization in the previous DISS model, where large errors were obtained, up to 27.1%). The global mean PRE and the mean maximum PREs have also decreased from 0.79% to 0.64% and from 7.31% to 3.59%, respectively.

PTC	1	2	3	4	5	6
$\bar{\epsilon}_{r_i}(\eta_{opt_i,0^\circ})$	0.41%	0.40%	0.67%	0.28%	0.42%	0.44%
$\epsilon^*_{r_i}(\eta_{opt_i,0^\circ})$	2.59%	2.58%	3.13%	2.00%	2.04%	2.05%
PTC	7	8	9	10	11	Mean
$\bar{\epsilon}_{r_i}(\eta_{opt_i,0^\circ})$	0.43%	0.45%	0.64%	1.29%	1.57%	0.64%
$\epsilon^*_{r_i}(\eta_{opt_i,0^\circ})$	2.07%	4.09%	4.53%	6.82%	10.72%	3.59%

Table 8.5: Mean and maximum PREs between experimental and simulated outlet temperatures for 9th July 2001



(a) Direct solar irradiance and ambient temperature

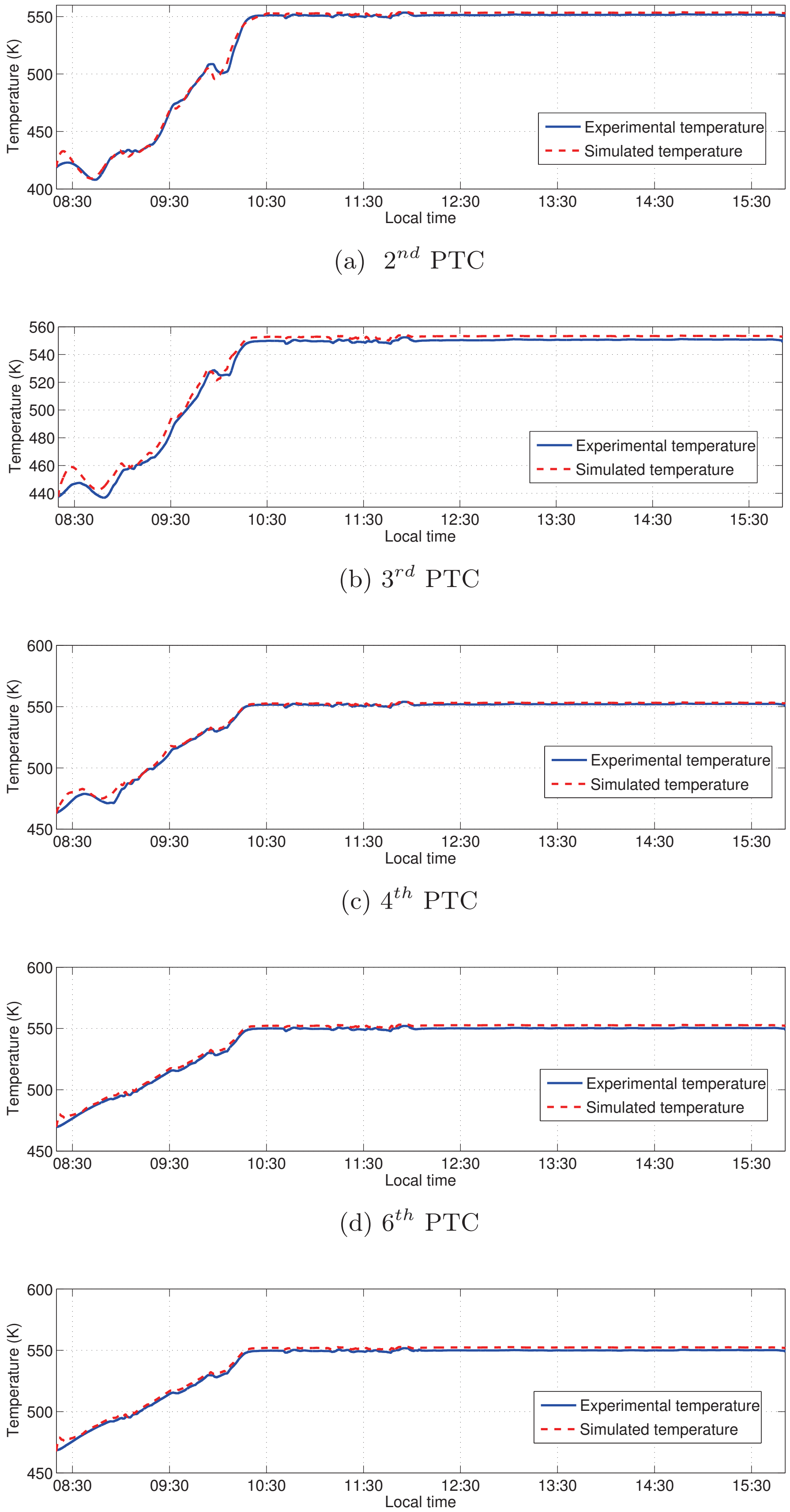
(b) Inlet temperature and mass flow rate

(c) Inlet and outlet pressures

(d) 11^{th} PTC injector inlet temperature and pressure

(e) 11^{th} PTC injector inlet mass flow rate

Figure 8.12: Inputs in the calibration - 9^{th} July 2001



(a) 2^{nd} PTC

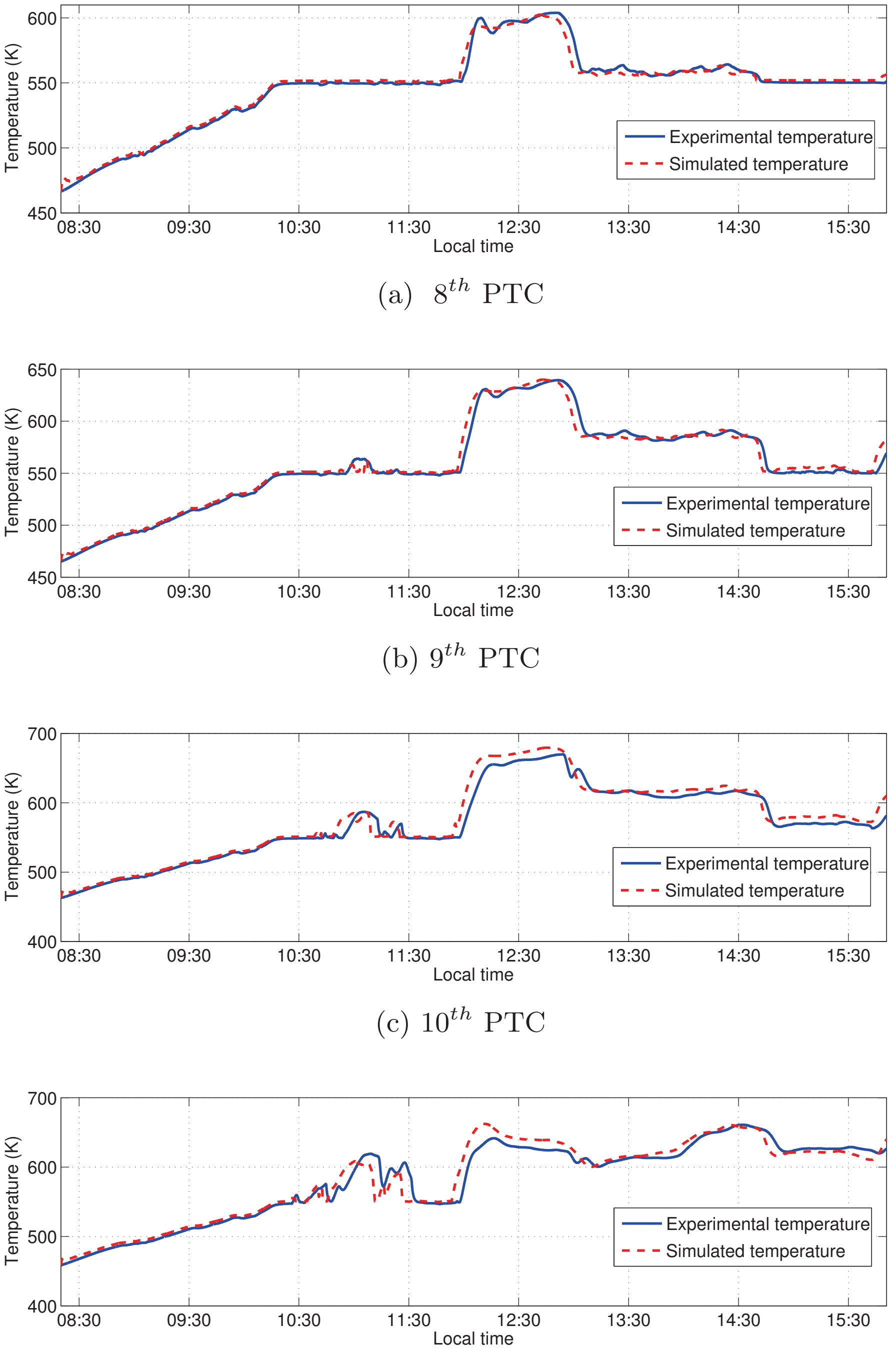
(b) 3^{rd} PTC

(c) 4^{th} PTC

(d) 6^{th} PTC

(e) 7^{th} PTC

Figure 8.13: Experimental and simulated temperatures at the output of evaporation PTCs in the calibration process for 9^{th} July 2001



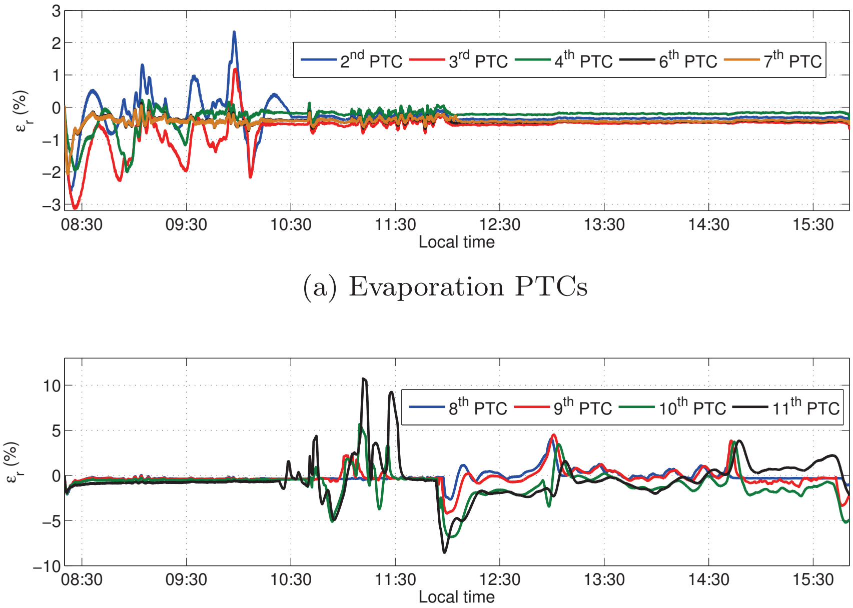
(a) 8^{th} PTC

(b) 9^{th} PTC

(c) 10^{th} PTC

(d) 11^{th} PTC

Figure 8.14: Experimental and simulated temperatures at the output of superheating PTCs in the calibration process for 9^{th} July 2001



(a) Evaporation PTCs

(b) Superheating PTCs

Figure 8.15: $\epsilon_{r_i}(\eta_{opt_i,0^\circ})$ between the experimental and simulated temperatures at the output of PTCs in the calibration process for 9^{th} July 2001

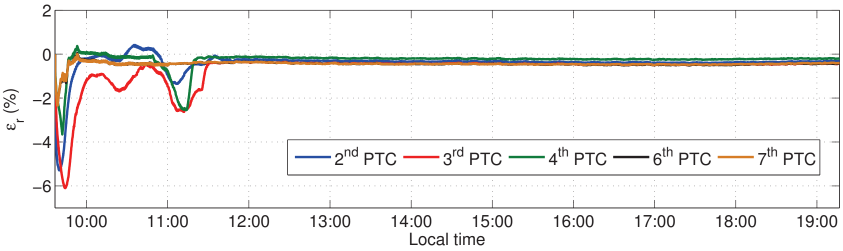
Fig. 8.15 shows the PREs between experimental and simulated outlet temperatures for some PTCs during the whole simulation. The PRE committed is lower when the outlet fluid is two-phase flow. This is because the temperature is only calculated by considering pressure for the two-phase flow. The pressure is properly estimated by the model, and the pressure drop is also low (cf. § 5.5).

8.6 Validation

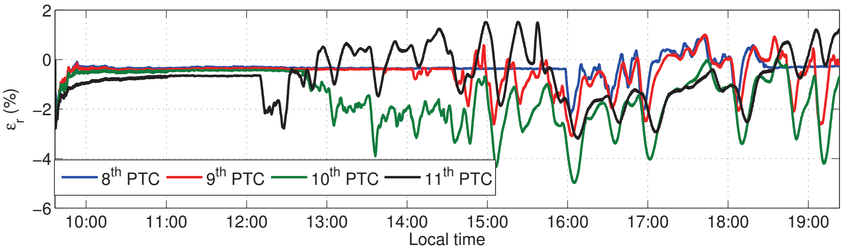
With the aim of validating the previously calibrated optical efficiencies shown in Tab. 8.4, the same optical efficiencies were used for a different operating day. The operating day selected was 10^{th} July 2001 because, as mentioned earlier, the DISS model is highly sensitive to optical efficiency. In order to use the closest optical efficiencies, the day following the calibration day was selected. On this operating day, inlet mass flow rate steps were applied in the experiment (cf. Fig. 8.17b).

8.6.1 Validation - 10^{th} July 2001

The model inputs mentioned in § 8.5.4.1 are shown in Fig. 8.17. Figs. 8.18 and 8.19 show the experimental and simulated outlet temperatures for some



(a) Evaporation PTCs



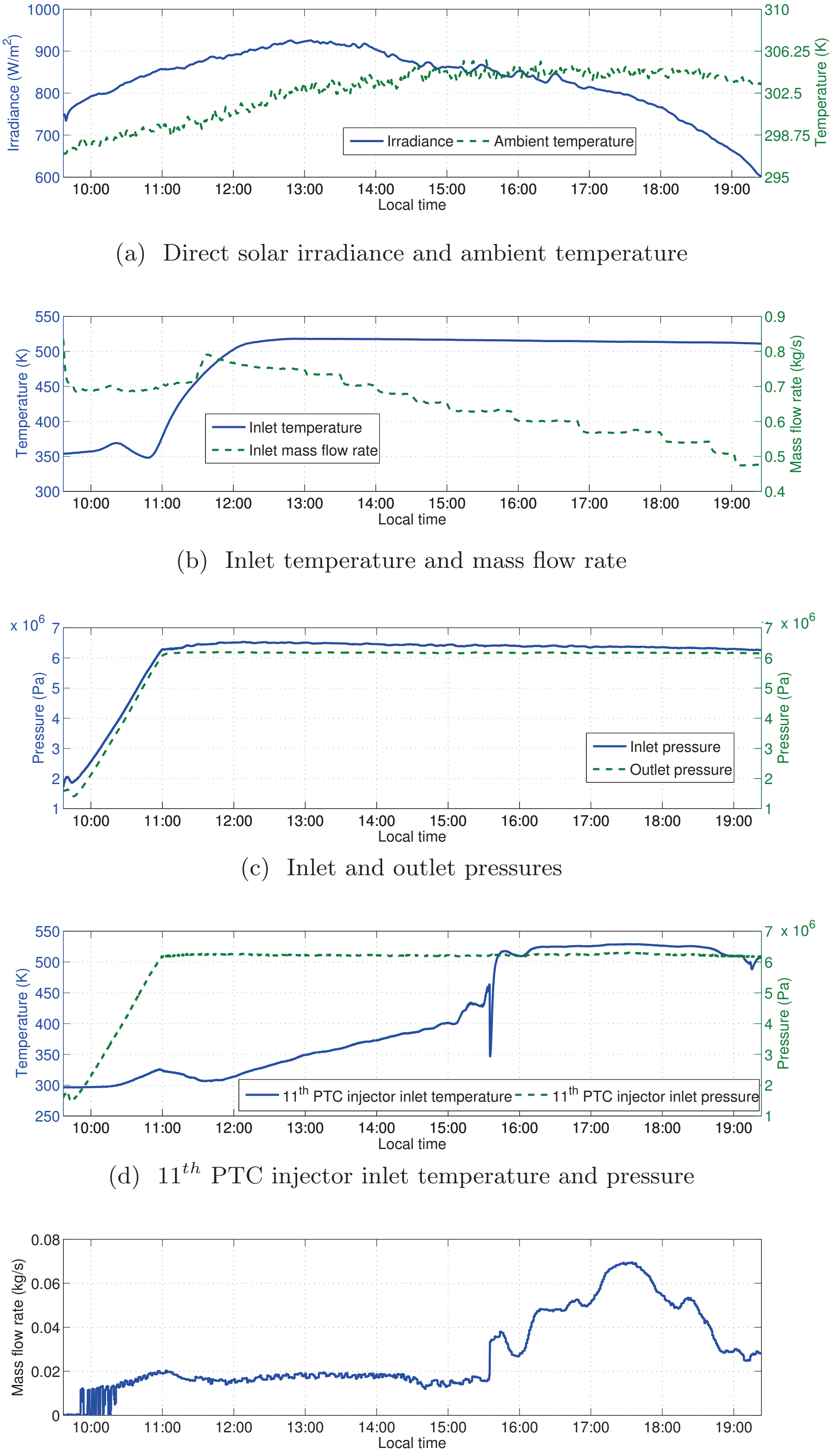
(b) Superheating PTCs

Figure 8.16: $\epsilon_{r_i}(\eta_{opt_i,0^\circ})$ between experimental and simulated temperatures at the output of PTCs in the validation process for 10^{th} July 2001

PTCs, where the solid line is the experimental temperature and the dashed line is the simulated temperature. It can be seen that the simulated temperatures follow the experimental temperatures' dynamics correctly and a delay is observed again, especially in the 10^{th} PTC (cf. Fig. 8.19c). In the 11^{th} PTC, the outlet temperature is overestimated (cf. Fig. 8.19d) when injected steam from the 11^{th} PTC injector increases in mass flow rate (15:33 in simulation), as can be seen in Figs. 8.19d and 8.17e. In Tab. 8.6, the mean and maximum PREs for each PTC and mean values for the field are shown. Fig. 8.16 shows the PREs for some PTCs during the whole simulation. The error committed in the validation is low, with a global mean PRE of 0.63% and a mean maximum PRE of 3.77%.

PTC	1	2	3	4	5	6
$\bar{\epsilon}_{r_i}(\eta_{opt_i,0^\circ})$	0.50%	0.51%	0.79%	0.32%	0.35%	0.48%
$\epsilon^*_{r_i}(\eta_{opt_i,0^\circ})$	5.27%	5.29%	6.10%	3.66%	2.97%	2.28%
PTC	7	8	9	10	11	Mean
$\bar{\epsilon}_{r_i}(\eta_{opt_i,0^\circ})$	0.47%	0.44%	0.63%	1.51%	0.99%	0.63%
$\epsilon^*_{r_i}(\eta_{opt_i,0^\circ})$	2.35%	2.29%	3.09%	4.99%	3.19%	3.77%

Table 8.6: Mean and maximum PREs between experimental and simulated outlet temperatures for 10^{th} July 2001



(a) Direct solar irradiance and ambient temperature

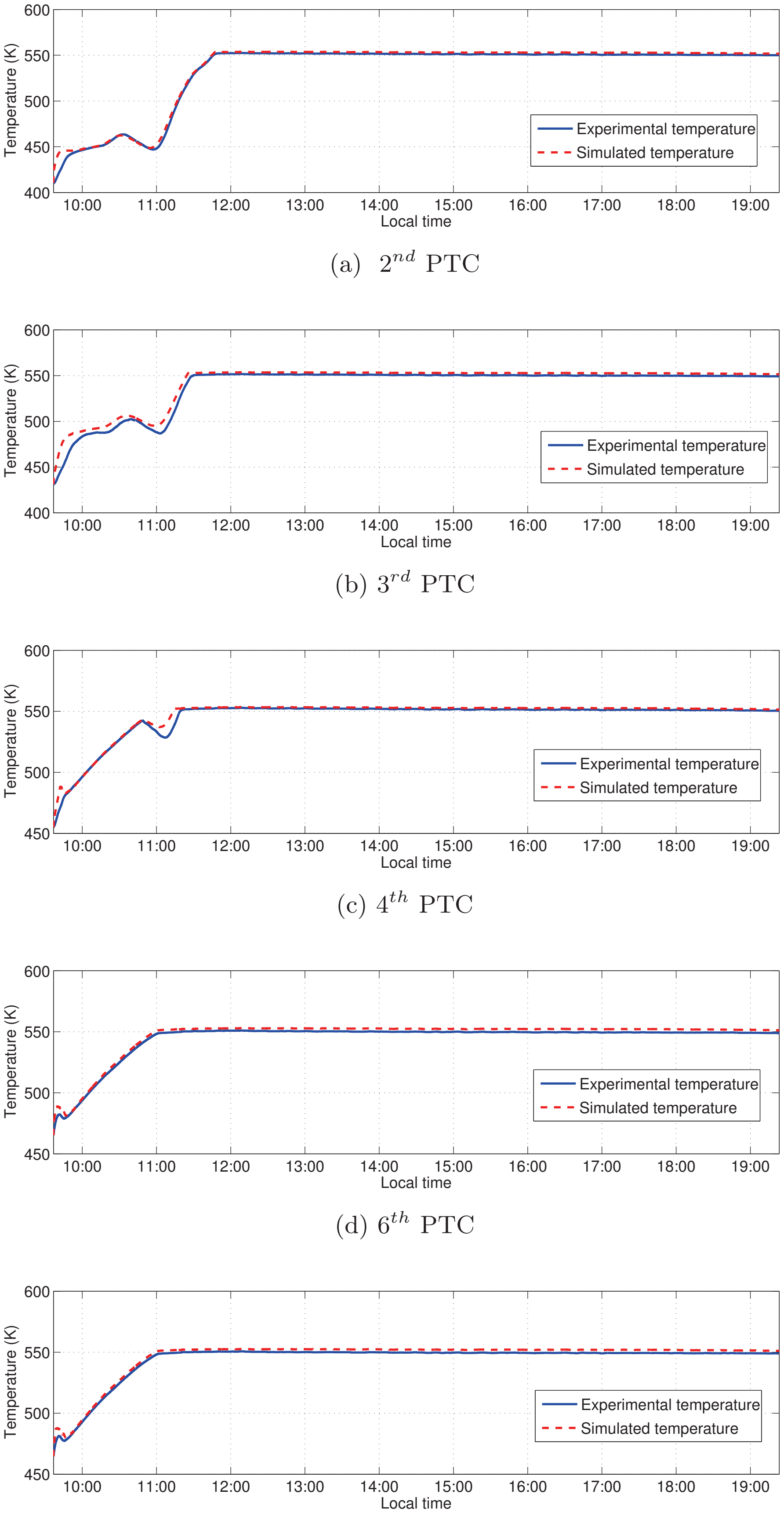
(b) Inlet temperature and mass flow rate

(c) Inlet and outlet pressures

(d) 11^{th} PTC injector inlet temperature and pressure

(e) 11^{th} PTC injector inlet mass flow rate

Figure 8.17: Inputs in the validation - 10^{th} July 2001



(a) 2^{nd} PTC

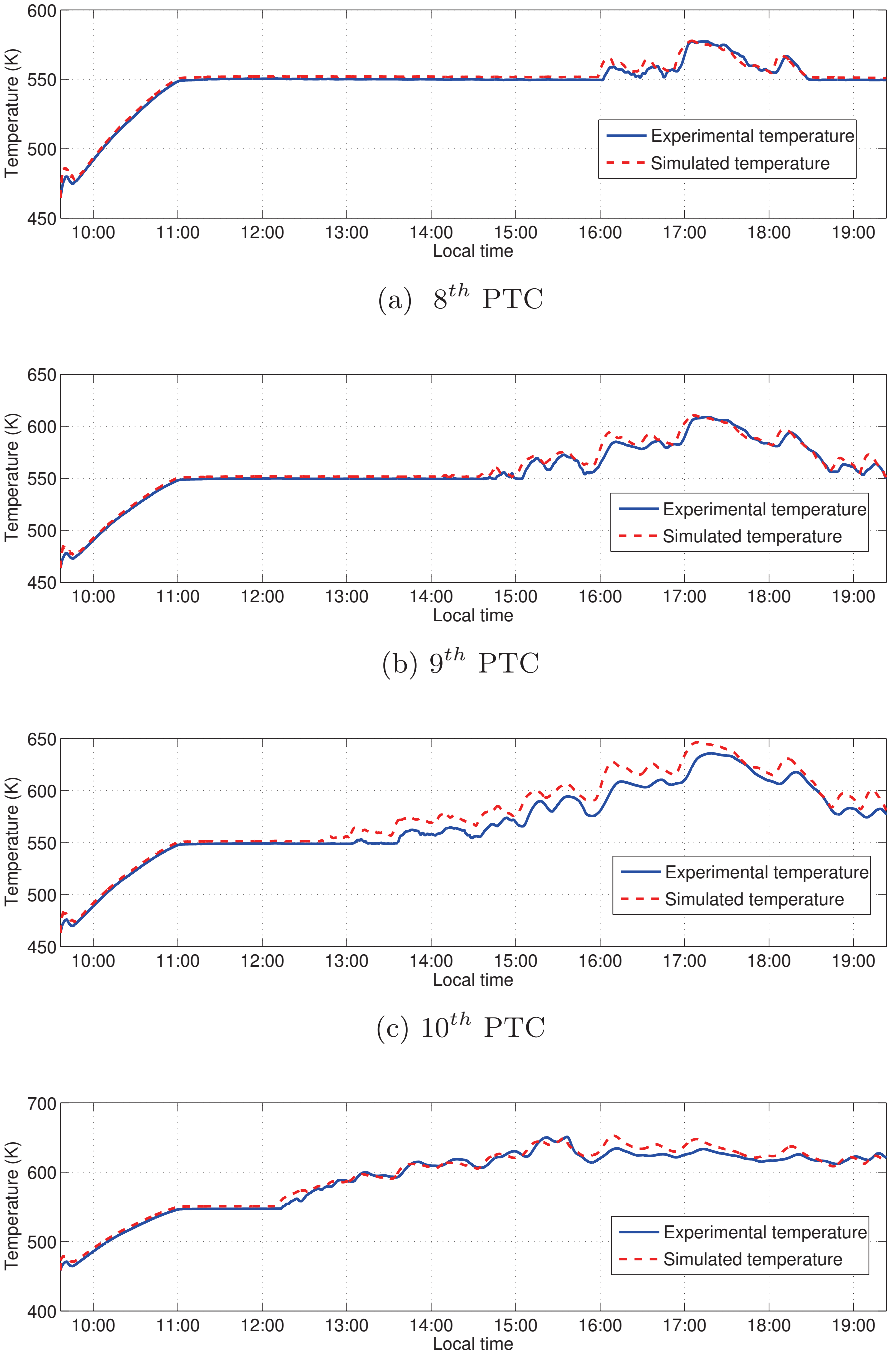
(b) 3^{rd} PTC

(c) 4^{th} PTC

(d) 6^{th} PTC

(e) 7^{th} PTC

Figure 8.18: Experimental and simulated temperatures at the output of evaporation PTCs in the validation process for 10^{th} July 2001



(a) 8th PTC

(b) 9th PTC

(c) 10th PTC

(d) 11th PTC

Figure 8.19: Experimental and simulated temperatures at the output of superheating PTCs in the validation process for 10th July 2001

8.7 Level of Discretization in the DISS Fluid Model

The optimized finite volume *Modelica Fluid* DISS model (FV-F-Opt) is studied in this section in terms of level of discretization, accuracy and simulation time.

Using experimental inputs from several operating days, the FV-F-Opt model was simulated with different numbers of CVs. Heat transfer driven by constant coefficients per fluid region was considered between the absorber tube and the HTF. The heat transfer coefficient values were: 2000 W/(m^2 · K) (SC liquid), 7500 W/(m^2 · K) (TP flow) and 2500 W/m^2/K) (SH vapor). These values were estimated in Yebra (2006). Peak optical efficiencies are detailed in Tab. 8.9.

Several simulations were performed, increasing the number of CVs from two CVs to ten CVs per PTC. Tab. 8.7 shows, for a particular operating day (16th April 2001), the number of CVs considered in each row, the maximum and mean PREs committed between those simulations in outlet temperature and their simulation times. This information gives an idea of the errors committed and time needed for the simulations. Tab. 8.8 shows the maximum and mean PREs against experimental data. The results are similar for other operating days.

Number of CVs	$\epsilon^*_{r_{11}}(\eta_{opt_{11},0^\circ})$	$\bar{\epsilon}_{r_{11}}(\eta_{opt_{11},0^\circ})$	CPU-time Int. (s)
22 - 33	0.6180%	3.4824%	21.2 - 29.7
33 - 44	0.3289%	1.5968%	29.7 - 68.9
44 - 55	0.2226%	0.9876%	68.9 - 108
55 - 66	0.1433%	0.8330%	108 - 157
66 - 77	0.0982%	0.6506%	157 - 215
77 - 88	0.0807%	0.5357%	215 - 274
88 - 99	0.0600%	0.2748%	274 - 323
99 - 110	0.0552%	0.2542%	323 - 500

Table 8.7: Results for the level of discretization studied in the FV-F-Opt model for 16th April 2001

Nº of CVs	22	33	44	55	66
$\bar{\epsilon}_{r_{11}}(\eta_{opt_{11},0^\circ})$	2.12%	1.65%	1.47%	1.32%	1.22%
$\epsilon^*_{r_{11}}(\eta_{opt_{11},0^\circ})$	13.59%	13.20%	13.24%	13.09%	12.90%
Nº of CVs	77	88	99	110	
$\bar{\epsilon}_{r_{11}}(\eta_{opt_{11},0^\circ})$	1.25%	1.18%	1.15%	1.10%	
$\epsilon^*_{r_{11}}(\eta_{opt_{11},0^\circ})$	12.99%	12.61%	12.61%	12.63%	

Table 8.8: Maximum and mean PRE in outlet temperature between experimental data and the FV-F-Opt model, considering different numbers of CVs

One way of compromising between accuracy and simulation efficiency would be to consider 55 CVs, because the maximum and mean PRE between the 55-CV and 66-CV simulations are lower than 0.9% and 0.15%, respectively. Additionally, the 55-CV model is 30% faster than the 66-CV model. The maximum and mean PRE committed against experimental data for the 55-CV model are 13.09% and 1.32%, respectively (cf. Tab. 8.8).

8.8 Comparison Between Different DISS Modelica Models

A comparison of four DISS Modelica models, previously introduced in § 8.2, is presented in this section. Experimental data, taken from the DISS project, was used in order to compare the accuracy and performance of the models.

8.8.1 DISS Modelica Models Considered in the Comparison

The DISS models and their configurations considered in this comparison are summarized in the following list.

- *Finite Volume ThermoFluid DISS model (FV-Th) (§ 8.2.1)*. In this model, the *Heuristic* approach (temporal restriction) is used together with the *ThermoFluid* library. The number of CVs considered in these simulations is five CVs per PTC, making a total of 55 CVs. The same number applies to the *Modelica Fluid* finite volume PTCs. The friction factor value was calibrated, as shown in Tab. 8.3.

- *Optimized Finite Volume Modelica Fluid DISS model (FV-F-Opt) (§ 8.2.2)*. Five CVs per PTC were used, making a total of 55 CVs. This number of CVs is justified in § 8.7.

- *Optimized Moving Boundary + Finite Volume DISS Modelica model (MB+FV-Opt) (§ 8.2.4)*. In the moving boundary evaporator there are up to three active CVs, whereas in the finite volume evaporator there are five CVs, making a total of eight CVs.

- *Optimized Moving Boundary DISS Modelica model (MB-Opt) (§ 8.2.3)*. Each moving boundary evaporator has up to three active CVs, making a total of six CVs.

All the models have constant heat transfer coefficients per fluid region, but they are joined smoothly by means of cubic splines (cf. § 6.10). Heat transfer coefficient values are mentioned in § 8.7 and were estimated in Yebra (2006).

The *Modelica Fluid* finite volume evaporators are configured with the A_V_B structure, because the best results were obtained using this set up (cf. § 8.2.2). The pressure loss model used for the *Modelica Fluid* finite volume evaporators is the detailed pressure loss model (Franke et al., 2009b), where pipe roughness is $3 \cdot 10^{-5}$ m (Zarza, 2000). The initialization configuration considered was the default configuration (cf. § 7.13.1).

Peak optical efficiencies are particular for each model and operating day (cf. § 8.8.2). This information is summarized in Tab. 8.9. Only one value is provided, meaning that all PTCs have that peak optical efficiency. The values in Tab. 8.9 have not been calibrated, except for values in Tab. 8.4. These values have been estimated in order to perform the simulations.

Operating day	FV-Th § 8.2.1	FV-F-Opt § 8.2.2	MB+FV-Opt § 8.2.4	MB-Opt § 8.2.3
1^{st} April 2001	0.610	0.610	0.608	0.598
16^{th} April 2001	0.610	0.610	0.608	0.598
9^{th} July 2001	Tab. 8.4	0.637	0.635	0.624
10^{th} July 2001	Tab. 8.4	0.637	0.635	0.624

Table 8.9: Peak optical efficiencies for each model and operating day

8.8.2 Operating Days Considered in the Comparison

The four models described in § 8.8.1 are compared against experimental data taken from the DISS project. The operating days were selected to expose the four models to a wide range of different operating conditions and disturbances by means of mass flow rate steps and cloud disturbances. Tab. 8.10 provides more information about the selected operating days.

Operating day	Pressure[b] (MPa)	Inlet $\dot{m}$ steps (kg/s)	Cloud disturbances	Model inputs
1^{st} April 2001	3	0.5 - 1.8	No	Fig. 8.20
16^{th} April 2001	6	0.5 - 1.7	Yes	Fig. 8.22
9^{th} July 2001	6	0.6 - 0.8	Yes	Fig. 8.12
10^{th} July 2001	6	0.5 - 0.8	No	Fig. 8.17

Table 8.10: Selected operating days for the comparison of the DISS models

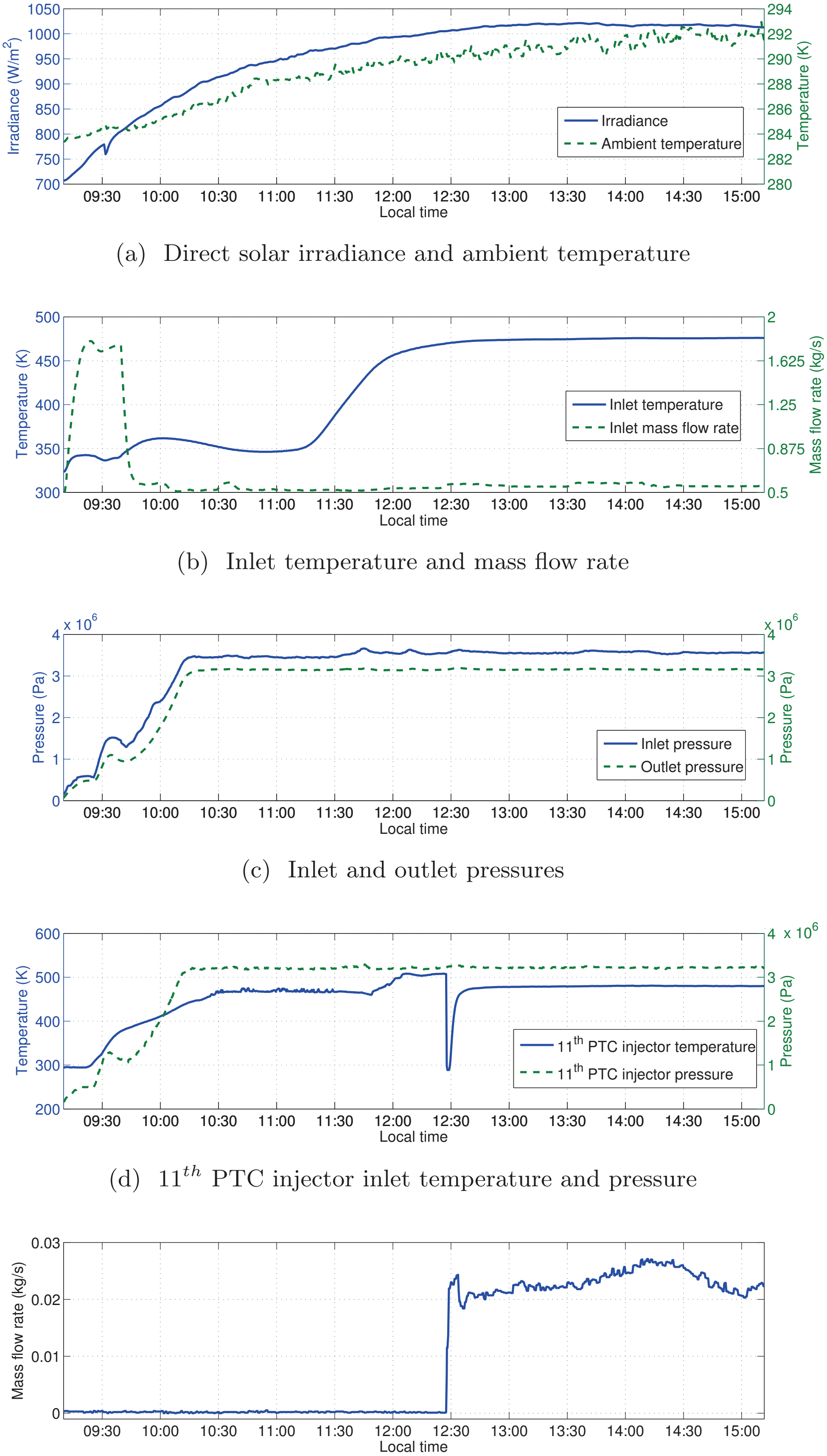
8.8.3 Simulation Results

In the following subsections, each of the operating days (cf. § 8.8.2) is simulated in the four models (cf. § 8.8.1). The model inputs are shown, together with the experimental and simulated DISS field outlet temperatures obtained by each model. The simulation times, the maximum and mean PREs committed against experimental data are also compared (not considering initialization). DASSL (cf. § 2.9.2.1) was the numerical solver employed in this instance, while the absolute and relative tolerances were set to 10^{-4}.

8.8.3.1 1^{st} April 2001

The model inputs mentioned in § 8.5.4.1 are shown in Fig. 8.20. The DISS field outlet experimental and simulated temperatures obtained by each model are shown in Fig. 8.21. The simulation times and the maximum and mean PREs committed against the experimental data are shown in Tab. 8.11.

[b] Nominal outlet pressure.



(a) Direct solar irradiance and ambient temperature

(b) Inlet temperature and mass flow rate

(c) Inlet and outlet pressures

(d) 11^{th} PTC injector inlet temperature and pressure

(e) 11^{th} PTC injector inlet mass flow rate

Figure 8.20: Inputs for the 1^{st} April 2001

Model	$\bar{\epsilon}_{r11}(\eta_{opt11,0°})$	$\epsilon^*_{r11}(\eta_{opt11,0°})$	CPU-time Int. (s)
FV-Th	2.08%	7.97%	95.4 [(d)]
FV-F-Opt	**1.85%**	**6.03%**	56.4
MB+FV-Opt	2.15%	7.48%	**46.8**
MB-Opt	2.44%	7.61%	50.6

Table 8.11: Comparison between DISS models for 1st April 2001

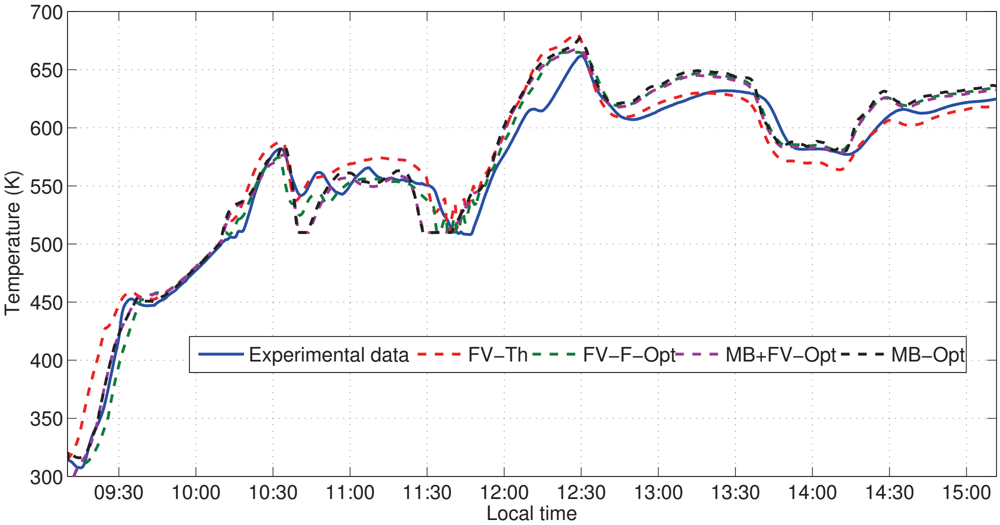


Figure 8.21: Comparison in outlet temperature between DISS models for 1st April 2001

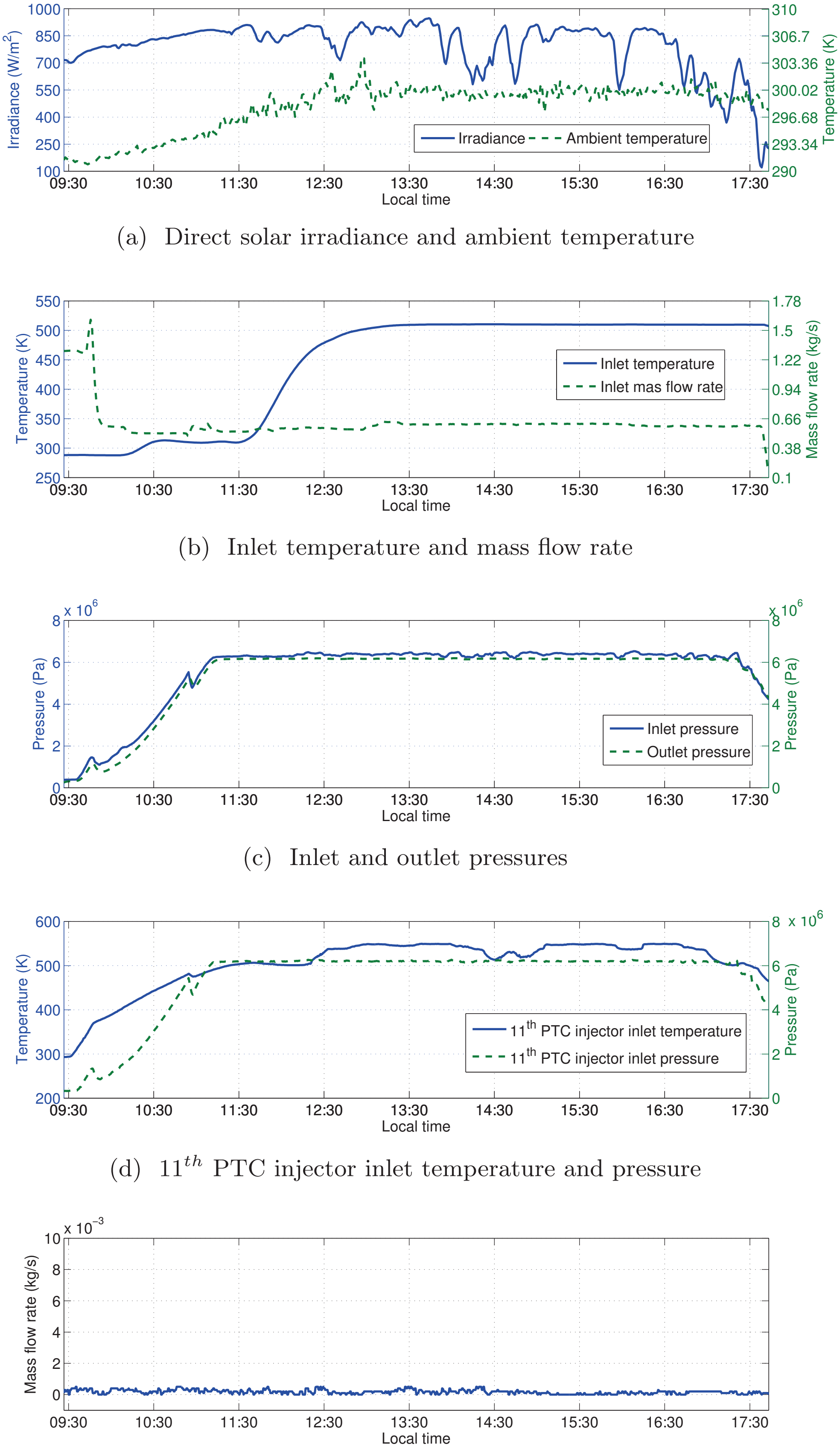
8.8.3.2 16th April 2001

The model inputs mentioned in § 8.5.4.1 are shown in Fig. 8.22. The DISS field outlet experimental and simulated temperatures obtained by each model are shown in Fig. 8.23. The simulation times and the maximum and mean PREs committed against the experimental data are shown in Tab. 8.12.

Model	$\bar{\epsilon}_{r11}(\eta_{opt11,0°})$	$\epsilon^*_{r11}(\eta_{opt11,0°})$	CPU-time Int. (s)
FV-Th	1.46%	13.31%	155
FV-F-Opt	**1.32%**	**13.09%**	108
MB+FV-Opt	1.43%	13.12%	**43.4**
MB-Opt	1.41%	13.38%	56.8

Table 8.12: Comparison between DISS models for 16th April 2001

[d] Chattering detected by the *Heuristic* approach, time: 13:25, 1st PTC, 4th CV, transition: sub-cooled liquid → two-phase flow.



(a) Direct solar irradiance and ambient temperature

(b) Inlet temperature and mass flow rate

(c) Inlet and outlet pressures

(d) 11^{th} PTC injector inlet temperature and pressure

(e) 11^{th} PTC injector inlet mass flow rate

Figure 8.22: Inputs for 16^{th} April 2001

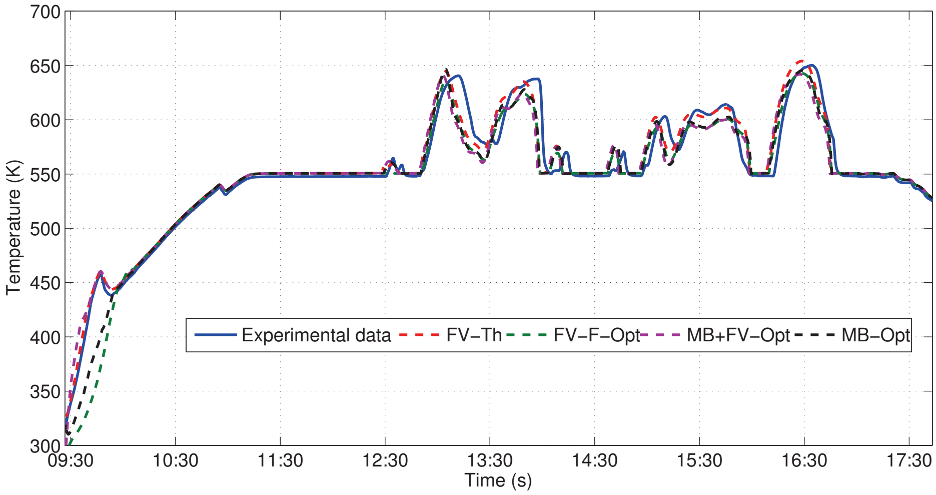


Figure 8.23: Comparison in outlet temperature between DISS models for 16th April 2001

8.8.3.3 9th July 2001

The model inputs were shown previously in Fig. 8.12. The DISS field outlet experimental and simulated temperatures obtained by each model are shown in Fig. 8.24. The simulation times and the maximum and mean PREs committed against the experimental data are shown in Tab. 8.13. Note that the optical efficiencies have been only calibrated for the FV-Th model (cf. § 8.9). The results obtained with this model are now better than the calibration results (cf. § 8.5), because 55 CVs have been considered. However, this requires more computational time.

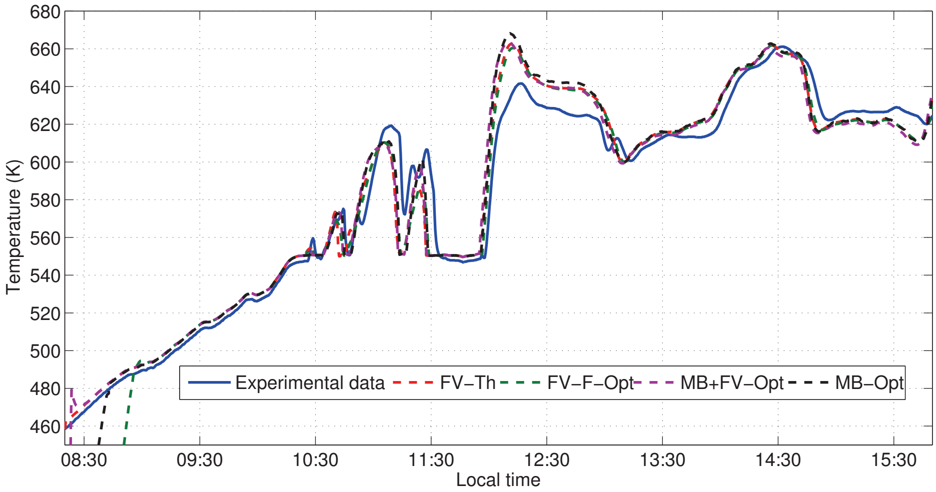


Figure 8.24: Comparison in outlet temperature between DISS models for 9th July 2001

Model	$\bar{\epsilon}_{r11}(\eta_{opt11,0^\circ})$	$\epsilon^*_{r11}(\eta_{opt11,0^\circ})$	CPU-time Int. (s)
FV-Th	1.43%	10.22%	84.0
FV-F-Opt	**1.25%**	**9.39%**	73.1
MB+FV-Opt	1.59%	10.58%	36.4
MB-Opt	1.69%	10.65%	**28.7**

Table 8.13: Comparison between DISS models for 9th July 2001

8.8.3.4 10th July 2001

The simulator inputs were shown previously in Fig. 8.17. The DISS field experimental and simulated outlet temperatures obtained by each model are shown in Fig. 8.25. The simulation times and the maximum and mean PREs committed against the experimental data are shown in Tab. 8.14.

Model	$\bar{\epsilon}_{r11}(\eta_{opt11,0^\circ})$	$\epsilon^*_{r11}(\eta_{opt11,0^\circ})$	CPU-time Int. (s)
FV-Th	**0.89%**	3.15%	70.8
FV-F-Opt	0.91%	**2.87%**	67.9
MB+FV-Opt	0.91%	3.18%	**31.6**
MB-Opt	1.00%	3.78%	45.2

Table 8.14: Comparison between DISS models for 10th July 2001

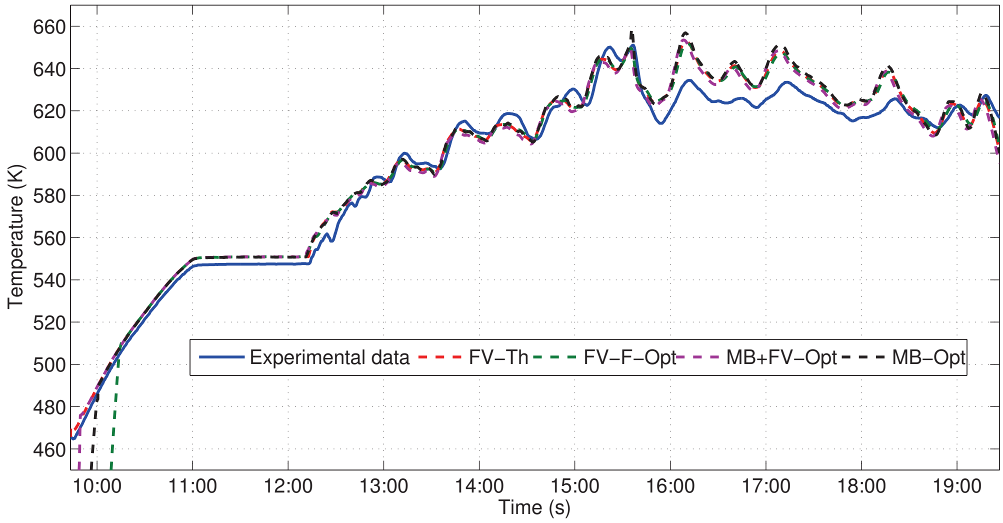


Figure 8.25: Comparison in outlet temperature between DISS models for 10th July 2001

8.8.4 Comparison Conclusions

From the experimental results obtained in § 8.8.3, the following conclusions can be inferred.

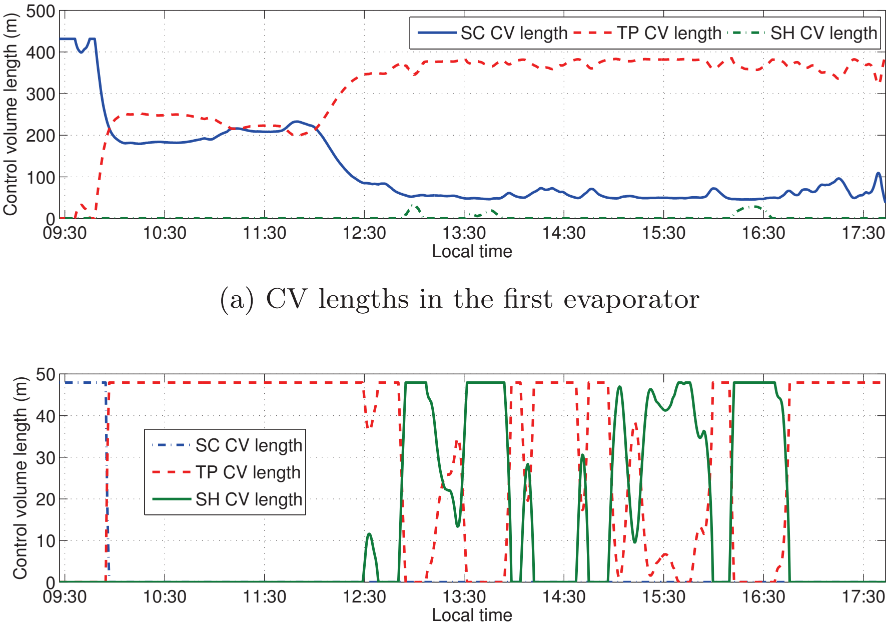
- *Accuracy.* The four models exhibit good agreement with the experimental data. The number of CV in finite volume models must be estimated in order to obtain accurate results, which was proven in the integrity tests performed in § 7.13.1 with the *Modelica Fluid* library. For this reason, the level of discretization in the DISS *Modelica Fluid* model was studied in § 8.7.

  The MB-Opt and MB+FV-Opt models are considerably accurate with only six and eight CVs, respectively, in comparison to 55 CV in the finite volume models. Using a fine grid in finite volume models, i.e. increasing the number of CVs, may result in more accuracy, but this would be at the expense of simulation time. The accuracy of MBMs was confirmed in Bendapudi et al. (2008).

- *Simulation time.* The FV-Th and FV-F-Opt models require more computational time than the other two models, because the number of CVs is higher. On the other hand, this number of CVs is required to obtain accurate results. The FV-Th model is the slowest in the comparison, and it implements the *Heuristic* approach, which is not implemented in the remaining models. Additionally, the FV-Th model was implemented in Modelica 2.2.1 and simulated in Dymola 6.0, whereas the remaining models were implemented in Modelica 3.2 and simulated in Dymola 2013. Improvements in the M&S tool might also be related to the gain in speed.

  The low number of CVs makes the MB-Opt and MB+FV-Opt faster. The hybrid MB+FV-Opt model exhibits better performance than the MB-Opt model in general because having two MBMs increases the number of state events (for dynamic switching). Additionally, the MB-Opt model includes a junction component (cf. § 8.2.3), which also increases computational time. The good performance of MBMs was demonstrated in Grald and MacArthur (1992); Bendapudi (2004); Bendapudi et al. (2008).

- *Initialization.* The FV-Th has good properties with respect to initialization in the steady-state. On the other hand, *Modelica Fluid* models (FV-F-Opt and MB+FV-Opt) are hard to initialize in a steady-state, so they had to be initialized with the default configuration (cf. § 8.8.1).



(a) CV lengths in the first evaporator

(b) CV lengths in the second evaporator

Figure 8.26: CV lengths in the MB-Opt model for 16th April 2001

MBMs are not easy to initialize, so the *MBMs* library facilitates this task by providing GUI options (cf. § 7.12.8), e.g. initially inactive CVs. However, if the inlet or outlet flow is two-phase, an estimation of the two-phase CV length is required, which is not an easy undertaking. When two MBMs are connected in a series, this situation is more difficult because the configurations in the MBMs must be coherent with each other.

Figs. 8.26a and 8.26b show examples of how switching between different configurations is performed and how the CV lengths evolve in the two evaporators (first ten PTCs and last PTC evaporators) in the MB-Opt model (cf. § 8.2.3) for 16th April 2001. On this operating day, there are cloud disturbances (cf. Fig. 8.22a), so the outlet flow varies between superheated steam and a two-phase flow (cf. Fig. 8.26b). The inlet and outlet flows were initially sub-cooled liquid.

- *Delay.* An unmodeled delay, which affects all the DISS models, seems to be a fluid transport delay. This delay is significantly higher in the last PTCs.

  In order to appreciate this delay, Fig. 8.27 shows the experimental and simulated outlet temperatures of the 10th PTC for 16th April 2001, using the MB-Opt model (cf. § 8.2.3). This PTC was selected in order to see the delay clearly, because less transitions between two-phase and superheated steam outlet flows occur (cf. Fig. 8.26a). The maximum and mean PREs are 4.64% and 0.75%, respectively. On the other hand, Fig. 8.28 shows the same outlet temperature, but it is delayed in time. In this case, the maximum and mean PREs are 3.61% and 0.51%, respectively, and only the outlet's superheated temperature has been delayed. This delay has been implemented using the *variableDelay* block from the MSL. The delay is not constant, as it has been set to 150 s in the first half of the simulation, whereas 300 s has been considered in the last part. This implementation is not satisfactory because, first of all, it is not based on first principles, and secondly, the delay was calculated by trial and error for each particular case.

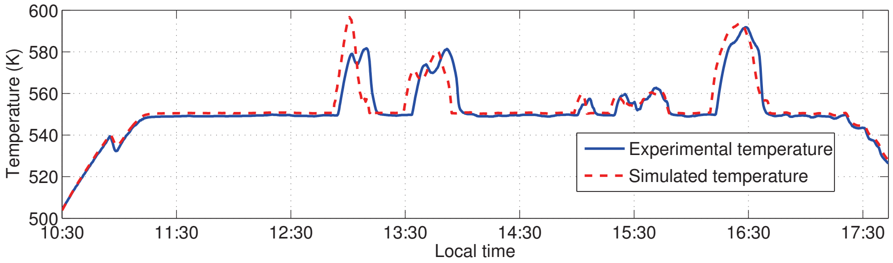


Figure 8.27: Experimental and simulated outlet temperatures of the 10th PTC for 16th April 2001

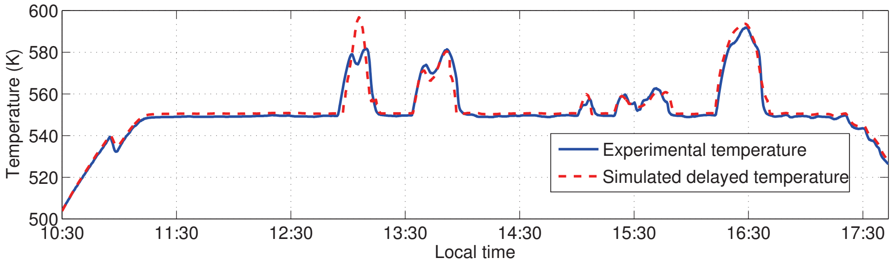


Figure 8.28: Experimental and delayed simulated outlet temperatures of the 10th PTC for 16th April 2001

- *Temperature overestimation.* The DISS outlet temperature is overestimated when injected steam from the 11th PTC injector increases in mass flow rate, as can be seen for 9th July 2001 (cf. Figs. 8.24 and 8.12e) from 12:00 to 13:00, and for 10th April 2001 (cf. Figs. 8.25 and 8.17e) from 15:30 to 18:30. The model commits a considerable error when the 11th PTC injector is active.

8.9 Summary and Conclusions

In this chapter, the simulation framework was introduced. Different DISS Modelica models were also developed from finite volume and moving boundary models, while unknown parameters were calibrated using multi-objective genetic algorithms. OpenMP was used to increase simulation speed. It was demonstrated that a linear specific enthalpy distribution in each CV in MBMs can be assumed, even in transients.

The four developed DISS Modelica models were validated against experimental data and exposed to a wide range of different operating conditions and disturbances, proving that they can take full advantage of dynamic simulations and can be used for the design of advanced ACSs, implementing a method to avoid chattering, if required.

8.10 Related Scientific Contributions

The scientific contributions related to this chapter are enumerated and briefly commented on in the following list. These contributions were previously listed in § 1.4.

1. Bonilla, J., Yebra, L. J., Dormido, S. and Zarza, E. Parabolic-trough solar thermal power plant simulation scheme, multi-objective genetic algorithm calibration and validation. *Solar Energy*, vol. 86(1), pages 531–540, 2012. ISSN 0038-092X.

Introduction of the simulation framework, calibration and validation of the DISS Modelica model.

7. Bonilla, J., Yebra, L. J. and Dormido, S. Exploiting OpenMP in the Initial Section of Modelica Models (Work in Progress). In *Proceedings of the 4th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools (EOOLT)*, oral communication, ETH Zurich, 2011. ISSN 1650-3686.

Exploiting OpenMP in the DISS Modelica Model.

VI

Conclusions

9

Contributions, Conclusions and Future Work

One never notices what has been done; one can only see what remains to be done.

Marie Curie

Abstract: The final chapter of this book lists in § 9.1 the main contributions and the principal conclusions reached. Promising ideas for future research are also sketched in § 9.2.

9.1 Contributions and Conclusions

The main contributions and conclusions derived from this work are presented, sorted by topic, in this section.

Two-phase Flow

- *Proposed calculation method for the time derivative of the mean void fraction.* This method was introduced in § 4.4. It assumes a linear distribution of specific enthalpy and uniform pressure in the two-phase flow region. The mean void fraction is calculated by integrating the void fraction. The time derivative of the mean void fraction is calculated by applying the chain rule, where mean void fraction partial derivatives are obtained from symbolic partial derivations of the mean void fraction. This method has been applied in the development of MBMs in Chap. 7.

Finite Volume Models

- *Chattering study.* A chattering study of finite volume homogeneous two-phase flow dynamic models was performed in § 6.6. It was illustrated, through a straightforward example, that discontinuities in the variables of the model which influence state variables (e.g. thermodynamic properties, heat transfer and friction factor coefficients) might cause chattering in dynamic simulations (cf. § 6.6.1). The general solutions applied to avoid the chattering problem have been analyzed (cf. § 6.6.4), and some particular implementations of these solutions have been performed: cubic splines (cf. Appx. C), *Mean Densities* (cf. § 6.7.1),the *Heuristic* approach (cf. § 6.7.2) and MBMs (cf. Chap. 7).

- *Cubic splines.* This attempt was unsuccessful, as the results obtained were inaccurate and oscillatory (cf. Appx. C.3). However, the attempt was important due gaining an understanding of the chattering problem, not about the applied method itself. This attempt was unsuccessful for the kind of sampling grid chosen, since it seems to be solved in Gräber et al. (2012); Wang et al. (2012) (cf. Appx. C.4).

- *Mean Densities.* One solution used to avoid chattering consists in modifying the model variables so that they are always continuous (cf. § 6.6.4). Considering the particular case of the two-phase flow, thermodynamic properties with discontinuities (which influence state variables) at phase changes are density and its partial derivatives with respect to pressure and specific enthalpy (cf. § 6.6.2). *Mean Densities* modifies these thermodynamic properties so that they are continuous and hence avoid chattering (cf. § 6.7.1). This approach is based on the separate integration of density in each flow region, where uniform pressure and linear specific enthalpy distribution in each CV are assumed (cf. § 6.7.1.1), therefore leading to a continuous density calculation (cf. § 6.7.1.3). Mean density partial derivatives are calculated symbolically from the mean density expressions (cf. § 6.7.1.4 and § 6.7.1.5).

- *Heuristic approach.* Another solution, which reduces chattering, consists in minimising the number of phase changes (cf. § 6.6.4). This is the main idea behind the *Heuristic* approach (cf. § 6.7.2), in which a chattering detector tries to identify chattering time instants by imposing lower bounds in the amount of time elapsed (temporal restriction) or a change in a certain variable (spatial restriction) between successive phase changes (cf. § 6.7.2.1). Once a chattering time instant has been identified, an action is taken to lead the system away from chattering. The action involves increasing or decreasing a certain magnitude for one of the state variables, in order to lead the state variables away from the attraction of the phase-change curves (cf. § 6.7.2.2).

- *Mean Densities vs. the Heuristic approach.* Both approaches were compared with respect to simulation speed and accuracy (cf. § 6.9). The comparison results are summarized in Tab. 6.9. The main conclusions are the following.

  When there is no chattering during simulation, both approaches require more computational time because they both involve extra code.

  The *Heuristic* approach is the fastest when there is no chattering, and it also has the lowest amount of errors. When there is chattering, the reduction in simulation time is considerable. On the other hand, it is a *Heuristic* approach – and chattering is detected by a heuristic rule.

  Mean Densities is the fastest method available when there is chattering in the original model. However, when there is no chattering it is slower than the *Heuristic* approach. Moreover, the errors committed, especially in relation to density, are high under certain circumstances. This approach is based on the separate integration of the mean density in each region and does not use a heuristic rule. Chattering, caused by thermodynamic properties, cannot occur with this approach, because there are no state events at phase changes.

  It is important to note that the *Heuristic* approach can deal with chattering at phase changes caused by discontinuities in any variable (e.g. the heat transfer coefficient), because this approach is not limited to discontinuities in thermodynamic properties, unlike *Mean Densities*. Although the *Heuristic* approach is based on a heuristic rule employed to detect and avoid chattering, the simulation results show that it is a good solution due to its ability to avoid chattering, its high accuracy and its good performance.

Moving Boundary Models

- *Switching moving boundary mathematical models for two-phase flow evaporators and condensers.* A study of the state-of-the-art on two-phase flow moving boundary evaporator and condenser models reveals that none of the existing models supports dynamic switching between all possible flow configurations (cf. § 7.2). With this in mind, a new mathematical formulation for two-phase flow moving boundary heat exchanger models was developed (cf. § 7.4). In order to develop low-order models, but still to reflect the principal dynamics, a number of assumptions were considered (cf. § 7.3), e.g. one-dimensional case, homogeneous two-phase flow, uniform pressure, linear specific enthalpy distribution in each CV, etc. Each flow region (CV) was modeled from first principles, considering the mass and energy balances (cf. § 7.7). HE models were developed by interconnecting CVs (cf. § 7.8), adding specific enthalpy constraints between the CVs and switching support. The developed MBMs support dynamic switching between all possible flow configurations in HEs (cf. § 7.10). Furthermore, all of the developed MBMs were thoroughly tested in steady-state and transient integrity and stability tests (cf. § 7.13).

Equation-based Object-oriented Modeling of Dynamic Thermofluid Systems

- *Equation-based object-oriented solar thermal components.* New Modelica components have been created considering the EOO modeling paradigm, and previous components have been updated using new interfaces. All these Modelica components can be parametrized and are intended to be used for future developments, which contributes to facilitating the maintenance and reuse of the models. Some of these components are: solar vector (cf. § 5.1), PTC (cf. § 5.2), PTCN (cf. § 5.2.1.1), absorber tube (cf. § 5.2.1.2), optical losses (cf. § 5.2.2.1), thermal losses (cf. § 5.2.2.2), geometrical losses (cf. § 5.2.2.3), IAM (cf. § 5.2.2.4) and evaporator (cf. § 5.3) components.

- *DISS facility models.* Several DISS facility models, considering different configurations of finite volume and moving boundary evaporators, were analyzed (cf. § 8.2) and their advantages and disadvantages compared (cf. § 8.2.6). Some of these models were also implemented. A comparison of the accuracy and performance between these models in simulation against experimental data was performed (cf. § 8.8).

Simulation, Calibration and Validation

- *DISS facility simulation framework.* The DISS facility simulation framework facilitates the set up of simulations dealing with obtaining inputs from experimental data, setting parameters and initial conditions (cf. § 8.1).

- *Parallelization of the initial section in the DISS model.* OpenMP (cf. § 8.3.1) was used to parallelize the initial section in the resulting C code obtained from the DISS Modelica models by the Dymola tool (cf. § 8.3.2). A gain in speed in the calibration and simulation processes was obtained (cf. § 8.3.3).

- *Specific enthalpy distribution in the evaporator.* The specific enthalpy distribution in the evaporator was studied by considering simulations of finite volumes models (and therefore considering first principles) with experimental inputs (cf. § 8.4). A linear specific enthalpy distribution can be assumed in each MBM region. With this configuration, MBMs can reproduce specific enthalpy distribution in the evaporator, which is obtained through finite volume models in steady-state and transient predictions (cf. Fig. 8.10).

- *Level of discretization in the DISS Modelica Fluid model.* The level of discretization in the DISS Modelica *Fluid* model was studied and analyzed with experimental inputs (cf. § 8.7). The level of discretization was selected by considering a compromise between accuracy and simulation performance (five CVs per PTC).

- *Calibration and validation of the DISS models.* A framework for calibration using genetic algorithms was developed (cf. § 8.5), while some unknown parameters were calibrated and validated globally (cf. § 8.5.3). On the other hand, peak optical efficiencies must be calibrated, particularly for each experiment, because the DISS Modelica model is highly sensitive to them (cf. § 8.5.4).

- *Simulations of DISS models.* The comparison between the developed DISS models in terms of accuracy and performance against experimental data revealed the following conclusions (cf. § 8.8).

  All the models exhibit suitable agreement with experimental data. Finite volume DISS models can achieve more accuracy when increasing the number of CV, but this is at the expense of simulation time. However, moving boundary DISS models are accurate while fast.

  Finite volume DISS models require more computational time than moving boundary DISS models, because the number of CVs is higher. On the other hand, this number of CVs is required to obtain similar accuracy as the moving boundary DISS models. The low number of CVs makes the moving boundary DISS models fast.

  All the developed DISS Modelica models composed of finite volume and moving boundary evaporator models were exposed to a wide range of different operating conditions and disturbances, proving that they can take full advantage of dynamic simulations and can be used for the design, testing and validation of advanced ACSs, by implementing a method to avoid chattering, if required.

9.2 Future Work

Promising ideas for future research and developments are sorted by topic as follows.

Modeling the DISS facility

- *Study unmodeled delays in the DISS model.* This improvement is proposed as the first future task in order to perform more accurate dynamic simulations.

- *Study temperature overestimation in the DISS model.* The model commits a considerable error when the 11^{th} PTC injector is active, as shown in § 8.8.4. This must be studied and analyzed in order to improve the accuracy of the model.

- *Model the injection and recirculation operating modes.* Only the once-through operating mode was considered in the DISS models. Injection and recirculation modes should be also considered in order to compare results under different configurations. Additionally, the recirculation mode was established as the best financial and technical option for the application of DSG PTC technology in commercial solar thermal power plants (Zarza et al., 2004). However, the once-trough operating mode has acquired relevance recently with the DUKE project (Feldhoff, 2012), which is attempting to demonstrate the once-trough mode on an industrial scale.

- *Model the solar tracking system of the PTCs.* This could help to identify tracking problems during experimentation. Tracking information during experimentation from the DAS of the DISS facility is available. Additionally, the influence of tracking problems in the outlet temperature could be studied in dynamic simulations.

- *Consider other possible models for the DISS facility.* From the analyzed models in § 8.2.5, there are two configurations which might be interesting for future research, namely the FV-MB and MBoFV models.

  Finite volume and moving boundary DISS Modelica model (FV-MB) (cf. § 8.2.5). Inspecting the experimental data from the DISS facility, the PTCs in which phase transitions occur can be identified. One solution for avoiding chattering involves using moving boundary evaporator models in PTCs where phase transitions occur, and finite volume evaporator models in the remaining PTCs. This model is highly customizable and it is a chattering-free model.

  Moving boundary of finite volumes DISS Modelica model (MBoFV) (cf. § 8.2.5). This idea involves dividing the evaporator into regions by applying the moving boundary formulation. Each region is then discretized in CVs by considering the FVM. The most remarkable advantages of this model are: the speed-accuracy trade-off can be easily configured, the model is highly customizable and it is a chattering-free model.

Finite Volume Models

- *Implement Mean Densities and the Heuristic approach in the Modelica Fluid library.* One of these approaches must be implemented in order to avoid chattering, when using the *Modelica Fluid* library in dynamic simulations.

- *Steady-state initialization of Modelica Fluid models.* All the *Modelica Fluid* models were initialized with dynamic balances for the mass and energy equations and steady-state for the momentum equation (default configuration), because it is tough to initialize them in a steady-state. It would therefore be worth studying the steady-state initialization.

- *Implement HTCs and FFCs in the Modelica Fluid library*, in order to estimate the heat transfer and friction factor coefficients in future developments.

Moving Boundary Models

- *Pressure loss.* Pressure loss, which has been neglected in MBMs, involves including the momentum balance equation in the mathematical formulation. The energy balance equation should also be modified to consider viscous stresses. It would also be worth comparing results in terms of speed and accuracy between MBMs with uniform and non-uniform pressure distribution in the evaporator.

- *Ports for the CV length and time derivative of the inlet specific enthalpy.* In order to design components for applying a strictly OO modeling approach, each variable must be shared between models, but only through ports. However, CV length is shared between the fluid and the pipe wall models by a class (Geometry class) (cf. § 7.12.3.3). The time derivative of the inlet specific enthalpy, if required, can be specified by code (cf. § 7.12.8), and this should be supported by the models through the GUI of the M&S tool.

- *Zero mass flow rate and flow reversal.* The developed MBMs are singular when the mass flow rate is zero. The support of a zero mass flow rate and flow reversal would be interesting contributions to the *MBMs* library.

- *More options for initialization.* The initialization of MBMs has been assumed in steady-state. However, new options could be added, e.g. by considering dynamic initialization and different sets of initial conditions. These new options should be supported by the models through the GUI of the M&S tool in order to help the modeler in this task.

Calibration and Validation

- *Parallelization in the calibration process*. The calibration process may be parallelized in order to take advantage of a 12-node cluster located at Centro de Investigación de Energía SOLar (CIESOL) in the UAL. This may significantly reduce calibration time.

- *Validation of the switching moving boundary evaporator model.* Although the switching moving boundary evaporator model was validated against experimental data from the DISS facility, it is planned to be used and validated in the modeling of a DSG LF in the ambit of the *HIBIOSOLEO* project - IPT-440000-2010-004 (cf. § 1.5).

- *Validation of the switching moving boundary condenser model.* It is planned to use and validate the switching moving boundary condenser model in the modeling of a double-effect absorption heat pump in the ambit of the *POWER* project - DPI2010-21589-C05-02 (cf. § 1.5).

Heat Transfer and Friction Factor Correlations

- *Multidimensional interpolation methods for HTCs and FFCs.* In order to use HTCs and FFCs in dynamic two-phase flow models, therefore avoiding chattering and numerical problems at phase changes, multidimensional interpolation methods could be applied to construct smooth surfaces. Previous experience (cf. Appx. C) and recent developments (Wang et al., 2012; Gräber et al., 2012) may be applied for this task.

VII

Appendices

A
Mathematical Background

Abstract: This appendix gathers, for the convenience of the reader, some mathematical theorems and rules used in this book.

A.1 Leibniz Integral Rule

Leibniz's rule for differentiating integrals with time varying limits is defined by Eq. A.1 (Woods, 1926), particularized for a time differential and a longitudinal integral (Todreas and Kazimi, 1993).

$$\frac{d}{dt}\int_{z_a}^{z_b} f(z,t)\,dz = \int_{z_a}^{z_b} \frac{\partial f(z,t)}{\partial t} dz + f(z_b,t)\frac{dz_b}{dt} - f(z_a,t)\frac{dz_a}{dt}. \qquad \text{(A.1)}$$

It should be noted that for this rule to be applicable, $f(z,t)$, z_a and z_b must be continuously differentiable with respect to t. Additionally, $f(z,t)$ and $\partial f(z,t)/\partial t$ should be continuous with respect to z in the $[z_a, z_b]$ interval.

A.2 Mean-value Theorem

The mean-value theorem states that if a function, $f(z,t)$, is continuous in the $[z_a, z_b]$ interval, then $\exists\, z_m \in [z_a, z_b]$, which satisfies Eq. A.2 (Piskunov, 1965).

$$\int_{z_a}^{z_b} f(z,t) = f(z_m,t)(z_b - z_a). \qquad \text{(A.2)}$$

A.3 Chain Rule

Considering a function, $F = f(z_1, z_2, \cdots, z_n)$, the chain rule states that its time derivative can be calculated by Eq. A.3.

$$\frac{dF}{dt} = \frac{\partial F}{\partial z_1}\frac{dz_1}{dt} + \frac{\partial F}{\partial z_2}\frac{dz_2}{dt} + \cdots + \frac{\partial F}{\partial z_n}\frac{dz_n}{dt} = \sum_{i=1}^{n} \frac{\partial F}{\partial z_i}\frac{dz_i}{dt}. \tag{A.3}$$

A.4 Product Rule for Derivatives

For the F function, a product of multiple functions f_i as shown in Eq. A.4, where each f_i function depends on a variable z, $f_i = f_i(z)$, the derivative of F is defined by Eq. A.5.

$$F = f_1 f_2 \cdots f_n = \prod_{i=1}^{n} f_i, \tag{A.4}$$

$$\frac{dF}{dz} = \frac{df_1}{dz} f_2 \cdots f_n + f_1 \frac{df_2}{dz} \cdots f_n + \cdots + f_1 f_2 \cdots \frac{df_n}{dz} = \sum_{i=1}^{n} \left(\frac{df_i}{dz} \prod_{\substack{j=1 \\ j \neq i}}^{n} f_j \right). \tag{A.5}$$

B
Partial Derivatives of the Mean Void Fraction

Abstract: The equations applied to the partial derivative of the mean void fraction with respect to pressure, inlet and outlet specific enthalpies are shown in this appendix. The Γ_1 and Γ_2 terms are calculated from Eqs. 4.30 and 4.31, respectively.

B.1 Inl. Spec. Enthalpy Part. Der. of Mean Void Fraction

The $\dfrac{\partial\bar{\gamma}}{\partial h_a}$ obtained symbolically from Eq. 4.29 is shown in Eq. B.1.

$$\frac{\partial\bar{\gamma}}{\partial h_a} = \frac{\rho'}{\left(h_a - h_b\right)^2\left(\rho' - \rho''\right)^2}\left(\left(h_b - h_a\right)\rho' + \frac{\left(h_a - h_b\right)\left(\rho' - \rho''\right)^2\left(h_a - h'\right)}{\rho'\left(h_a - h'\right) + \rho''\left(h'' - h_a\right)}\right.$$
$$\left. -\rho''\left(h_b - h_a + \left(h' - h''\right)\ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\right)\right). \tag{B.1}$$

B.2 Outl. Spec. Enthalpy Part. Der. of Mean Void Fraction

The $\dfrac{\partial\bar{\gamma}}{\partial h_b}$ obtained symbolically from Eq. 4.29 is shown in Eq. B.2.

$$\frac{\partial\bar{\gamma}}{\partial h_b} = \frac{\rho'}{\left(h_a - h_b\right)^2\left(\rho' - \rho''\right)^2}\left(\left(h_a - h_b\right)\rho' - \frac{\left(h_a - h_b\right)\left(\rho' - \rho''\right)^2\left(h_b - h'\right)}{\rho'\left(h_b - h'\right) + \rho''\left(h'' - h_b\right)}\right.$$
$$\left. +\rho''\left(h_b - h_a + \left(h' - h''\right)\ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\right)\right). \tag{B.2}$$

B.3 Pressure Partial Derivative of the Mean Void Fraction

The $\dfrac{\partial\bar{\gamma}}{\partial p}$ obtained symbolically from Eq. 4.29 is shown in Eq. B.3.

$$
\begin{aligned}
\frac{\partial\bar{\gamma}}{\partial p} = {} & \Biggl((\rho' - \rho'') \Biggl((h_a - h_b)\rho' + \rho'' \Biggl(h_b - h_a + (h' - h'') \ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\Biggr)\Biggr) \frac{\partial \rho'}{\partial p} \\
& - 2\rho' \Biggl((h_a - h_b)\rho' + \rho'' \Biggl(h_b - h_a + (h' - h'') \ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\Biggr)\Biggr) \left(\frac{\partial \rho'}{\partial p} - \frac{\partial \rho''}{\partial p} \right) \\
& + \rho'(\rho' - \rho'') \Biggl((h_a - h_b)\frac{\partial \rho'}{\partial p} + \Biggl(h_b - h_a + (h' - h'') \ln\left(\frac{\Gamma_1}{\Gamma_2}\right)\Biggr) \frac{\partial \rho''}{\partial p} \\
& + \rho'' \Biggl(\ln\left(\frac{\Gamma_1}{\Gamma_2}\right) \left(\frac{\partial h'}{\partial p} - \frac{\partial h''}{\partial p} \right) + \Biggl((h_a - h_b)(h' - h'') \Biggl(-\rho' \Biggl((h' - h'')\frac{\partial \rho''}{\partial p} \\
& + \quad \rho' \frac{\partial h'}{\partial p}\Biggr) - \rho''^2 \frac{\partial h''}{\partial p} + \rho'' \Biggl((h' - h'')\frac{\partial \rho'}{\partial p} + \rho' \left(\frac{\partial h'}{\partial p} + \frac{\partial h''}{\partial p} \right)\Biggr)\Biggr)\Biggr) \frac{-1}{\Gamma_1\Gamma_2}\Biggr)\Biggr)\Biggr) \\
& \frac{1}{(h_a - h_b)(\rho' - \rho'')^3}.
\end{aligned}
\tag{B.3}
$$

C

Cubic Splines

Knowledge rests not upon truth alone, but upon error also.

Carl Gustav Jung

Abstract: This appendix details the unsuccessful attempt to use cubic splines to avoid the chattering problem. This attempt is important because it helps to gain understanding about the chattering problem, not about the applied method itself.

C.1 Introduction

One chattering solution is based on the idea of constructing continuous surfaces for thermodynamic properties with discontinuities by using cubic splines, as mentioned in § 6.6.4. The thermodynamic properties that produce chattering (which have discontinuities) are density and its partial derivatives. This was discussed in § 6.6.2.

Cubic splines employed to interpolate these thermodynamic properties must be two-dimensional, due to the fact that they depend on two variables, namely specific enthalpy (h) and pressure (p), which are commonly state variables when modeling thermal-hydraulic systems, as described in § 4.3.3.1.

It is important to note that for constructing a continuous surface it is necessary to sample the entire thermodynamic property, and not only the discontinuous area. This is required because it is unlikely to assure a continuous transition between two surfaces if one of them has been sampled.

C.2 Hermite Bicubic Splines

With the aim of developing continuous thermodynamic properties, a well-documented interpolation method was used. The Hermite bicubic spline interpolation method (Gary, 1999) was chosen for several reasons. First of all, a two-dimensional interpolation method is required. Moreover, Hermite bicubic splines provide a smooth surface via a parametric curve using cubic polynomials. The slopes at each corner of the surface can also be specified using this interpolation method.

An interpolated surface is the union of several patches. The data required to construct a Hermite bicubic patch is as follows: the four corner points of the patch, the x-slope, the y-slope and twist vectors for each corner point. Four corner points $(P_{00}, P_{01}, P_{10}, P_{11})$ and the corresponding bicubic patch are shown in Fig. C.1.

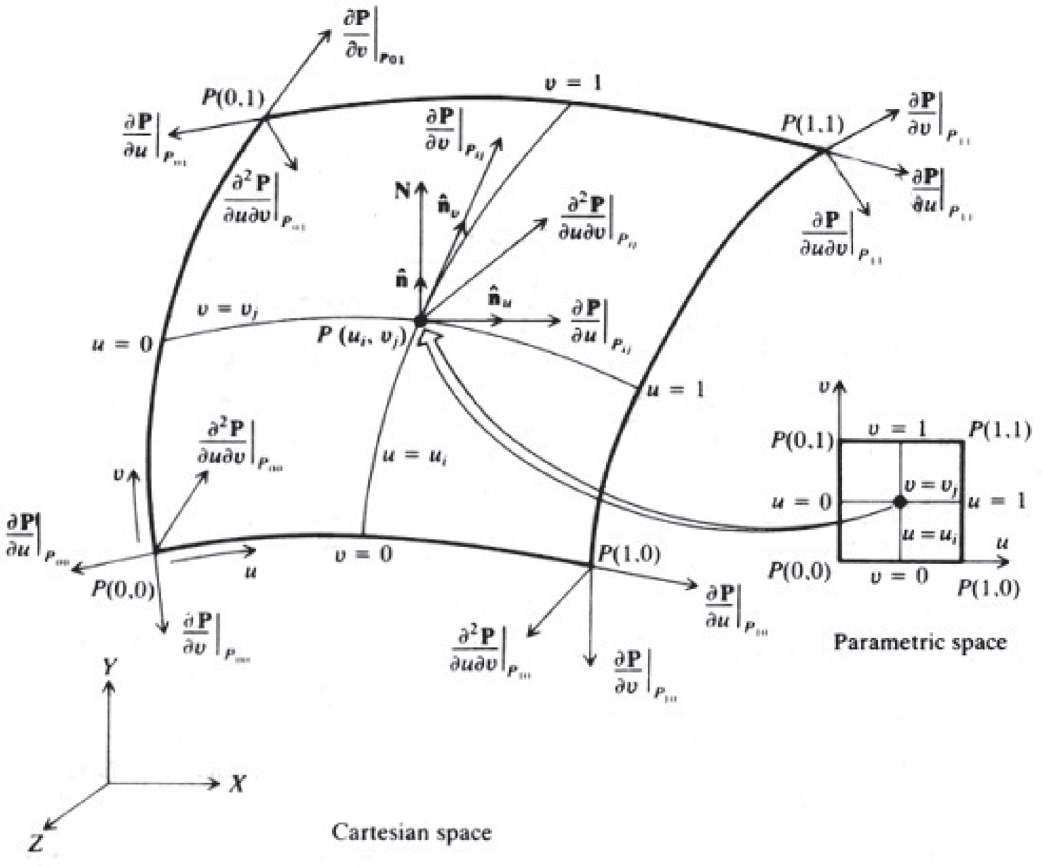


Figure C.1: Bicubic patch and its boundary conditions (Hsu, 2010)

Using the previously mentioned data in Eq. C.1, $P(u, w)$ is obtained, which is the Hermite bicubic patch, where P_{ij} is a corner point, P_{ij}^{u} is an x-slope, P_{ij}^{w} is a y-slope and P_{ij}^{uw} is a twist vector, $\forall i, j \in [0, 1]$.

$$P(u, w) = UHBH^{T}W^{T}, \tag{C.1}$$

$$U(u) = (u^{3}, u^{2}, u, 1), \tag{C.2}$$

$$B = \begin{pmatrix} P_{00} & P_{01} & P^{w}_{00} & P^{w}_{01} \\ P_{10} & P_{11} & P^{w}_{10} & P^{w}_{11} \\ P^{u}_{00} & P^{u}_{01} & P^{uw}_{00} & P^{uw}_{01} \\ P^{u}_{00} & P^{u}_{01} & P^{uw}_{10} & P^{uw}_{11} \end{pmatrix}, \tag{C.3}$$

$$H = \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \tag{C.4}$$

$$W(w) = (w^3, w^2, w, 1), \tag{C.5}$$

C.2.1 Sampling Grid

With the four corner points, x-slope, y-slope and twist vector for each corner point, continuous patches with a continuous union between adjacent patches can be constructed. Fig. C.2 shows different patches (rectangles in dashed lines), each of which is the union of four points (corner points) forming a continuous surface which avoids the discontinuous area at the saturation curves (in this case h'), thus maintaining a certain distance in the single-phase region. The thermodynamic properties were sampled using this grid in the interval $p \in [1, 100]$ bar and $h \in [1 \cdot 10^5, 100 \cdot 10^5]$ J/kg. For the sake of illustration, Fig. C.2 shows only a region and not the whole range. Furthermore, Fig. C.2 shows a low-density sampling grid (in order to see the patch distribution clearly). The interpolated surfaces are composed of 9,600 patches. It is important to note that the discontinuous area has to be avoided, as shown in Fig. C.2. Hence, the sample points cannot be in the area between the h' curve and a certain distance in the single-phase region.

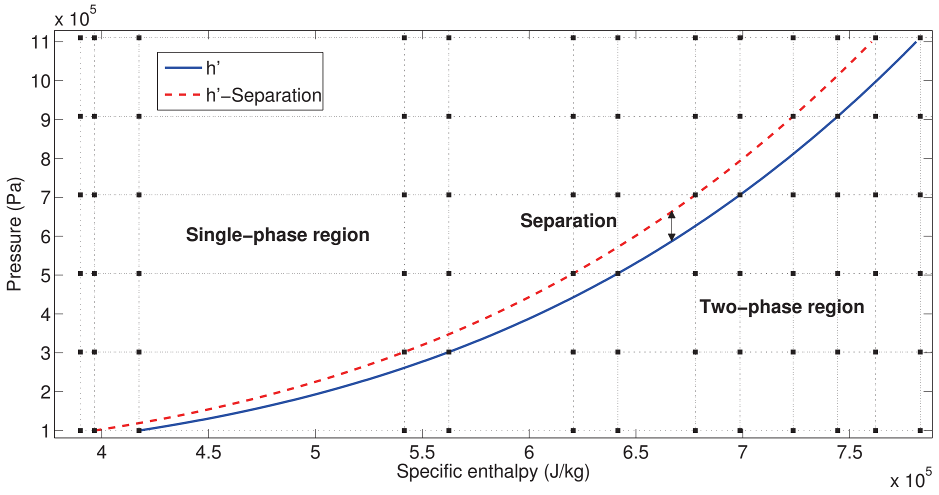
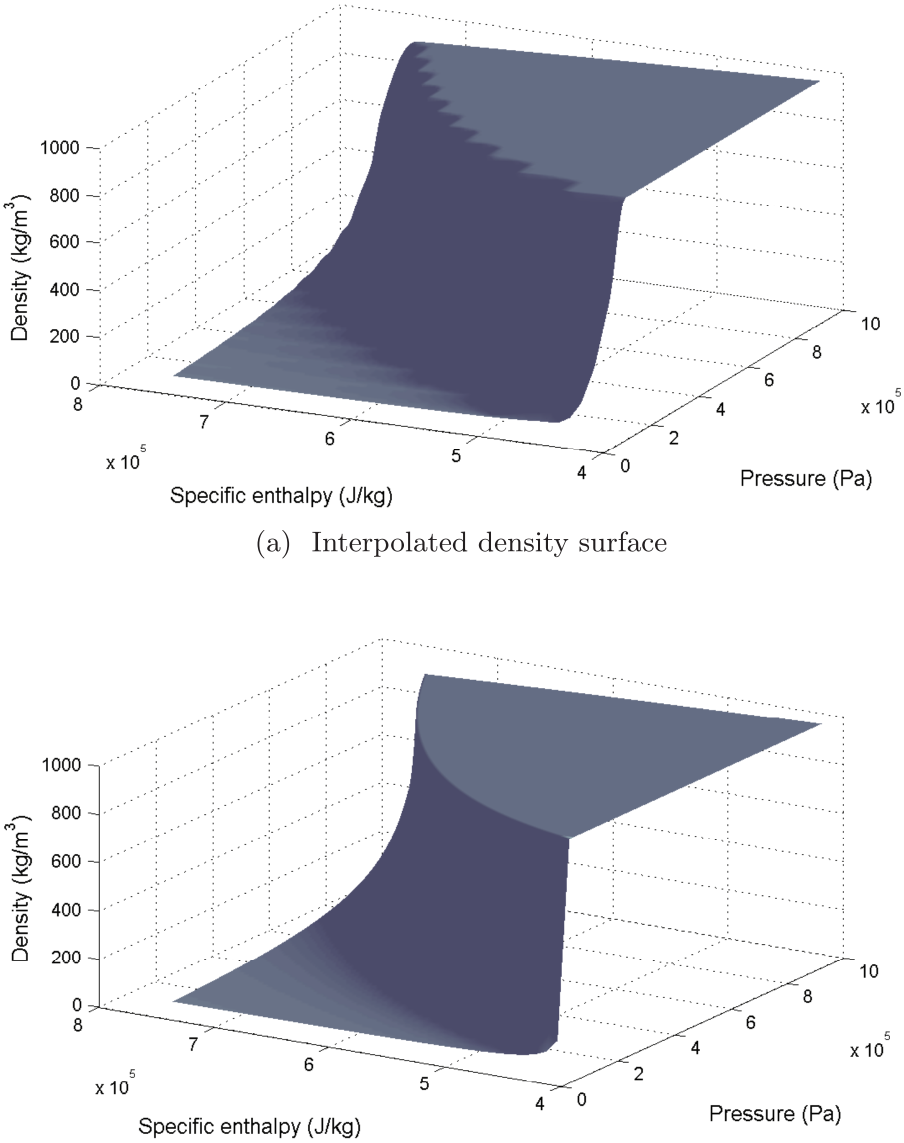


Figure C.2: Sampling grid

C.2.2 Interpolated Surfaces

Density, $\frac{\partial\rho}{\partial h}$ and $\frac{\partial\rho}{\partial p}$ were sampled using the sampling grid described in Appx. C.2.1, following which interpolated surfaces were constructed using Hermite bicubic splines. Fig. C.3a shows the interpolated density surface in the h' curve, in which different patches can be seen, especially in the discontinuous area. Fig. C.3b shows the original density surface.



(a) Interpolated density surface

(b) Original density surface

Figure C.3: Original and interpolated density surfaces

C.3 Simulation Results

The simulation results were not as good as expected. Figs. C.4 and C.5 show a comparison between the original model and the interpolated model in the

8^{th} and 9^{th} CV, respectively, considering the example model introduced in § 6.6.1. The solid line represents the original model and the dashed line the interpolated model. The results are inaccurate and oscillatory. There are large PREs, as shown in Tab. C.1, in each CV. Eq. 6.71 defines the PRE, where μ_o is density in the original model and μ_n is density in the interpolated model.

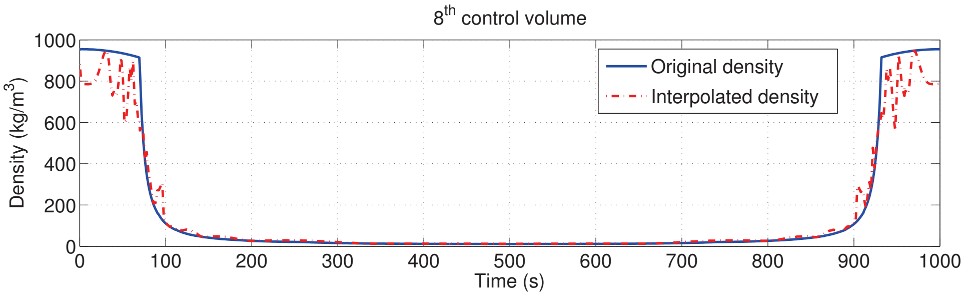


Figure C.4: Comparison between original and interpolated density, 8^{th} CV

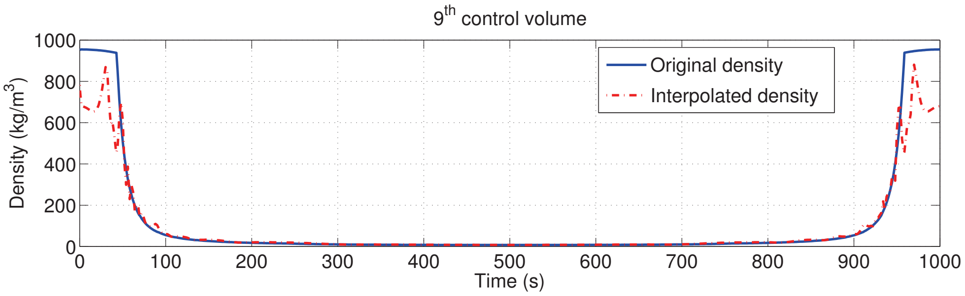


Figure C.5: Comparison between original and interpolated density, 9^{th} CV

CV	1	2	3	4	5	6	7	8	9	10
$\epsilon_r(\rho)$	1.7%	31%	25%	50%	49%	50%	46%	41%	50%	32%

Table C.1: Maximum density PRE in each CV using the interpolated model

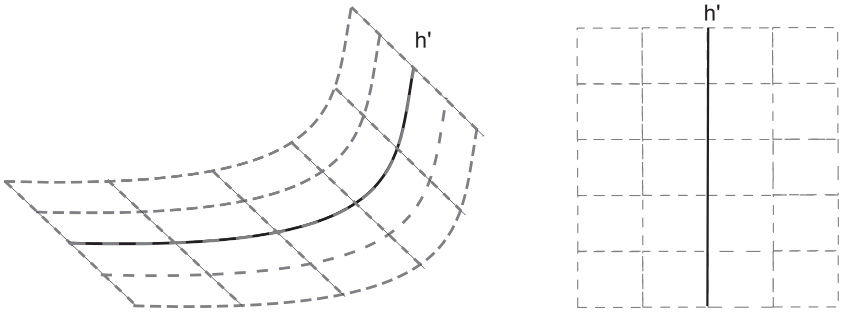
However, as can be seen in Tab. C.2, there is no chattering when using the interpolated model. The number of state events has decreased from 9,281 to 171, and as a consequence simulation time has also decreased by up to 40%, from 16.6 to 9.94 s. These results are encouraging because it means that continuous density and partial derivative thermodynamic properties wipe out chattering in the example model.

Metric	Original model	Interpolated model
Simulated time (s)	$1,000$	$1,000$
CPU-time (s)	16.6	9.94
State events	$9,281$	171
Min. integration stepsize	$5.45 \cdot 10^{-7}$	$5.07 \cdot 10^{-7}$
Max. integration stepsize	23.2	25.3
Max. integration order	5	5
Chattering	Yes	No

Table C.2: Original and interpolated models' simulation statistics

C.4 Summary and Conclusions

This appendix presents an attempt to use cubic splines in order to develop continuous surfaces for thermodynamic properties with discontinuities, which might cause chattering. The results obtained are inaccurate and oscillatory. Increasing the number of patches certainly leads to a more accurate surface but it also results in more oscillations, due to the nature of cubic splines, which are cubic polynomials. More patches mean more computational time to find the proper patch (even using quick sorting algorithms) in each evaluation of the thermodynamic properties for each CV. Furthermore, reducing the number of patches indeed leads to a faster model with less oscillations, but it also reduces accuracy. The main problem associated with this attempt is related to the interpolation grid not being a rectangular equidistant grid in order to not to sample in the discontinuous area. This causes computational time required to find a patch to increase considerably when increasing the number of patches. On the other hand, finding a particular patch is instantaneous in an equidistant grid. This problem seems to be solved in Gräber et al. (2012) and Wang et al. (2012), who remove the splines from the saturation curve, which solves oscillations in the solution, maintain an equidistant grid (cf. Fig. C.6a) and transform the grid coordinates to be rectangular (cf. Fig. C.6b) in order to find patches fast.



(a) Removing splines from h' (b) Rectangular grid

Figure C.6: Rectangular equidistant sampling grid

D

Pressure Partial Derivatives of Mean Densities

Abstract: Expressions for the density partial pressure derivatives of *Mean Densities* in each region are presented in this appendix. The symbolic integrations were done using software which can deal with symbolic manipulation and calculation, namely *Mathematica* (Wolfram, 2010).

D.1 Regions 1 and 5: Single-phase

The partial derivative of Eq. 6.53, with respect to pressure keeping specific enthalpy constant, is Eq. D.1.

$$\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{2}\left(\left.\frac{\partial \rho_{z_j}}{\partial p}\right|_h + \left.\frac{\partial \rho_{z_{j+2}}}{\partial p}\right|_h \right). \tag{D.1}$$

D.2 Region 2: Two-phase flow / Two-phase flow

The partial derivative of Eq. 6.60, with respect to pressure keeping specific enthalpy constant, is Eq. D.2.

$$\begin{aligned}\frac{\partial \bar{\rho}_j}{\partial p} = {} & \frac{1}{(h_{z_{j+1}} - h_{z_j})(\nu' - \nu'')^2} \left(\ln\left(\frac{\rho_{z_j}}{\rho_{z_{j+1}}}\right)(\nu'' - \nu')\left(\frac{\partial h''}{\partial p} - \frac{\partial h'}{\partial p}\right) \right. \\ & + \frac{(h' - h'')(\rho' - \rho'')\left(-\rho_{z_{j+1}}\frac{\partial \rho_{z_j}}{\partial p} + \rho_{z_{j+1}}\frac{\partial \rho_{z_{j+1}}}{\partial p}\right)}{\rho_{z_j}\rho_{z_{j+1}}\rho'\rho''} \\ & \left. - \ln\left(\frac{\rho_{z_j}}{\rho_{z_{j+1}}}\right)(h'' - h')\left(\frac{\frac{\partial \rho'}{\partial p}}{\rho'^2} - \frac{\frac{\partial \rho''}{\partial p}}{\rho''^2}\right) \right).\end{aligned} \tag{D.2}$$

D.3 Region 3: Sub-cooled liquid / Two-phase flow

The partial derivative of Eq. 6.63, with respect to pressure keeping specific enthalpy constant, is Eq. D.3.

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{h_{z_{j+1}} - h_{z_j}} \Bigg(& \frac{1}{2}(\rho_{z_j} + \rho')\frac{\partial h'}{\partial p} + \frac{\ln\left(\frac{\rho'}{\rho_{z_j}}\right)\left(-\frac{\partial h'}{\partial p} + \frac{\partial h''}{\partial p}\right)}{\nu'' - \nu'} \\
& + \frac{1}{2}(h' - h_{z_j})\left(\frac{\partial \rho_{z_j}}{\partial p} + \frac{\partial \rho'}{\partial p}\right) + \frac{(h' - h'')\rho''\left(\rho'\frac{\partial \rho_{z_{j+1}}}{\partial p} - \rho_{z_{j+1}}\frac{\partial \rho'}{\partial p}\right)}{\rho_{z_{j+1}}(\rho' - \rho'')} \\
& + \frac{\ln\left(\frac{\rho'}{\rho_{z_{j+1}}}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2} \Bigg) .
\end{aligned}
\tag{D.3}
$$

D.4 Region 4: Two-phase flow / Superheated steam

The partial derivative of Eq. 6.64, with respect to pressure keeping specific enthalpy constant, is Eq. D.4.

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{h_{z_{j+1}} - h_{z_j}} \Bigg(& -\frac{1}{2}(\rho_{z_{j+1}} + \rho'')\frac{\partial h''}{\partial p} + \frac{\ln\left(\frac{\rho_{z_j}}{\rho''}\right)\left(\frac{\partial h''}{\partial p} - \frac{\partial h'}{\partial p}\right)}{\nu'' - \nu'} \\
& + \frac{(h_{z_{j+1}} - h'')\left(\frac{\partial \rho_{z_{j+1}}}{\partial p} + \frac{\partial \rho''}{\partial p}\right)}{2} + \frac{(h' - h'')\rho'\left(\rho_{z_j}\frac{\partial \rho''}{\partial p} - \rho''\frac{\partial \rho_{z_j}}{\partial p}\right)}{\rho_{z_j}(\rho' - \rho'')} \\
& + \frac{\ln\left(\frac{\rho_{z_j}}{\rho''}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2} \Bigg) .
\end{aligned}
\tag{D.4}
$$

D.5 Region 6: Sub-cooled liquid / Superheated steam

The partial derivative of Eq. 6.65, with respect to pressure keeping specific enthalpy constant, is Eq. D.5.

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{2(h_{z_{j+1}} - h_{z_j})} \Bigg(& (\rho_{z_j} + \rho')\frac{\partial h'}{\partial p} - (\rho_{z_{j+1}} + \rho'')\frac{\partial h''}{\partial p} \\
& + \frac{2\ln\left(\frac{\rho'}{\rho''}\right)\left(\frac{\partial h'}{\partial p} + \frac{\partial h''}{\partial p}\right)}{\nu'' - \nu'} + (h' - h_{z_j})\left(\frac{\partial \rho_{z_j}}{\partial p} + \frac{\partial \rho'}{\partial p}\right) \\
& + (h_{z_{j+1}} - h'')\left(\frac{\partial \rho_{z_{j+1}}}{\partial p} + \frac{\partial \rho''}{\partial p}\right) + \frac{2(h' - h'')\left(\rho'\frac{\partial \rho''}{\partial p} - \rho''\frac{\partial \rho'}{\partial p}\right)}{\rho' - \rho''} \\
& + \frac{2\ln\left(\frac{\rho'}{\rho''}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2} \Bigg) .
\end{aligned}
\tag{D.5}
$$

D.6 Region 7: Superheated steam / Two-phase flow

The partial derivative of Eq. 6.66, with respect to pressure keeping specific enthalpy constant, is Eq. D.6.

$$
\begin{aligned}
\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{h_{z_{j+1}} - h_{z_j}} \Bigg(& \frac{1}{2}(\rho_{z_j} + \rho'')\frac{\partial h''}{\partial p} + \frac{\ln\left(\frac{\rho''}{\rho_{z_{j+1}}}\right)\left(\frac{\partial h''}{\partial p} - \frac{\partial h'}{\partial p}\right)}{\nu'' - \nu'} \\
& + \frac{(h'' - h_{z_j})\left(\frac{\partial \rho_{z_j}}{\partial p} + \frac{\partial \rho''}{\partial p}\right)}{2} + \frac{(h' - h'')\rho'\left(\rho''\frac{\partial \rho_{z_{j+1}}}{\partial p} - \rho_{z_{j+1}}\frac{\partial \rho''}{\partial p}\right)}{\rho_{z_{j+1}}(\rho' - \rho'')} \\
& + \frac{\ln\left(\frac{\rho''}{\rho_{z_{j+1}}}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2} \Bigg) .
\end{aligned}
\tag{D.6}
$$

D.7 Region 8: Two-phase flow / Sub-cooled liquid

The partial derivative of Eq. 6.67, with respect to pressure keeping specific enthalpy constant, is Eq. D.7.

$$\begin{aligned}\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{h_{z_{j+1}} - h_{z_j}} \Bigg(& -\frac{1}{2}(\rho_{z_{j+1}} + \rho')\frac{\partial h'}{\partial p} + \frac{\ln\left(\frac{\rho_{z_j}}{\rho'}\right)\left(\frac{\partial h''}{\partial p} - \frac{\partial h'}{\partial p}\right)}{\nu'' - \nu'} \\ & + \frac{1}{2}(h_{z_{j+1}} - h')\left(\frac{\partial \rho_{z_{j+1}}}{\partial p} + \frac{\partial \rho'}{\partial p}\right) + \frac{(h' - h'')\rho''\left(\rho_{z_j}\frac{\partial \rho'}{\partial p} - \rho'\frac{\partial \rho_{z_j}}{\partial p}\right)}{\rho_{z_j}(\rho' - \rho'')} \\ & + \frac{\ln\left(\frac{\rho_{z_j}}{\rho'}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2}\Bigg).\end{aligned} \tag{D.7}$$

D.8 Region 9: Superheated steam / Sub-cooled liquid

The partial derivative of Eq. 6.68, with respect to pressure keeping specific enthalpy constant, is Eq. D.8.

$$\begin{aligned}\frac{\partial \bar{\rho}_j}{\partial p} = \frac{1}{2(h_{z_{j+1}} - h_{z_j})} \Bigg(& (\rho_{z_j} + \rho'')\frac{\partial h''}{\partial p} - (\rho_{z_{j+1}} + \rho')\frac{\partial h'}{\partial p} \\ & + \frac{2\ln\left(\frac{\rho''}{\rho'}\right)\left(\frac{\partial h''}{\partial p} - \frac{\partial h'}{\partial p}\right)}{\nu'' - \nu'} + (h_{z_j} - h')\left(\frac{\partial \rho_{z_{j+1}}}{\partial p} + \frac{\partial \rho'}{\partial p}\right) \\ & + (h'' - h_{z_j})\left(\frac{\partial \rho_{z_j}}{\partial p} + \frac{\partial \rho''}{\partial p}\right) + \frac{2(h' - h'')\left(\rho''\frac{\partial \rho'}{\partial p} - \rho'\frac{\partial \rho''}{\partial p}\right)}{\rho' - \rho''} \\ & + \frac{2\ln\left(\frac{\rho''}{\rho'}\right)(h' - h'')\left(\rho''^2\frac{\partial \rho'}{\partial p} - \rho'^2\frac{\partial \rho''}{\partial p}\right)}{(\rho' - \rho'')^2}\Bigg).\end{aligned} \tag{D.8}$$

E

Moving Boundary Models' Modelica Code

ABSTRACT: This appendix describes some relevant Modelica code for the MBMs developed in this work. The Modelica code presented in this appendix corresponds to the switching criteria between different configurations and the initialization of MBMs.

E.1 Switching Criteria in the General Evaporator Model

The first block of code (lines 8-10) in List. E.1 sets flags (*no_cv*) if any CV disappears, i.e. the CV length is lower that the minimum length (cf. § 7.10.1). The *ltol* parameter is introduced in order to establish a hysteresis and to avoid convergence problems due to numerical errors, if the CV length during the simulation is close to the minimum length.

The second block of code (lines 15-17) sets flags (*new_cv*) if any CV appears, according to the events described in § 7.10.3 for evaporators. The *htol* parameter is introduced in order to establish a relay hysteresis in order to avoid convergence problems (cf. § 7.10.3).

Finally, the third block of code (lines 22-47) manages events that trigger the appearance or disappearance of any CV. The *when* clause checks all the flags. The *for* loop sets whether or not each particular CV in the *wcv* variable exists. Next, the CV state is accordingly set to inactive ($ki = 1$), complete ($kc = 1$) or partial ($kp = 1$).

Inactive CV equations are described in § 7.10.2, while complete and partial CV equations are described in § 7.7. The difference between the partial and complete states is that, in a complete state, the specific enthalpy constraint must be set, according to § 7.8. The partial state is not required for the TP CV because the constraints are set in the one-phase CVs. The stability of the switching criteria was previously tested in § 7.13.2.

The switching criteria Modelica codes for the flooded/dry evaporator and general/flooded/dry condenser models are similar to those described in this section relating to the general evaporator model.

```
parameter SIunits.Length lmin = 1e-4;
parameter SIunits.Length ltol = 1e-4;
parameter SIunits.SpecificEnthalpy htol = 1e-4;

// -----------------------------------
// Condition to make a CV disappear
// -----------------------------------
no_cv[1]= v1.Li<=lmin or (pre(no_cv[1]) and v1.Li<=lmin + ltol);
no_cv[2]= v2.Li<=lmin or (pre(no_cv[2]) and v2.Li<=lmin + ltol);
no_cv[3]= v3.Li<=lmin or (pre(no_cv[3]) and v3.Li<=lmin + ltol);

// -----------------------------------
// Condition to make a CV appear
// -----------------------------------
new_cv[1] = v2.ha + htol < v2.hl;
new_cv[2] = v2.ha - htol > v2.hl and v2.hb - htol < v2.hv;
new_cv[3] = v2.hb + htol > v2.hv;

// -------------------------------------------
// Event: Making a CV appear or disappear
// -------------------------------------------
when {edge(no_cv[1]), no_cv[1] and edge(new_cv[1]),
      edge(no_cv[2]), no_cv[2] and edge(new_cv[2]),
      edge(no_cv[3]), no_cv[3] and edge(new_cv[3])} then

    // Exist the CV
    for i in 1:n loop
      wcv[i] = (edge(new_cv[i]) and no_cv[i]) or cv[i];
    end for;

    // Set mode in CV
    // ki = true --> Inactive
    // kp = true --> Partial (evaporation or condensation)
    // kc = true --> Complete (constraints)
    //
    mode[1].ki = Functions.BoolToInt(not wcv[1]);
    mode[1].kp = Functions.BoolToInt(wcv[1] and not wcv[2]);
    mode[1].kc = Functions.BoolToInt(wcv[1] and wcv[2]);

    mode[2].ki = Functions.BoolToInt(not wcv[2]);
    mode[2].kc = Functions.BoolToInt(wcv[2]);

    mode[3].ki = Functions.BoolToInt(not wcv[3]);
    mode[3].kp = Functions.BoolToInt(wcv[3] and not wcv[2]);
    mode[3].kc = Functions.BoolToInt(wcv[3] and wcv[2]);

  end when;
```

Listing E.1: Switching Modelica code for the moving boundary general evaporator model

E.2 Initialization in the General Evaporator Model

The initialization Modelica code for a moving boundary general evaporator model is shown in List. E.2. The three first blocks of code (lines 6-14, 18-24 and 28-36) establish each CV as active (complete or partial) or inactive. When a CV is inactive, its length is set to a minimum length (cf. § 7.10.1) and the outlet specific enthalpy is initialized to the specific enthalpy inlet, thus adding a minimum specific enthalpy (*hmin*), in order to avoid inconsistencies when the CV becomes active. The next two blocks of code (lines 43-45, 49-51) establish initial saturation enthalpy values when required, according to § 7.8. The last four blocks of code provide additional initial equations for different initial configurations in the evaporator.

```
 1  constant SIunits.SpecificEnthalpy hmin = 1e-4;
    initial equation
 3    // --------------------------------------------- //
      // CV1 INITIALIZATION                            //
 5    // --------------------------------------------- //
      if cv1_active then
 7      if cv2_active then v1.mode = Constants.RM_COMPLETE;
        else v1.mode = Constants.RM_PARTIAL;
 9      end if;
      else
11      v1.mode       = Constants.RM_INACTIVE;
        v1.Li         = Constants.minLength;
13      v1.hb         = v1.ha + Constants.hmin;
      end if;
15    // --------------------------------------------- //
      // CV2 INITIALIZATION                            //
17    // --------------------------------------------- //
      if cv2_active then
19      v2.mode     = Constants.RM_COMPLETE;
      else
21      v2.mode     = Constants.RM_INACTIVE;
        v2.Li       = Constants.minLength;
23      v2.hb       = v2.ha + Constants.hmin;
      end if;
25    // --------------------------------------------- //
      // CV3 INITIALIZATION                            //
27    // --------------------------------------------- //
      if cv3_active then
29      if cv2_active then v3.mode = Constants.RM_COMPLETE;
        else v3.mode = Constants.RM_PARTIAL;
31      end if;
      else
33      v3.mode       = Constants.RM_INACTIVE;
        v3.Li         = Constants.minLength;
35      v3.hb         = v3.ha + Constants.hmin;
      end if;
37    // ------------------------------------------------------------------------------- //
      // CONSTRAINTS                                                                     //
      // ------------------------------------------------------------------ //
      // -------------------------------------
      // CV1 and CV2
      // -------------------------------------
      if cv1_active and cv2_active then
        v1.hb = Medium.bubbleEnthalpy(sat);
      end if;
      // -------------------------------------
      // CV2 and CV3
      // -------------------------------------
      if cv2_active and cv3_active then
        v2.hb = Medium.dewEnthalpy(sat);
      end if;
      // ------------------------------------------------------------------ //
      // INITIALIZATION FOR DIFFERENT CONFIGURATIONS                        //
      // ------------------------------------------------------------------ //
      // -------------------------------------
      // CV1, CV2 and CV3
      // -------------------------------------
      if cv1_active and cv2_active and cv3_active then
        der(v1.Li) = 0;
        der(v3.Li) = 0;
        der(v3.hb) = 0;
      end if;
      // -------------------------------------
      // CV1
      // -------------------------------------
      if cv1_active and not cv2_active and not cv3_active then
        v1.hb = Medium.specificEnthalpy_pT(p0, Tout, 1);
      end if;
      // -------------------------------------
      // CV3
      // -------------------------------------
      if cv3_active and not cv2_active and not cv1_active then
        v3.hb = Medium.specificEnthalpy_pT(p0, Tout, 1);
      end if;
      // -------------------------------------
      // CV1 and CV2
      // -------------------------------------
      if cv1_active and cv2_active and not cv3_active then
        v2.Li = cv2_length;
      end if;
      // -------------------------------------
      // CV2 and CV3
      // -------------------------------------
      if cv3_active and cv2_active and not cv1_active then
        v2.Li       = cv2_length;
        der(v3.hb) = 0;
      end if;
```

Listing E.2: Initialization Modelica code for the switching moving boundary general evaporator model

The initialization Modelica codes for the flooded/dry evaporator and general/flooded/dry condenser models are similar to those described in this section relating to the general evaporator model.

Initialization examples are given in § 7.13.1.1 considering the SC-TP initial configuration, in § 7.13.1.2 for the SC-TP-SH initial configuration and in § 7.13.2 for the TP initial configuration (although it seems to be the SH initial configuration. If Fig. 7.22 is inspected, the inlet and outlet specific enthalpies are slightly behind the specific enthalpy of the saturated vapor). For the remaining possible initialization configurations, examples are given in this section. The test model is the same as that described in § 7.13.2 – only the DSI and the inlet specific enthalpy have been changed.

In the SC initial configuration test case, the DSI is set to 600 W/m^2. Fig. E.1 shows the outlet specific enthalpy (h_{out}) in a switching general evaporator when varying sinusoidally the inlet specific enthalpy (h_{in}). The two constant lines shown in Fig. E.1 are the enthalpies of saturated liquid and vapor (h', h''). They are constant because the pressure is held constant. The CV lengths are shown in Fig. E.2

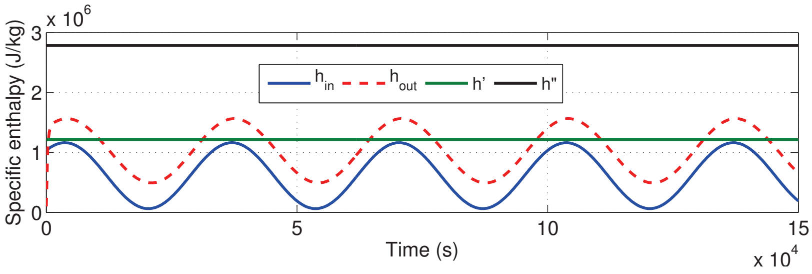


Figure E.1: Specific enthalpies in the SC initial configuration test

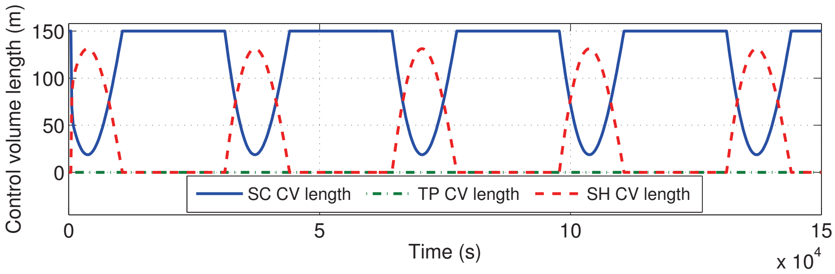


Figure E.2: CV lengths in the SC initial configuration test

Considering the SH initial configuration test case, the inlet and outlet specific enthalpies and the CV lengths are shown in Figs. E.3 and E.4, respectively. The DSI is fixed in this case to 700 W/m^2.

Considering the TP-SH initial configuration test case, the inlet and outlet specific enthalpies and the CV lengths are shown in Figs. E.5 and E.6 respectively. The DSI is fixed to 900 W/m^2.

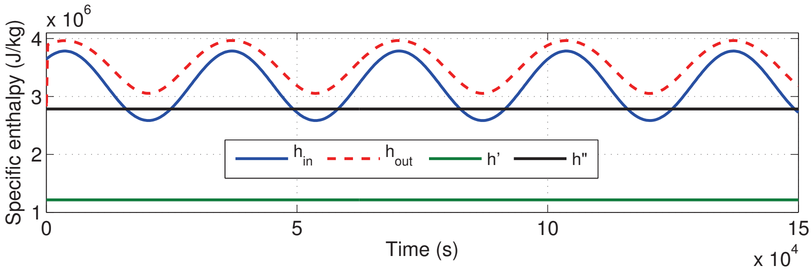


Figure E.3: Specific enthalpies in the SH initial configuration test

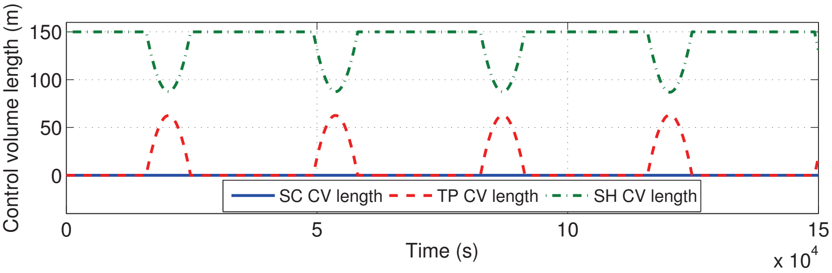


Figure E.4: CV lengths in the SH initial configuration test

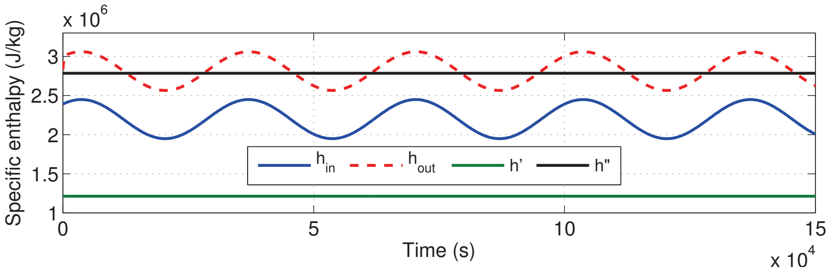


Figure E.5: Specific enthalpies in the TP-SH initial configuration test

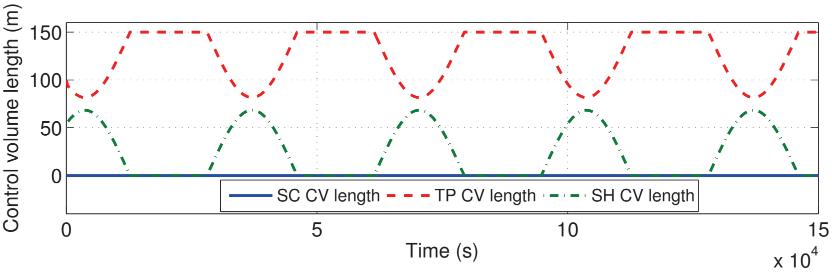


Figure E.6: CV lengths in the TP-SH initial configuration test

F

OpenMP Code for Initialising the DISS Modelica Model

Abstract: This appendix gives details about the procedure used to parallelize the initial section in the resulting *C* code, which was obtained by translating the DISS Modelica model using the Dymola tool.

F.1 Parallelization Procedure

The procedure used to parallelize the initial section in the resulting C code, obtained from translating the Modelica model by employing the Dymola tool, is as follows.

1. A reference to the OpenMP header (*omp.h*) has been included in the resulting C source code generated by the Dymola tool (*dsmodel.c*), as shown in List. F.1, in order to use the OpenMP API, directives and functions in the source code.

```
#include <omp.h>
```

Listing F.1: OpenMP header inclusion

2. Set the maximum number of threads to be executed using the *omp_set_num_threads()* function, as shown in List. F.2. In our particular case, this value was set considering the number of cores in the processor, in this case four cores.

```
constant int NCores = 4;
omp_set_num_threads(NCores);
```

Listing F.2: Setting the maximum number of threads

```
#pragma omp parallel
{
  #pragma omp sections
  {
    #pragma omp section
    {
      ........
      ........
      Tf[0]=getIniValue(iniTime,"TempFluidPTC1");
      Tf[1]=getIniValue(iniTime,"TempFluidPTC2");
      ........
      ........
    }
    #pragma omp section
    {
      ........
      ........
      Tp[0]=getIniValue(iniTime,"TempPipePTC1");
      Tp[1]=getIniValue(iniTime,"TempPipePTC2");
      ........
      ........
    }
    #pragma omp section
    {
      ........
      ........
      irr[0]=getIniValue(iniTime,"PressurePTC1");
      irr[1]=getIniValue(iniTime,"PressurePTC2");
      ........
      ........
    }
    #pragma omp section
    {
      ........
      ........
      mflow[0]=getIniValue(iniTime,"MassFlowRatePTC1");
      mflow[1]=getIniValue(iniTime,"MassFlowRatePTC2");
      ........
      ........
    }
  }
}
```

Listing F.3: OpenMP parallel sections

3. Locate the initial section(s) in the *dsmodel.c* file, which can be identified by the *InitialSection* identifier in the C source code.

Next, the sentences in the initial section must be manually distributed between the different threads to balance computational load between cores. For this purpose, the OpenMP *sections* and *section* directives were used. The source code is shown in List. F.3.

First, the *parallel* directive was used to indicate that the successive sentences must be executed in parallel. Then, the *sections* and *section* directives were used to define four sections corresponding with four threads. The reading operations were equally distributed between the four sections, each one executed by a different thread in each available processor core.

4. Compile the *dsmodel.c* including the OpenMP library. For the GNU Compiler Collection (GCC) in Linux operating systems, only the *fopenmp* argument is necessary to compile the *dsmodel.c* source code with OpenMP support. For this purpose, a script file, *compile.sh*, was created to include the previously mentioned argument in the compilation script. The default compilation script for the Dymola tool in Linux platforms is *dsbuild.sh*.

References

Adams, J., Clark, D. R., Louis, J. R. and Spanbauer, J. P. Mathematical modeling of once-through boiler dynamics. *IEEE Transactions on Power Apparatus and Systems*, vol. 84(2), pages 146 – 156, 1965. ISSN 0018-9510.

Ajona, J. I. *Electricity generation with distributed collector systems. Solar thermal electricity generation*. CIEMAT, Madrid, Spain, 1999. ISBN 84-7834-353-9.

Åkesson, J., Årzén, K. E., Gäfvert, M., Bergdahl, T. and Tummescheit, H. Modeling and optimization with Optimica and JModelica.org - Languages and tools for solving large-scale dynamic optimization problems. *Computers & Chemical Engineering*, vol. 34(11), pages 1737–1749, 2010. ISSN 0098-1354.

Anderson, J. D. *Computational fluid dynamics*. McGraw-Hill, 1995. ISBN 978-0-07-001685-9.

Andersson, M. *Omola - an object-oriented language for model representation*. Licentiate Thesis, Lund Institute of Technology, Lund, Sweden, 1990.

Architecture Review Board. The OpenMP API specification for parallel programming. http://openmp.org/, 2011.

Aronsson, P. *Automatic parallelization of equation-based simulation programs*. Ph.D. Thesis, Linköping University, Linköping, Sweden, 2006.

Aronsson, P. and Fritzson, P. A task merging technique for parallelization of Modelica models. In *Proc. 4th International Modelica Conference*, pages 123–128. Hamburg University of Technology, Hamburg-Harburg, Germany, 2005.

Åström, K. J. and Bell, R. D. Drum-boiler dynamics. *Automatica*, vol. 36(3), pages 363–378, 2000. ISSN 0005-1098.

Åström, K. J., Elmqvist, H. and Mattson, S. Evolution of continuous-time modeling and simulation. In *Proc. 12th European Simulation Multiconference*, pages 1–10. 1998.

Barton, P. and Pantelides, C. Modeling of combined discrete/continuous processes. *AIChE Journal*, vol. 40(6), pages 966–979, 1994. ISSN 1547-5905.

Bastian, J., Clauss, C., Enge-rosenblatt, O. and Schneider, P. MOSILAB – a Modelica solver for multi-physics problems with structural variability. http://publica.fraunhofer.de/eprints/urn:nbn:de:0011-n-1355711.pdf, 2010.

Beck, B. T. and Wedekind, G. L. A generalization of the system mean void fraction model for transient two-phase evaporating flows. *Journal of Heat Transfer*, vol. 103(1), pages 81–85, 1981. ISSN 0022-1481.

Bendapudi, S. *Development and evaluation of modeling approaches for transients in centrifugal chillers*. Ph.D. Thesis, Purdue University, 2004.

Bendapudi, S., Braun, J. and Groll, E. A comparison of moving-boundary and finite-volume formulations for transients in centrifugal chillers. *International Journal of Refrigeration*, vol. 31(8), pages 1437–1452, 2008. ISSN 0140-7007.

Beyer, H. and Deb, K. On self-adaptive features in real-parameter evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, vol. 5(3), pages 250–270, 2001. ISSN 1089-778X.

Blanco-Muriel, M., Alarcon-Padilla, D., López-Moratalla, T. and Lara-Coira, M. Computing the solar vector. *Solar Energy*, vol. 70(5), pages 431–441, 2001. ISSN 0038-092X.

Blochwitz, T., Otter, M., Arnold, M., Bausch, C., Clauss, C., Elmqvist, H., Junghanns, A., Mauss, J., Monteiro, M., Neidhold, T., Neumerkel, D., Olsson, H., Peetz, J.-V. and Wolf, S. The functional mockup interface for tool independent exchange of simulation models. In *Proc. 8th International Modelica Conference*. Dresden, Germany, 2010.

Bloomberg New Energy Finance. Global trends in renewable energy investment 2011. http://www.unep.org/pdf/BNEF_global_trends_in_renewable_energy_investment_2011_report.pdf, 2011.

Brenan, K. E., Campbell, S. L. and Petzold, L. R. *Numerical solution of initial-value problems in differential-algebraic equations*. Society for Industrial and Applied Mathematics (SIAM), 2nd edition, 1996. ISBN 978-0-89871-353-4.

Breunese, A. P. and Broenink, J. F. Modeling mechatronic systems using the SIDOPS language. In *Proc. 3rd International Conference on Bond Graph Modeling and Simulation (ICBGM-97)*, pages 301–306. Phoenix, Arizona, USA, 1997.

British Petroleum. BP statistical review of world energy June 2012. http://www.bp.com/assets/bp_internet/globalbp/globalbp_uk_english/reports_and_publications/statistical_energy_review_2011/STAGING/local_assets/pdf/statistical_review_of_world_energy_full_report_2012.pdf, 2012.

Brown, P., Hindmarsh, A. C. and Petzold, L. R. Using Krylov methods in the solution of large-scale differentialaAlgebraic systems. *SIAM Journal on Scientific Computing*, vol. 15(6), pages 1467–1488, 1994. ISSN 1064-8275.

Budd, C. and Dux, F. Chattering and related behavior in impact oscillators. *Philosophical transactions of the royal society of London series A - Mathematical physical and engineering sciences*, vol. 347(1683), pages 365–389, 1994. ISSN 0962-8428.

Bush, V. The differential analyzer: a new machine for solving differential equations. *Journal of the Franklin Institute*, vol. 212, pages 447–488, 1931. ISSN 0016-0032.

Camacho, E. F., Berenguel, M. and Rubio, F. R. *Advanced control of solar plants*. Springer-Verlag Berlin and Heidelberg GmbH & Co., 1997. ISBN 978-3-540-76144-0.

Camacho, E. F., Berenguel, M., Rubio, F. R. and Martínez, D. *Control of solar energy systems*. Springer-Verlag Berlin and Heidelberg GmbH & Co., 2012. ISBN 978-0-85729-915-4.

Casella, F. Object-oriented modelling of two-phase fluid flows by the finite volume method. In *Proc. 5th Vienna International Conference on Mathematical Modelling (Mathmod)*, pages 1–1 – 1–10. Vienna, Austria, 2006.

Casella, F. and Leva, A. Modelica open library for power plant simulation : design and experimental validation. In *Proc. 3rd International Modelica Conference*, pages 41–50. Linköping, Sweden, 2003.

Casella, F. and Leva, A. Modelling of thermo-hydraulic power generation processes using Modelica. *Mathematical and Computer Modelling of Dynamical Systems*, vol. 12(1), pages 19–33, 2006. ISSN 1387-3954.

Casella, F., Otter, M., Proelss, K., Richter, C. and Tummescheit, H. The Modelica Fluid and Media library for modeling of incompressible and compressible thermo-fluid pipe networks. In *Proc. 5th International Modelica Conference*, pages 631–640. Vienna, Austria, 2006.

Casella, F. and Richter, C. ExternalMedia: a library for easy re-use of external fluid property code in Modelica. In *Proc. 6th International Modelica Conference*, pages 157–161. Braunschweig, Germany, 2008.

Çengel, Y. A. and Boles, M. A. *Thermodynamics: an engineering approach*. McGraw-Hill series in mechanical engineering. McGraw-Hill Higher Education, 7th edition, 2010. ISBN 978-0-07-736674-2.

Cecchinato, L. and Mancini, F. An intrinsically mass conservative switched evaporator model adopting the moving-boundary method. *International Journal of Refrigeration*, vol. 35(2), pages 349–364, 2012. ISSN 0140-7007.

Cellier, F. E. *Combined continuous/discrete system simulation by use of digital computers: techniques and tools*. Ph.D. Thesis, ETH, 1979.

Cellier, F. E. *Continuous system modeling*. Springer-Verlag New York, Inc. Secaucus, NJ, USA, 1991. ISBN 978-0-387-97502-3.

Cellier, F. E. Object-oriented modeling: means for dealing with system complexity. In *Proc. 15th Benelux Meeting on Systems and Control*, pages 53–64. Mierlo, The Netherlands, 1996.

Cellier, F. E. and Elmqvist, H. Automated formula manipulation supports object-oriented continuous-system modeling. *IEEE Control Systems*, vol. 13(2), pages 28–38, 1993. ISSN 1066-033X.

Cellier, F. E., Elmqvist, H. and Otter, M. Modeling from physical principles. In *The Control Handbook*, pages 99–107. CRC Press, 1996.

Cellier, F. E. and Kofman, E. *Continuous system simulation*. Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2006. ISBN 978-0-387-26102-7.

Chapra, S. C. and Canale, R. P. *Numerical methods for engineers with programming and software applications*. McGraw-Hill Science/Engineering/Math, 1998. ISBN 978-0-07-243193-3.

Chen, J. C. Correlation for boiling heat transfer to saturated fluids in convective flow. *Industrial Engineering Chemistry Process Design and Development*, vol. 5(3), pages 322–329, 1966. ISSN 0196-4305.

Chen, N. H. An explicit equation for friction factor in pipe. *Industrial & Engineering Chemistry Fundamentals*, vol. 18(3), pages 296–297, 1979. ISSN 0196-4313.

Cheng, T. and Asada, H. H. Nonlinear observer design for a varying-order switched system with application to heat exchangers. In *Proc. 2006 American Control Conference*, pages 2898–2903. Minneapolis, Minnesota, USA, 2006. ISBN 978-1-4244-0210-6.

Chipperfield, A., Fleming, P., Pohlheim, H. and Fonseca, C. Genetic algorithm toolbox for use with MATLAB. 1994.

Clabaugh, J. A. ABACUSS II syntax manual. http://yoric.mit.edu/abacuss2/, 2001.

Colebrook, C. F. Turbulent flow in pipes, with particular reference to the transition region between the smooth and rough pipe laws. *Journal of the Institution of Civil engineers*, vol. 11(4), pages 133–156, 1939.

Collier, J. G. and Thome, J. R. *Convective boiling and condensation*. Oxford Engineering Science Series. Clarendon Press, 1996. ISBN 978-0-19-856296-2.

Courant, R., Isaacson, E. and Rees, M. On the solution of nonlinear hyperbolic differential equations by finite differences. *Communications on Pure and Applied Mathematics*, vol. 5, pages 243–255, 1952. ISSN 1097-0312.

Dassault Systemes. Dymola 2013. http://www.3ds.com/products/catia/portfolio/dymola, 2012.

Deb, K. *Multi-Objective optimization using evolutionary algorithms*. Wiley, 1^{st} edition, 2001. ISBN 978-0-471-87339-6.

Delthatheta. Converge: enabling model lifecycle management. http://www.deltatheta.com/products/converge/, 2012a.

Delthatheta. Vertex: a new approach to simulation. http://www.deltatheta.com/products/vertex/, 2012b.

Denn, M. *Process fluid mechanics*. Prentice-Hall, Englewood Cliffs, 1980. ISBN 978-0-13-723163-8.

Dhar, M. and Soedel, W. Transient analysis of a vapor compression refrigeration system. In *Proc. 15^{th} International Congress of Refrigeration*, pages 1031–1067. Venice, Italy, 1979.

Digiteo. Scilab - free open source software for numerical computation. http://www.scilab.org/, 2012.

Dittus, F. W. and Boelter, L. M. K. Heat transfer in automobile radiators of the tubular type. *University of California Publications in Engineering*, vol. 2(1), pages 443–461, 1930. ISSN 0735-1933.

Dynasim AB. *Dymola 6.0 user manual*, 2006.

EA International. EcosimPro - modeling and simulation software. http://www.ecosimpro.com/, 2011.

Eastman. Therminol - heat transfer fluids. http://www.therminol.com/, 2013.

Eborn, J. *On model libraries for thermo-hydraulic applications*. Ph.D. Thesis, Department of Automatic Control, Lund Institute of Technology, Sweden, 2001.

Eck, M., Zarza, E., Eickhoff, M., Rheinländer, J. and Valenzuela, L. Applied research concerning the direct steam generation in parabolic troughs. *Solar Energy*, vol. 74(4), pages 341–351, 2003. ISSN 0038-092X.

Eickhoff, M. Incident angle modifier of the LS-3-DISS-Collectors. Tech. report DISS-SC-SF-30, Plataforma Solar de Almería, Almería, Spain, 2002.

El Hefni, B. and Bouskela, D. Dynamic modelling of a condenser/water heater with the ThermoSysPro Library. In *Proc. 9^{th} International Modelica Conference*, pages 745–758. Munich, Germany, 2012.

Eldredge, B. D., Rasmussen, B. P. and Alleyne, A. G. Moving-boundary heat exchanger models with variable outlet phase. *Journal of Dynamic Systems, Measurement, and Control*, vol. 130(6), pages 061003–1 – 061003–12, 2008. ISSN 0022-0434.

Elmqvist, H. SIMNON - An interactive simulation program for nonlinear systems - User's manual. Tech. report TFRT-7502, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1975.

Elmqvist, H. *A structured model language for large continuous systems*. Ph.D. Thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1978.

Elmqvist, H. and Otter, M. Methods for tearing systems of equations in object-oriented modeling. In *Proc. European Simulation Multiconference 1994 (ESM94)*, vol. 2, pages 326–332. Barcelona, Spain, 1994.

Elmqvist, H., Tummescheit, H. and Otter, M. Object-oriented modeling of thermo-fluid systems. In *Proc. 3^{rd} International Modelica Conference*, pages 269–286. Linköping, Sweden, 2003.

Enermodal Engineering Limited. Cost reduction study for solar thermal power plants. Tech. report, Enermodal Engineering Limited, 650 Riverbend Drive, Kitchener, ON N 2K 3S2, 1999.

EuroSysLib. EuroSysLib - Modelica libraries for embedded systems modeling and simulation. http://www.eurosyslib.com/, 2010.

Eymard, R., Gallouet, T. and Herbin, R. Finite volume methods. In *Handbook of Numerical Analysis*, pages 713–1020. North-Holland, 2000. ISBN 978-0-444-81794-5.

Feldhoff, J. F. Direct Steam Generation (DSG) - Technology overview. Tech. report, SFERA Summer School, Almería, Spain, 2012.

Fernández-García, A., Zarza, E., Valenzuela, L. and Pérez, M. Parabolic-trough solar collectors and their applications. *Renewable and Sustainable Energy Reviews*, vol. 14(7), pages 1695–1721, 2010. ISSN 1364-0321.

Filippov, A. F. and Arscott, F. M. *Differential equations with discontinuous righthand sides*. Mathematics and Its Applications. Kluwer Academic Publishers, 1988. ISBN 978-90-277-2699-5.

Fowler, A. C. *Mathematical models in the applied sciences*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 1997. ISBN 978-0-521-46703-2.

Franke, R., Casella, F., Otter, M., Sielemann, M., Elmqvist, H., Mattson, S. and Olsson, H. Stream connectors – An extension of Modelica for device-oriented modeling of convective transport phenomena. In *Proc. 7th International Modelica Conference*, pages 108–121. Como, Italy, 2009a.

Franke, R., Casella, F., Sielemann, M., Proelss, K., Otter, M. and Wetter, M. Standardization of thermo-fluid modeling in Modelica.Fluid. In *Proc. 7th International Modelica Conference*, pages 122–131. Como, Italy, 2009b. ISBN 978-91-7393-513-5.

Frey, P. and O'Riordan, D. Verilog-AMS: mixed-signal simulation and cross domain connect modules. In *Proc. 2000 IEEE/ACM International Workshop on Behavioral Modeling and Simulation*, pages 103–108. Washington, DC, USA, 2000.

Fritzson, P. *Principles of object-oriented modeling and simulation with Modelica 2.1*. IEEE Press, John Wiley & Sons, 2004. ISBN 978-0-471-47163-9.

Fritzson, P., Viklund, L., Herber, J. and Fritzson, D. High-level mathematical modeling and programming. *IEEE Software*, vol. 12(4), pages 77–87, 1995. ISSN 0740-7459.

Gander, W. and Gruntz, D. Derivation of numerical method using computer algebra. *SIAM Review*, vol. 41(3), pages 577–593, 1999. ISSN 0036-1445.

Gary, D. K. *Interpolating cubic splines*. Birkhäuser Boston, 1999. ISBN 978-0-8176-4100-9.

Gear, C. W. Simultaneously numerical solution of differential-algebraic equations. *IEEE Transactions on Circuit Theory*, vol. 18, pages 217–225, 1971. ISSN 0018-9324.

Gerald, C. F. and Wheatley, P. O. *Applied numerical analysis*. Addison-Wesley, 6th edition, 1994. ISBN 978-0-321-13304-5.

Ghiaasiaan, S. M. *Two-phase flow, boiling and condensation in conventional and miniature systems*. Cambridge University Press, 1st edition, 2007. ISBN 978-05-2188-276-7.

Giorgidze, G. and Nilsson, H. Higher-order non-causal modelling and simulation of structurally dynamic systems. In *Proc. 7th International Modelica Conference*. Como, Italy, 2009. ISBN 978-91-7393-513-5.

Gnielinski, V. New equations for heat and mass transfer in turbulent pipe flow and channel flow. *International Chemical Engineering*, vol. 2(16), pages 359–368, 1976. ISSN 0020-6318.

Goebel, O. Thermohydraulics of direct steam generation. In *Proc. 9th International Symposium on Solar Thermal Concentrating Technologies*. Odeillo-Font-Romeu, 1998.

González, L., Zarza, E. and Yebra, L. J. Determinación del modificador por ángulo de incidencia de un colector LS-3, incluyendo la pérdidas geométricas por final de colector. Tech. report DISS-SC-SF-30, Plataforma Solar de Almería, Almería, Spain, 2001.

Gräber, M., Kirches, C., Schlöder, J. and Tegethoff, W. Nonlinear model predictive control of a vapor compression cycle based on first principle models. In *Proc. 7th Vienna International Conference on Mathematical Modelling (Mathmod)*. Vienna, Austria, 2012.

Gräber, M., Strupp, N. C. and Tegethoff, W. Moving boundary heat exchanger model and validation procedure. In *Proc. 8th EUROSIM Congress on Modelling and Simulation*. Prague, Czech Republic, 2010.

Grace, A. C. W. SIMULAB, an integrated environment for simulation and control. In *Proc. 1991 American Control Conference*, pages 1015–1020. Boston, MA, USA, 1991.

Grald, E. W. and MacArthur, J. W. A moving-boundary formulation for modeling time-dependent two-phase flows. *International Journal of Heat and Fluid Flow*, vol. 13(3), pages 266–272, 1992. ISSN 0142-727X.

Gungor, K. E. and Winterton, R. H. S. A general correlation for flow boiling in tubes and annuli. *International Journal of Heat and Mass Transfer*, vol. 29(3), pages 351–358, 1986. ISSN 0017-9310.

Haaland, S. Simple and explicit formulas for the friction factor in turbulent pipe flow. *Journal of Fluids Engineering*, vol. 105(1), pages 89–90, 1983. ISSN 0098-2202.

Harlow, F. H. and Welch, E. J. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, vol. 8(12), pages 2182–2189, 1965. ISSN 1070-6631.

He, X., Liu, S., Asada, H. H. and Itoh, H. Multivariable control of vapor compression systems. *HVAC&R Research*, vol. 4(3), pages 205–230, 1998. ISSN 1078-9669.

Hewitt, G. F. *Measurement of two phase flow parameters*. Academic Press, London, New York, 1978. ISBN 978-0-12-346260-2.

Hindmarsh, A. C. ODEPACK, A systematized collection of ODE solvers. *IMACS Transactions on Scientific Computation*, vol. 1, pages 55–64, 1983. ISSN 0169-7919.

Hongesombut, K., Mitani, Y. and Tsuji, K. An incorporated use of genetic algorithm and a Modelica library for simultaneous tuning of power system stabilizers. In *Proc. 2^{nd} International Modelica Conference*, pages 89–98. Oberpfaffenhofen, Germany, 2002.

Hsu, Y.-L. ME550 - Geometrical modeling and computer graphics. Course, Gerontechnology Research Center, Optimal Design Laboratory, Yuan Ze University, Taiwan. 2010.

IAPWS. Supplementary release on backward equations for pressure as a function of enthalpy and entropy $p(h, s)$ to the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. http://www.iapws.org/relguide/phs.pdf, 2001.

IAPWS. Guideline on the Tabular Taylor Series Expansion (TTSE) method for calculation of thermodynamic properties of water and steam applied to IAPWS-95 as an example. http://www.iapws.org/relguide/TTSE.pdf, 2003.

IAPWS. Revised supplementary release on backward equations for the functions $T(p, h)$, $v(p, h)$ and $T(p, s)$, $v(p, s)$ for region 3 of the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. http://www.iapws.org/relguide/RevisedBack3.pd, 2004a.

IAPWS. Supplementary release on backward equations $p(h, s)$ for region 3, equations as a function of h and s for the region boundaries, and an equation $T_{sat}(h, s)$ for region 4 of the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. http://www.iapws.org/relguide/Back3HS.pdf, 2004b.

IAPWS. Supplementary release on backward equations for specific volume as a function of pressure and temperature $v(p, T)$ for region 3 of the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. http://www.iapws.org/relguide/Back3VPT.pdf, 2005.

IAPWS. Release on the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. http://www.iapws.org/relguide/IF97-Rev.pdf, 2007.

IAPWS. Revised release on the IAPWS formulation 1995 for the thermodynamic properties of ordinary water substance for general and scientific use. http://www.iapws.org/relguide/IAPWS95-Rev.pdf, 2009.

IEEE. Standard VHDL analog and mixed-signal extensions. Tech. report 1076.1, IEEE, 1997.

IEEE. System - IEEE, conferences, publications and resources. http://technav.ieee.org/tag/1662/system, 2013.

International Energy Agency. Clean energy progress report, June 2011. http://iea.org/papers/2011/CEM_Progress_Report.pdf, 2011a.

International Energy Agency. *Solar energy perspectives*. IEA Publications, 2011b. ISBN 978-92-64-12457-8.

ITEA-2. ITEA-2, strategic pan-European programme for advanced precompetitive R&D in Software-intensive Systems and Services (SiSS). http://www.itea2.org/, 2012.

ITI. SimulationX. http://www.itisim.com/simulationx/, 2012.

Jackson, A. S. *Analog computation*. ASIN B0006AW3BS, McGraw-Hill, New York, 1960.

Jeandel A., Boudaud F. and Larivière E. ALLAN simulation release 3.1 description. Tech. report, M.DéGIMA.GSA1887. Gaz de France, Saint Denis La plaine, France, 1997.

Jensen, J. M. *Dynamic modeling of thermo-fluid systems with focus on evaporators for refrigeration*. Ph.D. Thesis, Technical University of Denmark, Lyngby, Denmark, 2003.

Jensen, J. M. and Tummescheit, H. Moving boundary models for dynamic simulations of two-phase flows. In *Proc. 2^{nd} International Modelica Conference*, vol. 3, pages 235 – 244. Oberpfaffenhofen, Germany, 2002.

Ji, J., He, H., Chow, T., Pei, G., He, W. and Liu, K. Distributed dynamic modeling and experimental study of PV evaporator in a PV/T solar-assisted heat pump. *International Journal of Heat and Mass Transfer*, vol. 52(5-6), pages 1365–1373, 2009. ISSN 0017-9310.

JModelica.org. JModelica.org - Open source Modelica platform for modeling, simulation and optimzation. http://www.jmodelica.org/, 2012.

Johansson, K. H., Lygeros, J., Sastry, S. and Egerstedt, M. Simulation of Zeno hybrid automata. In *Proc. 38th IEEE Conference on Decision and Control*, vol. 4, pages 3538–3543. Phoenix, Arizona, USA, 1999.

John Pye. Freesteam 2.0: steam tables, open source, IAPWS-IF97. http://freesteam.sourceforge.net/, 2010.

Kandlikar, S. G. A general correlation for saturated two-phase flow boiling heat transfer inside horizontal and vertical tubes. *Journal of Heat Transfer*, vol. 112, pages 219–228, 1990. ISSN 0022-1481.

Kloas, M., Friesen, V. and Simons, M. Smile: a simulation environment for energy systems. *Systems Analysis Modelling Simulation*, vol. 18-19, pages 503–506, 1995. ISSN 0232-9298.

Kretzschmar, H.-J., Cooper, J. R., Gallagher, J. S., Harvey, A. H., Knobloch, K., Mares, R., Miyagawa, K., Okita, N., Span, R., Stocker, I., Wagner, W. and Weber, I. Supplementary backward equations $p(h, s)$ for the critical and supercritical regions (region 3), and equations for the two-phase region and region boundaries of the IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. *Journal of Engineering for Gas Turbines and Power*, vol. 129(4), pages 1125–1137, 2007. ISSN 0742-4795.

Krüger, I., Mehlhase, A. and Schmitz, G. Variable structure modeling for vehicle refrigeration applications. In *Proc. 9th International Modelica Conference*, pages 927–933. Munich, Germany, 2012.

Kumar, M., Kar, I. and Ray, A. State space based modeling and performance evaluation of an air-conditioning system. *HVAC&R Research*, vol. 14(5), pages 797–816, 2008. ISSN 1078-9669.

Kunick, M. and Kretzschmar, H.-J. Fast calculation of thermodynamic properties of water and steam in process modelling using spline interpolation. In *Proc. 15th International Congress on the Properties of Water and Steam (ICPWS)*. Berlin, Germany, 2008.

Law, A. and Kelton, D. *Simulation modelling and analysis*. McGraw-Hill Education, New York, USA, 4th edition, 1991. ISBN 978-0-07-059292-6.

Leducq, D., Guilpart, J. and Trystram, G. Low order dynamic model of a vapor compression cycle for process control design. *Journal of Food Process Engineering*, vol. 26(1), pages 67–91, 2003. ISSN 1745-4530.

Lee, H. and Utkin, V. I. Chattering suppression methods in sliding mode control systems. *Annual Reviews in Control*, vol. 31(2), pages 179–188, 2007. ISSN 1367-5788.

Li, B. and Alleyne, A. G. A dynamic model of a vapor compression cycle with shut-down and start-up operations. *International Journal of Refrigeration*, vol. 33(3), pages 538–552, 2010. ISSN 0140-7007.

LMS. LMS Imagine.Lab AMESim, the integrated platform for multi-domain system simulation. http://www.lmsintl.com/imagine-amesim-1-d-multi-domain-system-simulation, 2012.

Lundvall, H. *Automatic parallelization using pipelining for equation-based simulation languages*. Licentiate Thesis, Linköping University, Linköping, Sweden, Department of Computer and Information Science, 2008.

Maier, R. S., Petzold, L. R. and Rath, W. Parallel solution of large-scale differential-algebraic systems. *Concurrency: practice and experience*, vol. 7(8), pages 795 – 822, 1995. ISSN 1096-9128.

Maly, T. and Petzold, L. R. Numerical methods and software for sensitivity analysis of differential-algebraic systems. *Applied Numerical Mathematics*, vol. 20(1–2), pages 57–79, 1996. ISSN 0168-9274.

Mancini, F. *Energy efficiency improvements of household heat pump systems*. Ph.D. Thesis, University of Padova, Padova, Italy, 2011.

MapleSoft. Maple 16 - Technical computing software for engineers, mathematicians, scientists, instructors and students. http://www.maplesoft.com/products/maple/, 2012a.

MapleSoft. MapleSim - High-performance physical modeling and simulation. http://www.maplesoft.com/products/maplesim//, 2012b.

MathCore Engineering AB. MathModelica - modeling, simulation, analysis and documentation of multi-engineering and life science systems. http://www.mathcore.com/products/mathmodelica/, 2012.

Mckinley, T. L. and Alleyne, A. G. An advanced nonlinear switched heat exchanger model for vapor compression cycles using the moving-boundary method. *International Journal of Refrigeration*, vol. 31(7), pages 1253–1264, 2008. ISSN 0140-7007.

Mehlhase, A. Varying the level of detail during simulation. In *Symposium Simulationstechnik (ASIM 2011)*. ASIM, Winterthur, Switzerland, 2011. ISBN 978-3-905745-44-3.

Mehlhase, A. A Python package for simulating variable-structure models with Dymola. In *Proc. 7th Vienna International Conference on Mathematical Modelling (Mathmod)*. Vienna, Austria, 2012.

Meier, H. F., Alves, J. J. N. and Mori, M. Comparison between staggered and collocated grids in the finite-volume method performance for single and multi-phase flows. *Computers & Chemical Engineering*, vol. 23(3), pages 247–262, 1999. ISSN 0098-1354.

Mitchell, E. E. L. and Gauthier, J. S. Advanced Continuous Simulation Language (ACSL). *Simulation*, pages 72–78, 1976. ISSN 0037-5497.

Miyagawa, K. and Hill, P. G. Rapid and accurate calculation of water and steam properties using the tabular Taylor series expansion method. *Journal of Engineering for Gas Turbines and Power*, vol. 123(3), pages 707–712, 2001. ISSN 0742-4795.

Modelica Association. Modelica standard library 2.2.1. http://www.modelica.org/libraries/Modelica/releases/2.2.1, 2007.

Modelica Association. Modelica standard library 3.2. https://www.modelica.org/libraries/Modelica, 2010.

Modelica Association. Modelica libraries. https://modelica.org/libraries, 2012a.

Modelica Association. Modelica tools. https://modelica.org/tools, 2012b.

Modelisar. FMI for co-simulation specification. http://www.modelisar.com/specifications/FMI_for_CoSimulation_v1.0.pdf, 2010a.

Modelisar. FMI for model exchange specification. http://www.modelisar.com/specifications/FMI_for_ModelExchange_v1.0.pdf, 2010b.

Modelisar. AUTOSAR - Interoperability to support vehicle functional mock-up. http://www.modelisar.com/, 2011.

Modelisar. FMI support in tools. http://www.modelisar.com/tools.html, 2012a.

Modelisar. Functional Mockup Interface (FMI). http://www.modelisar.com/fmi.html, 2012b.

Modelon AB. Modelica physical modeling toolbox for MATLAB. http://www.modelon.com/products/modelica-physical-modeling-toolbox-for-matlab/, 2012.

Mosterman, P. An overview of hybrid simulation phenomena and their support by simulation packages. In *Hybrid Systems: Computation and Control*, vol. 1569 of *Lecture Notes in Computer Science*, pages 165–177. Springer Berlin Heidelberg, 1999. ISBN 978-3-540-65734-7.

Mosterman, P. J. HYBRSIM - a modeling and simulation environment for hybrid bond graphs. *Journal of Systems and Control Engineering - Part I*, vol. 216, pages 35–46, 2002. ISSN 0959-6518.

OpenProd. OpenProd - Open model-driven whole-product development and simulation environment. http://www.openprod.com/, 2012.

Oregon State University. MIMIC, A digital-analog simulator. Tech. report, Computer Center, Oregon State Univeristy, Corvallis, Oregon, USA, 1969.

Östlund, P. *Simulation of Modelica models on the CUDA architecture*. Master Thesis, Linköping University, Linköping, Sweden, Department of Computer and Information Science, 2009.

Otter, M., Elmqvist, H., Blochwitz, T., Mauss, J., Junghanns, A. and Olsson, H. Functional mockup interface - Overview. http://synchronics.inria.fr/lib/exe/fetch.php/modelica-fmi-elmqvist.pdf, 2011.

Otter, M., Elmqvist, H. and Cellier, F. E. Relaxing : a symbolic sparse matrix method exploiting the model structure in generating efficient simulation code. In *Proc. Symposium on Modelling, Analysis, and Simulation, CESA'96, IMACS MultiConference on Computational Engineering in Systems Applications*, pages 1–12. Lille, France, 1996. ISBN 978-2-9502908-5-4.

Pantelides, C. The consistent initialization of differential-algebraic systems. *SIAM Journal on Scientific and Statistical Computing*, vol. 9(2), pages 213–231, 1988. ISSN 0196-5204.

Patankar, S. *Numerical heat transfer and fluid flow*. Hemisphere, Washington,D.C, 1980. ISBN 978-0-89116-522-4.

Peiró, J. and Sherwin, S. Finite difference, finite element and finite volume methods for partial differential equations. In *Handbook of Materials Modeling*, chapter 8 - Fluids, pages 1–32. Springer, 2005. ISBN 978-1-4020-3287-5.

Pelab. The OpenModelica project. http://www.openmodelica.org, 2012.

Pettit, N., Willatzen, M. and Ploug-Sørensen, L. A general dynamic simulation model for evaporators and condensers in refrigeration . Part

II : simulation and control of an evaporator. *International Journal of Refrigeration*, vol. 21(5), pages 404–414, 1998. ISSN 0140-7007.

Petukhov, B. S. Heat transfer and friction in turbulent pipe flow with variable physical properties. *Advances in Heat Transfer*, vol. 6(c), pages 504–564, 1970. ISSN 0065-2717.

Petzold, L. A description of DASSL: a diferential/algebraic system solver. *Scientific Computing*, pages 65–68, 1983. ISSN 0163-5964.

Piela, P., Epperly, T., Westerberg, K. and Westerberg, A. ASCEND: An object-oriented computer environment for modeling and analysis: the modeling language. *Computers & Chemical Engineering*, vol. 15(1), pages 53–72, 1991. ISSN 0098-1354.

Piskunov, N. S. *Differential and integral calculus*, vol. 2. Routledge, 1965. ISBN 978-0-677-20600-4.

Price, H., Lüpfert, E., Kearney, D., Zarza, E., Cohen, G., Gee, R. and Mahoney, R. Advances in parabolic trough solar power technology. *Journal of Solar Energy Engineering*, vol. 124(2), pages 109–125, 2002. ISSN 0199-6231.

Ragazzini, J. R., Randall, R. H. and Russell, F. A. Analysis of problems in dynamics by electronic circuits. *Proc. Institute of Radio Engineers (IRE)*, vol. 35(5), pages 444–452, 1947. ISSN 0096-8390.

Rasmussen, B. P. *Dynamic modeling and advanced control of air conditioning and refrigeration systems*. Ph.D. Thesis, University of Illinois, Urbana, Champaign, Illinois, USA, 2006.

Ratzel, A. C. and Simpson, C. E. Heat loss reduction techniques for annular solar receiver designs. Tech. report, Sandia Labs, Albuquerque, USA, 1979.

Ray, A. Dynamic modelling of once-through subcritical steam generator for solar applications. *Applied Mathematical Modelling*, vol. 4, pages 417 – 423, 1980. ISSN 0307-904X.

Ray, A. Nonlinear dynamic model of a solar steam generator. *IEEE Transactions Power Apparatus and Systems*, vol. 26(4), pages 297 – 306, 1981. ISSN 0302-9743.

Ray, A. and Bowman, H. A nonlinear dynamic model of a once-through subcritical steam generator. *Journal of Dynamic Systems, Measurement, and Control*, vol. 98(3), pages 322 – 329, 1976. ISSN 0022-0434.

Renewable Energy Policy Network for the 21^{st} Century. Renewables, global status report 2010. http://www.ren21.net/Portals/97/documents/GSR/REN21_GSR_2010_full_revised%20Sept2010.pdf, 2010.

Rhie, C. M. and Chow, W. L. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *The American Institute of Aeronautics and Astronautics Journal*, vol. 21, pages 1525–1532, 1983. ISSN 1727-1983.

Rimvall, M. and Cellier, F. E. Evolution and perspectives of simulation languages following the CSSL standard. *Modeling, Identification and Control*, vol. 6, pages 181–199, 1986. ISSN 0332-7353.

Romero-Alvarez, M. and Zarza, E. Concentrating solar thermal power. In *Handbook of Energy Efficiency And Renewable Energy*, chapter 21, pages 1–98. CRC Press, 2007. ISBN 978-0-8493-1730-9.

Rubinstein, R. Y. and Kroese, D. P. *Simulation and the Monte Carlo method*. Wiley Series in Probability and Statistics, 2^{nd} edition, 2008. ISBN 978-0-470-17794-5.

Runge, T. F. *A universal language for continuous network simulation*. Ph.D. Thesis, University of Illinois, Urbana, Champaign, Illinois, USA, 1977.

Sahlin, P., Bring, A. and Sowell, E. F. The neutral model format for building simulation, version 3.02. Tech. report, Department of Building Sciences, The Royal Institute of Technology, Stockholm, Sweden, 1996.

Saldamli, L., Fritzson, P. and Bachmann, B. Extending Modelica for partial differential equations. In *Proc. 2^{nd} International Modelica Conference*, pages 307 – 314. Oberpfaffenhofen, Germany, 2002.

Sanz, V. *Hybrid system modeling using the parallel DEVS formalism and the Modelica language*. Ph.D. Thesis, Escuela Técnica Superior de Ingeniería Informática, Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain, 2010.

Schlichting, H. and Gersten, K. *Boundary-layer theory*. Springer, 2000. ISBN 978-3-540-66270-9.

Selfridge, R. G. Coding a general-purpose digital computer to operate as a differential analyzer. In *Proc. Western Joint Computer Conference*, AFIPS '55 (Western), pages 82–84. ACM, New York, NY, USA, 1955.

Shah, M. M. Chart correlation for saturated boiling heat transfer: equations and further study. *ASHRAE Transactions*, vol. 88, pages 185–196, 1982. ISSN 0001-2505.

Shah, S. C., Floyd, M. A. and Lehman, L. L. Matrix: control design and model building CAE capability. In *Computer-Aided Control Systems Engineering*, chapter 2, pages 181–207. Elsevier Ltd, Amsterdarm, Holland, 1985.

SimForge. SimForge. http://trac.ws.dei.polimi.it/simforge/, 2012.

Skoglund, T., Arzen, K. and Dejmek, P. Dynamic object-oriented heat exchanger models for simulation of fluid property transitions. *International Journal of Heat and Mass Transfer*, vol. 49(13-14), pages 2291–2303, 2006. ISSN 0017-9310.

Spalding, D. B. A novel finite-difference formulation for different expressions involving both first and second derivatives. *International Journal for Numerical Methods in Engineering*, vol. 4(2), pages 551–559, 1972. ISSN 1097-0207.

Steinmann, W.-D. and Tamme, R. Latent heat storage for solar steam systems. *Journal of Solar Energy Engineering*, vol. 130(1), page 011004, 2008. ISSN 0199-6231.

Strauss, J. C. The SCi continuous system simulation language CSSL. *Simulation*, vol. 9, pages 281–303, 1967. ISSN 0037-5497.

Stuetzle, T. A. *Automatic control of the 30 MWe SEGS VI parabolic trough plant*. Master Thesis, University of Wisconsin-Madison, 2002.

Tarjan, R. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, vol. 1(2), pages 146–160, 1972. ISSN 0097-5397.

The American Society of Mechanical Engineers. Rebuilding steam tables. *Mechanical Engineering*, vol. 121(3), pages 78–111, 1999. ISSN 0025-6501.

The MathWorks Inc. *MATLAB R2011a*, 2012.

Thomas, P. J. *Simulation of industrial processes for control engineers*. Elsevier Science & Technology Books, 1999. ISBN 978-0-7506-4161-6.

Tiller, M. *Introduction to physical modelling with Modelica*. Kluwer Academic Publishers, Boston, 2004. ISBN 978-0-7923-7367-4.

Todreas, N. E. and Kazimi, M. S. *Nuclear systems I: thermal hydraulic fundamentals*. Taylor and Francis, 1993. ISBN 978-1-56032-051-7.

Tummescheit, H. *Design and implementation of object-oriented model libraries using Modelica*. Ph.D. Thesis, Lund Institute of Technology, Lund, Sweden, 2002.

UNI. EN 10273: Hot rolled weldable steel bars for pressure purposes with specified elevated temperature properties. Tech. report, ISE/103, 2008.

Utkin, V. I. and Lee, H. The chattering analysis. In *Proc. 12th International Power Electronics and Motion Control Conference EPE-PEMC 2006*, pages 2014–2019. Portoroz, Slovenia, 2006.

Valenzuela, L., Zarza, E., Berenguel, M. and Camacho, E. Control scheme for direct steam generation in parabolic troughs under recirculation operation mode. *Solar Energy*, vol. 80(1), pages 1–17, 2006. ISSN 0038-092X.

Valenzuela, L., Zarza, E., Berenguel, M. and Camacho, E. F. Direct steam generation in solar boilers. *IEEE Control Systems Magazine*, vol. 24, pages 15–29, 2004. ISSN 0272-1708.

Valenzuela, L., Zarza, E., Berenguel, M. and Camacho, E. F. Control concepts for direct steam generation in parabolic troughs. *Solar Energy*, vol. 78(2), pages 301–311, 2005. ISSN 0038-092X.

Van Beek, D. A. and Rooda, J. E. Languages and applications in hybrid modelling and simulation: positioning of Chi. *Control Engineering Practice*, vol. 8, pages 81–91, 2000. ISSN 0967-0661.

Verein Deutscher Ingenieure. *VDI - Wärmeatlas: berechnungsblätter für den wärmeübergang*. Springer-Verlag GmbH, 1997. ISBN 978-3-540-63024-1.

Wagner, F. J. *Object-oriented modeling of energy system*. Ph.D. Thesis, University of Denmark, Lyngby, Denmark, Denmark, 2000.

Wagner, W., Cooper, J. R., Dittmann, A., Kijima, J., Kretzschmar, H. J., Kruse, A., Mareš, R., Oguchi, K., Sato, H., Stöcker, I., Šifner, O., Takaishi, Y., Tanishita, I., Trübenbach, J. and Willkommen, T. The IAPWS Industrial Formulation 1997 for the thermodynamic properties of water and steam. *Journal of Engineering for Gas Turbines and Power*, vol. 122(1), pages 150–184, 2000. ISSN 0742-4795.

Wang, X.-D., An, B., Duan, Y.-Y., Wang, Z.-X. and Lee, D.-J. Efficient and accurate computation scheme of p–T thermodynamic properties of water and steam. *Journal of the Taiwan Institute of Chemical Engineers*, 2012. ISSN 1876-1070.

Waterson, N. P. Development of bounded high-order convection scheme for general industrial applications. Tech. report 1994-33, von Karman Institute for Fluid Dynamics, Sint-Genesius-Rode, Belgium, 1994.

Waterson, N. P. and Deconinck, H. A unified approach to the design and application of bounded high-order covection schemes. In *Proc. 9^{th} Int. Conf. on Numerical Methods in Laminar and Turbulent Flow*. Taylor and Durbetaki eds., Pineridge Press., Atlanta, USA, 1995.

Wedekind, G. L., Bhatt, B. L. and Beck, B. T. A system mean void fraction model for predicting various transient phenomena associated with two-phase evaporating and condensing flows. *International Journal of Multiphase Flow*, vol. 4(1), pages 97–114, 1978. ISSN 0301-9322.

Wedekind, G. L. and Stoecker, W. F. Theoretical model for predicting the transient response of the mixture-vapor transition point in horizontal evaporating flow. *Transactions of ASME Journal of Heat Transfer*, vol. 72, pages 165–174, 1965. ISSN 0961-5539.

Weustink, P. B. T., de Vries, T. J. A. and Breedveld, P. C. Object-oriented modeling and simulation of mechatronic systems with 20-sim 3.0. In *Proc. UK Mechatronics Forum's 6^{th} International Conference*, pages 873–878. Skvde, Sweden, 1998.

Whalley, P. B. *Boiling, condensation, and gas-liquid flow*. Oxford engineering science series. Clarendon Press, 1987. ISBN 978-0-19-856181-1.

Willatzen, M., Pettit, N. and Ploug-Sørensen, L. A general dynamic simulation model for evaporators and condensers in refrigeration. Part I: moving-boundary formulation of two-phase flows with heat exchange. *International Journal of Refrigeration*, vol. 21(5), pages 398–403, 1998. ISSN 0140-7007.

Winter, C. J., Sizmann, R. L. and Vant-Hull, L. L. *Solar power plants: fundamentals, technology, systems, economics*. Addison Wesley Longman Limited, 2003. ISBN 978-0-321-13304-5.

Wolfram. Mathematica 8 documentation. http://www.wolfram.com/mathematica/, 2010.

Wolfram. SystemModeler - High-fidelity modeling environment. http://www.wolfram.com/system-modeler/, 2012.

Woods, F. S. Integration under the integral sign. In *Advanced Calculus*, chapter 1, § 61, pages 145–146. Ginn, Boston, Massachusetts, USA, 1926.

Yebra, L. J. *Modelado orientado a objetos con Modelica de plantas de energía solar basadas en colectores cilindro-parabólicos*. Ph.D. Thesis, Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain, 2006.

Yebra, L. J., Berenguel, M. and Dormido, S. Extended moving boundary models for fwo-phase flows. In *Proc. 16th IFAC World Congress*. Prague, Czech Republic, 2005.

Zapata, J. I., Pye, J. and Lovegrove, K. A transient model for the heat exchange in a solar thermal once through cavity receiver. *Solar Energy*, vol. 93, pages 280–293, 2013. ISSN 0038-092X.

Zarza, E. *Generación directa de vapor con colectores cilindro-parabólicos. El proyecto DISS*. Ph.D. Thesis, Escuela Superior de Ingenieros Industriales de Sevilla, Seville, Spain, 2000.

Zarza, E., Rojas, M. E., González, L., Caballero, J. M. and Rueda, F. INDITEP: the first pre-commercial DSG solar power plant. *Solar Energy*, vol. 80(10), pages 1270–1276, 2006. ISSN 0038-092X.

Zarza, E. and Sánchez, R. Influencia de la posición que ocupan las secciones transversales de ensayos (S.T.E.) de los colectores DISS con respecto a los espejos reflectores. Tech. report DISS-SC-SF-31, Plataforma Solar de Almería, Almería, 2002.

Zarza, E., Valenzuela, L., León, J., Hennecke, K., Eck, M., Weyers, H.-D. and Eickhoff, M. Direct steam generation in parabolic troughs: final results and conclusions of the DISS project. *Energy*, vol. 29(5-6), pages 635–644, 2004. ISSN 0360-5442.

Zarza, E., Valenzuela, L., León, J., Weyers, D., Eickhoff, M., Eck, M. and Hennecke, K. The DISS project: direct steam generation in parabolic trough systems. Operation & maintenance experience and update on project status. *Journal of Solar Energy Engineering*, vol. 124, pages 126–133, 2002. ISSN 0199-6231.

Zhang, W. and Zhang, C. A generalized moving-boundary model for transient simulation of dry-expansion evaporators under larger disturbances. *International Journal of Refrigeration*, vol. 29(7), pages 1119–1127, 2006. ISSN 0140-7007.

Zimmer, D. *Equation-based modeling of variable-structure systems*. Ph.D. Thesis, Swiss Federal Institute of Technology (ETH), Zurich, 2010.

Index

9 788478 347056