Simulation and Optimization of Solar Thermal Power Plants

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Mathematiker
Diplom-Informatiker
Pascal Richter

aus Aachen

Berichter: Prof. Dr. rer. nat. Martin Frank
Prof. Dr. rer. nat. Siegfried Müller
Prof. Manuel J. Castro Díaz

Tag der mündlichen Prüfung: 17. März 2017

To Tina
And Our Kids
Leo, Fritz & Jule

Acknowledgments

There have been many people who supported and accompanied me during the development of this work.

First of all I would like to thank my advisor Martin Frank who was, from the first day, open to a self suggested topic in the area of renewable energy. He taught me how to work scientifically, kept interest in my work all the time, always had time to discuss progresses, and helped with good advice in times of setbacks – which have been a lot more often than I wished. Most of all it is his idealism which not only inspired me, but the whole group. All this makes him an outstanding advisor, maybe one of the best you could ever have.

I am very grateful to Erika Ábrahám for the exchange of insights, her confidence in the supervision of plenty of master students, and her commitment to my successful application for a scholarship. Erika supported me from the very first of my academic research. I would like to thank Manuel Castro Díaz and Carlos Parés for our conversations on path-conservative and entropy stable schemes and their hospitality during my stay in Málaga. I also want to extend my gratitude to Siegfried Müller, who helped me to find the closures out of the jungle of Baer-Nunziato type models. Especially, the joined work in Magdeburg helped me a lot.

The work also owes a great deal to the help of all my fellows at the MathCCES institute, who dedicated a considerable amount of time and patience explaining programming and math problems to me and discussing with me. In particular, I would like to thank Graham Alldredge, Thomas Camminady, Kerstin Küpper, Jonas Kusch, Philipp Otte, and Torsten Trimborn. Special thanks to my office-mate, the *Captain*, who had to withstand my course through the *PhD development curve*, with all peaks and valleys. Also thanks belongs to the former generation of colleagues – Richard Barnard, Kai Krycki, Philipp Monreal, and Edgar Olbrant – for many inspiring conversations. I will not forget the "four o'clock rule", those days it was not just a rule, it was a duty.

Thanks to Carmen-Ana Domínguez-Bravo for all the discussions we had. Maybe anytime she finds the way back to CSP such that we could follow on working together. Thanks to Matthias Cherek, Levin Gerdes, Gregor Heiming, and David Laukamp for all their time they invested in the solar project. Also, thanks to Nina Kopmann for her motivation and finally letting me win at least one bet.

Special thanks belongs to the daycares *Piccolino* and *Pusteblume* conducted by the Studierendenwerk of the RWTH Aachen University. In all, they invested 21 701 hours of patience, stress, and loving care for my children. As well, thanks to the AICES Graduate School for the financial support for child care.

I acknowledge funding by the *Friedrich-Naumann-Stiftung für die Freiheit* with financial support from the *Federal Ministry of Education and Research* of Germany. Their support started from the beginning of this project and made its realization easier or even possible. I will not forget the variety of seminars and events which have been offered.

I would like to thank my parents for their emotional support and honesty during the time of my studies. You set the initial impulse for making me sensitive and attentive to numerics and renewable energy. Dad, I could not make it faster. It was close, but still I was not fast enough. You left us too soon.

Last but not least I thank the most important persons in my life: my own family. Thanks to my life partner Tina Roeckerath who entirely supported me till the end of this project, which finally lasted nearly uncountable times "two weeks". Especially since the birth of our youngest kid Jule, Tina mainly was juggling her work and the kids. Without her encouragement I'm not sure what I'd be doing, probably something even more stressful! I'm well aware, that she surrendered her own interests to those of this project. Thanks for covering my back and for sacrificing so often our precious family time to the success of this never ending project.

I thank my three lovely children Leo, Fritz and Jule, for being with me, giving endless love and affection.

Abstract

The contribution of renewable energies to global energy use has significantly increased over the past decades – completely new industry branches have developed. Among the renewable energy technologies, concentrated solar thermal power plants are a promising option for power generation. Their basic technical idea is quite simple: Large mirrors are used to concentrate rays of sunlight on a receiver for heating up a fluid. The heat of the fluid transfers water into steam, such that the steam powers a turbine to generate electricity.

In the course of the technical progress of this young technology, permanently new issues occur. Mathematical methods and simulation sciences offer adequate techniques for understanding some of these complex processes. They can help to develop more efficient and thus more competitive solar power plants. Within this work, two problems out of the construction and operation of solar thermal power plants are regarded and are successfully solved with the help of numerics and optimization.

The first part deals with a solar tower power plant which consists of a field of hundreds or thousands of heliostats whose mirrors concentrate the direct solar radiation onto a receiver placed at the top of a tower. An open problem is to find the optimal placement of the heliostats around the tower. Because this global optimization problem has non-convex constraints a heuristic is needed to solve this problem. A forward solver is modeled as a deterministic ray-tracer using ideas from the convolution method. Due to its fast simulation speed compared to state of the art solvers, this model allows for more complex optimization techniques. Within this work, an evolutionary algorithm is developed, where modifications to the genotype representation and the evolutionary operators like recombination and mutation has been made to increase the convergence rate dramatically. Numerical results show the applicability of this approach. The optimization method developed within this work can be used to yield more efficient and thus more competitive heliostat fields. This tool was already used for the optimization of a test facility in South Africa.

In the second part, a solar thermal power plant with linear Fresnel collectors is regarded. Parallel rows of large mirrors are used to concentrate rays of sunlight on a long absorber tube of about 1000 m length. Different fluids can be used as heat transfer, e.g. thermal oil, water/steam, or molten salt. For optimal control of the power plant there is need of accurate knowledge about the ongoing processes in the absorber tubes. Here we regard the case of using water in the absorber tubes, like in the PE2 solar power plant in Spain. Current numerical approaches are lacking of necessary mathematical properties such as hyperbolicity or do not use thermodynamic properties like entropy dissipation. Mathematically, two-phase flow of water can be described by a Baer-Nunziato type PDE system. Thus, a two-velocity two-pressure seven-equations model is developed, such that several thermodynamic and mathematical properties are fulfilled. But here the problem occurs, that this system is in non-conservative form, such that appropriate numerical solvers have to be developed. Within this work, a new path-conservative entropy-preserving scheme and a Godunov solver of the Suliciu-relaxated model are developed and compared.

Contents

Preface

The exploitation of renewable energy supply is of increasing importance. Completely new industry branches have developed in the field of solar, wind, and biomass energy. Among the renewable energy technologies, concentrated solar thermal power plants are a promising option for power generation in regions with high direct solar irradiation.

The technical design of concentrated solar thermal power plants is based on a simple idea: Large mirrors are used to concentrate rays of sunlight on a receiver, such that a fluid is being heated up. Today's receiver types use thermal oil, water/steam, air or molten salt to transport the heat. The heat is used to transfer water into steam which finally powers a turbine to generate electricity.

Concentrated solar collectors for thermal power generation are subdivided into two types, with respect to the concentration principle of the sunlight:

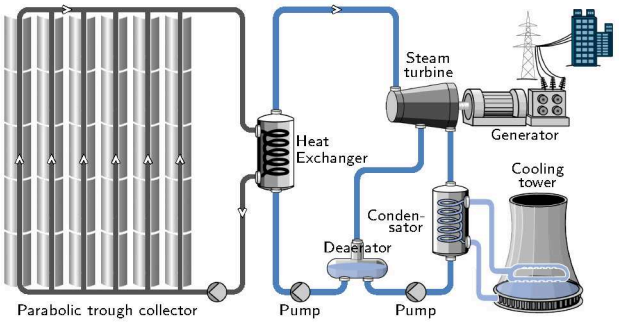
- Line-focusing systems, such as the parabolic trough collector and linear Fresnel collector. These systems track the sun position in one dimension, see Figures 1a and 1b.
- Point-focusing systems, such as the solar tower and solar dishes. These systems track the sun position in two dimensions, see Figure 1c.

From mathematical point of view, many problems show up which can be solved with the help of mathematical methods. Within this work, we regard two different problems out of the field of solar power engineering. Both problems are solved with the help of numerics and optimization.

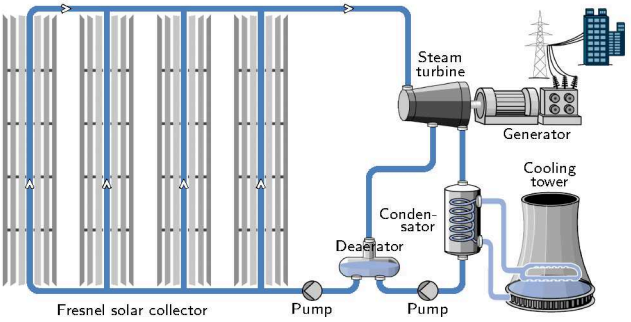
▶ Heliostat field layout optimization of solar tower power plants

Solar tower power plants consist of a receiver on top of a tower and a field of hundreds or thousands of heliostats. The heliostat field reflects and concentrates direct solar radiation onto a receiver placed at the top of the tower. At the receiver the sun light is absorbed and the resulting high-temperature thermal energy is transferred to the heat transfer fluid in order to either directly produce electricity through a conventional thermodynamic cycle or to be stored. Today four large tower plants are already operating in the US (*Ivanpah 1-3* and *Crescent Dunes*), three in Spain (*PS10*, *PS20* and *Gemasolar*) and one is under construction in South Africa (*Khi Solar One*). Numerous small-scale plants exist around the world for demonstration and research purposes (e.g. the *Solarturm Jülich* in Germany, and the facilities *CESA-1* and *SSTS-CRS* in Spain).

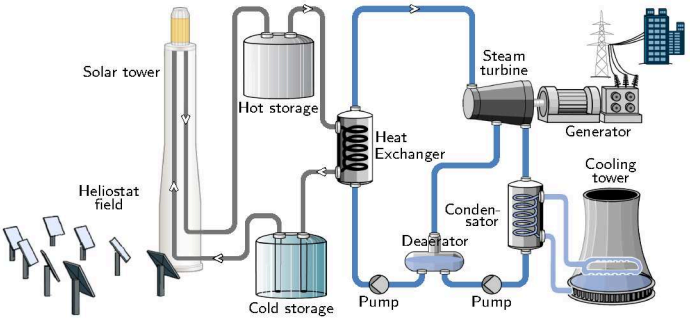
Solar tower power plants technology is very well suited for converting sunlight into dispatchable electricity. Dispatchability is important as the electricity demand hardly ever matches the production of renewable energies, such as wind and photovoltaics. While for small amounts of renewable energies the effect on the electric grid is negligible, countries with high shares of solar energy (such as Italy and Germany) face a challenge. Solar tower systems operate at high temperatures, making thermal storage systems very cost-



(a) Parabolic-trough solar collector at Plataforma Solar de Almería in Spain (left), and simplified sketch of a parabolic trough solar thermal power plant cycle using thermal oil as heat transfer fluid (right). Sources: DLR, Solar Millenium



(b) Linear Fresnel collector in 1.4 MW plant PE1 in Murcia, Spain (left), and simplified sketch of a Fresnel solar thermal power plant cycle using direct steam generation (right). Sources: Novatec Solar, Solar Millenium



(c) Solar tower plant PS10, 11 MW in Andalucía, Spain (left), and simplified sketch of a concentrated solar tower power plant cycle using molten salt and a storage. Sources: flickr, Solar Millenium

Figure 1: Concentrated solar collectors for thermal power generation.

efficient. Their storage capabilities help to even out fluctuations of other renewable plants and thus help to further increase the capacity of the non-dispatchable renewable energy technologies.

The design of the heliostat field layout is a challenging task of exceptional importance because it is the sub-system with the highest cost. Its optimal design highly depends on the specifications for each project. Note, that the position of the sun varies during a day, such that individual heliostats can be blocked or shaded by neighboring heliostats; this affects the efficiency of the power plant. As constraint of the placement we need to consider a minimum distance between neighbouring heliostats, such that they never can touch each other. As objective function usually the efficiency, received irradiation energy, thermal energy, or LCOE[1] is used, which can be provided by the most of the models in the literature, see [1]. All these models use meteorological data of a year and simulate the received power in the receiver mounted on top of the tower. The model plays a key role in the whole optimization process: first of all it should be ensured that its predictions have a high accuracy, such that in a comparison of different fields the "better" fields can be detected. Then additionally, special attention is paid to the simulation time, which finally decides about the optimization strategy. If the model is very time-consuming, then the optimization strategy should be chosen such that within a few steps an optimal solution can be found. This means, that the parameter space must be reduced.

Within this work a tool is developed from scratch which optimizes the field layout including new features and constraints as for instance tripod foundations for heliostats or multi-tower layouts. The developed optimizer has the possibility to consider pattern-based and pattern-free fields. The optimization method developed within this work can be used to yield more efficient and thus more competitive heliostat fields. This will hopefully lead to higher market penetration of this technology, benefiting the entire industry.

This task is solved in Part I, where parts of this work relies on papers and conference proceedings [2, 3, 4] which were published with coauthors. Parts of Section 2 are based on [2], which was published in the proceedings from the ECMI conference. I was mainly working on the ray-tracing model and the programing of the optimizer. The content of Sections 3 and 4 is based on [4], which was published with David Laukamp, Levin Gerdes, Martin Frank and Erika Ábrahám in the EPiC Series in Computing. I was mainly working on the different crossover strategies, their testing and comparison. Parts of Section 5 are based on [3], which was presented at the SolarPaces Conference 2016 and published in the AIP Publishing. In the paper, the layout optimization of the *Helio 100* test facility in South Africa was performed. The results from three different optimizers (each from a different institute) have been compared. I was working on the development of therein called *Aachen* optimizer to produce the shown results.

[1] Levelized costs of efficiency in Euro/kWh

▶ Direct steam generation in Fresnel solar collector systems

As second problem within this work, a Fresnel solar collector system is regarded. This technology uses parallel rows of large mirror to concentrate rays of sunlight on a long absorber tube of about 1000 m length. The receiver absorbs the heat of the sun and transfers it into thermal power using a fluid. Using water als heat transfer fluid means, that steam is directly generated in the tubes through phase change. This technique is also used in other solar thermal power plants, e.g. in line-focusing systems with Fresnel or parabolic trough collectors. Some examples for realized direct steam generating solar power plants are the *Thai Solar Energy 1 (TSE1)* parabolic trough power plant in Kanchanaburi (Thailand), the *Tubo Sol PE2* Fresnel power plant in Calasparra (Spain), and the *Solar One* tower power plant in Barstow (USA).

So far, the control of an operating direct steam generating power plants is still a challenging task. Especially during difficult cloud situations an optimal control is not known, so far. Because the processes at direct steam generation are very sensitive to local energy disturbances, strong transients can occur which are supposed to lead to super-heating events within the evaporator field. To avert a damage of the absorber tubes, the absorber tubes are completely defocused for safety reasons and energy is lost until the loop is in normal operation again. To increase the overall efficiency of a direct steam generating power plant efficient counter-acting control strategies must be found. The requirement for the development of optimal control strategies is an accurate and fast thermal-hydraulic model which predicts the ongoing processes in the absorber tubes. Current used numerical approaches are lacking of necessary mathematical properties such as hyperbolicity or do not use thermodynamic properties like entropy dissipation.

Thus, in Part II of this work a two-velocity two-pressure seven-equations model is developed which is consistent with the second law of thermodynamics and has some more mathematical properties. Parts of Section 2.5 have been published in the paper [5], which was published with Siegfried Müller and Maren Hantke in the Continuum Mechanics and Thermodynamics journal. I was mainly working on the closures of the inter-facial velocity and pressure with the usage of the entropy-entropy flux pairs, and on the entropy dissipation. The underlying model is a Baer Nunziato [6] partial differential equation system which is in non-conservative form, such that appropriate numerical solvers have to be developed. So far, all existing Godunov solvers for this problem simplified the Rankine Hugoniot conditions at the contact discontinuity of the volume fraction, e.g. [7] [8]. Within this work, a path-conservative entropy-preserving scheme and a Godunov solver of the Suliciu-relaxated model [9] are developed and compared. Both approaches have the drawback, that the acoustic waves determine the step size of the simulation time due to the CFL condition. Therefore, a semi-implicit scheme was developed, which enables us to control the approximate speeds of sound and thus, the time step. The schemes are tested successfully with multiphase flow problems from literature.

List of Figures

Nomenclature

Water properties

α	volume fraction	[–]
λ	thermal conductivity	[W/m-K]
μ	dynamic viscosity	[kg/m-s]
ρ	density	[kg/m^3]
σ	surface tension	[kg/s^2]
c	speed of sound	[m/s]
c_p	specific isobaric heat capacity	[J/kg-K]
c_v	specific isochoric heat capacity	[J/kg-K]
g	specific Gibbs free energy	[J/kg]
h	specific enthalpy	[J/kg]
p	pressure	[Pa]
$(p)_\rho$	derivative of pressure with respect to density for constant specifici inner energy	[J/kg]
$(p)_u$	derivative of pressure with respect to specific inner energy with constant density	[kg/m^3]
q	heat flux	[W/m^2]
s	specific entropy	[J/kg-K]
u	specific internal energy	[J/kg]
v	velocity	[m/s]
E	specific total energy	[J/kg]
T	temperature	[K]

Subscripts of water properties

prop_k	property of phase k, either liquid ℓ or steam g
prop_ℓ	property of liquid phase
prop_g	property of steam phase
prop_i	property at the vapour/liquid interphase
$\mathrm{prop}_\mathrm{sat}$	property of homogeneous fluid on saturation line
$\mathrm{prop}_\mathrm{satL}$	property of homogeneous fluid on saturated liquid line
$\mathrm{prop}_\mathrm{satV}$	property of homogeneous fluid on saturated vapour line
$\mathrm{prop}_{k\,\mathrm{sat}}$	property of phase k on saturation line
$\mathrm{prop}_{k\,\mathrm{satL}}$	property of phase k on saturated liquid line
$\mathrm{prop}_{k\,\mathrm{satV}}$	property of phase k on saturated vapour line
$\mathrm{prop}_{\mathrm{i}\,k}$	property of phase k molecules (droplets or bubbles) at the vapour/liquid interphase
$\widehat{\mathrm{prop}_{\mathrm{i}\,k}}$	source property of phase k molecules (droplets or bubbles) at the vapour/liquid interphase
$\mathrm{prop}_{\mathrm{i\,sat}}$	property at the vapour/liquid interphase on saturation line
$\mathrm{prop}_{\mathrm{i\,satL}}$	property at the vapour/liquid interphase on saturated liquid line
$\mathrm{prop}_{\mathrm{i\,satV}}$	property at the vapour/liquid interphase on saturated vapour line

Further greek symbols

$\alpha_{\mathrm{w}\,k}$	wetted volume fraction in the wall film of phase k	[–]
$\gamma_\mathrm{collector}$	azimuth angle of the collector-axis	[–]

γ_{solar}	solar azimuth angle	[–]
λ_{w}	tube wall thermal conductivity	[W/m-K]
ρ_{cont}	density of the continuous phase	[kg/m^3]
ρ_{w}	tube wall density	[kg/m^3]
θ	central angle between vertical and stratified liquid level	[–]
θ_{solar}	solar zenith angle	[–]
θ_{trans}	transversal sun angle	[–]
ε_{w}	surface roughness of the inner tube wall	[m]
ϑ	elevation angle of the tube with respect to the horizontal	[–]
ξ	dimensionless factor	[–]
Γ_{i}	interfacial mass flow rate in the bulk	[kg/m^3-s]
Γ_{w}	interfacial mass flow rate near the wall	[kg/m^3-s]

Further latin symbols

a_{i}	specific interfacial area	[m^2/m^3]
c_{pw}	specific isobaric heat capacity of the tube wall	[J/kg-K]
$f_{\text{w}\,k}$	Darcy friction factor of phase k with the wall	[–]
g	gravity acceleration	9.81 [m/s^2]
h_{osv}	specific enthalpy for significant net voids	[m^2/s^2]
$h_{\text{T,i}\,k}$	interfacial heat transfer coefficient of phase k	[W/m^2-K]
$h_{\text{T,w}\,k}$	wall heat transfer coefficient of phase k	[W/m^2-K]
p_{crit}	critical pressure of water	$22.064 \cdot 10^6$ [Pa]
Q_{ext}	external heat transfer rate density from the sun	[W/m^2]
Q_{loss}	losses of the heat transfer rate density at the outer tube	[W/m^2]
t	time dimension	[s]
v_{crit}	critical velocity between stratified and non-stratified flow	[m/s]
x	spatial dimension	[m]
C_{D}	drag coefficient	[–]
D_{in}	inner tube diameter	[m]
D_{out}	outer tube diameter	[m]
$F_{\text{i}\,k}$	interfacial friction density of phase k	[N/m^3]
$F_{\text{w}\,k}$	wall friction density of phase k	[N/m^3]
G	mass flux of homogeneous fluid	[kg/s-m^2]
$H_{\text{i}\,k}$	interfacial heat transfer coefficient per unit volume of phase k	[W/m^3-K]
$Q_{\text{i}\,k}$	interfacial heat transfer rate density of phase k in the bulk	[W/m^3]
$Q_{\text{w}\,k}$	wall heat transfer rate density of phase k	[W/m^3]
$Q_{\text{w}\,k}^{\text{boil}}$	interfacial heat transfer rate density of phase k near the wall due to boiling	[W/m^3]
$Q_{\text{w}\,k}^{\text{conv}}$	interfacial heat transfer rate density of phase k near the wall due to convection	[W/m^3]
Pr	Prandtl number	[–]
Ra	Rayleigh number	[–]
Re	Reynolds number	[–]
T_{crit}	critical temperature of water	647.096 [K]
T_{w}	wall temperature around the tube	[K]
$T_{\text{i sat}}$	interphasic temperature at saturation	[K]
$T_{\text{w}\,k}$	temperature in the wall film of phase k	[K]
We	Weber number	[–]

Part I.
Heliostat Field Layout Optimization

1. State of the art

For a fixed tower position $(x_t, y_t) \in \mathbb{R}^2$, a number N of heliostats and a given set $\Omega \subseteq \mathbb{R}^2$ of possible two-dimensional[2] heliostat positions, a *layout configuration* is defined as a collection of positions $\mathcal{I} = \{(x_1, y_1), \ldots, (x_N, y_N)\} \subseteq \Omega$ for the heliostat centers. To avoid collisions with neighboring heliostats, valid layout configurations must satisfy the constraints $\|(x_i, y_i) - (x_j, y_j)\| \geq d$ for each $1 \leq i < j \leq N$, where d is the expansion size (incl. safety distance) of all heliostats and $\|\cdot\|$ is the Euclidean distance. The goodness of a layout configuration $\mathcal{I}$ can be measured by some objective function $\mathcal{F}(\mathcal{I})$ (e.g. annual performance) which can be computed by an annual simulation of the sun irradiation. The *layout optimization problem* can be specified as follows:

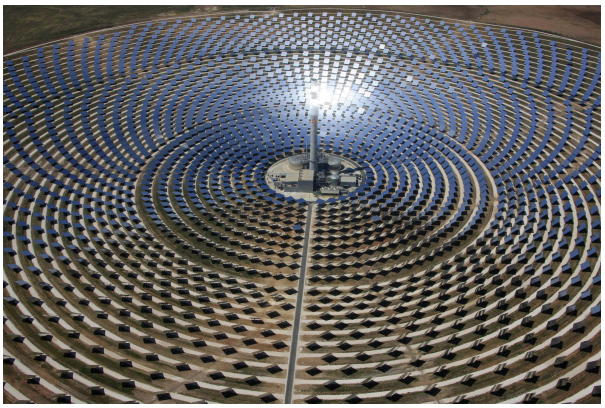


Figure 2: The solar power plant *Gemasolar* is located in the province of Seville in Spain.

Sources: Torresol Energy Investments S.A.

$$\begin{aligned} \max_{\mathcal{I}} \quad & \mathcal{F}(\mathcal{I}) \\ \textit{such that} \quad & \mathcal{I} = \{(x_1, y_1), \ldots, (x_N, y_N)\} \subseteq \Omega \textit{ and} \\ & \|(x_i, y_i) - (x_j, y_j)\| \geq d \textit{ for all } 1 \leq i < j \leq N \ . \end{aligned}$$

This global optimization problem has non-convex constraints, such that this problem may have multiple feasible regions within each region. Furthermore, this problem has multiple locally optimal points, due to the interchangeability of a pair of heliostats (x_i, y_i) and (x_j, y_j) which gives two different configrations $\mathcal{I}$ with the same goodness. Therefore it is extremely hard to find the global optimal solution. Thus, there is need of an heuristic to solve this problem in appropriate time.

In the following, existing tools for simulation of solar tower plants and optimization of the heliostat layout problem are listed. The summary is inspired by the reviews of Garcia, Ferriere and Bezian [1] and Bode and Gauché [16].

1.1. Optical models for solar tower power plants

Starting in the 1970s, several different codes have been developed which simulate the irradiation power of a central receiver system. They mainly differ in the flux calculation method, where ray tracing and mathematical simulation techniques such as Hermite

[2]The third dimension (height) is also relevant, however, for a given parcel of land on which the plant should be built the coordinate in the third dimension is determined by the coordinates in the first two dimensions.

polynomial expansion or convolution are used [17].

Monte-Carlo ray-tracers generate millions of randomized rays, where the directions of the rays are perturbed with a certain probability. This makes the obtained results very accurate. On the other hand, the calculations are computationally expensive and therefore slow.

Analytical codes consider the Gaussian distribution of the reflected sun rays analytical, such that deterministic results are obtained. To avoid the increase of computational costs, simplifications of the models are made. Depending on these simplifications, the obtained results may not be as accurate as the results obtained by Monte-Carlo ray-tracers.

▶ Ray-tracers

One of the first Monte-Carlo ray-tracers was *MIRVAL* [18]. The development of the code started in 1978 at the Sandia National Laboratories. Today, it is commercially available as *SPRAY* through the German Aerospace Center (DLR).

SolTrace [19] is a more recent tool. It is developed by the US National Renewable Energy Laboratory (NREL) since 1999 and is freely available. This Monte-Carlo ray-tracer automatically parallelizes the simulation. It returns the simulated rays for possible post-processing steps, but it is also capable of directly showing the flux distribution and obtained power.

In 2004, Manuel Blanco began the development of an open-source ray-tracer called *Tonatiuh* [20], together with teams from the University of Texas at Brownsville and the Spanish National Renewable Energy Center (CENER). *Tonatiuh* only returns the rays, i.e. the resulting power and flux distribution need to be obtained by a post-processing script [16].

The DLR recently developed a backward ray-tracer called *STRAL* [21]. The rays are generated on the mirror surface instead of a surface above the heliostat field, thus no rays get lost, such that the code is fast compared to the previously mentioned Monte-Carlo ray-tracers. This tool is capable of considering highly resolved mirror surface geometries. It is available commercially and through possible collaborations with DLR [16].

In 2011, the company Tietronix developed a tool called *TieSOL*, which is commercially available. The Monte-Carlo tool makes use of Graphic Processing Units (GPUs) to decrease the simulation time [16]. *TieSOL* has an advanced visualization tool.

HPC-SA and PROMES-CNRS developed a Monte-Carlo code *SOLFAST* (SOLar FAcilities Simulation Tools), which uses an integral formulation instead of collision-based ray-tracing. The tool was validated using *SolTrace* and *Tonatiuh* [22].

▶ Analytical simulation models

In 1974 the University of Houston started developing the code *UHC*, which is also called *RCELL suite* [1]. The software suite was used to design the Solar One tower power

plant [23]. An updated version of *UHC* is distributed commercially as *TieSOL*, see above.

DELSOL is a code from 1978, developed at Sandia [24]. The model is based on Hermite polynomial convolution. In contrast to most other first generation codes, it can optimize additional parameters such as tower height and receiver size, since it implements an economical model. There is a Windows adaption of the software called *WINDELSOL* with more features [1].

Another code developed at Sandia is *HELIOS* [25]. It is based on cone optics for flux calculation and is used where accurate flux is desired, as it uses detailed heliostat surface descriptions. The code is difficult to use and not available anymore.

The development of *HFLCAL* (Heliostat Field Layout CALculation) [26] started in the eighties at the company Interatom. Since the nineties, it is further developed and used at DLR [27], where it is also commercially available. This computationally efficient approach is based on the simplified convolution of the heliostats' flux [1]. As it was permanently improved since the eighties, *HFLCAL* has some appreciable features like automatic multi-aiming and several different receiver models with secondary concentrators etc. [27].

FIAT LUX [28] is a relatively new code, developed at CIEMAT [1]. Only few information are known about the model.

At the National Autonomous University of Mexico (Universidad Nacional Autónoma de México), the code *ISOS* was developed [29]. It calculates the 3D flux from a single heliostat, such that the flux at some height above the heliostat is known. The code requires input from an external ray-tracer [16].

HFLD, for Heliostat Field Layout Design, is a model, developed at the Chinese Academy of Sciences [30], where it is also commercially available [16]. The code is based on the edge-ray principle, i.e. exactly four rays per heliostat are generated, which makes this code sufficiently fast for the use in optimization.

The model of *CRS4-2* by the CRS4 research center is based on tessellation of the heliostats [31].

The above listed tools are either not freely available, or are Monte Carlo implementations. The problem with Monte Carlo codes is the large runtime, e.g. later it is shown that SolTrace needs about factor 100 more computation time than the model which is developed within this work. Due to the slow runtime of Monte Carlo codes it would be better to avoid them in an optimization cycle. Therefore a new tool needs to be created, which is accurate and fast enough to describe the received annual power of a solar tower power plant.

1.2. Heliostat layout optimization

Several approaches were proposed to solve the layout optimization problem, using different concepts [32]:

- The *field growth method* is a concept where the heliostats are added step by step on pre-defined points of the field. The algorithm terminates when the system requirements (e.g. minimum power output) are met. The efficiency and the runtime of this algorithm highly depends on the number of the pre-defined points of the field. Additionally, due to the successive approach each heliostat allocation depends on the preceding allocations, such that the optimization can hardly be parallelized.

  Sánchez and Romero [33] employed this concept by using a greedy heuristic. The algorithm starts with an empty field. The whole field is discretized in a set of possible points for placing heliostats. Each point in the field is evaluated, such that the points can be rated by their energy contribution. The best point is chosen as position for placing the next heliostat. Due to shading and blocking effects of the new heliostat, all free points in the field have to be evaluated again. Sánchez and Romero called this algorithm YNES, an abbreviation for *yearly normalized energy surfaces*.

- Much research has been done in the field of *pattern-based method*, where all heliostats are arranged in geometric patterns which can be described by certain adjustable parameters. With this approach the search domain is highly reduced from hundreds or thousands of x and y coordinates to a handful parameters. So, instead of optimizing the x-y coordinates, here now the pattern parameters are optimized which though influence the x-y coordinates. Thus, the pattern method essentially determines the best adaptation of the pattern for the problem and not necessarily the best x-y coordinates for optimal plant performance [32]. In literature, several different patterns have been used: rows [34], radial staggered [35], and biomimetic patterns [36]. The disadvantage of these optimizers is the small search space by construction.

- The *free variable method* follows a more classical optimization approach by directly optimizing the x-y coordinates. Due to the complexity of the problem an appropriate heuristic is needed. There exists a large variety of optimization approaches which could be used, such as non-linear programming, general gradient-based methods, to nature-inspired genetic, evolutionary, viral, simulated annealing, and particle swarm algorithms.

  So far, we just know from a gradient-based method [32] which was developed for the heliostat layout optimization problem. This approach starts with a random layout which iteratively adjusts each x-y coordinate by following the gradient in the direction of a better function value until a certain objective is achieved. The gradient of the simulation may be obtained by finding the partial derivatives of

the ray tracer function with respect to each variable. In this optimization concept the heliostats are not limited to a pattern, which means that they can freely move through the field during the optimization process.

- The *multi-step optimization strategy* consists of a combination of two or more optimization methods. First a meta-heuristic is used, able to search on a huge solution space and to move towards the global maximum. Afterwards, subordinated methods like a greedy heuristic or a linear programming refine the solution locally. The work [37] uses a pattern based optimization method and refined the results with a greedy heuristic by perturbing each heliostat position locally. This strategy has shown to give better results when compared to each of the two algorithms alone.

1.3. Related work

Despite this wide spectrum of achieved results, there is still a strong need for new approaches to solve the layout optimization problem to further improve the solutions. Because it is extremely hard to find the global optimal solution, methods from artificial intelligence may help to successfully find a good solution. In this work, we propose a classical optimization approach by using an *evolutionary algorithm* (which belongs to the above introduced class of free variable methods). We show that it is necessary to modify the crossover and mutation step, to increase the slow convergence rate. The new genetic operators are tailored to the underlying problem of two-dimensional genes (the x and y positions of the heliostats). We give a comparison of a pattern-based algorithm with an evolutionary algorithm (with classical and modified genetic operators) to show the efficiency of our approach.

The rest of this work paper is structured as follows. In Section 2 the hierarchical ray-tracer is presented. Section 3 describes the optimization algorithm using the evolutionary approach. The importance of adapting the classical operators used in evolutionary algorithms by providing some experimental results is shown in Section 4.

2. Model

A solar field consists of a solar tower and N heliostats H_i, each having a mirror area A_i. The geographical location of the tower is usually given by latitude ϕ and longitude θ. Locally a cartesian coordinate system is used, with $\boldsymbol{x}$-axis facing from west to east, $\boldsymbol{y}$-axis from south to north, and the $\boldsymbol{z}$-axis facing from ground to the sky, see Figure 3. The tower is placed at the origin.

The relative position of the sun to the geographical location is time-dependent and can be expressed by two solar angles, azimuth $\gamma_{\text{solar}}(t)$ and zenith $\theta_{\text{solar}}(t)$. The intensity of the sun is given by the time-dependent direct normal irradiation $I_{\text{DNI}}(t)$. The task of

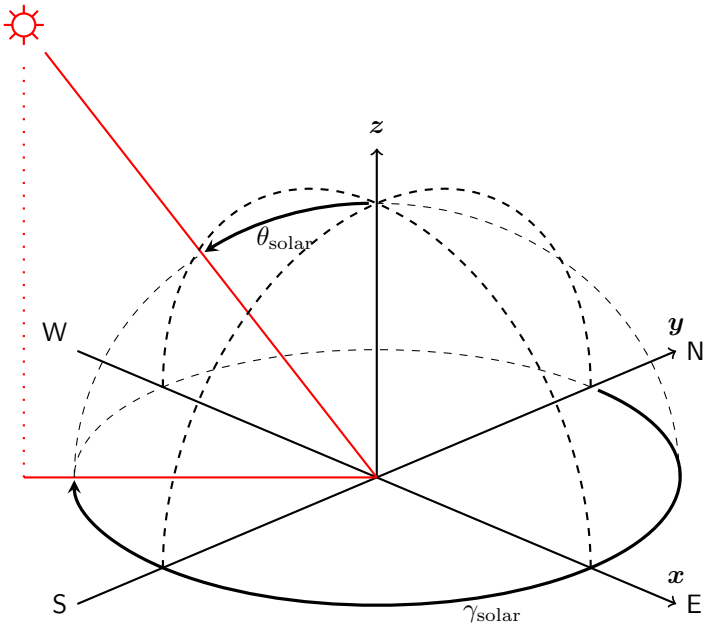


Figure 3: The solar position is given by solar zenith θ_{solar} and solar azimuth γ_{solar}.

the optical model is to compute the received energy over a year E_{year},

$$E_{\text{year}} = \int_0^{8760} P(t)\ \mathrm{d}t = \int_0^{8760} \left(\sum_{i=1}^{N} P_i(t) \right)\ \mathrm{d}t, \tag{2.1}$$

where 8760 is the number of hours in a year. For each heliostat H_i the time dependent received power is defined by

$$P_i(t) = A_i \cdot I_{\text{DNI}}(t) \cdot \eta_{\cos,i}(t) \cdot \eta_{\text{sb},i}(t) \cdot \eta_{\text{ref},i}(t) \cdot \eta_{\text{aa},i}(t) \cdot \eta_{\text{spl},i}(t), \tag{2.2}$$

while taking cosine effects $\eta_{\cos}(t)$, shading and blocking $\eta_{\text{sb}}(t)$, heliostat reflectivity $\eta_{\text{ref}}(t)$, atmospheric attenuation $\eta_{\text{aa}}(t)$ and spillage losses $\eta_{\text{spl}}(t)$ into account. All these effects are described below in more detail.

The value of the received optical radiation over a year E_{year} is the basis for each objective

function in the optimization process, see Section 3. For each different configuration of the solar field, this value has to be computed by a simulation. The time integral is solved by regarding each day separately (just the interesting part when the sun shines),

$$E_{\text{year}} = \int_0^{8760} P(t)\ \mathrm{d}t = \sum_{d=1}^{365} \left(\int_{\text{sunrise}(d)}^{\text{sunset}(d)} P(t)\ \mathrm{d}t \right), \tag{2.3}$$

and then using numerical quadrature rules. In common practice, an iteration with midpoint rule using hourly time step [24, 38, 39] is performed. For higher accuracy other numerical quadrature rules are recommended, e.g. the Gauss-Legendre quadrature rule, which uses non-constant time steps. Another possibility to reduce the number of evaluation points is to select a subset of the number of days. Each selected day is weighted with a factor (for the days which are sorted out), such that the sum of these weights is 365. In Section 2.6 an investigation of the needed number of evaluation points is done.

In the following, the above mentioned effects are developed.

2.1. Hierarchical ray-tracing method

The rays have their origin in the sun, hit the surface of a heliostat and are reflected in direction of the receiver. We are interested in the reflected power of a heliostat, which is hitting the receiver. To detect the optical flux over the heliostat's surface we are using a hierarchical approach of ray-tracing methods [36, 21], where the complete flux is computed by numerical integration with the use of Gauss-Legendre quadrature rule. Thus, each mirror surface is partitioned in a number of regions, each with a representative ray, see Figure 4. Each ray is weighted with the area of its representative region. The influence on the reflection by shading, blocking and ray interception at the receiver is determined just for this single ray as representative for the whole region. Finally all values are summed to get the power of the heliostat.

The number of representative rays per heliostat facet is given by the selected order of the Gaussian quadrature rule. Because the effect of shading and blocking is typically small and thus confined to the edges [36], it is advantageous to have a fine discretization near the heliostat edge and a coarse discretization in the middle of the surface. With the choice of Gauss-Legendre quadrature rule for placing the representative rays, this advantage is performed.

2.2. Sun position and direct normal irradiation

There exist several models, which compute the relative position of the sun according to the geographical location, e.g. [40, 41, 42, 43]. In an investigation of Armstrong and Izygon [44] it was shown that all these models show a good agreement with the high accuracy model of Meeus [43]. Now, if the solar angles are computed, the solar vector

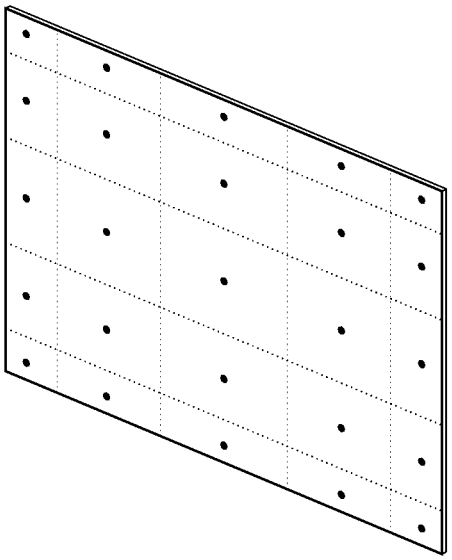


Figure 4: Numerical integration of the mirrors' flux with Gaussian quadrature rule. Each region has a representative ray, which is weighted by its area.

facing in direction of the sun is then given by

$$\boldsymbol{\tau}_{\text{solar}} = \begin{pmatrix} \sin(\gamma_{\text{solar}}) \cdot \sin(\theta_{\text{solar}}) \\ \cos(\gamma_{\text{solar}}) \cdot \sin(\theta_{\text{solar}}) \\ \cos(\theta_{\text{solar}}) \end{pmatrix}. \tag{2.4}$$

The solar radiation can be subdivided into diffuse and direct radiation. As diffuse radiation cannot be concentrated, the direct normal irradiation $I_{\text{DNI}}(t)$ is the only usable part for concentrating solar thermal power plants. The DNI can either be determined by meteorological models like the clear sky model MRM [45], or given by some measurement data (e.g. EnergyPlus[3] or Meteonorm TMY3[4]), see for instance Figure 5.

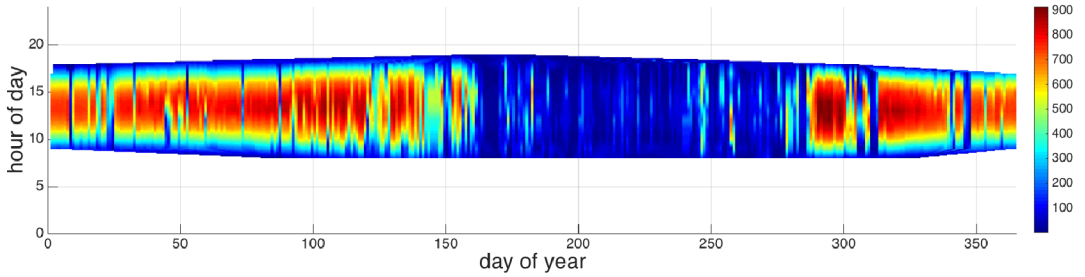


Figure 5: TMY data of the DNI (in Watt per square meter) of a place near Mumbai (India) at a latitude of 18.97° and a longitude of 72.83°. The monsoon period around July is easy to detect.

[3] `https://energyplus.net/weather`
[4] `http://www.meteonorm.com/`

2.3. Tower and receiver

The tower is placed at the origin, where the receiver is mounted on top of the tower. The position of the position is labeled with $\boldsymbol{p}_{\text{tower}}$. The sun rays are collected in the receiver, which transfers the radiation into heat. In our model we distinguish between three concepts:

- **Flat tilted cavity receiver.** This type represents a cavity or volumetric receiver, which can be found for example in the CESA-1 central receiver facility in Andalusia, Spain or the solar tower Jülich in Germany. The receiver is modeled as a bounded plane in the $\boldsymbol{x}$-$\boldsymbol{z}$ plane, which is tilted by a zenith angle θ_{rec} in the $\boldsymbol{y}$ direction. The receiver has a width of w_{rec} and height h_{rec}, see Figure 6a.

- **Cylindric cavity receiver** An internal cylindric cavity receiver has the form of a half-cylinder, such as the PS10 receiver in Andalusia, Spain. The receiver has a diameter of d_{rec} in the $\boldsymbol{x}$-$\boldsymbol{y}$ area and a height of h_{rec}, see Figure 6b.

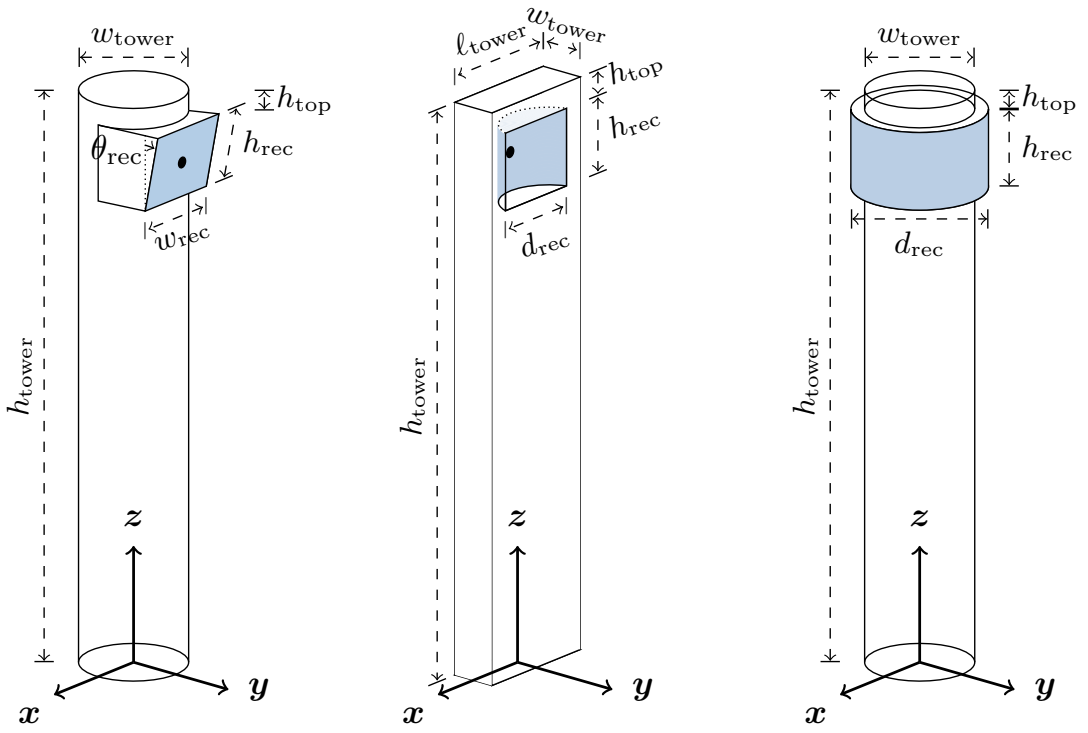
- **Cylindric external receiver** This type represents a 360° external receiver, which can be found for example in the Solar One and Solar Two central receiver facilities at Barstow in California, USA or the 19 MW plant Gemasolar in Andalusia, Spain. The receiver is modeled as a curved surface area of a cylinder with diameter d_{rec} in the $\boldsymbol{x}$-$\boldsymbol{y}$ area and height h_{rec}, see Figure 6c.

For the first two receiver types, the model assumes that each heliostat H_i aims towards the center of the aperture. For the cylindrical external receiver the model assumes that each heliostat H_i aims towards the closest point at the center of the aperture. The aiming point in the receiver is labeled with $\boldsymbol{p}_{\text{rec},i}$. Usually, the aiming points are not fixed for the whole time: Strategies distribute the flux over the full receiver area, to avoid dangerous flux levels at the center.

2.4. Heliostats and pod systems

The heliostats are tracking the sun position, to concentrate the sun light on a central, tower-mounted receiver. Each heliostat H_i is raised on a pedestal. It's mirror center-position is called $\boldsymbol{p}_i$. A heliostat can consist of many small mirrors, called facets, which are arranged horizontally and vertically on a mirror frame, see Figure 7. Each facet of a heliostat has the same length and width, where the overall mirror area A_i of heliostat H_i is then given as sum of all facet areas. Between the facets there can be horizontal and vertical gaps. For an overall width w_i and length ℓ_i the heliostat's expansion d_i is the diameter of the minimum bounding sphere, $d_i = \sqrt{\ell_i^2 + w_i^2}$.

The heliostat facets can either be flat or focused with a focal length f_i. Usually the focal length is chosen such that it corresponds to the distance of the heliostat position to the aiming point on the receiver.
The positions and alignment of the heliostat's facets are saved in terms of local heliostat coordinates, with axes $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ and origin $\boldsymbol{p}_i$. The $\boldsymbol{z}$-axis is the normal vector $\boldsymbol{n}_i$ of



(a) Flat tilted cavity receiver (b) Cylindric cavity receiver (c) Cylindric external receiver

Figure 6: Different receiver types

the heliostat scaffold.

For security reasons or to make sure every heliostat is accessible for cleaning and maintenance, it may be desired to have a minimal distance between two heliostats, which is at least the diameter of the heliostats, such that they do not touch.

The heliostats can be placed everywhere in a given area Ω. This area considers also places where no heliostats are allowed to construct, e.g. if service roads are needed, or a pipeline crosses the area (which is the case for the solar tower in Jülich).

2.4.1. Heliostat canting

On a large heliostat, the facets are positioned and aligned in a certain way on the heliostat scaffold. This is called canting. There are different approaches. In our model, on-axis canting and off-axis canting are implemented.

▶ On-axis canting

With on-axis canting, the facets are aligned, such that they perfectly reflect the sun rays for the case that the sun shines out of the receiver. Heliostat, receiver and sun are then on one common axis. The heliostat facets are then positioned around the symmetry axis of a paraboloid, where the receiver is the focus.

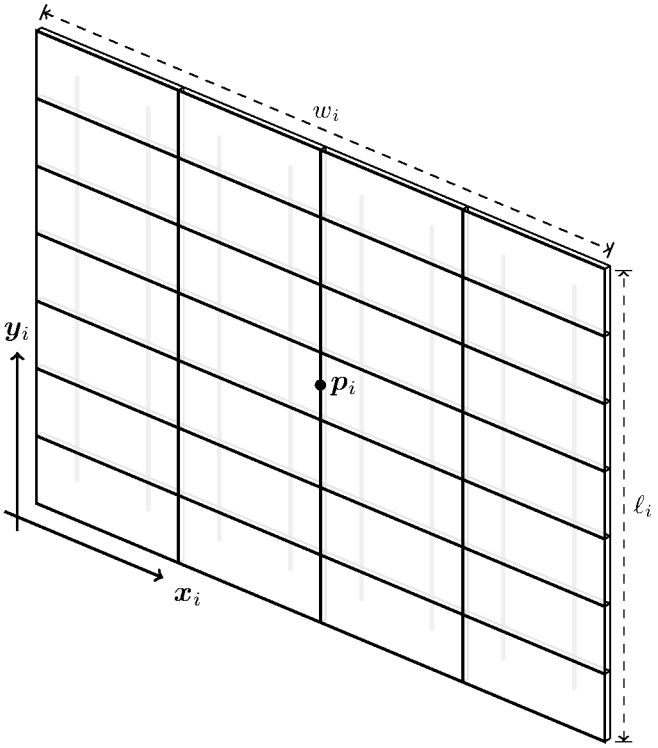


Figure 7: Local heliostat coordinate system of heliostat H_i with axes $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ and origin $\boldsymbol{p}_i$.

▶ Off-axis canting

Off-axis canting means, the facets are aligned for perfect reflection at the design point, which can be defined by date and time or by the sun position (azimuth and zenith). In this case, the heliostat position is not on the sun-receiver axis (off-axis) and the facets are therefore positioned on the side of a paraboloid.

2.4.2. Alignment and tracking

In the model the heliostats are aligned in such a way that they aim the reflected ray through $\boldsymbol{p}_i$ always towards the aiming point $\boldsymbol{p}_{\text{rec},i}$ in the receiver aperture. Thus, the normalized reflective vector is defined by

$$\boldsymbol{r}_i = \frac{\boldsymbol{p}_{\text{rec},i} - \boldsymbol{p}_i}{|\boldsymbol{p}_{\text{rec},i} - \boldsymbol{p}_i|}. \tag{2.5}$$

Following the law of reflection, the normal vector $\boldsymbol{n}_i$ of heliostat H_i is given in dependency of the incoming and outgoing vectors

$$\boldsymbol{n}_i = \frac{\boldsymbol{r}_i + \boldsymbol{\tau}_{\text{solar}}}{|\boldsymbol{r}_i + \boldsymbol{\tau}_{\text{solar}}|}. \tag{2.6}$$

The local coordinate system of the aligned heliostat is then determined by

$$\boldsymbol{x}_i := \frac{\boldsymbol{n}_i \times (0,0,1)^\mathsf{T}}{|\boldsymbol{n}_i \times (0,0,1)^\mathsf{T}|}, \quad \boldsymbol{y}_i := \boldsymbol{n}_i \times \boldsymbol{x}_i, \quad \text{and} \quad \boldsymbol{z}_i := \boldsymbol{n}_i \tag{2.7}$$

Notice that just the heliostat scaffold is aligned – the single facets are fixated on the scaffold of the heliostat.

2.4.3. Clustering in pod systems

The heliostats can be grouped by a joint pod system, where they are positioned on an arbitrary truss construction, see Figure 8. So, instead of positioning single heliostats, groups of heliostats with fixed relative positions are placed on the field. A pod system S_i is characterized by a truss construction, truss length, center position $\boldsymbol{p}_i$ and a rotation angle α_i. The rotation angle can be restricted to a periodic rotation angle, e.g. a triangular pod may only be rotated in steps of 60 degrees. The pod systems are not allowed to touch each other, this includes all heliostats and the truss construction.

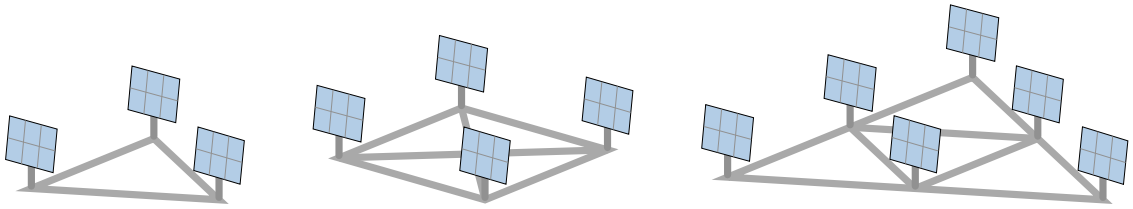


Figure 8: Examples of different pod systems to group heliostats on a truss construction.

2.5. Efficiencies and losses

2.5.1. Cosine effects

The heliostats are tracking the sun in such a way that the rays are reflected on the surface to hit the receiver. Due to the tilt of the heliostat surface, the projected area is reduced. This effect is called cosine effect and is numbered as $\eta_{\cos}$. Cosine effects depend on the solar position and the alignment of the heliostat. Related to the law of reflection, the heliostat surface normal bisects the angle between the solar rays and a line from the heliostat to the tower [17]. Thus, the effective reflection area of the mirror is reduced by the the angle of incidence [36], so that

$$\eta_{\cos,i} = \langle \boldsymbol{\tau}_{\text{solar}}, \boldsymbol{n}_i \rangle. \tag{2.8}$$

2.5.2. Shading and blocking

For each heliostat, a number of rays determine if a region of a heliostat is blocked or shaded by neighbouring heliostats, the tower, or the terrain. This is the most expensive part of a simulation. Especially Monte Carlo tools therefore have to use a high number

of rays. With the herein used hierarchical approach of a ray-tracing method, for each ray shading and blocking effects must be detected.

▶ Tower shadow

The tower is assumed to be a cuboid or cylinder. Each ray of a heliostat must be checked, if it hits the tower shape. A subset of potentially tower-shaded heliostats can be computed by selecting just those heliostats, which are placed in a simplified shadow of the tower, a corridor with the width of the tower's expansion facing in opposite direction of the sun. Just these few heliostats in this corridor have to be checked for tower shading. Therefore the minimum distance between the tower center (represented as a line through the tower position $\boldsymbol{p}_{\text{tower}}$ facing into the sky) and an incoming sun ray through the mirror center $\boldsymbol{p}_i$ has to be computed. If this distance is greater than half of tower expansion plus the half of heliostats expansion, then the whole mirror is not shaded by the tower. This analysis corresponds to an examination, if two infinite long cylinders touch each other. One cylinder is the heliostat in direction of the sun with a diameter of the heliostats expansion, and the other cylinder is the tower with a diameter of the tower expansion. If a heliostat is possibly shaded, then each single ray of a heliostat has to be checked.

▶ Heliostat shading

The algorithm for detecting shading heliostats is similar as for tower shading. For each heliostat H_i, a subset of potentially shading heliostats can be computed by selecting all heliostats which are placed in a corridor starting from the heliostat and facing in opposite direction of the sun. The corridor has the width of the heliostat's expansion. Just these few heliostats H_j of this subset have to be checked for shading. Therefore the minimum distance between the mirror center $\boldsymbol{p}_j$ of each neighboring mirror H_j and an incoming sun ray through the mirror center $\boldsymbol{p}_i$ has to be computed. If this distance is greater than half of heliostat H_i expansion plus the half of heliostat H_j expansion, then the whole mirror is not shaded by heliostat H_j.

This analysis corresponds to an examination, if two infinite long cylinders touch each other. Both cylinders are parallel and facing in direction of the sun. One is running through $\boldsymbol{p}_i$ with a diameter of heliostat H_i expansion, the other one is running through $\boldsymbol{p}_j$ with a diameter of heliostat H_j expansion

If a heliostat H_i is possibly shaded by another heliostat H_j, then each single ray of a heliostat has to be checked. Therefore the line-plane intersection point of the incoming sun ray and the neighboring heliostat's normal plane with $\boldsymbol{n}_j$ has to be computed. If the hit point lies inside the heliostat's borders, the heliostat H_i is partially shaded in the ray's represented region.

▶ Heliostat blocking

The algorithm for detecting blocking heliostats is similar as for heliostat shading. But

instead of taking the sun vector as direction, here the reflected ray with target vector $\boldsymbol{p}_{\mathrm{rec},i} - \boldsymbol{p}_i$ is used.

▶ Terrain blocking and shading

On hilly terrain, it is possible that rays are shaded or blocked by the terrain. To efficiently detect terrain shading and blocking, a multi-step filter and refinement algorithm is used. The idea is to first compute for each heliostat the set of terrain grid cells that can potentially shade (or block) rays, and then perform a fast intersection check for each ray with these candidate cells only.

In the first step, a 2-D corridor starting from each heliostat in direction of the sun (or the tower, respectively) is considered, where the width of the corridor is given by the heliostat expansion. The resulting set of candidate grid cells is refined using a 3-D filter by placing a capsule around each candidate grid cell and checking whether the capsule intersects the ray cylinder from the heliostat to the sun (or tower). This intersection test boils down to detecting the closest points of the ray cylinder axis and the capsule axis. If the distance between the closest points is larger than capsule radius plus ray cylinder radius, the cell does not shade or block a ray from the ray cylinder, and it can be removed from the set of candidates, see Figure 9. Only these grid cells that cannot

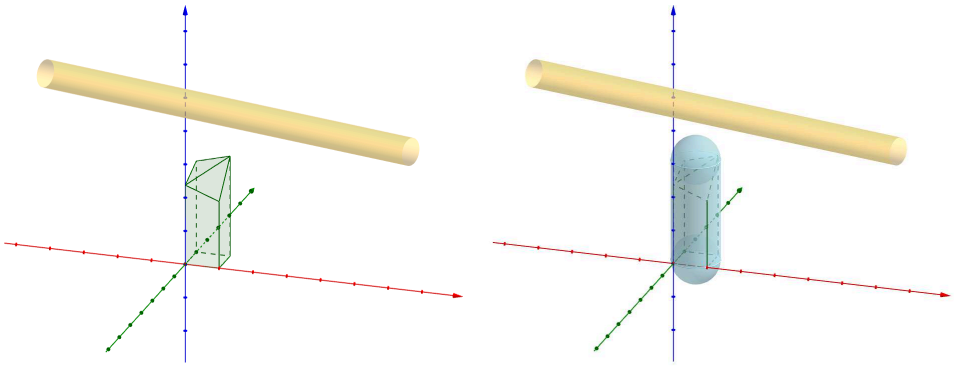


Figure 9: 3-D filter to refine candidate cells, depicting an isolated cell; (left) grid cell with the true terrain, (right) bounding capsule

be pruned using these two filters are used to check the actual intersections between individual rays and the terrain. Candidate cells between heliostats and the tower are the same for every simulated moment, so they have to be computed only once for each heliostat before simulating. Candidate cells between heliostats and the sun are different for every simulated moment, since the sun moves over the terrain. They have to be computed for every heliostat and every moment.

After having determined the sets of possibly shading and blocking cells, individual rays need to be checked for intersection with the terrain from each candidate grid cell (in 3-D). Again, this is done in two steps. First, a bounding box is placed around the grid cell, and the intersection between ray and bounding box is tested. If the ray does not hit the bounding box, it does not hit the terrain and the intersection test terminates. If,

however, the ray hits the bounding box, two cases can occur: (i) one of the two hitting points lies below the actual terrain at the cell boundary, (ii) both hitting points lie above the actual terrain, see Figure 10. In case (i), the ray definitely hits the terrain and the test terminates. In case (ii), it has to be additionally checked whether the ray hits one of the two triangles obtained from triangulating the four grid cell corner points.

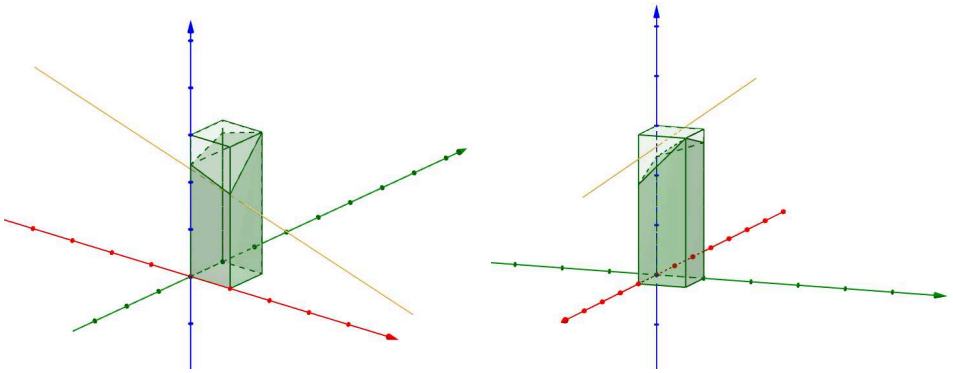


Figure 10: 3-D intersection test between rays and terrain cells; (left) both hitting points lie below the terrain at the grid cell boundary, (right) both hitting points lie above the terrain at the boundary and the ray does not intersect one of the two triangles.

2.5.3. Heliostat reflectivity

At the surface of a mirror the rays from the sun are reflected in direction of the receiver. But some radiation is scattered in a wrong direction due to slightly cleanliness or absorbance at the surface of the mirror. The reflectivity of a mirror depends on the incidence angle and the solar spectrum. But in the open literature, often a constant value is used for the heliostat reflectivity, e.g. [46]

$$\eta_{\mathrm{ref},i} \equiv 0.87, \tag{2.9}$$

which means that 87% of the energy is reflected as mentioned and 13% is lost at the surface of the heliostat. In our model the reflectivity can be set to be dependent on the incidence angle.

2.5.4. Atmospheric attenuation efficiency

The atmospheric attenuation efficiency considers the effect that the atmosphere transmits progressively less light. This radiation loss depends on the distance d_i between the heliostat H_i and the receiver aim point,

$$d_i = |\boldsymbol{p}_i - \boldsymbol{p}_{\mathrm{rec},i}|. \tag{2.10}$$

Leary and Hankins [18] adopted a simple formula for a distance of less than 1000 meter. This formula was extended for larger distances by Schmitz et al. [47], with the goal to

agree well with the model of Pitman and Vant-Hull [48] for a visual range of about 40 km:

$$\eta_{\mathrm{aa},i} = \begin{cases} 0.99321 - 1.176 \cdot 10^{-4}\ d_i + 1.97 \cdot 10^{-8}\ d_i^2 & ,\ d_i \leq 1000\,\mathrm{m} \\ \exp(-1.106 \cdot 10^{-4}\ d_i) & ,\ d_i > 1000\,\mathrm{m} \end{cases}. \tag{2.11}$$

2.5.5. Optical errors

Each rays's origin is the sun. When a ray hits a heliostat at a certain point, the insolation angle with the surface is not unique due to the sun shape, see Figure 11. This perturbation (compared to an ideal ray coming from the center of the sun) is amplified by reflection on the heliostat's surface due to tracking errors. Also slope errors could appear, where an irregular surface can cause a ray reflection in a different direction.

The sun shape is modeled as an angular Gaussian distribution with a standard deviation of σ_{sun}, based on the idea of Rabl [49]. The distribution is describing the probability that a ray has its origin in a certain location of the sun. Also the tracking and slope errors with a standard deviation of $\sigma_{\mathrm{tracking}}$ and σ_{slope} are modelled as a Gaussian distribution. The deviations are combined, using the Euclidean norm,

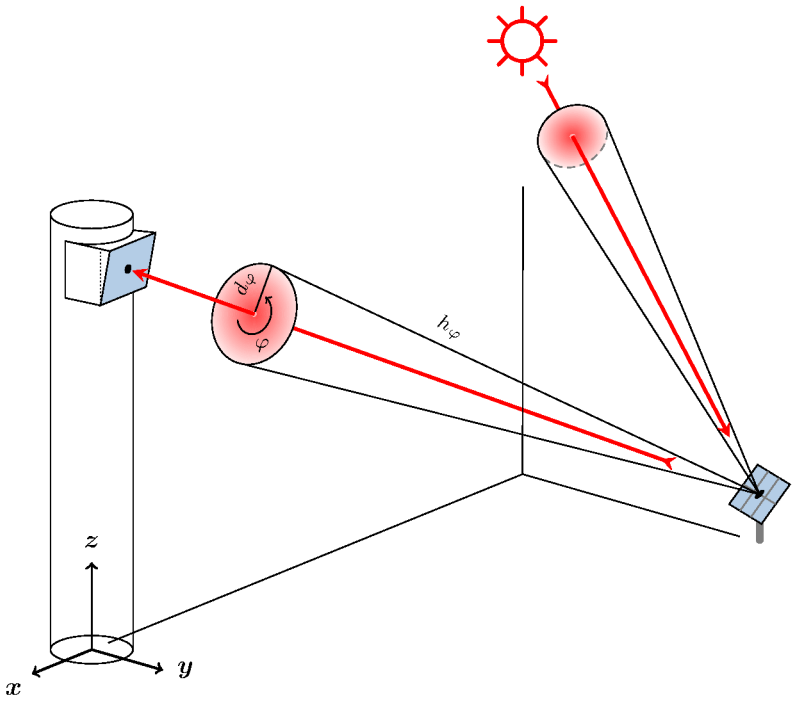


Figure 11: Error cone

$$\sigma_{\mathrm{beam}} = \sqrt{\sigma_{\mathrm{sun}}^2 + \sigma_{\mathrm{tracking}}^2 + \sigma_{\mathrm{slope}}^2}. \tag{2.12}$$

Note that the tracking error appear to both axes. So, the tracking error is a combination of errors in azimuthal and elevation direction using the Euclidean norm. As error angles, the following parameters are used [50, 51]

$$\sigma_{\text{sun}} = 2.35 \text{ mrad}, \quad \sigma_{\text{tracking}} = \sqrt{0.919^2 + 0.919^2} = 1.3 \text{ mrad} \quad \text{and} \quad \sigma_{\text{slope}} = 2.6 \text{ mrad}, \tag{2.13}$$

which results in a standard deviation of

$$\sigma_{\text{beam}} = 3.738 \text{ mrad}. \tag{2.14}$$

2.5.6. Interception efficiency

The interception efficiency of a ray is the probability that the ray hits the receiver. The ray, represented as an error cone, causes a flux at the receivers' surface around it's ideal hit point, see Figure 12. The interception efficiency (or often called spillage losses) of a ray is described by a two-dimensional integral of the standard normal distribution,

$$\eta_{\text{spl}}^{\text{ray}} = \frac{1}{2\pi\sigma_{\text{beam}}^2} \iint \exp\left(-\frac{x^2+y^2}{2\sigma_{\text{beam}}^2}\right) \, \mathrm{d}x \, \mathrm{d}y. \tag{2.15}$$

Notice that here the parameters x and y are the distances from the ideal ray in horizontal and vertical direction, measured as multiples of σ_{beam}. By converting into polar coordinates with $x = r\cos\varphi$ and $y = r\sin\varphi$, we get

$$\eta_{\text{spl}}^{\text{ray}} = \frac{1}{2\pi\sigma_{\text{beam}}^2} \int_0^{2\pi} \int_0^{r_\varphi} r \cdot \exp\left(-\frac{r^2}{2\sigma_{\text{beam}}^2}\right) \, \mathrm{d}r \, \mathrm{d}\varphi. \tag{2.16}$$

Also here, the radius r is the distance from the ideal ray, measured as multiples of σ_{beam}. As upper integration limit r_φ of the radius, the border of the receiver is used, which depends on the radial direction φ. This ensures that for a given φ all perturbed rays with $r \in [0, r_\varphi]$ are hitting the receiver. r_φ is given as arcus sinus of the minimum distance d_φ from the ideal ray to the receiver border point, divided by the slant height h_φ of the error cone through the border point, see also Figure 11,

$$r_\varphi = \operatorname{asin} \frac{d_\varphi}{h_\varphi}. \tag{2.17}$$

The inner integral of equation (2.16) can be computed analytically, where the outer integral is solved numerically with the use of the trapezoidal rule using n evaluation points:

$$\begin{aligned} \eta_{\text{spl}}^{\text{ray}} &= \frac{1}{2\pi} \int_0^{2\pi} \int_0^{r_\varphi} \frac{r}{\sigma_{\text{beam}}^2} \cdot \exp\left(-\frac{r^2}{2\sigma_{\text{beam}}^2}\right) \, \mathrm{d}r \, \mathrm{d}\varphi \\ &= \frac{1}{2\pi} \int_0^{2\pi} \left(-\exp\left(-\frac{r^2}{2\sigma_{\text{beam}}^2}\right)\Big|_0^{r_\varphi}\right) \, \mathrm{d}\varphi \end{aligned}$$

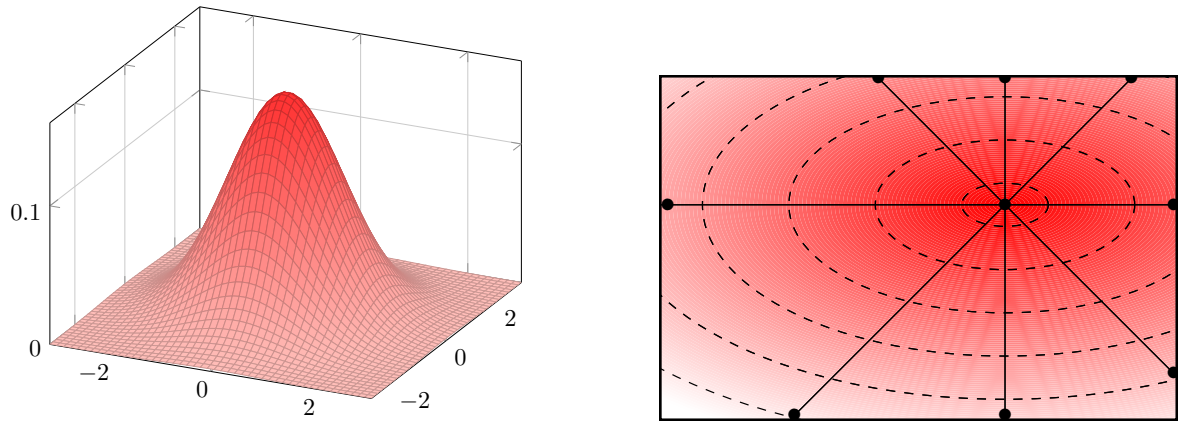


Figure 12: Gaussian normal distribution on a receiver

$$
\begin{aligned}
&= \frac{1}{2\pi} \int_0^{2\pi} \left(1 - \exp\left(-\frac{r_\varphi^2}{2\sigma_{\text{beam}}^2}\right)\right) \, \mathrm{d}\varphi \\
&= 1 - \frac{1}{2\pi} \int_0^{2\pi} \exp\left(-\frac{r_\varphi^2}{2\sigma_{\text{beam}}^2}\right) \, \mathrm{d}\varphi \\
&\approx 1 - \frac{1}{n} \cdot \sum_{j=0}^{n-1} \exp\left(-\frac{r_{\varphi_j}^2}{2\sigma_{\text{beam}}^2}\right)
\end{aligned} \tag{2.18}
$$

with equidistant angles $\varphi_j = j \cdot \frac{2\pi}{n}$. For non-equidistant angles $\varphi_j < \varphi_{j+1}$, $j = 0, \ldots, n-1$, the efficiency is approximated by

$$
\eta_{\text{spl}}^{\text{ray}} \approx 1 - \frac{1}{2\pi} \cdot \sum_{j=0}^{n-1} \frac{\varphi_{j+1} - \varphi_j}{2} \cdot \left(\exp\left(-\frac{r_{\varphi_j}^2}{2\sigma_{\text{beam}}^2}\right) + \exp\left(-\frac{r_{\varphi_{j+1}}^2}{2\sigma_{\text{beam}}^2}\right)\right), \tag{2.19}
$$

where $\varphi_n := \varphi_0 + 2\pi$ and $r_{\varphi_n} := r_{\varphi_0}$. Note that rays that shortly miss the receiver are not counted at all, although parts of their error cones would hit the receiver.

2.6. Reducing computational complexity

▶ Numerical approximation of the time integral

As shown in Equation (2.3), the time integral can be approximated by considering different amounts of days in a year and using quadrature rules with different amounts of time steps per day.

The time integral in Equation (2.3) is solved numerically. In common practice, an iteration with constant time step [38, 24] is used, which corresponds to the midpoint rule. Noone et al. [36] propose an iteration with constant solar angle step, which allows the same accuracy with fewer iterations. There exist other approaches which just regard the sun angle instead of the time integral [52]. In our model, different quadrature methods can be used with either time or solar angle as integration variable, where the Gauss-Legendre quadrature gives the most promising results. An investigation regarding the number of simulated points in a year was performed. The relative error (in %) of the computed annual performance for a place in Almerá (Spain) is shown in Figure 13. After this investigation, for our simulations we use 30 days in a year each evaluated at 5 instants of time per day.

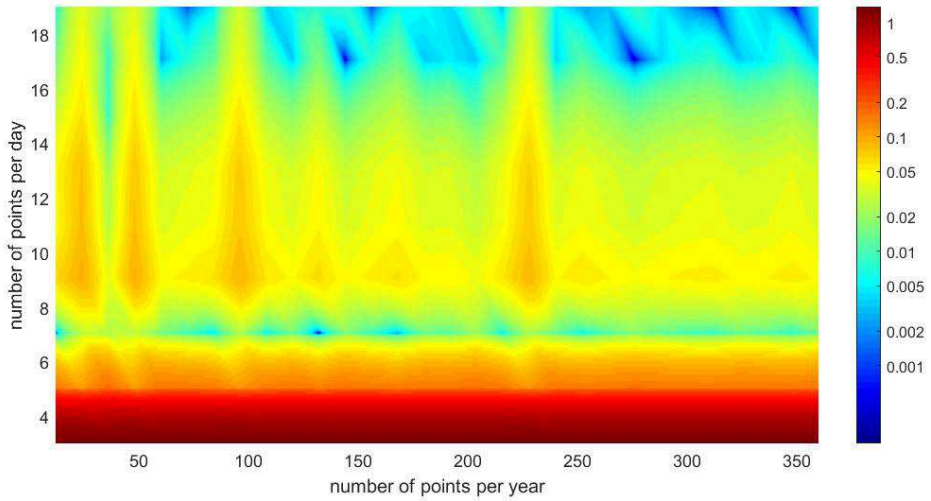


Figure 13: Accuracy for computing the annual energy using Gaussian quadrature rule, with different amounts of days and time points. The colors represents the relative error (in %) of the annual performance related to the reference solution, which considers about every minute of the year (every day of the year, each with 1000 points).

▶ Improving computational time for shading & blocking effects

For each ray it must be determined if is shaded or blocked by neighboring heliostats. The brute-force approach of a pairwise comparison of each ray with all heliostats is computationally expensive. Therefore a faster approach is needed.

One possibility is to consider just a subset of heliostats that can potentially shade or block a heliostat. To determine this subset, a data structure is needed which is fast in nearest-neighbor search and in range-search. Therefore, for better performance, a two-dimensional bitboard index structure is used. The idea is to cover the two-dimensional $\boldsymbol{x}$-$\boldsymbol{y}$ space with an equidistant grid such that the space is sub-divided in distinct quadratic cells. Inside these cells the information is stored if nearby is a heliostat, see Figure 14.

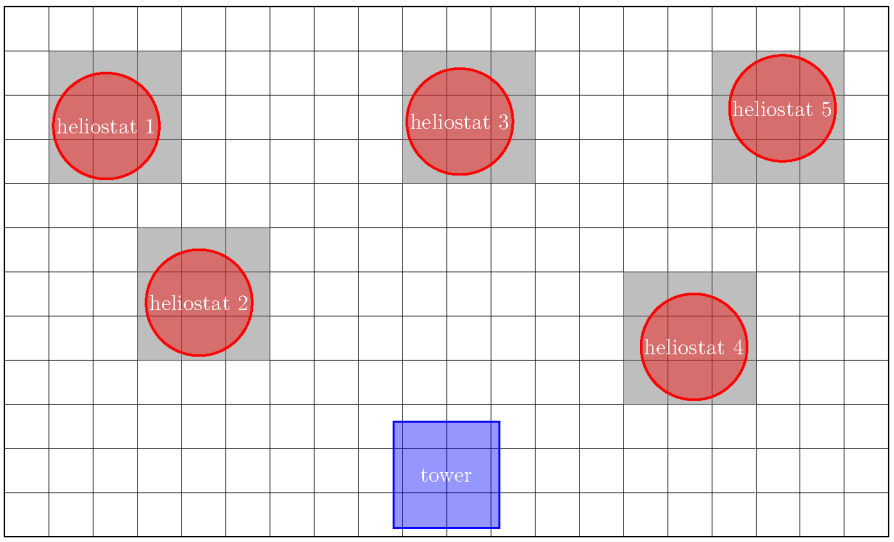


Figure 14: Bitboard index structure: In each cell the information is stored, if nearby is a heliostat. Here the grey cells have such an information, the white cells are empty.

For a nearest-neighbor search, just the surrounding cells around a cell have to be checked, instead of all heliostats. The same holds for a range-search, where e.g. all heliostats in one direction are wanted. Just the containing cells of the range have to be checked. The questin now is, how to choose the size of the quadratic cells. If the size is too small, the search time is very high because of checking too many cells. On the other hand, if the cell size is too big, too many heliostats are stored in the cells such that the advantage of a bitboard field is lost and we are ending up in a pairwise comparison. The optimal grid size was investigated, again using the solar tower power plant PS10 as reference field. The benefit of using such an index structure is analyzed in Figure 15. Best results are reached with a cell size which has approximately the heliostats size. With this cell size, the blocking and shading method can be accelerated by up to factor 10.

▶ Cross-Validation of the model

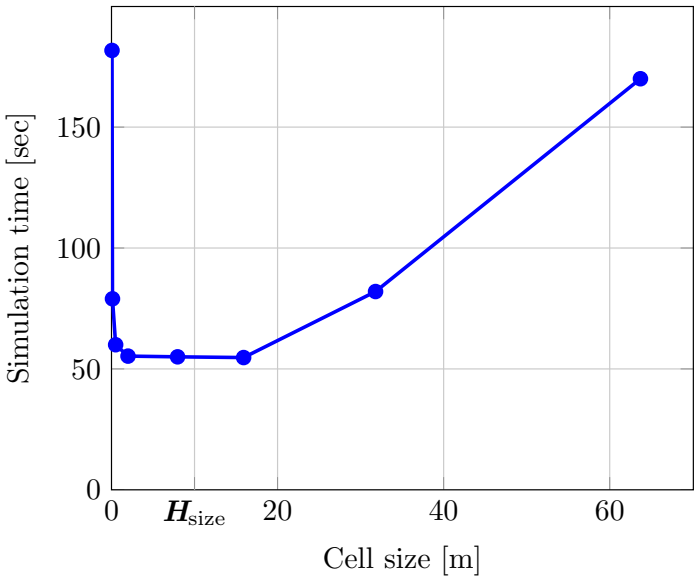


Figure 15: Comparison of different cell sizes of the bitboard index structure.

The model is compared to the Monte Carlo ray-tracing tool SolTrace [19], see Figure 16. Using once again the original PS10 heliostat field layout, the accuracy vs. the needed number of rays (measured in runtime) is investigated, see Figure 17. To compute the accuracy, the solutions are compared to a reference solution, which was computed with 10 million rays in SolTrace. It can be seen, that SunFlower is at least 10 times faster than SolTrace at the same accuracy.

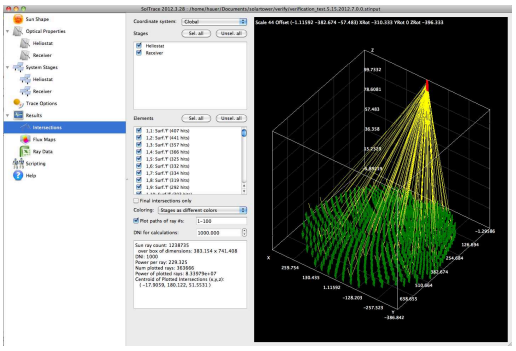


Figure 16: The tool SolTrace is a Monte Carlo ray-tracer which needs millions of rays for enough accuracy.

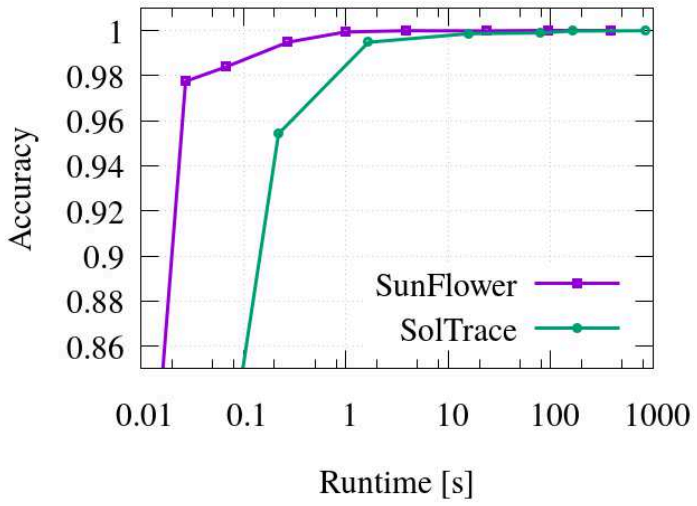


Figure 17: Comparing the accuracy of SolTrace and SunFlower for the PS10 solar tower power plant: The reference value corresponds to the SolTrace result with 10 million rays. For this test case, solutions for a different amount of rays is used (measured in time), where accuracy (in %) is defined as 1 minus the relative error.

Figure 18 shows the projection of one heliostat of 1.83 m $\times$ 1.22 m onto a 4 m^2 receiver. The model here uses 50 $\times$ 50 rays, while SolTrace simulated the same configuration using a million rays. The previously described Gauss-Legendre ray distribution does not change with optical errors as it does in SolTrace. Optical errors are considered by assuming error cones instead of simple rays. The obtained flux from the same test case is presented in Figure 19 for the case, where no optical errors are considered. Figure 20 shows the flux maps for the test case with considering optical errors. The test cases show good agreement of both models in the eyeball norm.

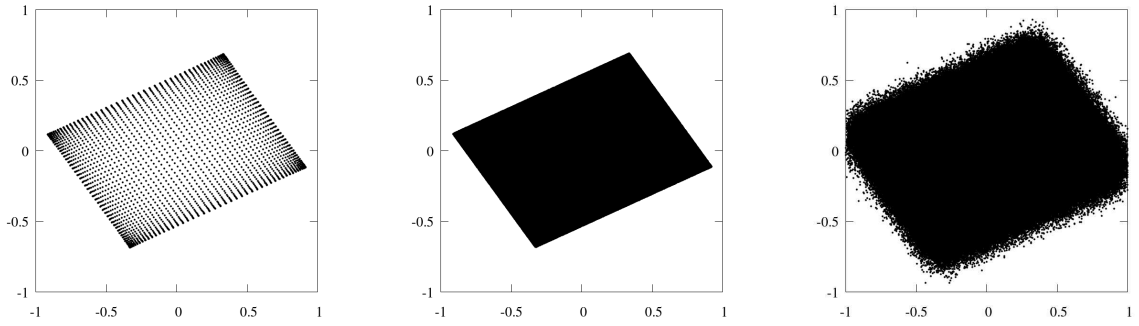


Figure 18: Ray distribution on the receiver in a test case with one single heliostat obtained by the SunFlower model (left) and SolTrace without optical errors (center) and SolTrace with optical errors (right).

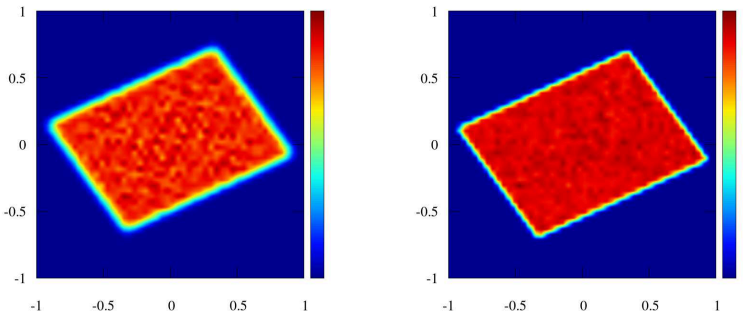


Figure 19: Flux maps without considering optical errors in the model (left) and SolTrace (right).

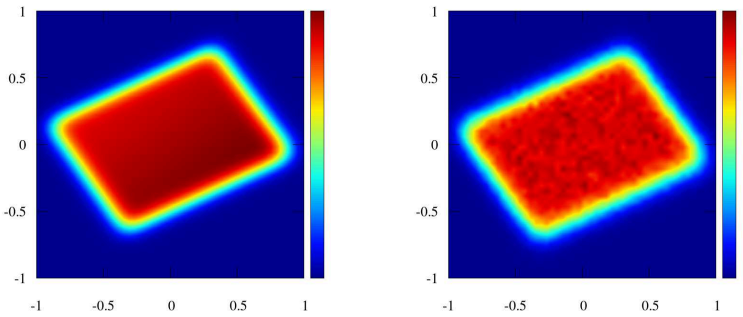


Figure 20: Flux maps with considering optical errors in the model (left) and SolTrace (right).

2.7. Web application

The model is available as a web application on `www.solar.rwth-aachen.de`, see Figure 21. Here the user can simulate a solar tower plant, i.e. either calculating the power at one moment or calculating the annual energy. Some additional features are supported, such as loading the topography from google maps, different meteorological models or importing measured irradiation data, scalable spatial and temporal resolutions, several heliostat canting methods, export tool for SolTrace, 3D visualizations with WebGL, etc.

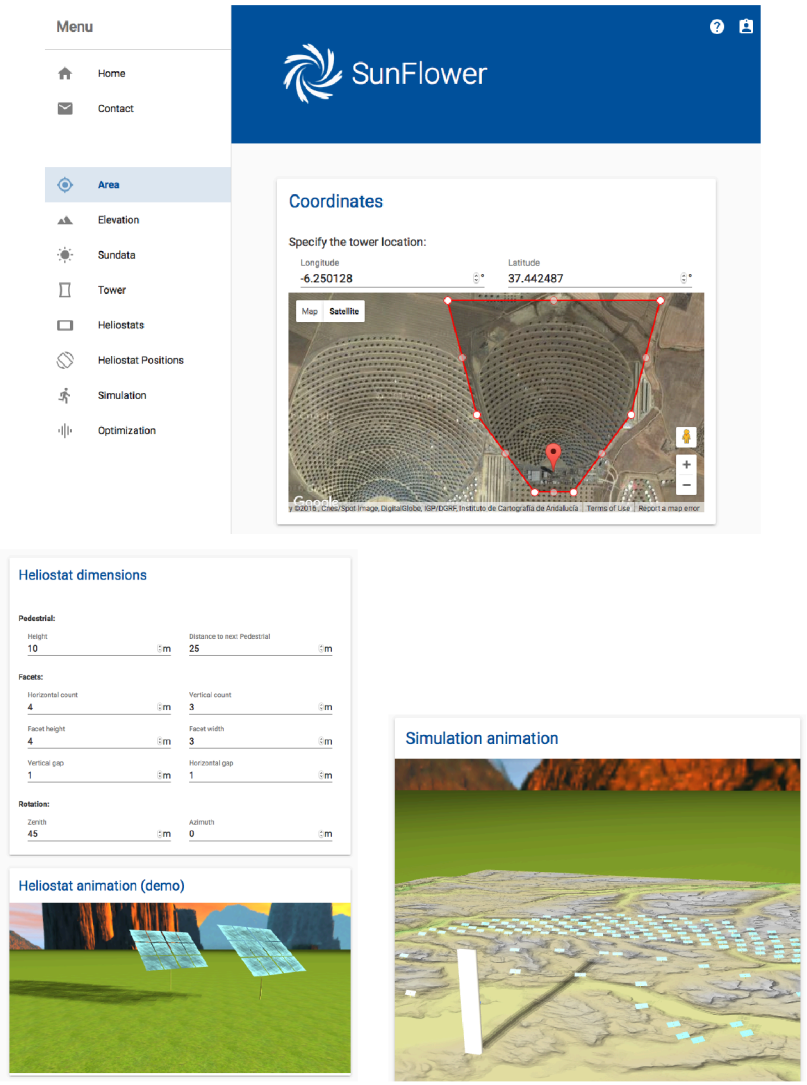


Figure 21: Screenshot of the new web application.

In the following two sections the model is used for the layout optimization using evolutionary algorithms. Both upcomig Sections have already been published in the joint work [4].

3. Layout optimization using evolutionary algorithms

The approach of using artificial intelligence for the optimization of solar power plant analysis in general is still rather new. There exist just a few works on this topic, e.g. [53, 54, 2, 55].

To solve the layout optimization problem, within this thesis an *evolutionary algorithm* as a free variable method was developed from scratch. This means that the optimizer is not based on any fixed pattern but offers the possibility to freely position heliostats inside a given area, as long as they have a sufficient distance to each other. The advantage is that a larger search space might contain more efficient solutions. If for any reasons a pattern is requested, our approach could be applied to any fixed topology in a quite straightforward manner.

The functionality of an evolutionary algorithm is inspired by the nature: In our setting, each layout configuration (*individual*) is specified by its properties (*genotype*), each property (*gene*) being either the x or the y coordinate value of a heliostat position. A *population* is a set of individuals. To measure the goodness of a population, we simulate all of its layout configurations to determine their *fitness values* (e.g. efficiency or received irradiation). The fitness of a population is its highest individual fitness value, which serves as the objective function for the optimization. The optimization algorithm starts from an initial population (e.g. random) and iteratively derives a new population from the previous one until some *termination criterion* is fulfilled. This could either be a maximum number of iterations, the convergence of the last rounds, or just a time limit. Upon termination, the best individual that was generated during the whole optimization process is returned.

To not loose the best solutions at the transition from one population to the next and thus to assure monotonicity of the population fitness, we initialize a new population to contain a certain number of fittest individuals from the previous population (*elitism*). Additionally, in order to avoid settling in a local optimum, it is also possible to introduce a certain number of new *random individuals* to each population. Afterwards these steps, we iteratively derive new individuals from the previous population and add them to the new population if they satisfy the minimal distance requirements (otherwise they are discarded). This procedure is repeated until the new population has the same size as the previous population.

To derive a new individual, three major operations are used:

1. *Selection:* Two or more individuals are randomly selected from the previous population according to their fitness values.

2. *Crossover:* The properties of the selected individuals are recombined according to their fitness values.

3. *Mutation:* Some genes of the new recombined individual might be modified before adding it to the new population.

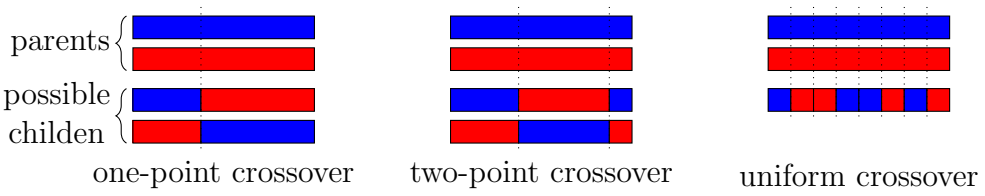


Figure 22: Classical crossover operators.

Due to the strong constraints, it is not ensured that classical crossover and mutation operators will generate valid individuals. Notice that an individual is not valid anymore, if just one pair of heliostats has a collision. To avoid this problem of inadmissibility, the main idea is to adapt the operators to the problem constraints. In the following we describe the selection, crossover and mutation algorithms along with the termination criterion that we use in our layout optimization approach.

3.1. Selection

Different techniques were proposed in literature for the selection of individuals that should be recombined. In our work, we utilize one of the most common methods, called the *roulette wheel method*. The objective of the roulette wheel method is to select potentially useful individuals to contribute to a new population with improved fitness. For that purpose, the fitness values are used in order to associate a probability of selection with each individual. This means that from a population $\mathcal{P}$ an individual $\mathcal{I} \in \mathcal{P}$ with fitness value $F(\mathcal{I})$ is selected with probability

$$p(\mathcal{I}) = \frac{F(\mathcal{I})}{\sum\limits_{\mathcal{I}' \in \mathcal{P}} F(\mathcal{I}')} \in [0, 1]. \tag{3.1}$$

3.2. Crossover

Using the above selection technique, we choose two *parent* individuals to be recombined into a new configuration (*child*). The classical crossover operators are *one-point crossover*, *two-point crossover* or *uniform crossover*. All three operators, illustrated on Figure 22, assume that the genes are stored as an ordered sequence of values. One-point crossover determines a sequence index, and generates children having genes from one of the parents up to the given index and from the other parents for larger indices. Two-point crossover works similarly but cuts the gene sequences at two points; children inherit genes from one of the parents for the indices between the two points and from the other parent for the remaining parts. Finally, uniform crossover determines for each gene in the sequence a parent, from which the gene is inherited, randomly using a uniform distribution (probability 50% for both parents).

The drawback of these crossover approaches is that many generated children violate

the constraint of minimum distance and need to be sorted out. Additionally, these approaches are highly sensitive to the order of the heliostats in their genotype representation. Finally, when applying these approaches for layout optimization, it is meaningful to encapsulate heliostat positions, represented by two genes (one for the x and one for the y coordinate value). Therefore we need to adapt both the genotype representation as well as the crossover operators to the layout optimization problem.

▶ Genotype representation

Note that in our setting each gene is either an x or a y coordinate value. The classical genotype representation is an *ordered sequence of genes*. Instead of sequences, in our approach, the genotype representation is a *set of genes*, where a gene is a position (x, y). Additionally, the set is combined with an order relation: The value of the objective function of a configuration is determined by an annual simulation of the sun irradiation, based on meteorological data. Besides the objective function value, the simulation also provides information for each single heliostat, e.g., its power contribution to the overall received power. Based on this information, we order the heliostats of a configuration by their goodness. Using this genotype representation, we define three different crossover operators, which are adapted to the layout optimization problem. They all base on the strategy of a *Greedy algorithm*.

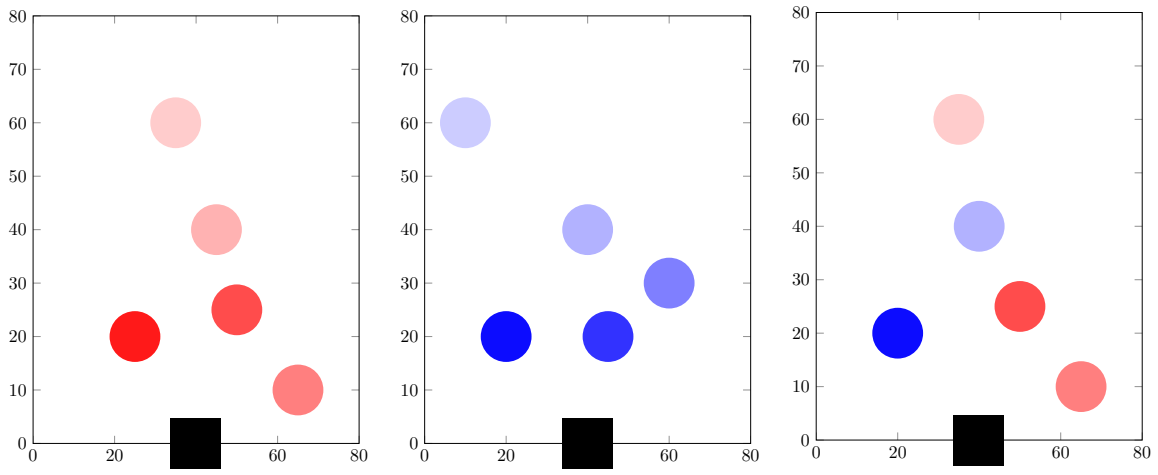
▶ Zero-step crossover

First the genes (the heliostat positions) from the genotype representations of both parents are sorted in descending order according to their fitness. From this sorted base list, the heliostats with the highest goodness values are step-wise inserted into the child individual (and popped from the base list). If an inserted heliostat causes a conflict, it is skipped and the next heliostat is chosen. If there are no more heliostats left in the base list, the child configuration is completed with randomly generated heliostats. This way we generate only valid individuals. Figure 23 illustrates the a zero-step crossover, where each of the parent genotypes contains five heliostat positions.

▶ One-step crossover

One weakness of the zero-step crossover operator is that the heliostats are weighted with their goodness in the parent individual, which does not guarantee to be a good choice in the child configuration due to new upcoming neighboring effects like blocking and shading. So, the heliostats' goodness for the generated child may not correlate to the one in the parent configuration.

In the one-step algorithm, we tackle this problem by placing all parent heliostats in decreasing goodness order into one field (skipping those which would affect collisions) and compute a new goodness value for each single heliostat. Based on these values, we select the desired number of heliostats for the child individual by choosing the best heliostats, similarly to the zero-step crossover (but now based on more appropriate goodness values).



(a) A mother individual

(b) A father individual

(c) A new child individual with the zero-step crossover strategy.

Figure 23: Recombination of two parent individuals in crossover. The saturation of the color (red or blue) is a simplified measure for the goodness of the single heliostat.

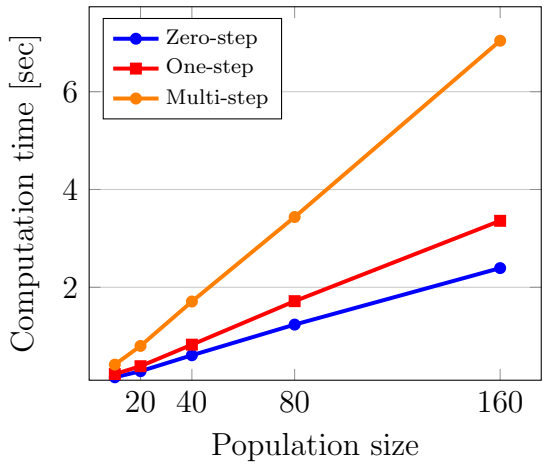


Figure 24: Crossover speed comparison.

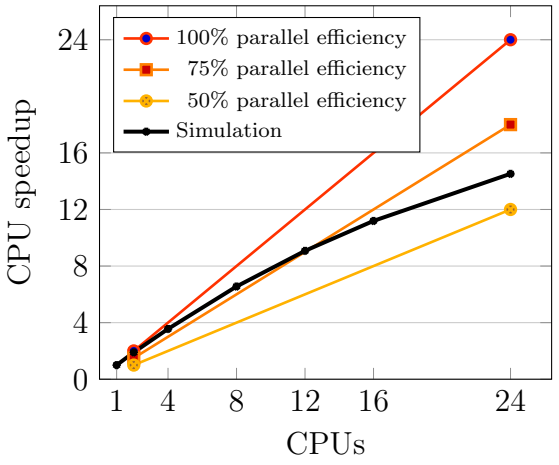


Figure 25: CPU speedup of an optimization with our evolutionary algorithm using a population size of 100 individuals.

▶ Multi-step crossover

The one-step crossover is based on improved goodness values, however, as not all the parent heliostats will be contained in the child configuration, these values still do not fully reflect the unique heliostat contributions to the final fitness value of the child. Consequently, with this approach heliostats in densely placed groups could be completely sorted out due to neighboring effects.

Therefore, we define a third operator called multi-step crossover, which uses several evaluation steps. First the goodness value of each single heliostat is computed as if it would be the only one on the field, i.e., without considering neighboring effects. Based on this ranking, the best heliostat is chosen and added to the (initially empty) child configuration. Now for each remaining heliostat that does not collide with the already added one we re-compute its contribution to the child fitness if it would be added to the current (incomplete) child configuration, and insert the best one into the child's genotype. This process is repeated iteratively until the required number of heliostats is reached. Again, if there are no more parent genes, we complete the child genotype with random genes.

It is obvious that the number of "steps" reflects the computational effort for the different crossover operators; the multi-step crossover is far more expensive to compute than the zero- and one-step versions. The differences in the computation time are illustrated in Figure 24.

3.3. Mutation

After the recombination of two parent individuals to one child individual by applying a crossover operator, some child genes might be modified by random mutation. Mutation leads to additional diversity of individuals in the new population. The classical mutation would effect single genes in isolation, which would lead to a modification of either the x- or the y-position of single heliostats. In our application, we encapsulate the (x, y)-position as one gene, such that in the case of mutation the whole position is shifted with random distance in a random direction. If a conflict appears, the mutation is repeated again.

4. Numerical results

We implemented our evolutionary algorithms with adapted genotype representation and crossover and mutation operators. For testing the quality of the different crossover operators, we first replaced the complex simulation-based fitness evaluation by using more simple objective functions: For a given individual $\mathcal{I}$ with its genotype representation $\mathcal{I} = \{(x_1, y_1), \ldots, (x_N, y_N)\} \subseteq \Omega$ we need to compute the objective function value $\mathcal{F}(\mathcal{I})$. Instead of using the simulation model we replace it by summing up function values $f : \mathbb{R}^2 \to \mathbb{R}$ for each heliostat:

$$\mathcal{F}(\mathcal{I}) = \sum_{i=1}^{N} f(x_i, y_i). \tag{4.1}$$

This test was successfully applied to all mentioned crossover operators. Figure 26 shows the results for the one-step crossover operator.

As the simulation-based fitness value computation is very time-consuming, we parallelized the optimization algorithm using `OpenMP`. Because the main workload of our optimization process is the calculation of the objective function values for every individual, we can apply the parallelization to this step. Due to the fact that each individual can be processed independently, the introduced parallelization overhead due to blocking is negligible. After these parallelized computations, an additional single-threaded pass over all individuals is performed to compute global values describing the whole population (e.g. min/max energy). The achieved speed-up by using parallelization is depicted in Figure 25.

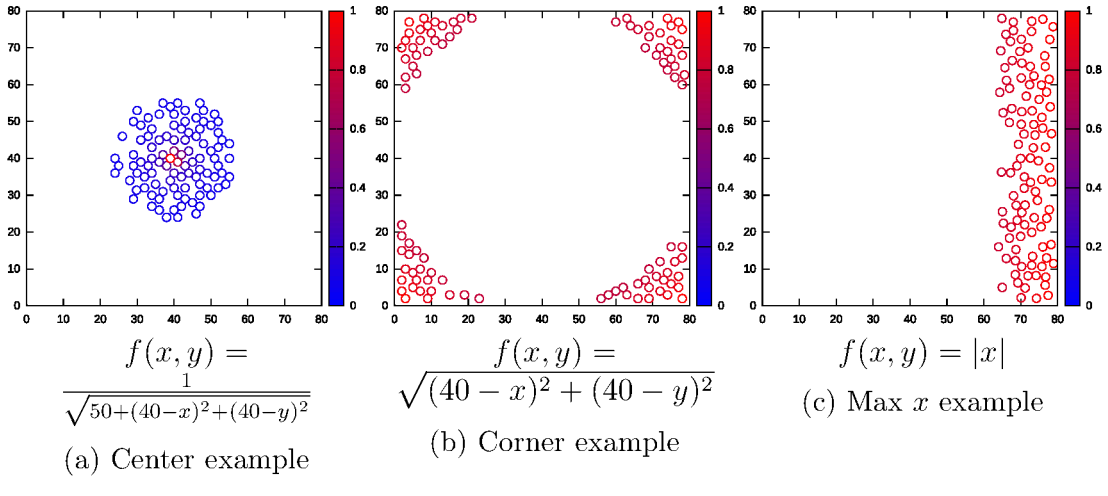
To show the applicability of our approach, we applied our algorithms to optimize two real solar tower power plants. As objective function the annual performance is used, which is defined as the fraction of irradiation energy received at the tower and the total energy reaching the mirrors without shading,

$$\mathcal{F}(\mathcal{I}) = \frac{\int_0^{8760} A \cdot \mathrm{DNI}(t) \cdot \eta(t)\ \mathrm{d}t}{\int_0^{8760} A \cdot \mathrm{DNI}(t)\ \mathrm{d}t}, \tag{4.2}$$

where 8760 is the number of hours in a year, A is the mirror area, $\mathrm{DNI}(t)$ the time-dependent direct normal irradiation, and $\eta(t)$ the time-dependent efficiency of the field, considering cosine effects, blocking & shading of neighboring heliostats, interception efficiency and atmospheric attenuation [36, 2].

▶ Planta Solar 10 (PS10)

The PS10 solar tower power plant is placed near Seville, in Andalusia, Spain. Since 2007, the 11 megawatt (MW) solar power tower produces electricity with 624 large heliostats. Each heliostat has a mirror surface of 120 square meters, where the receiver is placed on top of a tower at 115 meters height. More details about the configuration can be found in [36].



$$f(x,y) = \frac{1}{\sqrt{50+(40-x)^2+(40-y)^2}}$$

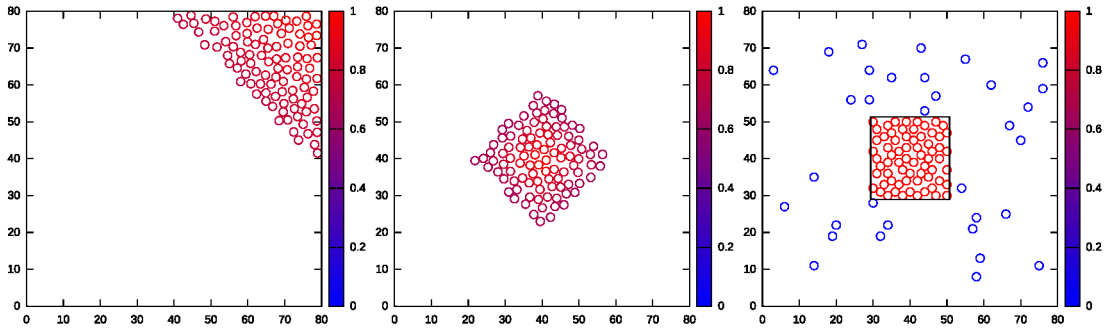
(a) Center example

$$f(x,y) = \sqrt{(40-x)^2+(40-y)^2}$$

(b) Corner example

$$f(x,y) = |x|$$

(c) Max x example



$$f(x,y) = x + y$$

(d) Triangle example

$$f(x,y) = 50 - |40-x| - |40-y|$$

(e) Rhombus example

Within rectangle

(f) Sphere packing example

Figure 26: Overview of the different test function results.

We used our evolutionary algorithms to optimize the PS10 power plant, using a random initial population. As shown in Figure 27, using different crossover operators leads to different convergence rates for the optimization where the zero-step crossover shows a convergence behaviour similar to the standard crossover operators, the one-step crossover shows a very fast convergence rate. The multi-step crossover converges, against our expectations, much slower. We suspect that this phenomenon is due to the fact that the multi-step crossover prefers higly efficient heliostat positions in the context of the current incomplete child genotype. Such heliostats are usually free-standing without other heliostats in their neighborhood where it pays off for the first few heliostats, this heuristics possibly reduces the possibilities for adding further efficient heliostat positions. According to the convergence rate of the multi-step crossover, we expect that problems with smaller fields show better convergence rates in optimization time.

We also compared the results of our approach to results using two different pattern-based optimization approaches (Figure 28). The first one is the original PS10 field layout, which was optimized using the radially staggered grid approach. Additionally

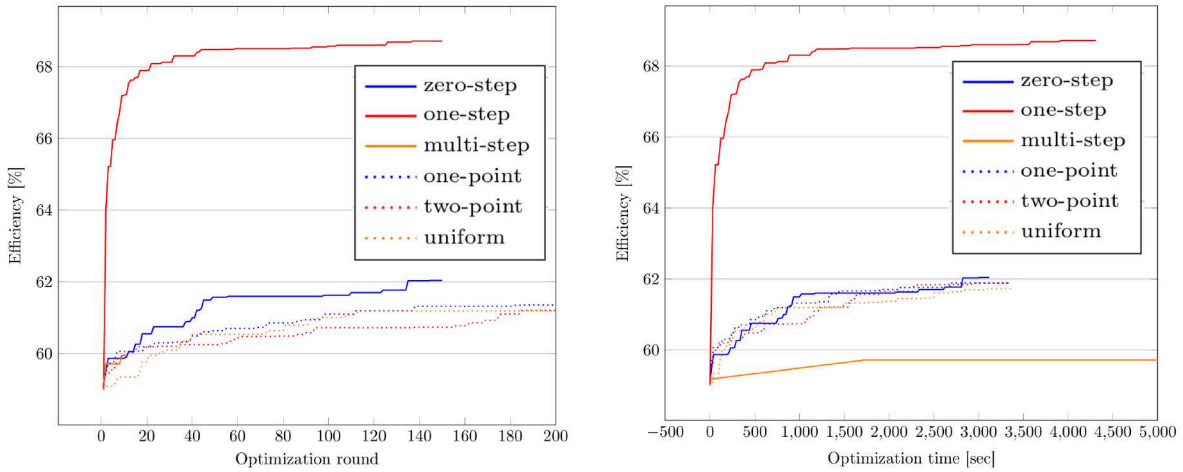
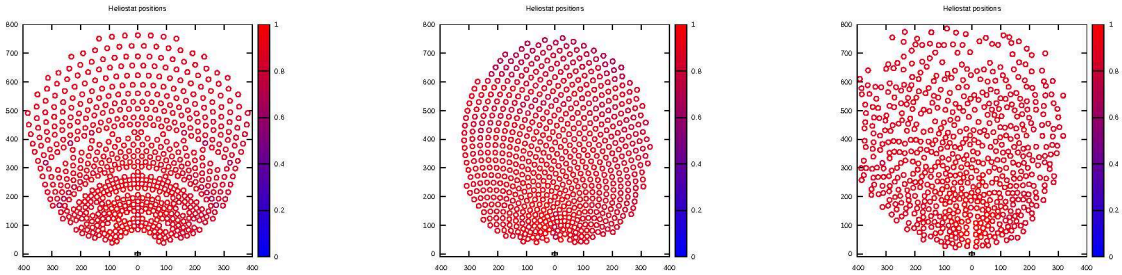


Figure 27: Comparison of crossover operators on PS10, plotting optimization progress vs. optimization steps (left) and vs. optimization time (right).



(a) Original PS10 layout as radially staggered grid: 66.5189 % efficiency.

(b) Optimized layout using a biomimetic approach: 67.518 % efficiency.

(c) Optimized layout using evolutionary algorithm with one-step crossover: 68.7194 % efficiency.

Figure 28: Comparison of optimization algorithms on PS10.

we compare our results with the biomimetic approach that was applied to optimize the PS10 plant in [36]. The original field layout collects less energy than the biomimetic approach, and even less than our evolutionary algorithms. Our best approach using one-step crossover collects 2.2 % more energy than the original layout, and 1.2014 % more energy than the biomimetic "sun flower" approach.

It is interesting to see that our approach shows a shift of the heliostats to the top left corner. There can be at least two reasons for this effect: At first, the topography rises to the top left corner for about 40 m in comparison to the position of the tower, such that these positions may be more attractive. Another reason is the solar irradiation, which may not be symmetric over a full day. Altogether, it is interesting to see that our approach has the possibility to adjust the results on these external influences.

▶ Helio100

The Helio100 solar tower facility is placed near Stellenbosch, South Africa. 120 heliostats are used, each with a mirror surface of 2.2 square meters, where the receiver is placed on top of a tower at 12 meters height. The plant is in operation since 2015.

The behaviour of different standard and adapted crossover operators is investigated in Figure 29. All three adapted crossover operators perform better than the classical ones. The approach using the one-step crossover operator reaches the fastest convergence rate. On this Helio100 application the multi-step crossover shows somewhat better performance than on PS10, possibly due to the smaller number of heliostats.

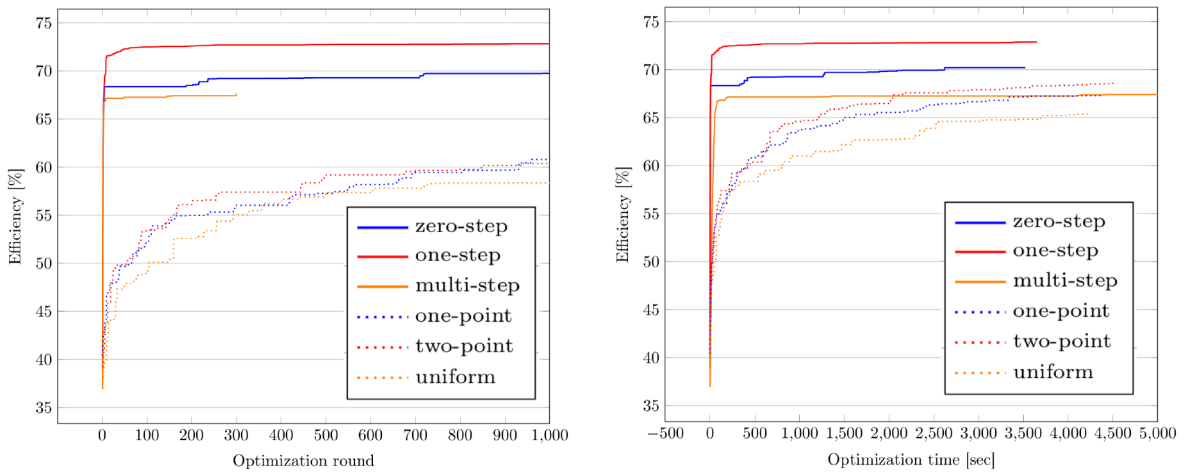
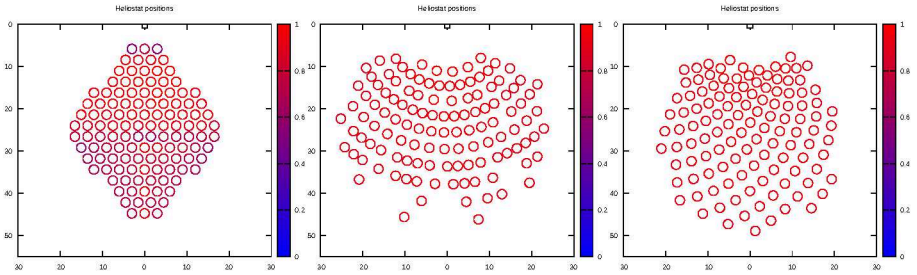


Figure 29: Comparison of crossover algorithms on Helio100, plotting optimization progress vs. optimization steps (left) and vs. optimization time (right).

We compare the layout results of our optimization approaches to pattern-based optimization approaches and to the classical crossover operators in Figure 30. Our best result using the one-step crossover collects 7.5335 % more energy than the original layout, and 0.0595 % more energy than the biomimetic "sun flower" approach.

Pattern-based layouts

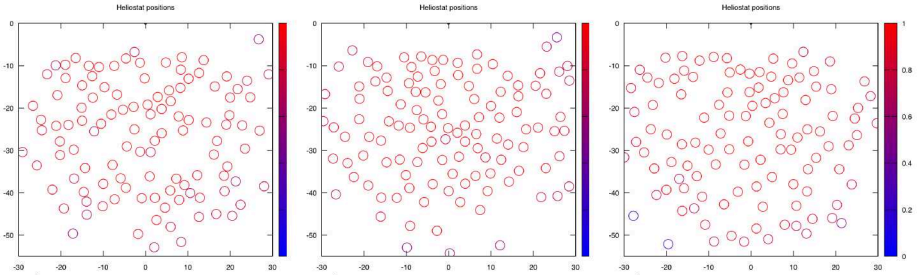


(a) Row-pattern-based original field layout 65.3512 % efficiency

(b) Radially staggered grid approach 72.127 % efficiency

(c) Biomimetic approach 72.8252 % efficiency

Evolutionary methods: classical crossover operators

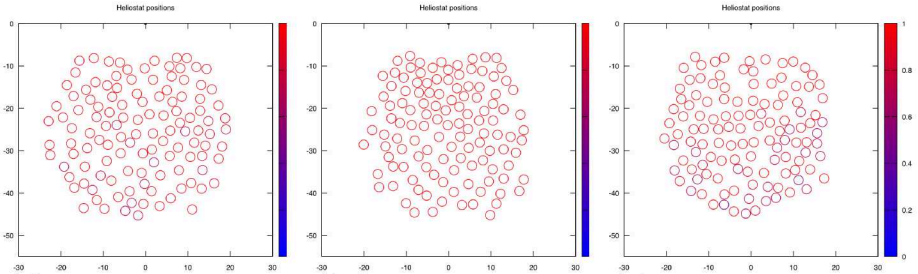


(d) One-point crossover 67.2945 % efficiency

(e) Two-point crossover 68.6612 % efficiency

(f) Uniform crossover 65.5853 % efficiency

Evolutionary methods: adapted crossover operators



(g) Zero-step crossover 70.2242 % efficiency

(i) One-step crossover 72.8847 % efficiency

(h) Multi-step crossover 67.6485 % efficiency

Figure 30: Results for different layout optimization algorithms on Helio100.

5. Extensions and post-processing

▶ Helio100 with triangular heliostat pod systems

The Helio100 test facility originally uses 20 heliostat pods, where six heliostats are grouped on one triangular structure (instead of 120 single heliostats). As introduced in Section 2.4, a heliostat pod position is defined by the center position (x, y) and additionally by a rotation angle γ. Thus, the number of free variables is reduced from twelve (six times x and y) to three [3].

This new rotation parameter γ has been incorporated in the evolutionary algorithm. The optimization delivers a configuration with an efficiency of 68.0635 %, see Figure 31. Due to the new constraint of using a local triangular structure, the obtained results are worse compared to the above results with 120 single heliostats.

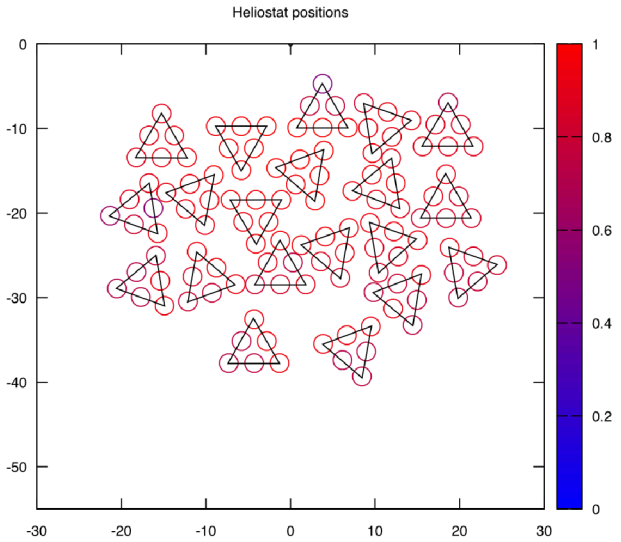


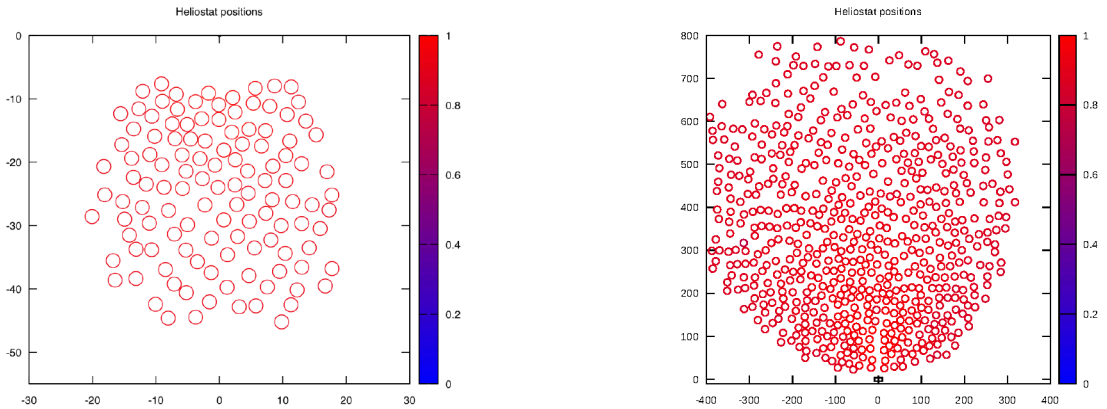
Figure 31: Helio100 with triangular heliostat pod systems. Using the evolutionary algorithm delivers an efficiency of 68.0635 %.

▶ Post-Processing

The evolutionary optimization algorithm is a global approach to achieve an optimal field layout which is ideal with respect to the given objective function. As can be seen in the optimization examples in the last section, the results still offer room for further improvements, to advance the placement of individual heliostats. To benefit from this potential, a local optimization strategy as post-processing step can be used.

The straight-forward way is to use a *greedy heuristic*. The idea of this approach is to optimize the field with slight local position changes. Therefore, one after another randomly selected heliostat is attempt to be moved in random direction for random distances. If a new position yields a higher fitness, we move it to this position, following the Greedy principle. This approach was used for the Helio100 field and the PS10 solar field (see Figure 32). The efficiency of both fields increases for about 0.5 %.

Before post-processing:



After post-processing:

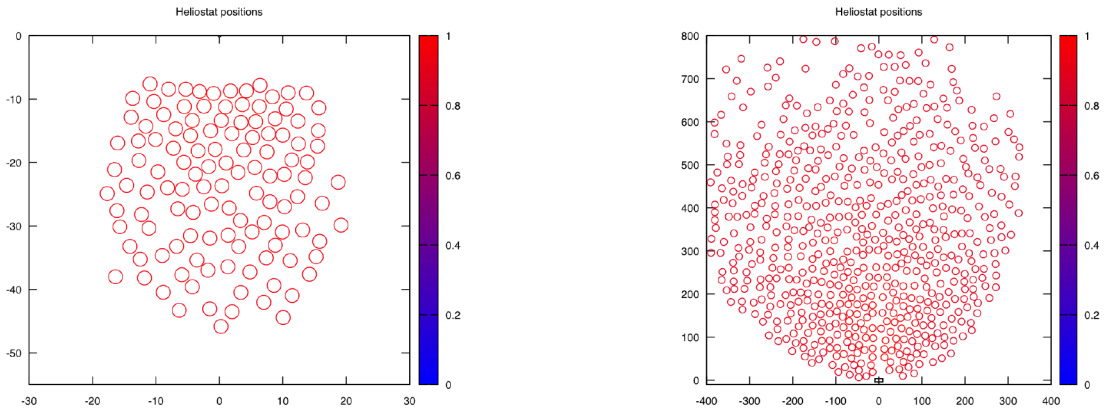


Figure 32: Applying a post-processing step for Helio100 after using the evolutionary algorithm, an efficiency of 72.9622 % is delivered (left). For PS10, an efficiency of 69.1735 % is delivered (right).

Part II.
Modeling and Simulation of Direct Steam Generation

1. State of the art

To model a two-phase flow of water in the absorber tube, the conservation of mass, momentum and energy is supposed for liquid and steam phase. Different model approaches exist in the literature:

- The *homogeneous equilibrium model* [56] is based on the assumption that the two-phase mixture behaves as a single-phase fluid. It uses mean fluid properties that are weighted relatively to vapor and liquid content and it is assumed, that both phases have equal velocities. Because all three conservative equations are modeled homogeneously, the model is a so-called three-equation model.

- The *drift-flux model* [57] enhances the homogeneous equilibrium model by taking different densities for both phases into account. Hence, the three partial differential equations are extended by a fourth equation by segregating the homogeneous mass balance equation.

- The *five-equation drift-flux model* is additionally segregating the energy balance equation [58], to obtain relaxed energy equations.

- The *separated two-phase flow model* [59] considers the phases to be artificially segregated into a liquid and a vapor stream. Therefore the model bases on a system of six equations – three equations for each phase. These six-equations models are widely used in nuclear thermal-hydraulic codes (RELAP5[5], TRAC[6], CATHARE[7], SPACE[8]). As closure condition, equal pressure in both phases is assumed. Following Saurel and Abgrall [60], this choice yields ill-posed mathematical models and results in numerical instabilities.

- As extension of the six-equations two-fluid model, Drew and Passman [61] and Saurel and Abgrall [60] propose the *two-pressure two-phase model*. An additional equation for volume fraction completes the system of equations. This hyperbolic model allows simulations of liquid phase at negative pressure, while the pressure of the vapor phase remains positive, and vice versa.

Related work

The aim of this work is to describe the direct-steam generation in the absorber tubes of a solar thermal power plant. Often, the two-phase flow of water is modelled using the

[5] Reactor Excursion and Leak Analysis Program by Idaho National Laboratory `www.inl.gov/relap5`.

[6] Transient Reactor Analysis Code by Los Alamos National Laboratory `nuclear.lanl.gov/nrc.shtml`.

[7] Code for Analysis of Thermalhydraulics during an Accident of Reactor and safety Evaluation by Commisariat á l'Energie Atomique `www-cathare.cea.fr`.

[8] Safety and Performance Analysis Code by Korea Atomic Energy Research Institute `www.kaeri.re.kr`.

homogeneous equilibrium model [62], [63], [64], [65]. Just in some recent publications, more complicated models are used by making use of nuclear thermal-hydraulic codes [66], [67]. But notice that the therein used six-equations models are ill-posed, as already mentioned above.

This is the motivation of this work: to derive a well-posed model and to develop an accurate solver which is fast enough, such that it can be used for the control of solar tower power plants. The solver has to be chosen carefully, because the underlying problem is quite hard: We have an inhomogeneous non-conservative system, with low Mach number and stiff source terms [68].

The rest of this work is structured as follows. In Section 2 a *two-pressure two-phase model* is developed, such that thermodynamical and mathematical properties hold. This model is compared with the widespread *homogeneous equilibrium model*. The problem is solved using a finite volume method which is introduced in Section 3. Because the underlying problem is in non-conservative form, this problem is solved using an entropy-preserving path-conservative scheme in Section 4. In Section 5 a Godunov-type method is developed by finding a Riemann solution of the model. To accelerate the run-time, the relaxation model is solved by using a semi-implicit scheme 6. All solvers are validated and compared against well-defined test cases in Section 7.

2. Two-phase flow model

Within this section, a two-phase flow model is derived from the three-dimensional compressible Navier-Stokes equations. Due to the underlying application of a flow in a thin tube, two spatial dimensions are neglected. The lost of information about the local fluid structure is compensated by replacing diffusive second order terms by empirical laws in dependency of the local flow pattern. The model fulfilles several mathematical and thermodynamical properties, which deliver closure conditions for the freely selectable model parameters.

In Section 2.1 the fundamental two-phase flow model is derived, averaged, and simplified. Out of the developed one-dimensional balance laws two models are derived: a homogeneous equilibrium and a two-velocity two-pressure model, see Section 2.3. In Section 2.5 the models are closed by deriving properties of the models. Finally, in Section 2.4 constitutive relations for the empirical laws are derived.

2.1. Ensemble averaging

The herein presented model bases on the work of Drew and Passman [61]. While in their original work they do not give closures for the interfacial properties, this is developed in this section.

▶ Fundamental of two-phase flow

The motion of fluids can mathematically be described by a system of non-linear partial differential equations of second order [61], based on the canonical form of the local balance equations

$$\frac{\partial(\rho\psi)}{\partial t} + \nabla \cdot (\rho \boldsymbol{v} \psi) = \nabla \cdot \mathbb{J} + \rho\phi, \tag{2.1}$$

with fluid density ρ and fluid velocity vector $\boldsymbol{v} \in \mathbb{R}^d$ in a d-dimensional space. Balance equations are derived by choosing the conserved quantity ψ, diffusive flux $\mathbb{J}$ and source density ϕ, The fluid flow is governed by the three balance equations per phase of mass, momentum and energy. Their usual values for ψ, $\mathbb{J}$ and ϕ are given in Table 1.

	ψ	$\mathbb{J}$	ϕ
Mass	1	0	0
Momentum	$\boldsymbol{v}$	$(\mathbb{S} - p\mathcal{I})$	$\boldsymbol{g}$
Energy	E	$(\mathbb{S} - p\mathcal{I}) \cdot \boldsymbol{v} - \boldsymbol{q}$	$\boldsymbol{g} \cdot \boldsymbol{v} + \frac{\dot{q}}{\rho}$

Table 1: Parameters of the general balance equation for conservation of mass, momentum and energy.

$p, T, E, \boldsymbol{q}$ are the pressure, temperature, specific total energy and the heat flux of the fluid, $\mathcal{I}$ is the identity matrix, $\dot{q}$ is the body heating, $\boldsymbol{g}$ is the gravitational body force, and $\mathbb{S}$ the viscous stress tensor.

In a multi-phase flow, the phasic flows can be very heterogeneous: Each phase may locally consist of many small disjoint sets, e.g. bubbles or droplets. This causes strong disequilibria between the phases. Because small scales neither can be handled by experiments nor by numerics, there is need of an averaging process.

Therefore each phase k has to be isolated theoretically. To derive the equations for multi-phase flow, it is necessary to describe the local characteristics of the flow, where the macroscopic properties should be obtained by means of an approriate averaging procedure [69].

▶ Ensemble averaging

Ensemble averaging is a generalisation to the elementary averaging of Baer and Nunziato [6] in which the observed values are added and divided by the number of observations. To identify each phase separately, Drew and Passman [61, 70] introduced the component indicator function $X_k(\boldsymbol{x}, t)$ as characteristic function with

$$X_k(\boldsymbol{x}, t) = \begin{cases} 1, & \text{if phase } k \text{ is present at position } \boldsymbol{x} \text{ and time } t \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

In our case, we consider just two components, the liquid phase (ℓ) and the steam phase (g). The two-phase indicator functions are related by

$$X_\ell + X_g = 1. \tag{2.3}$$

With the help of the Reynolds transport theorem [71], Drew and Passman derive the topological equation,

$$\frac{\partial X_k}{\partial t} + \boldsymbol{v}_\mathrm{i} \nabla X_k = 0, \tag{2.4}$$

where $\boldsymbol{v}_\mathrm{i}$ is the velocity of the interface. The ensemble average $\overline{\cdot}$ of the component indicator function X_k in a small region is the local ratio of the volume of phase k to the total volume of that region [61]. This ensemble average is called volume or void fraction and is labeled with α_k,

$$\alpha_k := \overline{X_k}. \tag{2.5}$$

It is obvious that all phasic volume fractions sum to one, see (2.3), so that for a liquid ℓ and steam g phase it holds

$$\alpha_\ell + \alpha_g = 1. \tag{2.6}$$

By multiplying the averaging procedure to the canonical form of the balance equations (2.1) with X_k, applying the averaging procedure and using the Gauss rule for (2.4), the averaged canonical form of the balance equation for multi-phase flow is obtained:

$$\frac{\partial \overline{X_k \rho \psi}}{\partial t} + \nabla \cdot \overline{X_k \rho \boldsymbol{v} \psi} = \nabla \cdot \overline{X_k \mathbb{J}} + \overline{X_k \rho \phi} + \overline{\big(\rho \psi (\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) - \mathbb{J}\big) \cdot \nabla X_k}, \tag{2.7}$$

The last term at the right-hand side is the interfacial source of ψ. Three types of averaged variables appear:

- The component-weighted average of a thermodynamic variable f is given by

$$\alpha_k f_k := \overline{X_k f}. \tag{2.8}$$

- The mass-weighted average of a thermodynamic variable f is given by

$$\alpha_k \rho_k f_k := \overline{X_k \rho f}. \tag{2.9}$$

- The interfacial sources of (2.7) are interphase-weighted variables. They split in a molecular flux $\overline{-\mathbb{J} \cdot \nabla X_k}$ and a convective flux $\overline{\rho \psi (\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k}$. The convective flux describes the average flow rate, where the difference $(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i})$ expresses the velocity fluctuations which may be due to turbulence, see [72] and [61]. Following [61] (see equations (11.38) to (11.42)), we define

$$\overline{\rho \psi (\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k} = \overline{\rho_{\mathrm{i}k}} \widehat{\psi}_{\mathrm{i}k} \cdot \underbrace{\overline{(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k}}_{:= \Gamma_{\mathrm{i}k} / \widehat{\rho}_{\mathrm{i}k}} = \widehat{\psi}_{\mathrm{i}k} \Gamma_{\mathrm{i}k} \tag{2.10}$$

  with interfacial mass flow rate $\Gamma_{\mathrm{i}k}$ from bulk. The parameters $\widehat{\rho}_{\mathrm{i}k}$ and $\widehat{\psi}_{\mathrm{i}k}$ describe the source quantities.

▶ Volume fraction

Applying the averaging operator to the component indicator function (2.4), and using (2.5), we get [61]:

$$\begin{aligned}
\overline{\frac{\partial X_k}{\partial t}} + \overline{\boldsymbol{v}_\mathrm{i} \cdot \nabla X_k} &= \frac{\partial \overline{X_k}}{\partial t} + \overline{\boldsymbol{v}_\mathrm{i} \cdot \nabla X_k} \\
&= \frac{\partial \overline{X_k}}{\partial t} + \overline{(\boldsymbol{v} - (\boldsymbol{v} - \boldsymbol{v}_\mathrm{i})) \cdot \nabla X_k} \\
&= \frac{\partial \overline{X_k}}{\partial t} + \overline{\boldsymbol{v} \cdot \nabla X_k} - \overline{(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k} \\
&= \frac{\partial \alpha_k}{\partial t} + \boldsymbol{v}_{\mathrm{i}k} \cdot \nabla \alpha_k - \frac{\Gamma_{\mathrm{i}k}}{\widehat{\rho}_{\mathrm{i}k}} \;=\; 0,
\end{aligned} \tag{2.11}$$

where $\boldsymbol{v}_{\mathrm{i}k}$ is the phasic velocity of the interface. The convective flux $\Gamma_{\mathrm{i}k}/\widehat{\rho}_{\mathrm{i}k}$ describes the interfacial volume fraction source. Finally, the void fraction equation is given by

$$\partial_t \alpha_k + \boldsymbol{v}_{\mathrm{i}k} \cdot \nabla \alpha_k = \frac{\Gamma_{\mathrm{i}k}}{\widehat{\rho}_{\mathrm{i}k}}. \tag{2.12}$$

This volume fraction equation is similiar to the compaction equation of Baer and Nunziato [6] (see for instance equations 53 and 59). Due to (2.6), it is clear that the gain of one phase in time and space is equal to the loss of the other phase:

$$\partial_t \alpha_\ell = -\partial_t \alpha_g \quad \text{and} \quad \nabla \alpha_\ell = -\nabla \alpha_g. \tag{2.13}$$

▶ Averaged balance equations

The averaged balance equations are derived from the averaged canonical form of the balance equation for multi-phase flow (2.7), combined with the values given in Table 1.

Mass
The averaged mass equation is

$$\frac{\partial \overline{X_k \rho}}{\partial t} + \nabla \cdot \overline{X_k \rho \boldsymbol{v}} = \overline{\rho(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k}. \tag{2.14}$$

The averaged variables are converted by using (2.8) and (2.9):

$$\overline{X_k \rho} = \alpha_k \rho_k \quad \text{and} \quad \overline{X_k \rho \boldsymbol{v}} = \alpha_k \rho_k \boldsymbol{v}_k. \tag{2.15}$$

The convective flux at the right-hand side describes the interfacial mass flow rate, see (2.10), such that

$$\overline{\rho(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k} = \Gamma_{\mathrm{i}\,k}. \tag{2.16}$$

Finally, the averaged mass equation is given by

$$\partial_t(\alpha_k \rho_k) + \nabla \cdot (\alpha_k \rho_k \boldsymbol{v}_k) = \Gamma_{\mathrm{i}\,k}. \tag{2.17}$$

Momentum
The averaged momentum equation is

$$\frac{\partial \overline{X_k \rho \boldsymbol{v}}}{\partial t} + \nabla \cdot \overline{X_k \rho \boldsymbol{v} \boldsymbol{v}} = \nabla \cdot \overline{X_k (\mathbb{S} - p\mathcal{I})} + \overline{X_k \rho \boldsymbol{g}} + \overline{\big(\rho \boldsymbol{v}(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) - (\mathbb{S} - p\mathcal{I})\big) \cdot \nabla X_k}. \tag{2.18}$$

The averaged variables are converted by using (2.8) and (2.9):

$$\overline{X_k \rho \boldsymbol{v}} = \alpha_k \rho_k \boldsymbol{v}_k, \quad \overline{X_k \rho \boldsymbol{v} \boldsymbol{v}} = \alpha_k \rho_k \boldsymbol{v}_k \boldsymbol{v}_k, \quad \text{and} \quad \overline{X_k \rho \boldsymbol{g}} = \alpha_k \rho_k \boldsymbol{g}_k. \tag{2.19}$$

Similarly, we convert

$$\nabla \cdot \overline{X_k (\mathbb{S} - p\mathcal{I})} = \nabla \cdot (\alpha_k \mathbb{S}_k) - \nabla \cdot (\alpha_k p_k \mathcal{I}). \tag{2.20}$$

The convective interfacial momentum source is

$$\overline{\rho \boldsymbol{v}(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k} =: \widehat{\boldsymbol{v}_{\mathrm{i}\,k}} \cdot \overline{\rho(\boldsymbol{v} - \boldsymbol{v}_\mathrm{i}) \cdot \nabla X_k} \overset{(2.10)}{=} \widehat{\boldsymbol{v}}_{\mathrm{i}\,k} \Gamma_{\mathrm{i}\,k} \tag{2.21}$$

where $\widehat{\boldsymbol{v}}_{\mathrm{i}\,k}$ is a phase change term describing the phasic source velocity. The molecular interfacial momentum source is given by [61]

$$-\overline{(\mathbb{S}-p\mathcal{I})\cdot\nabla X_k} = p_{\mathrm{i}\,k}\nabla\alpha_k - \mathbb{S}_{\mathrm{i}\,k}\cdot\nabla\alpha_k, \tag{2.22}$$

with interfacial pressure $p_{\mathrm{i}\,k}$ and interphase shear stress $\mathbb{S}_{\mathrm{i}\,k}$. So finally, the averaged momentum equation is given by

$$\begin{aligned}\partial_t(\alpha_k\rho_k\boldsymbol{v}_k) + \nabla\cdot(\alpha_k\rho_k\boldsymbol{v}_k\boldsymbol{v}_k) \;=\; & -\nabla\cdot(\alpha_k p_k\mathcal{I}) + \nabla\cdot(\alpha_k\mathbb{S}_k) + \alpha_k\rho_k\boldsymbol{g}_k + \widehat{\boldsymbol{v}}_{\mathrm{i}\,k}\Gamma_{\mathrm{i}\,k}\\ & + p_{\mathrm{i}\,k}\nabla\alpha_k - \mathbb{S}_{\mathrm{i}\,k}\cdot\nabla\alpha_k.\end{aligned} \tag{2.23}$$

Energy

The averaged total energy equation is

$$\begin{aligned}\frac{\partial\overline{X_k\rho E}}{\partial t} + \nabla\cdot\overline{X_k\rho\boldsymbol{v}E} = \nabla\cdot\overline{X_k\big((\mathbb{S}-p\mathcal{I})\cdot\boldsymbol{v}-\boldsymbol{q}\big)} + \overline{X_k(\rho\boldsymbol{g}\cdot\boldsymbol{v}+\dot{q})}\\ + \overline{\big(\rho E(\boldsymbol{v}-\boldsymbol{v}_{\mathrm{i}}) - (\mathbb{S}-p\mathcal{I})\cdot\boldsymbol{v} + \boldsymbol{q}\big)\cdot\nabla X_k}.\end{aligned} \tag{2.24}$$

The averaged variables are converted by using (2.8) and (2.9):

$$\overline{X_k\rho E} = \alpha_k\rho_k E_k, \quad \overline{X_k\rho\boldsymbol{v}E} = \alpha_k\rho_k\boldsymbol{v}_k E_k \quad\text{and}\quad \overline{X_k(\rho\boldsymbol{g}\cdot\boldsymbol{v}+\dot{q})} = \alpha_k\rho_k\boldsymbol{g}_k\cdot\boldsymbol{v}_k + \alpha_k\dot{q}_k. \tag{2.25}$$

Similarly, we convert with (2.20)

$$\begin{aligned}\nabla\cdot\overline{X_k\big((\mathbb{S}-p\mathcal{I})\cdot\boldsymbol{v}-\boldsymbol{q}\big)} &= \nabla\cdot(\alpha_k\boldsymbol{v}_k\mathbb{S}_k) - \nabla\cdot(\alpha_k\boldsymbol{v}_k p_k\mathcal{I}) - \nabla\cdot(\alpha_k\boldsymbol{q}_k)\\ &= \boldsymbol{v}_k\cdot\nabla\cdot(\alpha_k\mathbb{S}_k) + \alpha_k\mathbb{S}_k : \nabla\boldsymbol{v}_k - \nabla\cdot(\alpha_k\boldsymbol{v}_k p_k\mathcal{I}) - \nabla\cdot(\alpha_k\boldsymbol{q}_k),\end{aligned} \tag{2.26}$$

The convective interfacial energy generation source is given by

$$\overline{\rho E(\boldsymbol{v}-\boldsymbol{v}_{\mathrm{i}})\cdot\nabla X_k} =: \widehat{E}_{\mathrm{i}\,k}\cdot\overline{\rho(\boldsymbol{v}-\boldsymbol{v}_{\mathrm{i}})\cdot\nabla X_k} \overset{(2.10)}{=} \widehat{E}_{\mathrm{i}\,k}\Gamma_{\mathrm{i}\,k}, \tag{2.27}$$

where $\widehat{E}_{\mathrm{i}\,k}$ is a phase change term describing the phasic source specific total energy. The molecular interfacial work is given by (2.22)

$$-\overline{(\mathbb{S}-p\mathcal{I})\cdot\boldsymbol{v}\cdot\nabla X_k} \overset{(2.22)}{=} p_{\mathrm{i}\,k}\cdot\boldsymbol{v}_{\mathrm{i}\,k}\cdot\nabla\alpha_k - \mathbb{S}_{\mathrm{i}\,k}\cdot\boldsymbol{v}_{\mathrm{i}\,k}\cdot\nabla\alpha_k, \tag{2.28}$$

The interfacial heat source is defined by

$$\overline{\boldsymbol{q}\cdot\nabla X_k} = \boldsymbol{q}_{\mathrm{i}\,k}\cdot\nabla\alpha_k, \tag{2.29}$$

We finally get the averaged energy equation,

$$\begin{aligned}\partial_t(\alpha_k\rho_k E_k) + \nabla\cdot(\alpha_k\rho_k\boldsymbol{v}_k E_k) \;=\; & -\nabla\cdot(\alpha_k\boldsymbol{v}_k p_k\mathcal{I}) + \nabla\cdot(\alpha_k\mathbb{S}_k)\cdot\boldsymbol{v}_k + \alpha_k\mathbb{S}_k : \nabla\boldsymbol{v}_k\\ & - \nabla\cdot(\alpha_k\boldsymbol{q}_k) + \alpha_k\rho_k\boldsymbol{g}_k\cdot\boldsymbol{v}_k + \alpha_k\dot{q}_k + \widehat{E}_{\mathrm{i}\,k}\Gamma_{\mathrm{i}\,k}\\ & + p_{\mathrm{i}\,k}\cdot\boldsymbol{v}_{\mathrm{i}\,k}\cdot\nabla\alpha_k - \mathbb{S}_{\mathrm{i}\,k}\cdot\boldsymbol{v}_{\mathrm{i}\,k}\cdot\nabla\alpha_k + \boldsymbol{q}_{\mathrm{i}\,k}\cdot\nabla\alpha_k.\end{aligned} \tag{2.30}$$

▶ Averaged system

The volume fraction equation (2.12) and the three balance equations of mass (2.17), momentum (2.23) and energy (2.30) describe the two-phase flow model:

$$
\begin{aligned}
\partial_t \alpha_k \;+\; \boldsymbol{v}_{\mathrm{i}\,k} \cdot \nabla \alpha_k \qquad &= \frac{\Gamma_{\mathrm{i}\,k}}{\widehat{\rho}_{\mathrm{i}\,k}} \\
\partial_t (\alpha_k \rho_k) \;+\; \nabla \cdot (\alpha_k \rho_k \boldsymbol{v}_k) \qquad &= \Gamma_{\mathrm{i}\,k} \\
\partial_t (\alpha_k \rho_k \boldsymbol{v}_k) \;+\; \nabla \cdot (\alpha_k \rho_k \boldsymbol{v}_k \boldsymbol{v}_k) \;&=\; -\nabla \cdot (\alpha_k p_k \mathcal{I}) + \nabla \cdot (\alpha_k \mathbb{S}_k) + \alpha_k \rho_k \boldsymbol{g}_k + \widehat{\boldsymbol{v}}_{\mathrm{i}\,k} \Gamma_{\mathrm{i}\,k} \\
&\quad + p_{\mathrm{i}\,k} \nabla \alpha_k - \mathbb{S}_{\mathrm{i}\,k} \cdot \nabla \alpha_k \\
\partial_t (\alpha_k \rho_k E_k) \;+\; \nabla \cdot (\alpha_k \rho_k \boldsymbol{v}_k E_k) \;&=\; -\nabla \cdot (\alpha_k \boldsymbol{v}_k p_k \mathcal{I}) + \nabla \cdot (\alpha_k \mathbb{S}_k) \cdot \boldsymbol{v}_k + \alpha_k \mathbb{S}_k : \nabla \boldsymbol{v}_k \\
&\quad - \nabla \cdot (\alpha_k \boldsymbol{q}_k) + \alpha_k \rho_k \boldsymbol{g}_k \cdot \boldsymbol{v}_k + \alpha_k \dot{q}_k + \widehat{E}_{\mathrm{i}\,k} \Gamma_{\mathrm{i}\,k} \\
&\quad + p_{\mathrm{i}\,k} \cdot \boldsymbol{v}_{\mathrm{i}\,k} \cdot \nabla \alpha_k - \mathbb{S}_{\mathrm{i}\,k} \cdot \boldsymbol{v}_{\mathrm{i}\,k} \cdot \nabla \alpha_k + \boldsymbol{q}_{\mathrm{i}\,k} \cdot \nabla \alpha_k .
\end{aligned}
\tag{2.31}
$$

▶ Boundary conditions

For the flow of a medium through a heated tube we need to consider the friction of the fluid at the wall, and the heat transfer from the heated tube wall to the fluid. In the underlying problem, as external source the concentrated sun irradiation is used which is reflected on the surface of several mirrors: for solar tower power plants heliostats are used, for parabolic trough collectors one mirror with a parabolic shape is used, and for linear Fresnel collectors several flat (or slightly curved) mirrors are used. For all three concentration techniques, a non-uniform solar flux around the receiver tube is obtained.

The heat transfer from the heated tube to the fluid is modeled with the use of Fourier's heat diffusion equation for in-compressible medium,

$$
\partial_t T_{\mathrm{w}} = \frac{1}{c_{\mathrm{p\,w}} \rho_{\mathrm{w}}} \nabla \left(\lambda_{\mathrm{w}} \nabla T_{\mathrm{w}} \right) , \tag{2.32}
$$

with wall temperature T_{w}, thermal conductivity λ_{w}, constant specific heat capacity $c_{\mathrm{p\,w}}$ and density ρ_{w} of the material (here steel is used as material). The Laplacian describes the change of the heat transfer density through the tube wall, which can be understood as the difference between the *external* heat transfer density (from the sun into the wall) and the *wall* heat transfer density (from the wall into the fluid). The latter one is used as source term for the energy balance law.

2.2. Simplifications and closures

First of all, the complexity of the model is reduced by reducing the dimension. Subsequently the system is closed with additional constitutive equations where we have to ensure that the balance laws conserve mass, momentum and energy of the total mixture.

▶ Dimension reduction

Due to the difference in spatial scale of tube diameter (some centimeters) versus tube length (several hundreds of meters) and in order to reduce the complexity of the problem, we model the two-phase as one-dimensional flow in intended flow direction.

Thus, the velocity and heat flux vectors $\boldsymbol{v}_k$, $\boldsymbol{v}_{\mathrm{i}\,k}$, $\widehat{\boldsymbol{v}}_{\mathrm{i}\,k}$, $\boldsymbol{q}_k$, $\boldsymbol{q}_{\mathrm{i}\,k}$ are reduced to the one-dimensional values v_k, $v_{\mathrm{i}\,k}$, $\widehat{v}_{\mathrm{i}\,k}$, q_k, $q_{\mathrm{i}\,k}$. The gravitational body force $\boldsymbol{g}_k$ is reduced to the one-dimensional quantity

$$g \sin(\vartheta), \tag{2.33}$$

with gravitational acceleration $g = 9.81$ m/s and elevation angle ϑ of the tube with respect to the horizontal.

▶ Volume fraction

With (2.6) and (2.13) the liquid void fraction α_ℓ can be expressed by the vapor void fraction α_g. Because α_ℓ is obsolete, we directly define

$$\alpha := \alpha_g = 1 - \alpha_\ell. \tag{2.34}$$

We assume that the spatial change of the volume fraction happens with the same speed, such that

$$v_{\mathrm{i}} := v_{\mathrm{i}\,g} = v_{\mathrm{i}\,\ell}. \tag{2.35}$$

This equality agrees with the assumption of Gallouët, Hérard and Seguin [8]. Due to total conservation of volume fraction, the phasic production terms of the void fraction balance equations have to sum to zero:

$$-\frac{\Gamma_{\mathrm{i}\,\ell}}{\widehat{\rho}_{\mathrm{i}\,\ell}} \overset{!}{=} \frac{\Gamma_{\mathrm{i}\,g}}{\widehat{\rho}_{\mathrm{i}\,g}}. \tag{2.36}$$

Below in (2.38) we see that the interfacial mass flow rate of one phase must be equal to the loss of the other phase, which finally leads to the equality of the source densities,

$$\widehat{\rho}_{\mathrm{i}} := \widehat{\rho}_{\mathrm{i}\,\ell} = \widehat{\rho}_{\mathrm{i}\,g}. \tag{2.37}$$

▶ Total conservation of mass at the interface

Due to the conservation of mass in the mixture, we claim that the convective interfacial mass production of one phase is equal to the loss of the other phase [72]. Because $\Gamma_{\mathrm{i}\,\ell}$ is obsolete we directly define

$$\Gamma_{\mathrm{i}} := \Gamma_{\mathrm{i}\,g} = -\Gamma_{\mathrm{i}\,\ell}. \tag{2.38}$$

▶ Total conservation of momentum at the interface

The viscous shear stress term $\nabla \cdot (\alpha_k \mathbb{S}_k)$ describes the friction density between the two phases, considering the form drag, lift force and skin drag. This is modeled by a constitutive equation,

$$F_{\mathrm{i}\,k} := \nabla \cdot (\alpha_k \mathbb{S}_k). \tag{2.39}$$

Following [73], interfacial viscous stresses are neglected for the one-dimensional case,

$$\mathbb{S}_{\mathrm{i}\,k} \equiv 0. \tag{2.40}$$

So finally, the 1D momentum balance equations are given by

$$\begin{aligned} \partial_t(\alpha_k \rho_k v_k) + \partial_x \cdot (\alpha_k \rho_k v_k v_k) \;=\; & -\partial_x \cdot (\alpha_k p_k) + F_{\mathrm{i}\,k} + \alpha_k \rho_k \, g \, \sin(\vartheta) \\ & + \widehat{v}_{\mathrm{i}\,k} \Gamma_{\mathrm{i}\,k} + p_{\mathrm{i}\,k} \, \partial_x \alpha_k. \end{aligned} \tag{2.41}$$

Due to the conservation of momentum at the interface, we expect that the interfacial forces of both phases sum to zero, where the external forces $\alpha_k \rho_k \, g \, \sin(\vartheta)$ are not taken into account. Therefore we claim that the force terms associated with interface mass and momentum exchange sum to zero [73],

$$(F_{\mathrm{i}\,\ell} + F_{\mathrm{i}\,g}) + (\widehat{v}_{\mathrm{i}\,g} - \widehat{v}_{\mathrm{i}\,\ell}) \cdot \Gamma_{\mathrm{i}} + (p_{\mathrm{i}\,g} - p_{\mathrm{i}\,\ell}) \, \partial_x \alpha \overset{!}{=} 0. \tag{2.42}$$

We assume that this happens independently for each term, such that interfacial friction density of one phase is equal to the loss of the other phase, the source velocities must be equal, and the interfacial pressures must be equal,

$$F_{\mathrm{i}} := -F_{\mathrm{i}\,g} = F_{\mathrm{i}\,\ell}, \quad \widehat{v}_{\mathrm{i}} := \widehat{v}_{\mathrm{i}\,g} = \widehat{v}_{\mathrm{i}\,\ell}, \quad \text{and} \quad p_{\mathrm{i}} := p_{\mathrm{i}\,g} = p_{\mathrm{i}\,\ell}. \tag{2.43}$$

This equality agrees with the assumption of Gallouët, Hérard and Seguin [8].

▶ Total conservation of total energy at the interface

The viscous shear stress term $\nabla \cdot (\alpha_k \mathbb{S}_k) \cdot \boldsymbol{v}_k$ describes the specific frictional heat between the two phases. As already suggested above in (2.39), this term is given by a constitutive equation.

$$F_{\mathrm{i}\,k} \cdot v_k = \nabla \cdot (\alpha_k \mathbb{S}_k) \cdot \boldsymbol{v}_k. \tag{2.44}$$

The divergence term $\alpha_k \mathbb{S}_k : \nabla \boldsymbol{v}_k$ is describing the frictional heat of phase k with the tube wall due to shear forces. We follow [73] who assume that the frictional work due to shear forces fully provides heat to the fluid, such that the term vanishes:

$$\alpha_k \mathbb{S}_k : \nabla \boldsymbol{v}_k \equiv 0. \tag{2.45}$$

With (2.40), also the term $\mathbb{S}_{\mathrm{i}\,k} \cdot \boldsymbol{v}_{\mathrm{i}\,k} \cdot \nabla \alpha_k$ vanishes.

The second order term $\nabla \cdot (\alpha_k \boldsymbol{q}_k)$ describes the heat transfer rate density between the two phases. This quantity is modelled by a constitutive equation,

$$-\nabla \cdot (\alpha_k q_k) =: Q_{\mathrm{i}\,k} \tag{2.46}$$

Following Ishii and Hibiki [69] and Drew and Passman [61], the body heating $\dot{q}_k$ and the the interfacial energy flux $q_{\mathrm{i}\,k}$ are negligibly small in comparison with the heat flux q_k of the fluid. Therefore we assume

$$\begin{aligned} \dot{q}_k &\equiv 0, \\ q_{\mathrm{i}\,k} &\equiv 0. \end{aligned} \tag{2.47}$$

So finally, the 1D energy balance equations are given by

$$\begin{aligned} \partial_t(\alpha_k \rho_k E_k) + \nabla \cdot (\alpha_k \rho_k \boldsymbol{v}_k E_k) \;=\; & -\nabla \cdot (\alpha_k v_k p_k) + F_{\mathrm{i}\,k} v_k + Q_{\mathrm{i}\,k} \\ & + \alpha_k \rho_k v_k \, g \, \sin(\vartheta) + \widehat{E}_{\mathrm{i}\,k} \Gamma_{\mathrm{i}\,k} + p_{\mathrm{i}} v_{\mathrm{i}} \cdot \nabla \alpha_k. \end{aligned}$$

Due to the conservation of total energy at the interface, we expect that the interfacial forces of both phases sum to zero, where the external forces $\alpha_k \rho_k v_k \, g \, \sin(\vartheta)$ are not taken into account. Therefore we claim that the force terms associated with interface mass and energy exchange sum to zero [73],

$$(F_{\mathrm{i}} v_\ell - F_{\mathrm{i}} v_g) + (Q_{\mathrm{i}\,g} + Q_{\mathrm{i}\,\ell}) + (\widehat{E}_{\mathrm{i}\,g} - \widehat{E}_{\mathrm{i}\,\ell})\Gamma_{\mathrm{i}} + (p_{\mathrm{i}} v_{\mathrm{i}} - p_{\mathrm{i}} v_{\mathrm{i}}) \cdot \nabla \alpha = 0 \tag{2.48}$$

If we choose the interphase mass flow rate as

$$\Gamma_{\mathrm{i}} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}} \big(F_{\mathrm{i}}(v_\ell - v_g) + Q_{\mathrm{i}\,\ell} + Q_{\mathrm{i}\,g} \big), \tag{2.49}$$

the conservation of total energy is satisfied. This choice agrees with Berry et al. [74] who call this a heat conduction limited model. The interphasic specific total energies $\widehat{E}_{\mathrm{i}\,\ell}$ and $\widehat{E}_{\mathrm{i}\,g}$ are chosen in Section 2.5.1, where we ensure that both quantities are distinct[9].

▶ Boundary conditions

Due to the changes of the metric, also the original boundary conditions need to be adjusted:

- The momentum balance equation gets an additional wall friction density term $F_{\mathrm{w}\,k}$, which causes an additional dissipative heat transfer rate density $v_k F_{\mathrm{w}\,k}$ in the energy balance equation.

[9] Later, for each time step it has to be ensured that Γ_{i} does not represent a larger mass of liquid (steam) than is available to boil (condensate).

- As external source the sun irradiation is concentrated onto the tube, which generates a heat transfer from the tube into the fluid. Thus, Fourier's heat diffusion equation from (2.32) is approximated by [63]

$$\partial_t T_\mathrm{w} = \frac{1}{c_{\mathrm{p\,w}}\rho_\mathrm{w}} \left(Q_\mathrm{ext} - Q_\mathrm{loss} - Q_{\mathrm{w}\,k}\right), \tag{2.50}$$

  where Q_ext describes the external heat transfer rate density coming from the sun, Q_loss are the losses at the outer tube wall due to convection and radiation, and $Q_{\mathrm{w}\,k}$ is the heat transfer rate density through the tube wall into phase k of the fluid. The latter term $Q_{\mathrm{w}\,k}$ is used as additional heat transfer rate density in the energy balance equation.

▶ Additional modeled source terms

The heat transfer rate density through the tube wall $Q_{\mathrm{w}\,k}$ causes an exchange between both phases: Due to the hot wall, boiling or condensation appears for parts of the fluid near the wall [73]. Therefore, $Q_{\mathrm{w}\,k}$ is partitioned into a boiling and convection part,

$$Q_{\mathrm{w}\,k} = Q_{\mathrm{w}\,k}^\mathrm{boil} + Q_{\mathrm{w}\,k}^\mathrm{conv}. \tag{2.51}$$

For the vapor phase, the boiling part is always zero. Because of this near-wall boiling, additional void fraction $\frac{\Gamma_{\mathrm{w}\,k}}{\widehat{\rho}_\mathrm{i}}$, mass $\Gamma_{\mathrm{w}\,k}$, momentum $\widehat{v}_\mathrm{i}\Gamma_{\mathrm{w}\,k}$, and energy $\widehat{E}_{\mathrm{i}\,k}\Gamma_{\mathrm{w}\,k}$ is generated. As already discussed in (2.38), due to the conservation of mass at the near-wall interface the interfacial mass flow rate $\Gamma_{\mathrm{w}\,k}$ near the wall must hold

$$\Gamma_\mathrm{w} := \Gamma_{\mathrm{w}\,g} = -\Gamma_{\mathrm{w}\,\ell}. \tag{2.52}$$

The rate of vapor generated by boiling at the wall due to wall heat flux is then

$$\Gamma_\mathrm{w} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}} \big(- Q_{\mathrm{w}\,\ell}^\mathrm{boil} - Q_{\mathrm{w}\,g}^\mathrm{boil}\big). \tag{2.53}$$

▶ Closure of the pressure

A model for the pressure is needed. The phase rule of Gibbs [75] says that thermal properties of a fluid are described by two independent, intensive properties. Because our system delivers the density ρ and specific internal energy u, the fluid equations are supplemented by the following equation of state

$$p_k = p_k(\rho_k, u_k). \tag{2.54}$$

The following relation holds,

$$\mathrm{d}p_k = \underbrace{\left.\frac{\partial p_k}{\partial \rho_k}\right|_{u_k}}_{=:(p_k)_\rho} \mathrm{d}\rho_k + \underbrace{\left.\frac{\partial p_k}{\partial u_k}\right|_{\rho_k}}_{=:(p_k)_u} \mathrm{d}u_k, \tag{2.55}$$

where the derivatives of p_k with respect to the two state variables depend on the underlying equation of state, e.g. Tait, stiffened gas, or real gas, see Section 2.4.1. Incidentally, the speed of sound c can be expressed in dependency of these derivatives [76],

$$c_k = \sqrt{\left.\frac{\partial p_k}{\partial \rho_k}\right|_{s_k}} = \sqrt{(p_k)_\rho + (p_k)_u \cdot \frac{p_k}{\rho_k^2}}. \tag{2.56}$$

2.3. Two-velocity two-pressure seven-equations model

With the above introduced simplifications, the following one-dimensional balance laws describe the *two-velocity two-pressure seven-equations* model:

$$
\begin{aligned}
&\partial_t \alpha_g && && + v_\mathrm{i}\,\partial_x \alpha_g && = \frac{\Gamma_\mathrm{i}+\Gamma_\mathrm{w}}{\widehat{\rho}_\mathrm{i}} \\
&\partial_t(\alpha_\ell\rho_\ell) && + \partial_x(\alpha_\ell\rho_\ell v_\ell) && && = -(\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&\partial_t(\alpha_\ell\rho_\ell v_\ell) && + \partial_x\big(\alpha_\ell(\rho_\ell v_\ell^2+p_\ell)\big) && + p_\mathrm{i}\,\partial_x\alpha_g && = \alpha_\ell\rho_\ell\, g\,\sin(\vartheta) - \widehat{v}_\mathrm{i}(\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&&&&&&& \quad + F_\mathrm{i} - F_{\mathrm{w}\,\ell} \\
&\partial_t(\alpha_\ell\rho_\ell E_\ell) && + \partial_x\big(\alpha_\ell v_\ell(\rho_\ell E_\ell+p_\ell)\big) && + p_\mathrm{i} v_\mathrm{i}\,\partial_x\alpha_g && = \alpha_\ell\rho_\ell v_\ell\, g\,\sin(\vartheta) - \widehat{E}_{\mathrm{i}\,\ell}(\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&&&&&&& \quad + F_\mathrm{i} v_\ell - F_{\mathrm{w}\,\ell} v_\ell + Q_{\mathrm{i}\,\ell} + Q^{\mathrm{conv}}_{\mathrm{w}\,\ell} \\
&\partial_t(\alpha_g\rho_g) && + \partial_x(\alpha_g\rho_g v_g) && && = (\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&\partial_t(\alpha_g\rho_g v_g) && + \partial_x\big(\alpha_g(\rho_g v_g^2+p_g)\big) && - p_\mathrm{i}\,\partial_x\alpha_g && = \alpha_g\rho_g\, g\,\sin(\vartheta) + \widehat{v}_\mathrm{i}(\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&&&&&&& \quad - F_\mathrm{i} - F_{\mathrm{w}\,g} \\
&\partial_t(\alpha_g\rho_g E_g) && + \partial_x\big(\alpha_g v_g(\rho_g E_g+p_g)\big) && - p_\mathrm{i} v_\mathrm{i}\,\partial_x\alpha_g && = \alpha_g\rho_g v_g\, g\,\sin(\vartheta) + \widehat{E}_{\mathrm{i}\,g}(\Gamma_\mathrm{i}+\Gamma_\mathrm{w}) \\
&&&&&&& \quad - F_\mathrm{i} v_g - F_{\mathrm{w}\,g} v_g + Q_{\mathrm{i}\,g} + Q^{\mathrm{conv}}_{\mathrm{w}\,g}.
\end{aligned}
\tag{2.57}
$$

These equations are generic for all materials or fluids. In order to apply the system to liquid water and steam, additional closure equations have to be defined. Therefore, in subsection 2.4, the equation of state for water and the closure conditions for fricition density $F_{\mathrm{w}\,k}$, F_i and heat transfer rate density $Q_{\mathrm{i}\,k}$, $Q^{\mathrm{conv}}_{\mathrm{w}\,k}$, $Q^{\mathrm{boil}}_{\mathrm{w}\,k}$ are developed. The algebraic expressions, which base on measured data, are used to restore at least some information that was lost in the averaging routine to simplify the model equations [74].

Because the equations for liquid and vapor volume fraction (first and fifth equation) are describing the same variable, one equation can be neglected, see also (2.13) and (2.34). Therefore we just use the vapor volume fraction with $\alpha_g := \alpha$. The model can be written in the general non-conservative form [77],

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\,\partial_x \mathbf{u} = \mathbf{s}(\mathbf{u}), \tag{2.58}$$

with independent variables $\mathbf{u}$ and flux vector $\mathbf{f}(\mathbf{u})$

$$\mathbf{u} := \begin{pmatrix} \alpha \\ (1-\alpha)\rho_\ell \\ (1-\alpha)\rho_\ell v_\ell \\ (1-\alpha)\rho_\ell E_\ell \\ \alpha\rho_g \\ \alpha\rho_g v_g \\ \alpha\rho_g E_g \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} 0 \\ (1-\alpha)\rho_\ell v_\ell \\ (1-\alpha)(\rho_\ell v_\ell^2 + p_\ell) \\ (1-\alpha)(\rho_\ell E_\ell + p_\ell)v_\ell \\ \alpha\rho_g v_g \\ \alpha(\rho_g v_g^2 + p_g) \\ \alpha(\rho_g E_g + p_g)v_g \end{pmatrix}, \tag{2.59}$$

and non-conservative system matrix $B(\mathbf{u})$

$$B(\mathbf{u}) = \begin{pmatrix} v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ p_\mathrm{i} v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -p_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ -p_\mathrm{i} v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{2.60}$$

The source vector $\mathbf{s}(\mathbf{u}) = \mathbf{s}_\mathrm{i}(\mathbf{u}) + \mathbf{s}_\mathrm{w}(\mathbf{u})$ is split into internal sources and external sources, with

$$\mathbf{s}_\mathrm{i}(\mathbf{u}) = \begin{pmatrix} \frac{\Gamma_\mathrm{i}}{\widehat{\rho}_\mathrm{i}} \\ -\Gamma_\mathrm{i} \\ -\widehat{v}_\mathrm{i}\Gamma_\mathrm{i} + F_\mathrm{i} \\ -\widehat{E}_{\mathrm{i}\,\ell}\Gamma_\mathrm{i} + F_\mathrm{i} v_\ell + Q_{\mathrm{i}\,\ell} \\ \Gamma_\mathrm{i} \\ \widehat{v}_\mathrm{i}\Gamma_\mathrm{i} - F_\mathrm{i} \\ \widehat{E}_{\mathrm{i}\,g}\Gamma_\mathrm{i} - F_\mathrm{i} v_g + Q_{\mathrm{i}\,g} \end{pmatrix}, \tag{2.61}$$

and

$$\mathbf{s}_\mathrm{w}(\mathbf{u}) = \begin{pmatrix} \frac{\Gamma_\mathrm{w}}{\widehat{\rho}_\mathrm{i}} \\ -\Gamma_\mathrm{w} \\ \alpha_\ell \rho_\ell\, g\, \sin(\vartheta) - \widehat{v}_\mathrm{i}\Gamma_\mathrm{w} - F_{\mathrm{w}\,\ell} \\ \alpha_\ell \rho_\ell v_\ell\, g\, \sin(\vartheta) - \widehat{E}_{\mathrm{i}\,\ell}\Gamma_\mathrm{w} - F_{\mathrm{w}\,\ell} v_\ell + Q Q^{\mathrm{conv}}_{\mathrm{w}\,\ell} \\ \Gamma_\mathrm{w} \\ \alpha_g \rho_g\, g\, \sin(\vartheta) + \widehat{v}_\mathrm{i}\Gamma_\mathrm{w} - F_{\mathrm{w}\,g} \\ \alpha_g \rho_g v_g\, g\, \sin(\vartheta) + \widehat{E}_{\mathrm{i}\,g}\Gamma_\mathrm{w} - F_{\mathrm{w}\,g} v_g + Q^{\mathrm{conv}}_{\mathrm{w}\,g} \end{pmatrix}. \tag{2.62}$$

All solutions for $\mathbf{u}$ are in the set of admissible states,

$$\Omega = \{\mathbf{u} \in \mathbb{R}^n \mid \alpha \in (0,1), \rho_k > 0, u_k > 0, c_k > 0\} \tag{2.63}$$

with $n = 7$, where u_k is the specific internal energy, given by the physical law

$$u_k = E_k - \tfrac{1}{2} v_k^2. \tag{2.64}$$

To ensure that both phases are always everywhere present, the volume fraction α needs to be in the open intervall from 0 to 1. For a transformation of the system in a closed quasilinear form

$$\partial_t \mathbf{u} + A(\mathbf{u})\, \partial_x \mathbf{u} = \mathbf{s}(\mathbf{u}), \tag{2.65}$$

with system matrix

$$A(\mathbf{u}) = \partial_{\mathbf{u}} \mathbf{f}(\mathbf{u}) + B(\mathbf{u}), \tag{2.66}$$

we need to differentiate the flux vector $\mathbf{f}(\mathbf{u})$ with respect to our independent variables $\mathbf{u}$. Therefore we need the partial derivatives of the primitive variables,

$$\begin{aligned}
\partial_x \rho_k &= -\frac{\rho_k}{\alpha_k}\, \partial_x \alpha_k + \frac{1}{\alpha_k}\, \partial_x(\alpha_k \rho_k) \\
\partial_x v_k &= -\frac{v_k}{\alpha_k \rho_k}\, \partial_x(\alpha_k \rho_k) \frac{1}{\alpha_k \rho_k}\, \partial_x(\alpha_k \rho_k v_k) \\
\partial_x u_k &= -\frac{E_k - v_k^2}{\alpha_k \rho_k}\, \partial_x(\alpha_k \rho_k) - \frac{v_k}{\alpha_k \rho_k}\, \partial_x(\alpha_k \rho_k v_k) + \frac{1}{\alpha_k \rho_k}\, \partial_x(\alpha_k \rho_k E_k)
\end{aligned} \tag{2.67}$$

and the partial derivatives of the pressure, which can be derived from the pressure relation (2.55), such that

$$\begin{aligned}
\partial_x(\alpha_k p_k) &= p_k \partial_x \alpha_k + \alpha_k \partial_x p_k \\
&\overset{(2.55)}{=} p_k \partial_x \alpha_k + \alpha_k \cdot (p_k)_\rho\, \partial_x \rho_k + \alpha_k \cdot (p_k)_u\, \partial_x u_k \\
&\overset{(2.67)}{=} \left(p_k - \rho_k \cdot (p_k)_\rho\right) \partial_x \alpha_k + \left((p_k)_\rho - (p_k)_u \cdot \frac{E_k - v_k^2}{\rho_k} \right) \partial_x(\alpha_k \rho_k) \\
&\quad - \frac{(p_k)_u \cdot v_k}{\rho_k}\, \partial_x(\alpha_k \rho_k v_k) + \frac{(p_k)_u}{\rho_k}\, \partial_x(\alpha_k \rho_k E_k),
\end{aligned} \tag{2.68}$$

$$\begin{aligned}
\partial_x(\alpha_k v_k p_k) &= v_k\, \partial_x(\alpha_k p_k) + \alpha_k p_k \partial_x v_k \\
&\overset{(2.68)}{=} v_k \left(p_k - \rho_k \cdot (p_k)_\rho\right) \partial_x \alpha_k \\
&\quad + v_k \left((p_k)_\rho - (p_k)_u \cdot \frac{E_k - v_k^2}{\rho_k} - \frac{p_k}{\rho_k} \right) \partial_x(\alpha_k \rho_k) \\
&\quad + \frac{p_k - (p_k)_u \cdot v_k^2}{\rho_k}\, \partial_x(\alpha_k \rho_k v_k) + \frac{v_k \cdot (p_k)_u}{\rho_k}\, \partial_x(\alpha_k \rho_k E_k).
\end{aligned} \tag{2.69}$$

Then we can compute the derivative of $\mathbf{f}(\mathbf{u})$ with respect to $\mathbf{u}$, such that the system matrix $A(\mathbf{u})$ is given by:

$$A(\mathbf{u}) = \partial_{\mathbf{u}}\mathbf{f}(\mathbf{u}) + B(\mathbf{u}) =$$
$$\begin{pmatrix}
v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
-\left(p_\ell - p_{\mathrm{i}} - \rho_\ell \cdot (p_\ell)_\rho\right) & A_{\mathrm{mom},\rho_\ell} & A_{\mathrm{mom},\rho_\ell v_\ell} & A_{\mathrm{mom},\rho_\ell E_\ell} & 0 & 0 & 0 \\
-\left(p_\ell v_\ell - p_{\mathrm{i}} v_{\mathrm{i}} - \rho_\ell v_\ell \cdot (p_\ell)_\rho\right) & A_{\mathrm{ener},\rho_\ell} & A_{\mathrm{ener},\rho_\ell v_\ell} & A_{\mathrm{ener},\rho_\ell E_\ell} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
p_g - p_{\mathrm{i}} - \rho_g \cdot (p_g)_\rho & 0 & 0 & 0 & A_{\mathrm{mom},\rho_g} & A_{\mathrm{mom},\rho_g v_g} & A_{\mathrm{mom},\rho_g E_g} \\
p_g v_g - p_{\mathrm{i}} v_{\mathrm{i}} - \rho_g v_g \cdot (p_g)_\rho & 0 & 0 & 0 & A_{\mathrm{ener},\rho_g} & A_{\mathrm{ener},\rho_g v_g} & A_{\mathrm{ener},\rho_g E_g}
\end{pmatrix}, \tag{2.70}$$

with

$$\begin{aligned}
A_{\mathrm{mom},\rho_k} &:= -v_k^2 + (p_k)_\rho - (p_k)_u \cdot \frac{E_k - v_k^2}{\rho_k} \\
A_{\mathrm{mom},\rho_k v_k} &:= 2v_k + \frac{-(p_k)_u \cdot v_k}{\rho_k} \\
A_{\mathrm{mom},\rho_k E_k} &:= \frac{(p_k)_u}{\rho_k} \\
A_{\mathrm{ener},\rho_k} &:= v_k \left((p_k)_\rho - (p_k)_u \cdot \frac{E_k - v_k^2}{\rho_k} - \frac{p_k}{\rho_k} - E_k \right) \\
A_{\mathrm{ener},\rho_k v_k} &:= E_k + \frac{p_k - (p_k)_u \cdot v_k^2}{\rho_k} \\
A_{\mathrm{ener},\rho_k E_k} &:= v_k + \frac{v_k \cdot (p_k)_u}{\rho_k}
\end{aligned} \tag{2.71}$$

2.4. Two-phase flow constitutive models

The source vector $\mathbf{s}(\mathbf{u})$ depends on thermodynamical models, which describe the exchange in-between the phases and between the fluid and the wall (subscripts $_{\mathrm{i}}$ and $_{\mathrm{w}}$). Within this subsection constitutive models for the equation of state, tube wall friction density $F_{\mathrm{w}\,k}$, interfacial friction density F_{i}, wall heat transfer rate density $Q_{\mathrm{w}\,k}$ with $Q_{\mathrm{w}\,k}^{\mathrm{boil}}$, $Q_{\mathrm{w}\,k}^{\mathrm{conv}}$, and interfacial heat transfer rate density in the bulk $Q_{\mathrm{i}\,k}$, are presented.

2.4.1. Equation of state

As already introduced at the end of Section 2.1, we use the density ρ and specific internal energy u as independent, intensive properties to provide each phase with thermal properties as equation of state (EOS). This means that we use these two quantities to compute other thermodynamical properties, e.g. pressure and temperature. In the following different EOS models are presented.

▶ Stiffened gas EOS

For a compressible fluid the following stiffened gas equation of state was performed for pressure, temperature, speed of sound and specific Gibbs free energy:

$$\begin{aligned} p(\rho,u) &= (\gamma-1)\rho(u-q)-\gamma\pi, \\ T(\rho,u) &= \frac{1}{c_{\mathrm{v}}}\left(u-q-\frac{\pi}{\rho}\right), \\ c(\rho,u) &= \sqrt{\gamma\,\frac{p+\pi}{\rho}}, \\ g(\rho,u) &= g(p,T) = (\gamma c_{\mathrm{v}}-q')T - c_{\mathrm{v}}T\log\left(\frac{T^{\gamma}}{(p+\pi)^{\gamma-1}}\right)+q. \end{aligned} \tag{2.72}$$

The five constants, $\gamma > 0$, $q \geq 0$ as binding energy, $c_{\mathrm{v}} \geq 0$ as specific heat at constant volume, $\pi \geq 0$, and $q' \geq 0$ are given for each fluid particularly. The derivatives of the pressure and the temperature with respect to ρ and u are given by

$$\left.\frac{\partial p}{\partial \rho}\right|_u = (\gamma-1)(u-q), \qquad \left.\frac{\partial p}{\partial u}\right|_\rho = (\gamma-1)\rho, \qquad \left.\frac{\partial T}{\partial \rho}\right|_u = \frac{\pi}{c_{\mathrm{v}}\rho^2} \quad \text{and} \quad \left.\frac{\partial T}{\partial u}\right|_\rho = \frac{1}{c_{\mathrm{v}}}. \tag{2.73}$$

▶ Ideal gas EOS

The ideal gas equation of state is performed for calorically ideal gas. It is a particular form of the stiffened gas equation of state with $q = \pi = 0$.

▶ Data-based EOS for water and steam

For water and steam the constitutive equations of the IAPWS[10] industrial formulation

[10] International Association for the Properties of Water and Steam

1997 [78] can be used. These equations of state base on a fundamental equation for the specific Gibbs free energy. In dependency of the pressure p and the temperature T all other thermal properties are computed, where the following range of validity is covered:

$$\begin{aligned} 273.15\ \mathrm{K} \leq T \leq 1073.15\ \mathrm{K},\ \text{for } p \leq 100\ \mathrm{MPa} \\ 1073.15\ \mathrm{K} < T \leq 2273.15\ \mathrm{K},\ \text{for } p \leq 50\ \mathrm{MPa}. \end{aligned}$$

But the equations of state cannot be closed explicitly for given density ρ and specific inner energy u. Thus an inner loop is required to determine the temperature T and the pressure p using a Newton-Raphson procedure. By using the temperature $T_0 = 700$ K as initial guess the procedure converges within approximately three to five iterations to a residual of the order 10^{-8}. Finally, we get the expression

$$p = p(\rho, u) \ \text{ and } \ T = T(\rho, u).$$

The derivatives of p_k with respect to the two state variables are given by [76]

$$\begin{aligned} \left.\frac{\partial p}{\partial \rho}\right|_u &= \frac{p^2 \alpha_\mathrm{p}^2 T + p \beta_\mathrm{p} c_\mathrm{v} - p^2 \alpha_\mathrm{p}}{c_\mathrm{v} \rho^2}, \\ \left.\frac{\partial p}{\partial u}\right|_\rho &= \frac{p \alpha_\mathrm{p}}{c_\mathrm{v}}, \end{aligned} \tag{2.74}$$

for temperature T, relative pressure coefficient α_p, isothermal stress coefficient β_p and specific isochoric heat capacity c_v. Incidentally, the speed of sound can be expressed in dependency of these derivatives [76],

$$c(\rho, u) = \sqrt{\left.\frac{\partial p}{\partial \rho}\right|_s} = \sqrt{\left.\frac{\partial p}{\partial u}\right|_\rho + \left.\frac{\partial p}{\partial \rho}\right|_u \cdot \frac{p}{\rho^2}} = \sqrt{\frac{p^2 \alpha_\mathrm{p}^2 T + p \beta_\mathrm{p} c_\mathrm{v}}{c_\rho^2}}. \tag{2.75}$$

2.4.2. Interfacial mass transfer rate

The interfacial mass transfer rate is given by a heat conduction limited model. This means that the mass transfer rates in the bulk and near the wall are chosen in such a way that the conservation of total energy at the interface holds, see (2.49) and (2.53):

$$\Gamma_\mathrm{i} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}} \big(F_\mathrm{i}(v_\ell - v_g) + Q_{\mathrm{i}\,\ell} + Q_{\mathrm{i}\,g} \big),$$

and

$$\Gamma_\mathrm{w} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}} \big(- Q_{\mathrm{w}\,\ell}^{\mathrm{boil}} - Q_{\mathrm{w}\,g}^{\mathrm{boil}} \big).$$

2.4.3. Flow-regime

For flow in tubes, usually two flow regime maps are used: a horizontal and a vertical map. The horizontal volume flow regime map is valid for an inclination angle of $0° \leq |\vartheta| \leq 30°$, and consists of six flow regimes: stratified flow, bubble flow, slug (and plug) flow, interpolation flow, annular flow and mist flow, see Figure 33. The vertical volume flow regime map is valid for an inclination angle of $60° \leq |\vartheta| \leq 90°$ [73]. For tubes with an absolute inclination angle between 30° and 60° an interpolation region between horizontal and vertical flow regimes is used.

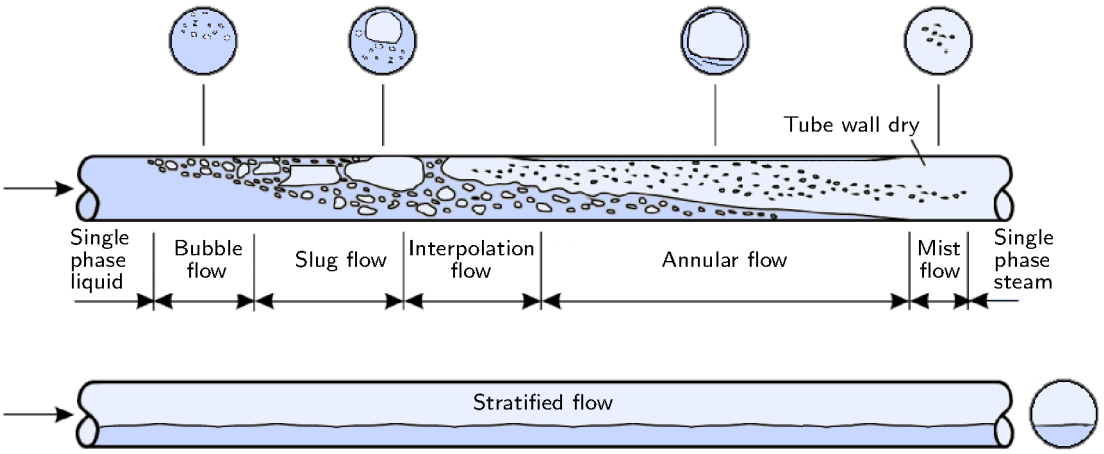


Figure 33: Two-phase flow patterns in an evaporated horizontal tube.
Sources: [10] and [11] with modifications

For the horizontal regime map, a flow pattern is defined by the vapor void fraction α_g, the relative velocity $|v_g - v_\ell|$ and the homogeneous mass flux G. For the vertical regime map, a flow pattern is defined by the vapor void fraction α_g, the average mixture velocity and the temperature (pre- or post-critical heat flux temperature).

2.4.4. Wall friction density

The description of the wall friction density is based on the Darcy-Weisbach equation [79],

$$F_{\mathrm{w}\,k} = \frac{1}{2}\rho_k v_k |v_k| \, f_{\mathrm{w}\,k} \, \frac{\alpha_{\mathrm{w}\,k}}{D_{\mathrm{in}}}. \tag{2.76}$$

The last term describes the specific wetted wall area, given by the wetted volume fraction in the wall film $\alpha_{\mathrm{w}\,k}$ and the inner tube diameter D_{in}. It holds $\alpha_{\mathrm{w}\,g} = 1 - \alpha_{\mathrm{w}\,\ell}$. The two parameters $\alpha_{\mathrm{w}\,k}$ and the wall friction factor $f_{\mathrm{w}\,k}$ depend on the flow pattern and need to be modelled related to the underlying problem.

2.4.5. Interphase friction density

According to [56, 73, 80] the interphase friction density for the liquid and vapor phase are calculated by the drag coefficient method:

$$F_{\mathrm{i}} \overset{(2.43)}{:=} -F_{\mathrm{i}\,g} = F_{\mathrm{i}\,\ell} = \frac{1}{2}\rho_{\mathrm{cont}}(v_g - v_\ell)|v_g - v_\ell| \cdot C_{\mathrm{D}} \cdot \frac{1}{4}a_{\mathrm{i}}. \tag{2.77}$$

In dependency of the flow pattern, the density of continuous phase ρ_{cont}, the specific interfacial area a_{i}, and the drag coefficient C_{D} have to be considered.

2.4.6. Wall temperature

The wall temperature is given as ordinary differential equation in (2.50). This can be solved numerically by simply using the explicit Euler method. Thus, the wall temperature at a new instant of time $t_{n+1} = t_n + \Delta t$ is given by

$$T_{\mathrm{w}}(t_{n+1}) = T_{\mathrm{w}}(t_n) + \frac{\Delta t}{c_{\mathrm{p\,w}}\rho_{\mathrm{w}}}\big(Q_{\mathrm{ext}}(t_n) - Q_{\mathrm{loss}}(t_n) - Q_{\mathrm{w}\,k}(t_n)\big). \tag{2.78}$$

The three heat transfer rate densities are evaluated at the old instant of time t_n. The external heat transfer rate density Q_{ext} is given by an optical model which depends on the collector system, e.g. [81, 82, 19]. To consider non-homogeneous flux around the tube wall, an azimuthal discretization of the tube with different temperatures and corresponding specific heat transfer coefficients in each tube segment has to be performed [63].

The convective and radiative losses Q_{loss} at the outer tube wall depend on the tube wall temperature. The Stefan-Boltzmann law states that irradiance losses are proportional to the fourth power of the temperature. According to this law, the losses can be described by [83],

$$Q_{\mathrm{loss}} = \frac{1}{\pi r_{\mathrm{out}}^2 - \pi r_{\mathrm{in}}^2} \cdot \Big(c_1(T_{\mathrm{w}} - 273.15[\mathsf{K}]) + c_4(T_{\mathrm{w}} - 273.15[\mathsf{K}])^4\Big). \tag{2.79}$$

where the first term describes the wall cross-section area. The constants c_1 and c_4 are particular for every receiver and must be determined by experiments. For example, for the absorber tubes used in the DISS[11] test facility the constant values are given by $c_1 = 0.16155\mathrm{W}\ /\ \mathrm{m\,K}$ and $c_4 = 6.4407 \cdot 10^{-9}\mathrm{W}\ /\ \mathrm{m\,K}^4$ [63].

Now, the heat transfer rate density $Q_{\mathrm{w}\,k}$ through the tube wall into phase k of the fluid is described.

2.4.7. Wall heat transfer rate density

The wall heat transfer rate density is approximated by an empirical law in dependency of the difference of the wall temperature T_{w} and the temperature of the fluid near the

[11] Direct Solar Steam is a European research project at the Plataforma Solar de Almería in Spain.

wall $T_{\mathrm{w}\,k}$ [84],

$$Q_{\mathrm{w}\,k} := \frac{\alpha_{\mathrm{w}\,k}}{D_{\mathrm{in}}} \cdot h_{\mathrm{T,w}\,k} \cdot \big(T_{\mathrm{w}} - T_{\mathrm{w}\,k}\big), \tag{2.80}$$

The first term describes the specific wetted wall area, given by the wetted volume fraction in the wall film $\alpha_{\mathrm{w}\,k}$ and the inner tube diameter D_{in}. The convective heat transfer coefficient $h_{\mathrm{T,w}\,k}$ from wall to phase k and the corresponding temperature $T_{\mathrm{w}\,k}$ in the wall film depend on Nukiyama's boiling curve [85], see Figure 34. Usually these parameters are determined under considering natural convection, forced convection, condensation, sub-cooled nucleate boiling, saturated nucleate boiling, and film boiling.

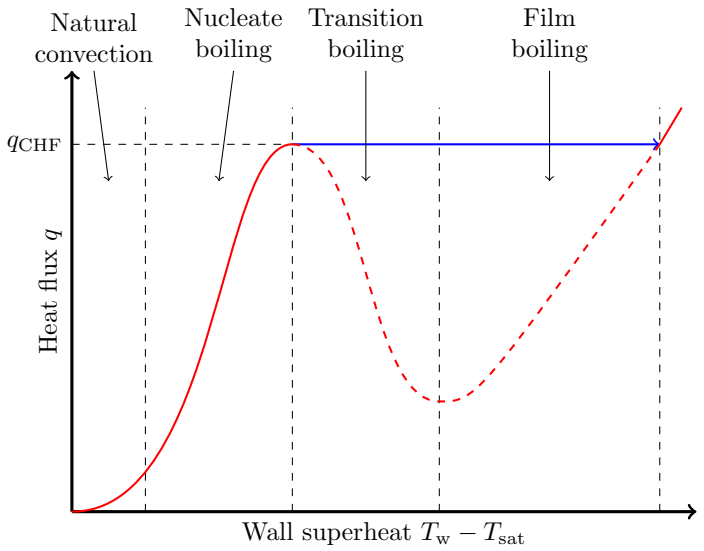


Figure 34: A typical boiling curve showing the variation of the heat flux as a function of the wall superheat, from [12] and [13].

The wall heat transfer rate density $Q_{\mathrm{w}\,k}$ can be partitioned into a boiling and convection part (2.51),

$$Q_{\mathrm{w}\,k} = Q_{\mathrm{w}\,k}^{\mathrm{boil}} + Q_{\mathrm{w}\,k}^{\mathrm{conv}}.$$

This is usually modeled as a fraction β_k which depends on the wall temperature and the saturated temperature,

$$Q_{\mathrm{w}\,k}^{\mathrm{boil}} = (1 - \beta_k)\, Q_{\mathrm{w}\,k} \quad \text{and} \quad Q_{\mathrm{w}\,k}^{\mathrm{conv}} = \beta_k\, Q_{\mathrm{w}\,k}. \tag{2.81}$$

2.4.8. Interphase heat transfer rate density (in the bulk)

The interfacial heat transfer rate density $Q_{\mathrm{i}\,k}$ resulted from bulk energy exchange is modeled as an approximation of Fourier's law of heat conduction,

$$Q_{\mathrm{i}\,k} = a_{\mathrm{i}}\, h_{\mathrm{T,i}\,k}\big(\widehat{T}_{\mathrm{i\,sat}} - T_k\big), \tag{2.82}$$

with fluid temperature T_k, source saturated temperature $\widehat{T}_{\mathrm{i\,sat}}$ and specific interfacial area a_{i} of droplets or bubbles. The corresponding convective heat transfer coefficient $h_{\mathrm{T,i}\,k}$ between the interphase and phase k depends on the flow pattern and on its change of aggregate state (either boiling or condensation). Usually the following models are used: Lee-Ryley [86], Plesset-Zwick [87], Unal [88], Lahey [89], Brown [90], Theofanous [91] and Dittus-Boelter [92].

2.5. Closures for the two-phase flow model

So far, for the two-phase flow model, the following parameters can be chosen arbitrarily: interfacial velocity v_i, interfacial pressure p_i, source velocity $\widehat{v}_\mathrm{i}$, source density $\widehat{\rho}_\mathrm{i}$, source specific total energies $\widehat{E}_{\mathrm{i}\,k}$, and source saturated temperature $\widehat{T}_{\mathrm{i\,sat}}$. The source parameters are modeled as an own phase.

To find closures for these parameters, we develop thermodynamical and mathematical properties which our models should fulfill. In Section 2.5.1 it is shown that the model is consistent with the second law of thermodynamics. The entropy compatibility condition delivers a closure for the interfacial pressure p_i, where the entropy dissipation helps to develop closures for the source parameters. In Section 2.5.2 the hyperbolicity of the model is examined, which delivers an additional constraint for the interfacial velocity v_i. This parameter is then closed in Section 2.5.3 by choosing it in such a way that the v_i-contact discontinuity is associated with a linearly degenerate field. Finally, in Section 2.5.4 it is verified that the system is symmetrizable.

2.5.1. Second law of thermodynamics

From a physical perspective, a model has to be consistent with the second law of thermodynamics. Therefore the entropy law is derived and the entropy production terms are determined to be non-negative. So, the entropy inequality can be considered as a restriction on the constitutive laws.

Due to the presence of shocks, there does not exist a continuous or smooth solution. Hence, the definition of a solution is weakened, so that also non-smooth solutions are allowed. These weak solutions form the basis of numerical solvers, for finding an approximate solution of the system (2.65).

An additional condition is required to select the physically relevant entropy solution [93]. Such a condition is called admissibility condition, or more often entropy condition in analogy with thermodynamics. The physical quantity called entropy is known to be constant in smooth flow and to jump to a higher value for arising shocks. It can never jump to a lower value. This is guaranteed by the second law of thermodynamics. The behavior of such a function can be used to test a weak solution for admissibility.

We have to find Lax's entropy-entropy flux pair (η, ψ) with an entropy function $\eta(\mathbf{u})$ and its corresponding entropy flux $\psi(\mathbf{u})$. For smooth solutions It should hold the additional conservation law,

$$\partial_t \eta(\mathbf{u}) + \partial_x \psi(\mathbf{u}) = 0. \tag{2.83}$$

Additionally we demand on the entropy function to be convex, $\eta''(\mathbf{u}) > 0$.Note that due to the demand, the mathematical entropy has an opposed behavior than the physical entropy. For discontinuous solutions $\mathbf{u}$ the above conservation law becomes an inequality,

$$\partial_t \eta(\mathbf{u}) + \partial_x \psi(\mathbf{u}) \leq 0. \tag{2.84}$$

A weak solution is said to be an *entropy solution*, if it satisfies the above inequality in the distributional sense. The entropy flux ψ is chosen such that it satisfies the compatibility condition

$$\partial_{\mathbf{u}}\psi(\mathbf{u})^{\mathsf{T}} \overset{!}{=} \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}} A(\mathbf{u}). \tag{2.85}$$

By multiplying our equation system $\partial_t \mathbf{u} + A(\mathbf{u})\,\partial_x \mathbf{u} = \mathbf{s}_{\mathrm{i}}(\mathbf{u})$ (without boundary conditions) from (2.65) with the so-called entropy variables $\mathbf{v}(\mathbf{u}) := \partial_{\mathbf{u}}\eta(\mathbf{u})$, the above entropy inequality is obtained

$$\partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}}\,\partial_t \mathbf{u} + \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}} A(\mathbf{u})\,\partial_x \mathbf{u} \overset{(2.85)}{=} \partial_t \eta(\mathbf{u}) + \partial_x \psi(\mathbf{u}) = \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}}\,\mathbf{s}_{\mathrm{i}}(\mathbf{u}) \overset{(2.84)}{\leq} 0. \tag{2.86}$$

So, the entropy inequality (2.84) holds iff the entropy production is negative, i.e.,

$$\partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}} \cdot \mathbf{s}_{\mathrm{i}}(\mathbf{u}) \leq 0. \tag{2.87}$$

For each model, the entropy-entropy flux pair $\eta(\mathbf{u})$ and $\psi(\mathbf{u})$ has to be chosen. Usually η depends on the physical specific entropy s of the fluid, which can be expressed by its physical law,

$$sT = u + \frac{p}{\rho} - g, \tag{2.88}$$

with g as specific Gibbs free energy. Due to the constitutive equation of state for the fluid (2.54) the specific entropy can be described as a function relating the density ρ and specific internal energy u. Its partial derivatives are then given by [76],

$$\left.\frac{\partial s}{\partial \rho}\right|_u = -\frac{p}{\rho^2 T} \quad \text{and} \quad \left.\frac{\partial s}{\partial u}\right|_\rho = \frac{1}{T}. \tag{2.89}$$

Due to the needed Hessian of the entropy function, we also need the derivatives of the temperature with respect to the two state variables, which are given by [76],

$$\left.\frac{\partial T}{\partial \rho}\right|_u = -\frac{p - \alpha_{\mathrm{p}} p T}{c_{\mathrm{v}} \rho^2} \quad \text{and} \quad \left.\frac{\partial T}{\partial u}\right|_\rho = \frac{1}{c_{\mathrm{v}}}, \tag{2.90}$$

with relative pressure coefficient α_{p} and specific isochoric heat capacity c_{v}.

In the following, we choose an entropy-entropy flux and show its properties to derive conditions for the freely selectable model parameters. The compatibility condition delivers an expression of the interfacial pressure. For the chosen entropy-entropy flux pair we need to show the convexity of the entropy function $\eta(\mathbf{u})$, the compatibility condition (2.85), and the entropy inequality of the entropy production terms (2.87).

▶ Entropy-entropy flux pair

Motivated by the thermodynamics of the system, a candidate for an entropy-entropy flux pair for our model is the physical entropy of the mixture,

$$\eta(\mathbf{u}) = -((1-\alpha)\rho_\ell s_\ell + \alpha \rho_g s_g) \quad \text{and} \quad \psi(\mathbf{u}) = -((1-\alpha)\rho_\ell v_\ell s_\ell + \alpha \rho_g v_g s_g). \tag{2.91}$$

The physical entropy s_k can be expressed by its physical law, see (2.88).

▶ Entropy compatibility condition

The entropy variables are given by

$$\mathbf{v}(\mathbf{u}) := \eta'(\mathbf{u}) = \begin{pmatrix} \frac{p_\ell}{T_\ell} - \frac{p_g}{T_g} \\ \frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} - s_\ell \\ \frac{v_\ell}{T_\ell} \\ -\frac{1}{T_\ell} \\ \frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} - s_g \\ \frac{v_g}{T_g} \\ -\frac{1}{T_g} \end{pmatrix} \overset{(2.88)}{=} \begin{pmatrix} \frac{p_\ell}{T_\ell} - \frac{p_g}{T_g} \\ \frac{g_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\ \frac{v_\ell}{T_\ell} \\ -\frac{1}{T_\ell} \\ \frac{g_g - \frac{1}{2}v_g^2}{T_g} \\ \frac{v_g}{T_g} \\ -\frac{1}{T_g} \end{pmatrix}. \tag{2.92}$$

The entropy compatibility condition (2.85) delivers a condition for the unknown interfacial pressure:

$$\begin{aligned} & \partial_{\mathbf{u}}\psi(\mathbf{u})^\mathsf{T} && \overset{!}{=} && \partial_{\mathbf{u}}\eta(\mathbf{u})^\mathsf{T} \cdot A(\mathbf{u}) \\ \Leftrightarrow \quad & \begin{pmatrix} \frac{p_\ell v_\ell}{T_\ell} - \frac{p_g v_g}{T_g} \\ v_\ell \cdot \frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\ \frac{v_\ell^2}{T_\ell} - s_\ell \\ -\frac{v_\ell}{T_\ell} \\ v_g \cdot \frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} \\ \frac{v_g^2}{T_g} - s_g \\ -\frac{v_g}{T_g} \end{pmatrix} && \overset{!}{=} && \begin{pmatrix} \frac{p_\ell v_\ell}{T_\ell} - \frac{p_g v_g}{T_g} + \frac{(p_g - p_\mathrm{i})(v_g - v_\mathrm{i})}{T_g} - \frac{(p_\ell - p_\mathrm{i})(v_\ell - v_\mathrm{i})}{T_\ell} \\ v_\ell \cdot \frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\ \frac{v_\ell^2}{T_\ell} - s_\ell \\ -\frac{v_\ell}{T_\ell} \\ v_g \cdot \frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} \\ \frac{v_g^2}{T_g} - s_g \\ -\frac{v_g}{T_g} \end{pmatrix}. \end{aligned} \tag{2.93}$$

Solving the first equation for the interfacial pressure, we get the expression

$$\boxed{p_\mathrm{i} := \frac{p_g T_\ell (v_g - v_\mathrm{i}) + p_\ell T_g (v_\mathrm{i} - v_\ell)}{T_\ell (v_g - v_\mathrm{i}) + T_g (v_\mathrm{i} - v_\ell)}.} \tag{2.94}$$

▶ Convexity of the entropy function

In order to verify that $\eta(\mathbf{u})$ is a convex function of $\mathbf{u}$, we follow the proof of [94]. The key idea is to exploit the fact that the phasic entropy functions,

$$S_k(\mathbf{u}_k) := -\rho_k s_k \quad \text{with} \quad \mathbf{u}_k^{\mathsf{T}} = (\rho_k, \rho_k v_k, \rho_k E_k) \tag{2.95}$$

are strictly convex functions. At first we showe that $S_k(\mathbf{u}_k)$ is a strictly convex function, if the specific internal energy u_k is a convex function of the specific volume $\tau_k := \frac{1}{\rho_k}$ and the specific entropy s_k. This can be shown by computing the Hessian matrix of the specific internal energy,

$$u''(\tau, s) = \begin{pmatrix} \frac{\partial^2 u}{\partial \tau^2} & \frac{\partial^2 u}{\partial \tau \partial s} \\ \frac{\partial^2 u}{\partial s \partial \tau} & \frac{\partial^2 u}{\partial s^2} \end{pmatrix} = \begin{pmatrix} \frac{p\beta_{\mathrm{p}} c_{\mathrm{v}} + \alpha_{\mathrm{p}}^2 p^2 T}{c_{\mathrm{v}}} & -\frac{\alpha_{\mathrm{p}} p T}{c_{\mathrm{v}}} \\ -\frac{\alpha_{\mathrm{p}} p T}{c_{\mathrm{v}}} & \frac{T}{c_{\mathrm{v}}} \end{pmatrix}. \tag{2.96}$$

for relative pressure coefficient α_{p}, isothermal stress coefficient β_{p} and specific isochoric heat capacity c_{v}. The quadratic and symmetric Hessian matrix is positive-definite, because the determinants associated with all upper-left sub-matrices are positive (Sylvester's criterion),

$$\frac{\partial^2 u}{\partial \tau^2} = \frac{p\beta_p c_v + \alpha_p^2 p^2 T}{c_v} > 0 \quad \text{and} \quad \det(u'') = \frac{\partial^2 u}{\partial \tau^2} \cdot \frac{\partial^2 u}{\partial s^2} - \left(\frac{\partial^2 u}{\partial \tau \partial s}\right)^2 = \frac{\beta_p p T}{c_v} > 0. \tag{2.97}$$

For stiffened gas, the pressure can be admissible. Thus (2.97) is a constraint for the pressure. Therefore u_k is a convex function, which implies the convexity of $S_k(\mathbf{u}_k)$. Each phase of the *two-velocity two-pressure seven-equations model* has the following convex entropy function:

$$S_k(\mathbf{u}_k) := -\rho_k s_k \quad \text{with} \quad \mathbf{u}_k^{\mathsf{T}} = (\rho_k, \rho_k v_k, \rho_k E_k) \tag{2.98}$$

The derivatives of $S_k(\mathbf{u}_k)$ are given by

$$S_k'(\mathbf{u}_k) = \begin{pmatrix} \frac{p_k + \rho_k(E_k - v_k^2)}{\rho_k T_k} - s_k \\ \frac{v_k}{T_k} \\ -\frac{1}{T_k} \end{pmatrix}, \tag{2.99}$$

and

$$S_k''(\mathbf{u}_k) = \begin{pmatrix} \frac{T_k (p_k)_\rho - \left(p_k + 2\rho_k(E_k - v_k^2)\right)(T_k)_\rho + \left(E_k - v_k^2\right)^2 (T_k)_u + v_k^2 T_k}{\rho_k T_k^2} & -v_k\left(S_{13}'' + \frac{1}{\rho_k T_k}\right) & S_{13}'' \\ -v_k\left(S_{13}'' + \frac{1}{\rho_k T_k}\right) & \frac{v_k^2 (T_k)_u + T_k}{\rho_k T_k^2} & \frac{-v_k (T_k)_u}{\rho_k T_k^2} \\ S_{13}'' & \frac{-v_k (T_k)_u}{\rho_k T_k^2} & \frac{(T_k)_u}{\rho_k T_k^2} \end{pmatrix}, \tag{2.100}$$

with $S''_{13} = \frac{\rho_k (T_k)_\rho - (E_k - v_k^2)(T_k)_u}{\rho_k T_k^2}$ The needed derivatives of the temperature are given in (2.90).

In order to verify now that $\eta(\mathbf{u})$ is a convex function of $\mathbf{u}$, we rewrite the entropy function from (2.91) and express it in terms of the phasic entropy function S_k,

$$\begin{aligned}\eta(\mathbf{u}) &= (1-\alpha)\cdot S_\ell(\rho_\ell, \rho_\ell v_\ell, \rho_\ell E_\ell) + \alpha \cdot S_g(\rho_g, \rho_g v_g, \rho_g E_g) \\ &= (1-\mathbf{u}_1)\cdot S_\ell\left(\frac{\mathbf{u}_2}{1-\mathbf{u}_1}, \frac{\mathbf{u}_3}{1-\mathbf{u}_1}, \frac{\mathbf{u}_4}{1-\mathbf{u}_1}\right) + \mathbf{u}_1 \cdot S_g\left(\frac{\mathbf{u}_5}{\mathbf{u}_1}, \frac{\mathbf{u}_6}{\mathbf{u}_1}, \frac{\mathbf{u}_7}{\mathbf{u}_1}\right). \end{aligned} \tag{2.101}$$

For convexity of $\eta(\mathbf{u})$ we need to check the positive semi-definiteness of the Hessian

$$\eta''(\mathbf{u}) = \begin{pmatrix} \eta_{\alpha,\alpha} & \eta^\mathsf{T}_{(1-\alpha)\mathbf{u}_\ell,\alpha} & \eta^\mathsf{T}_{\alpha\mathbf{u}_g,\alpha} \\ \eta_{(1-\alpha)\mathbf{u}_\ell,\alpha} & \frac{1}{1-\alpha} S''_\ell(\mathbf{u}_\ell) & 0 \\ \eta_{\alpha\mathbf{u}_g,\alpha} & 0 & \frac{1}{\alpha} S''_g(\mathbf{u}_g) \end{pmatrix} \tag{2.102}$$

with

$$\begin{aligned}\eta_{\alpha,\alpha} &= \frac{1}{1-\alpha}\mathbf{u}_\ell^\mathsf{T} S''_\ell(\mathbf{u}_\ell)\mathbf{u}_\ell + \frac{1}{\alpha}\mathbf{u}_g^\mathsf{T} S''_g(\mathbf{u}_g)\mathbf{u}_g, \\ &= \frac{\rho_g T_g (p_g)_\rho - \rho_g p_g (T_g)_\rho}{\alpha T_g^2} + \frac{\rho_\ell T_\ell (p_\ell)_\rho - \rho_\ell p_\ell (T_\ell)_\rho}{(1-\alpha)T_\ell^2} \end{aligned}$$

$$\eta_{(1-\alpha)\mathbf{u}_\ell,\alpha} = \frac{1}{1-\alpha} S''_\ell(\mathbf{u}_\ell)\mathbf{u}_\ell = \begin{pmatrix} \frac{T_\ell (p_\ell)_\rho - \left(p_\ell + \rho_\ell (E_\ell - v_\ell^2)\right)(T_\ell)_\rho}{(1-\alpha)T_\ell^2} \\ -\frac{\rho_\ell v_\ell (T_\ell)_\rho}{(1-\alpha)T_\ell^2} \\ \frac{\rho_\ell (T_\ell)_\rho}{(1-\alpha)T_\ell^2} \end{pmatrix},$$

$$\eta_{\alpha\mathbf{u}_g,\alpha} = -\frac{1}{\alpha} S''_g(\mathbf{u}_g)\mathbf{u}_g. = \begin{pmatrix} -\frac{T_g (p_g)_\rho - \left(p_g + \rho_g (E_g - v_g^2)\right)(T_g)_\rho}{\alpha T_g^2} \\ \frac{\rho_g v_g (T_g)_\rho}{\alpha T_g^2} \\ -\frac{\rho_g (T_g)_\rho}{\alpha T_g^2} \end{pmatrix}. \tag{2.103}$$

For a non-null vector $\mathbf{x}^\mathsf{T} = (a, b^\mathsf{T}, c^\mathsf{T}) \in \mathbb{R}^7$, $a \in \mathbb{R}$ and $b, c \in \mathbb{R}^3$, it holds

$$\begin{aligned}\mathbf{x}^\mathsf{T}\,\eta''(\mathbf{u})\,\mathbf{x} &= (a, b^\mathsf{T}, c^\mathsf{T})\,\eta''(\mathbf{u}) \begin{pmatrix} a \\ b \\ c \end{pmatrix} \\ &= \frac{1}{1-\alpha}(b + a\mathbf{u}_\ell)^\mathsf{T} S''_\ell(\mathbf{u}_\ell)\,(b + a\mathbf{u}_\ell) \;+\; \frac{1}{\alpha}(c - a\mathbf{u}_g)^\mathsf{T} S''_g(\mathbf{u}_g)\,(c - a\mathbf{u}_g) \\ &\overset{!}{\geq} 0. \end{aligned} \tag{2.104}$$

Due to the convexity of $S''_\ell(\mathbf{u}_\ell)$ and $S''_g(\mathbf{u}_g)$ the last inequality holds, such that finally the entropy function $\eta(\mathbf{u})$ is convex.

▶ Entropy dissipation

In order to ensure the entropy inequality (2.84) it remains to verify the entropy production inequality (2.87). This will give us conditions for the unknown source parameters $\widehat{v}_\mathrm{i}$, $\widehat{\rho}_\mathrm{i}$, $\widehat{E}_{\mathrm{i}\,k}$, and source saturated temperature $\widehat{T}_{\mathrm{i\,sat}}$.

The entropy production (including entropy production from the boundary) is given by

$$\begin{aligned}\partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}}\mathbf{s}_\mathrm{i}(\mathbf{u}) = &-\frac{Q_{\mathrm{i}\,\ell}}{T_\ell}+\frac{\widehat{E}_{\mathrm{i}\,\ell}-E_\ell+v_\ell^2-v_\ell\widehat{v}_\mathrm{i}+\frac{p_\ell}{\widehat{\rho}_\mathrm{i}}-\frac{p_\ell}{\rho_\ell}+s_\ell T_\ell}{T_\ell}\cdot\Gamma_\mathrm{i}\\ &-\frac{Q_{\mathrm{i}\,g}}{T_g}-\frac{\widehat{E}_{\mathrm{i}\,g}-E_g+v_g^2-v_g\widehat{v}_\mathrm{i}+\frac{p_g}{\widehat{\rho}_\mathrm{i}}-\frac{p_g}{\rho_g}+s_g T_g}{T_g}\cdot\Gamma_\mathrm{i}\overset{!}{\leq}\ 0. \end{aligned}\tag{2.105}$$

To fulfill the entropy inequality, we expect that the interfacial entropy production terms of both phases sum to less or equal zero. We have the freedom to choose our unknown source parameters $\widehat{v}_\mathrm{i}$, $\widehat{\rho}_\mathrm{i}$, $\widehat{E}_{\mathrm{i}\,k}$, and $\widehat{T}_{\mathrm{i\,sat}}$, such that the above condition holds. For this purpose we proceed in four steps:

(1) The specific total energy E_k and specific entropy s_k are expanded by their physical laws (2.64) and (2.88). Additionally the source specific total energy $\widehat{E}_{\mathrm{i}\,\ell}$ and $\widehat{E}_{\mathrm{i}\,g}$ have to be derived. By physical law, a specific total energy consists of a specific internal energy part and a specific kinetic energy part, such that we assume

$$\boxed{\widehat{E}_{\mathrm{i}\,k}\ :=\ \widehat{u}_{\mathrm{i}\,k}+\tfrac{1}{2}\widehat{v}_\mathrm{i}^2\ =\ \widehat{h}_{\mathrm{i}\,k}-\frac{\widehat{p}_\mathrm{i}}{\widehat{\rho}_\mathrm{i}}+\tfrac{1}{2}\widehat{v}_\mathrm{i}^2,}\tag{2.106}$$

where the source specific enthalpy is chosen on the saturation line [74]:

$$\boxed{\widehat{h}_{\mathrm{i}\,\ell}:=h_{\mathrm{satL}}(\widehat{p}_\mathrm{i})\quad\text{and}\quad\widehat{h}_{\mathrm{i}\,g}:=h_{\mathrm{satV}}(\widehat{p}_\mathrm{i}).}\tag{2.107}$$

As mentioned in Section 2.3, we have to ensure that both quantities $\widehat{E}_{\mathrm{i}\,\ell}$ and $\widehat{E}_{\mathrm{i}\,g}$ are distinct. This is necessary for the definition of the interphase mass flow rate Γ_i in (2.49), such that the conservation of total energy in the balance equations is satisfied. With the above chosen source specific internal energies, the denominator of the interphase mass flow rate Γ_i is defined by the latent heat of vaporization, $\widehat{h}_{\mathrm{i}\,g}-\widehat{h}_{\mathrm{i}\,\ell}$. With these assumptions, the entropy production simplifies to

$$\begin{aligned}\mathbf{s}_{\alpha\rho s} = &-\frac{Q_{\mathrm{i}\,\ell}}{T_\ell}+\frac{\frac{1}{2}(v_\ell-\widehat{v}_\mathrm{i})^2+\left(\widehat{h}_{\mathrm{i}\,\ell}+\frac{p_\ell-\widehat{p}_\mathrm{i}}{\widehat{\rho}_\mathrm{i}}-g_\ell\right)}{T_\ell}\cdot\Gamma_\mathrm{i}\\ &-\frac{Q_{\mathrm{i}\,g}}{T_g}-\frac{\frac{1}{2}(v_g-\widehat{v}_\mathrm{i})^2+\left(\widehat{h}_{\mathrm{i}\,g}+\frac{p_g-\widehat{p}_\mathrm{i}}{\widehat{\rho}_\mathrm{i}}-g_g\right)}{T_g}\cdot\Gamma_\mathrm{i}\overset{!}{\leq}\ 0.\end{aligned}\tag{2.108}$$

(2) We expect that the factors of the interphasic mass flow Γ_i describe interfacial specific entropy terms. Due to the physical meaning of specific entropy, it is clear that these

factors should not depend on a velocity, thus the velocity terms in (2.108) must vanish,

$$\frac{(v_\ell - \widehat{v}_\mathrm{i})^2}{T_\ell} - \frac{(v_g - \widehat{v}_\mathrm{i})^2}{T_g} \overset{!}{=} 0. \tag{2.109}$$

This gives us a condition on the unknown source velocity. The above quadratic equation delivers two choices for $\widehat{v}_\mathrm{i}$, where just one is a convex combination of v_ℓ and v_g:

$$\boxed{\widehat{v}_\mathrm{i} := \frac{\sqrt{T_g}\, v_\ell + \sqrt{T_\ell}\, v_g}{\sqrt{T_g} + \sqrt{T_\ell}}.} \tag{2.110}$$

With these assumptions, the entropy production simplifies to

$$\mathbf{s}_{\alpha\rho s} = -\frac{Q_{\mathrm{i}\,\ell}}{T_\ell} + \frac{\widehat{h}_{\mathrm{i}\,\ell} + \frac{p_\ell - \widehat{p}_\mathrm{i}}{\widehat{\rho}_\mathrm{i}} - g_\ell}{T_\ell} \cdot \Gamma_\mathrm{i} - \frac{Q_{\mathrm{i}\,g}}{T_g} - \frac{\widehat{h}_{\mathrm{i}\,g} + \frac{p_g - \widehat{p}_\mathrm{i}}{\widehat{\rho}_\mathrm{i}} - g_g}{T_g} \cdot \Gamma_\mathrm{i} \overset{!}{\leq} 0. \tag{2.111}$$

(3) As mentioned at the begin of this Subsection, the source parameters are modeled as an own phase. Thus, the source saturated temperature $\widehat{T}_{\mathrm{i\,sat}}$ is chosen on the saturation line,

$$\boxed{\widehat{T}_{\mathrm{i\,sat}} = T_\mathrm{sat}(\widehat{p}_\mathrm{i}).} \tag{2.112}$$

The source saturated temperature is used within the wall and the interface heat transfer rate density, see Sections 2.4.7 and 2.4.8.

(4) It remains to select the source pressure $\widehat{p}_\mathrm{i}$ and source density $\widehat{\rho}_\mathrm{i}$. Müller et al. [5] developed a closure for the source specific internal energy and source density, such that both parameter stay positive and that the entropy production inequality is satisfied. But this solution holds just for the chemical potential relaxation[12]. Saurel, Petitpas, and Abgrall [95] assume that phase transfer happens with an isentropic acoustic wave, such that they choose

$$\widehat{\rho}_\mathrm{i} = \frac{\frac{\rho_\ell c_\ell^2}{(1-\alpha)} + \frac{\rho_g c_g^2}{\alpha}}{\frac{c_\ell^2}{(1-\alpha)} + \frac{c_g^2}{\alpha}}. \tag{2.113}$$

Berry et al. [74] assume the source density on the liquid saturation line, where the source pressure is chosen in dependency of the acoustic impedance:

$$\widehat{\rho}_\mathrm{i} := \rho_\mathrm{satL}(\widehat{p}_\mathrm{i}) \quad \text{and} \quad \widehat{p}_\mathrm{i} := \frac{\rho_g c_g p_\ell + \rho_\ell c_\ell p_g}{\rho_\ell c_\ell + \rho_g c_g}.$$

A suitable choice is very difficult to find and might also depend on the relevant physical region. Therefore the choice of an appropriate closure for $\widehat{p}_\mathrm{i}$ and $\widehat{\rho}_\mathrm{i}$ still remains as open problem and is therefore left to the reader.

[12] See [5], Section 7.3.1, equations (7.24) to (7.25).

2.5.2. Hyperbolicity

Because the models only describe transport effects, they should be hyperbolic to ensure that all wave speeds are finite and the system may be locally decoupled [5].

In order to characterize this mathematical property, the equation system neglecting dissipative effects is transformed in quasi-conservative form in terms of primitive quantities,

$$\partial_t \tilde{\mathbf{u}} + \tilde{A}(\tilde{\mathbf{u}})\partial_x \tilde{\mathbf{u}} = 0, \tag{2.114}$$

with primitive system vector

$$\tilde{\mathbf{u}} := \begin{pmatrix} \alpha & \rho_\ell & v_\ell & p_\ell & \rho_g & v_g & p_g \end{pmatrix}^\mathsf{T}. \tag{2.115}$$

and primitve system matrix [94]

$$\tilde{A}(\tilde{\mathbf{u}}) = \begin{pmatrix} v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\rho_\ell(v_\ell - v_\mathrm{i})}{1-\alpha} & v_\ell & \rho_\ell & 0 & 0 & 0 & 0 \\ -\frac{p_\ell - p_\mathrm{i}}{(1-\alpha)\rho_\ell} & 0 & v_\ell & \frac{1}{\rho_\ell} & 0 & 0 & 0 \\ -\frac{c_\ell^2 \rho_\ell (v_\ell - v_\mathrm{i})}{(1-\alpha)} & 0 & c_\ell^2 \rho_\ell & v_\ell & 0 & 0 & 0 \\ \frac{\rho_g(v_g - v_\mathrm{i})}{\alpha} & 0 & 0 & 0 & v_g & \rho_g & 0 \\ \frac{p_g - p_\mathrm{i}}{\alpha \rho_g} & 0 & 0 & 0 & 0 & v_g & \frac{1}{\rho_g} \\ \frac{c_g^2 \rho_g (v_g - v_\mathrm{i})}{\alpha} & 0 & 0 & 0 & 0 & c_g^2 \rho_g & v_g \end{pmatrix}. \tag{2.116}$$

Then, the eigenvalue decomposition of the matrix $\tilde{A}(\tilde{\mathbf{u}})$ is given by

$$L\,\tilde{A}\,R = \Lambda,$$

where L and R are defined by the left and right eigenvectors and Λ is a diagonal matrix with eigenvalues on the diagonal:

$$\Lambda = \begin{pmatrix} v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & v_\ell - c_\ell & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & v_\ell & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & v_\ell + c_\ell & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & v_g - c_g & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & v_g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & v_g + c_g \end{pmatrix}, \tag{2.117}$$

$$R^{\mathsf{T}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{p_\ell - p_\mathrm{i} - \rho_\ell (v_\ell - v_\mathrm{i})^2}{(1-\alpha)\left(c_\ell^2 - (v_\ell - v_\mathrm{i})^2\right)} & \frac{1}{c_\ell^2} & 1 & \frac{1}{c_\ell^2} & 0 & 0 & 0 \\ -\frac{\left(p_\ell - p_\mathrm{i} - \rho_\ell c_\ell^2\right)(v_\ell - v_\mathrm{i})}{(1-\alpha)\rho_\ell\left(c_\ell^2 - (v_\ell - v_\mathrm{i})^2\right)} & -\frac{1}{\rho_\ell c_\ell} & 0 & \frac{1}{\rho_\ell c_\ell} & 0 & 0 & 0 \\ \frac{\left(p_\ell - p_\mathrm{i} - \rho_\ell (v_\ell - v_\mathrm{i})^2\right) c_\ell^2}{(1-\alpha)\left(c_\ell^2 - (v_\ell - v_\mathrm{i})^2\right)} & 1 & 0 & 1 & 0 & 0 & 0 \\ -\frac{p_g - p_\mathrm{i} - \rho_g (v_g - v_\mathrm{i})^2}{\alpha\left(c_g^2 - (v_g - v_\mathrm{i})^2\right)} & 0 & 0 & 0 & \frac{1}{c_g^2} & 0 & \frac{1}{c_g^2} \\ \frac{\left(p_g - p_\mathrm{i} - \rho_g c_g^2\right)(v_g - v_\mathrm{i})}{\alpha\rho_g\left(c_g^2 - (v_g - v_\mathrm{i})^2\right)} & 0 & 0 & 0 & -\frac{1}{\rho_g c_g} & 0 & \frac{1}{\rho_g c_g} \\ -\frac{\left(p_g - p_\mathrm{i} - \rho_g (v_g - v_\mathrm{i})^2\right) c_g^2}{\alpha\left(c_g^2 - (v_g - v_\mathrm{i})^2\right)} & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}, \tag{2.118}$$

and

$$L = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{(p_\ell - p_\mathrm{i} - \rho_\ell c_\ell (v_\ell - v_\mathrm{i}))c_\ell}{2\,(1-\alpha)(c_\ell - (v_\ell - v_\mathrm{i}))} & 0 & -\frac{\rho_\ell c_\ell}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & -\frac{1}{c_\ell^2} & 0 & 0 & 0 \\ -\frac{(p_\ell - p_\mathrm{i} + \rho_\ell c_\ell (v_\ell - v_\mathrm{i}))c_\ell}{2\,(1-\alpha)(c_\ell + (v_\ell - v_\mathrm{i}))} & 0 & \frac{\rho_\ell c_\ell}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{(p_g - p_\mathrm{i} - \rho_g c_g (v_g - v_\mathrm{i}))c_g}{2\alpha(c_g - (v_g - v_\mathrm{i}))} & 0 & 0 & 0 & 0 & -\frac{\rho_g c_g}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 & 0 & -\frac{1}{c_g^2} \\ \frac{(p_g - p_\mathrm{i} + \rho_g c_g (v_g - v_\mathrm{i}))c_g}{2\alpha(c_g + (v_g - v_\mathrm{i}))} & 0 & 0 & 0 & 0 & \frac{\rho_g c_g}{2} & \frac{1}{2} \end{pmatrix}. \tag{2.119}$$

The system admits seven real eigenvalues which are all real but not necessarily distinct: v_i, v_k and $v_k \pm c_k$. The corresponding eigenvectors are linearly independent, as soon as the non-resonance condition [94, 5] is fulfilled:

$$\boxed{v_\mathrm{i} \neq v_k \pm c_k.} \tag{2.120}$$

Otherwise, the first eigenvector will not be defined due to the denominator which would become zero. The interfacial velocity will be chosen below in (2.123) as convex combination of the phasic velocities, such that $v_\mathrm{i} \in [v_\ell, v_g]$. With this choice and the fact that our application deals with subsonic flow of liquid water and steam in the absorber tubes, $v_k \ll c_k$, the non-resonance condition is fulfilled.

Because the model provides real eigenvalues and the corresponding eigenvectors are linearly independent, the system matrix $\tilde{A}$ is diagonalizable and therefore the quasilinear system is hyperbolic. In the following, we use λ_j as j-th eigenvalue of Λ, and the corresponding eigenvector R_j as j-th column of matrix R^{T}.

2.5.3. Characteristic fields

The solution of the $n \times n$ system (2.65) is determined by n characteristic fields, where each consists of a characteristic speed λ_j and a corresponding j-wave. Depending on the type

of the particular characteristic field, the corresponding wave has different properties [7]. T.-P. Liu classified the waves of characteristic fields into two different waves: a contact discontinuitiy (*linearly degenerated field*), and a shock or rarefaction wave (*genuinely nonlinear field*) [96]. For the classification we need to compute the scalar product $\partial_{\tilde{\mathbf{u}}}\lambda_j \, R_j$ of the derivative of the eigenvalue λ_j and the corresponding eigenvector R_j. If this scalar product is zero, then the associated field is linearly degenerate, otherwise it is genuinely nonlinear.

As known for the Euler equations, just the field associated with the fluid velocity is linearly degenerate, where the other fields are genuinely nonlinear:

$$\begin{aligned}
\partial_{\tilde{\mathbf{u}}}\lambda_2 \, R_2 &= -\frac{1}{\rho_\ell c_\ell}, & \partial_{\tilde{\mathbf{u}}}\lambda_5 \, R_5 &= -\frac{1}{\rho_g c_g}, \\
\partial_{\tilde{\mathbf{u}}}\lambda_3 \, R_3 &= 0, & \partial_{\tilde{\mathbf{u}}}\lambda_6 \, R_6 &= 0, \\
\partial_{\tilde{\mathbf{u}}}\lambda_4 \, R_4 &= \frac{1}{\rho_\ell c_\ell}, & \partial_{\tilde{\mathbf{u}}}\lambda_7 \, R_7 &= \frac{1}{\rho_g c_g}.
\end{aligned} \tag{2.121}$$

The field associated with the first eigenvalues v_i depends on the choice of the interfacial velocity v_i. This wave corresponds to the non-conservative term $v_\mathrm{i}\partial_x\alpha$ in the volume fraction equation, see first equation in (2.57). Because the volume fraction α should be preserved through the v_i-contact discontinuity, v_i should be chosen in such a way that this discontinuity is associated with a linearly degenerate field [97].

From a physical point of view it makes sense to assume the interfacial velocity v_i as convex combination of v_ℓ and v_g [60, 8], such that we define:

$$\boxed{v_\mathrm{i} := \beta v_\ell + (1-\beta) v_g \quad \text{with} \quad \beta \in [0,1].} \tag{2.122}$$

Following Saleh [97], we set β as mass fraction,

$$\boxed{\beta := \frac{\xi \alpha_\ell \rho_\ell}{\xi \alpha_\ell \rho_\ell + (1-\xi)\alpha_g \rho_g} \quad \text{with} \quad \xi \in [0,1].} \tag{2.123}$$

The derivative of the interfacial velocity with respect to $\tilde{\mathbf{u}}$ is then given by

$$\partial_{\tilde{\mathbf{u}}}\lambda_1 = \begin{pmatrix}
-\frac{\xi(1-\xi)\rho_\ell\rho_g(v_g-v_\ell)}{(\xi(1-\alpha)\rho_\ell+(1-\xi)\alpha\rho_g)^2} \\
-\frac{\xi(1-\xi)(1-\alpha)\alpha\rho_g(v_g-v_\ell)}{(\xi(1-\alpha)\rho_\ell+(1-\xi)\alpha\rho_g)^2} \\
\frac{\xi(1-\alpha)\rho_\ell}{\xi(1-\alpha)\rho_\ell+(1-\xi)\alpha\rho_g} \\
0 \\
\frac{\xi(1-\xi)(1-\alpha)\alpha\rho_\ell(v_g-v_\ell)}{(\xi(1-\alpha)\rho_\ell+(1-\xi)\alpha\rho_g)^2} \\
\frac{(1-\xi)\alpha\rho_g}{\xi(1-\alpha)\rho_\ell+(1-\xi)\alpha\rho_g} \\
0
\end{pmatrix}, \tag{2.124}$$

which is orthogonal to its corresponding eigenvector, $\partial_{\tilde{\mathbf{u}}}\lambda_1 R_1 = 0$. This means that with the above choice of β the field of the v_i-wave is linearly degenerate.

In literature, mostly the three interfacial velocities,

$$v_\mathrm{i} = v_g, \quad v_\mathrm{i} = v_\ell, \quad \text{and} \quad v_\mathrm{i} = \frac{\alpha_\ell \rho_\ell v_\ell + \alpha_g \rho_g v_g}{\alpha_\ell \rho_\ell + \alpha_g \rho_g}, \tag{2.125}$$

are used, see [98]. They correspond to the above defined interfacial velocity with $\xi = 0$, $\xi = 1$, and $\xi = \frac{1}{2}$. For our application of one fluid (with two aggregate states) we do not want to give one phase an advantage, such that we choose v_i symmetric with $\xi = \frac{1}{2}$.

2.5.4. Symmetrization of the hyperbolic system

A hyperbolic system should provide the property that locally in time there exists a smooth solution of the corresponding Cauchy problem. Godlewski and Raviart [99] showed that for conservative problem this property comes with the existence of an entropy-entropy flux pair. But for non-conservative problems we additionally need to show that the system is symmetrizable. For two-phase flow this was first proven in [94] and was extended for multi-components fluids in [5].

This means, we need to find a symmetric positive definite matrix $P(\tilde{\mathbf{u}})$ such that also the matrix $P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})$ is symmetric,

$$P(\tilde{\mathbf{u}})\partial_t \tilde{\mathbf{u}} + \underbrace{P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})}_{\text{symmetric}} \partial_x \tilde{\mathbf{u}} = P(\tilde{\mathbf{u}})\tilde{\mathbf{s}}(\tilde{\mathbf{u}}) \tag{2.126}$$

with primitive variables $\tilde{u}$ given in (2.115). Symmetry for the matrices $P(\tilde{\mathbf{u}})$ and $P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})$ is given by choosing

$$P(\tilde{\mathbf{u}}) := \begin{pmatrix} P_\alpha & P_{\ell\alpha}^\mathsf{T} & P_{g\alpha}^\mathsf{T} \\ P_{\ell\alpha} & P_\ell & 0 \\ P_{g\alpha} & 0 & P_g \end{pmatrix} \tag{2.127}$$

with vector

$$P_{k\alpha} := L_k^\mathsf{T}(\Lambda_k - v_\mathrm{i}\mathcal{I}_3)^{-1} R_k^\mathsf{T} P_k \tilde{A}_{k\alpha} \tag{2.128}$$

and symmetric positiv definite matrices [5]

$$P_k := R_k^{-1} L_k. \tag{2.129}$$

Hereby, Λ_k, R_k and L_k describe the 3×3 sub-matrices from L, R and Λ, with rows/columns 2–4 for liquid, and 5–7 for steam phase. $\tilde{A}_{k\alpha}$ describes the 3×1 sub-matrix from the the first column of $\tilde{A}$, rows 2–4 for liquid, and 5–7 for steam phase. So, finally we get

$$P_\ell = \begin{pmatrix} 1 & 0 & -\frac{1}{c_\ell^2} \\ 0 & \frac{\rho_\ell^2 c_\ell^2}{2} & 0 \\ -\frac{1}{c_\ell^2} & 0 & \frac{1}{2} + \frac{1}{c_\ell^4} \end{pmatrix}, \quad P_g = \begin{pmatrix} 1 & 0 & -\frac{1}{c_g^2} \\ 0 & \frac{\rho_g^2 c_g^2}{2} & 0 \\ -\frac{1}{c_g^2} & 0 & \frac{1}{2} + \frac{1}{c_g^4} \end{pmatrix}, \tag{2.130}$$

and

$$P_{\ell\,\alpha} = \begin{pmatrix} 0 \\ \frac{\rho_\ell c_\ell^2 (p_\ell - p_\mathrm{i} - \rho_\ell c_\ell^2)(v_\ell - v_\mathrm{i})}{2(1-\alpha)(c_\ell^2 - (v_\ell - v_\mathrm{i})^2)} \\ -\frac{c_\ell^2 (p_\ell - p_\mathrm{i} - \rho_\ell (v_\ell - v_\mathrm{i})^2)}{2(1-\alpha)(c_\ell^2 - (v_\ell - v_\mathrm{i})^2)} \end{pmatrix}, \quad P_{g\,\alpha} = \begin{pmatrix} 0 \\ -\frac{\rho_g c_g^2 (p_g - p_\mathrm{i} - \rho_g c_g^2)(v_g - v_\mathrm{i})}{2\alpha(c_g^2 - (v_g - v_\mathrm{i})^2)} \\ \frac{c_g^2 (p_g - p_\mathrm{i} - \rho_g (v_g - v_\mathrm{i})^2)}{2\alpha(c_g^2 - (v_g - v_\mathrm{i})^2)} \end{pmatrix}. \tag{2.131}$$

This choice for $P(\tilde{\mathbf{u}})$ is realizable if the non-resonance condition (2.120) holds. It remains to choose P_α in such a way that $P(\tilde{\mathbf{u}})$ is positive definite. With a non-null vector $\mathbf{x}^\mathsf{T} = (a, b^\mathsf{T}, c^\mathsf{T}) \in \mathbb{R}^7$ with $a \in \mathbb{R}$ and $b, c \in \mathbb{R}^3$, it holds

$$\begin{aligned} \mathbf{x}^\mathsf{T} P(\tilde{\mathbf{u}})\, \mathbf{x} &= (a, b^\mathsf{T}, c^\mathsf{T})\, \eta''(\mathbf{u}) \begin{pmatrix} a \\ b \\ c \end{pmatrix} \\ &= a^2 P_\alpha + 2a \left(P_{\ell\,\alpha}^\mathsf{T} b + P_{g\,\alpha}^\mathsf{T} c \right) + b^\mathsf{T} P_\ell b + c^\mathsf{T} P_g c \\ &\overset{!}{\geq} 0. \end{aligned} \tag{2.132}$$

This polynomial is of degree 2 in a. We adapt the proof of Coquel et al. [94] who considered a different set of variables than $\tilde{u}$, and determine the discriminant of this polynomial

$$\begin{aligned} D &= 4 \left| P_{\ell\,\alpha}^\mathsf{T} b + P_{g\,\alpha}^\mathsf{T} c \right|^2 - 4 P_\alpha \left(b^\mathsf{T} P_\ell b + c^\mathsf{T} P_g c \right) \\ &= 4 \left| (P_\ell^{-1/2} P_{\ell\,\alpha})^\mathsf{T} P_\ell^{1/2} b + (P_g^{-1/2} P_{g\,\alpha})^\mathsf{T} P_g^{1/2} c \right|^2 - 4 P_\alpha \left(\left| P_\ell^{1/2} b \right|^2 + \left| P_g^{1/2} c \right|^2 \right) \\ &= 4 \left(\left| P_\ell^{-1/2} P_{\ell\,\alpha} \right|^2 + \left| P_g^{-1/2} P_{g\,\alpha} \right|^2 - P_\alpha \right) \left(\left| P_\ell^{1/2} b \right|^2 + \left| P_g^{1/2} c \right|^2 \right) \\ &\quad - 4 \left| (P_\ell^{-1/2} P_{\ell\,\alpha})^\mathsf{T} P_g^{1/2} c - (P_g^{-1/2} P_{g\,\alpha})^\mathsf{T} P_\ell^{1/2} b \right|^2 \\ &\overset{!}{>} 0 \end{aligned} \tag{2.133}$$

Because P_ℓ and P_g are symmetric positive definite, there exist symmetric positive definite matrices $P_\ell^{1/2}$ and $P_g^{1/2}$ and their inverses $P_\ell^{-1/2}$ and $P_g^{-1/2}$, such that $P_\ell^{1/2} P_\ell^{1/2} = P_\ell$ and $P_g^{1/2} P_g^{1/2} = P_g$. The discriminant D is positive, if we choose

$$P_\alpha > \left| P_\ell^{-1/2} P_{\ell\,\alpha} \right|^2 + \left| P_g^{-1/2} P_{g\,\alpha} \right|^2. \tag{2.134}$$

2.5.5. Summary of the chosen closures

Within the above subsections the interfacial and source parameters are developed in such a way that the second law of thermodynamics holds for the quasilinear system.

- Interfacial velocity (2.122) and (2.123)

$$v_\mathrm{i} := \frac{\xi \alpha_\ell \rho_\ell v_\ell + (1-\xi) \alpha_g \rho_g v_g}{\xi \alpha_\ell \rho_\ell + (1-\xi) \alpha_g \rho_g}$$

  with an arbitrary $\xi \in [0, 1]$. For our application we choose $\xi = \frac{1}{2}$.

- Interfacial pressure (2.94) as convex combination of p_ℓ and p_g

$$p_{\mathrm{i}} := \frac{p_g T_\ell (v_g - v_{\mathrm{i}}) + p_\ell T_g (v_{\mathrm{i}} - v_\ell)}{T_\ell (v_g - v_{\mathrm{i}}) + T_g (v_{\mathrm{i}} - v_\ell)} = \frac{\xi \alpha_\ell \rho_\ell T_\ell \, p_g + (1-\xi) \alpha_g \rho_g T_g \, p_\ell}{\xi \alpha_\ell \rho_\ell T_\ell + (1-\xi) \alpha_g \rho_g T_g}.$$

  with the same ξ as above.

- Source specific total energy (2.106)

$$\widehat{E}_{\mathrm{i}\,k} := \widehat{u}_{\mathrm{i}\,k} + \tfrac{1}{2}\widehat{v}_{\mathrm{i}}^2 = \widehat{h}_{\mathrm{i}\,k} - \frac{\widehat{p}_{\mathrm{i}}}{\widehat{\rho}_{\mathrm{i}}} + \tfrac{1}{2}\widehat{v}_{\mathrm{i}}^2$$

- Source specific enthalpy (2.107)

$$\widehat{h}_{\mathrm{i}\,\ell} := h_{\mathrm{satL}}(\widehat{p}_{\mathrm{i}}) \quad \text{and} \quad \widehat{h}_{\mathrm{i}\,g} := h_{\mathrm{satV}}(\widehat{p}_{\mathrm{i}}).$$

- Source velocity (2.110)

$$\widehat{v}_{\mathrm{i}} := \frac{\sqrt{T_g}\, v_\ell + \sqrt{T_\ell}\, v_g}{\sqrt{T_g} + \sqrt{T_\ell}}.$$

- Source saturated temperature (2.112)

$$\widehat{T}_{\mathrm{i\,sat}} = T_{\mathrm{sat}}(\widehat{p}_{\mathrm{i}}).$$

- The choice of an appropriate closure for the source density $\widehat{\rho}_{\mathrm{i}}$ and source pressure $\widehat{p}_{\mathrm{i}}$ still remains as open problem.

2.6. Homogeneous equilibrium two-phase flow model

In the literature, related to the underlying solar thermal problem, it is often found that the two-phase flow is described by a simplified *homogeneous equilibrium* model [62], [63], [64], [65]. For the sake of completeness, a homogeneous model is derived from the basic equations (2.57), and its properties are shown. For better readability, the same notation as introduced in 2.3 is used, despite that the system dimension n is now 3 instead of 7. At first, the following mixture quantities are defined [100],

$$\begin{aligned}
\rho_{\mathrm{h}} &:= \alpha_\ell \rho_\ell + \alpha_g \rho_g, \\
u_{\mathrm{h}} &:= \frac{1}{\rho_{\mathrm{h}}} \left(\alpha_\ell \rho_\ell u_\ell + \alpha_g \rho_g u_g \right), \\
E_{\mathrm{h}} &:= \frac{1}{\rho_{\mathrm{h}}} \left(\alpha_\ell \rho_\ell E_\ell + \alpha_g \rho_g E_g \right), \\
s_{\mathrm{h}} &:= \frac{1}{\rho_{\mathrm{h}}} \left(\alpha_\ell \rho_\ell s_\ell + \alpha_g \rho_g s_g \right),
\end{aligned} \tag{2.135}$$

for density, specific internal energy, specific total energy and specific entropy. The subscript $_{\mathrm{h}}$ is used for homogeneous properties. Usually the model is assumed to be at mechanical and thermodynamical equilibrium. With this assumption no contributions corresponding to the slip of both phases occur during the homogenization process. Thus, the velocities, pressures and temperatures for each phase are equal [100]:

$$v_{\mathrm{h}} := v_\ell = v_g, \quad p_{\mathrm{h}} := p_\ell = p_g, \quad T_{\mathrm{h}} := T_\ell = T_g. \tag{2.136}$$

Because the temperatures for both phase are equal the phasic densities must be on the saturation line, such that we define

$$\rho_\ell := \rho_{\mathrm{satL}}(p_{\mathrm{h}}) \quad \text{and} \quad \rho_g := \rho_{\mathrm{satV}}(p_{\mathrm{h}}). \tag{2.137}$$

Then we also get an explicit expression for the volume fraction,

$$\alpha = \frac{\rho_{\mathrm{h}} - \rho_\ell}{\rho_g - \rho_\ell}. \tag{2.138}$$

Now, the *homogeneous equilibrium model* is obtained by summing up the corresponding equations of void fraction, mass, momentum and energy equation for each phase in (2.57):

$$\begin{aligned}
&\partial_t \rho_{\mathrm{h}} && + \partial_x(\rho_{\mathrm{h}} v_{\mathrm{h}}) && = 0 \\
&\partial_t(\rho_{\mathrm{h}} v_{\mathrm{h}}) && + \partial_x(\rho_{\mathrm{h}} v_{\mathrm{h}}^2) && = -\partial_x p_{\mathrm{h}} + \rho_{\mathrm{h}}\, g\, \sin(\vartheta) - F_{\mathrm{w}\,\ell} - F_{\mathrm{w}\,g} \\
&\partial_t(\rho_{\mathrm{h}} E_{\mathrm{h}}) && + \partial_x(\rho_{\mathrm{h}} v_{\mathrm{h}} E_{\mathrm{h}}) && = -\partial_x(v_{\mathrm{h}} p_{\mathrm{h}}) + \rho_{\mathrm{h}} v_{\mathrm{h}}\, g\, \sin(\vartheta) - (F_{\mathrm{w}\,\ell} + F_{\mathrm{w}\,g}) v_{\mathrm{h}} + Q_{\mathrm{w}\,\ell} + Q_{\mathrm{w}\,g}.
\end{aligned} \tag{2.139}$$

We note that the homogenized void fraction disappears due to (2.34). Due to the choice of the interfacial mass flow rates Γ_{i} and Γ_{w} in (2.49) and (2.53) also the interfacial

exchange terms in the momentum and energy equation disappear. The *homogeneous equilibrium model* is given by three balance laws in the form,

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = \mathbf{s}(\mathbf{u}), \tag{2.140}$$

the independent variables $\mathbf{u}$ and the flux vector $\mathbf{f}(\mathbf{u})$ are given by

$$\mathbf{u} := \begin{pmatrix} \rho_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h} \\ \rho_\mathrm{h} E_\mathrm{h} \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} \rho_\mathrm{h} v_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h}^2 + p_\mathrm{h} \\ (\rho_\mathrm{h} E_\mathrm{h} + p_\mathrm{h}) v_\mathrm{h} \end{pmatrix}, \tag{2.141}$$

where the source vector $\mathbf{s}(\mathbf{u}) = \mathbf{s}_\mathrm{i}(\mathbf{u}) + \mathbf{s}_\mathrm{w}(\mathbf{u})$ is given by

$$\mathbf{s}_\mathrm{i}(\mathbf{u}) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{s}_\mathrm{w}(\mathbf{u}) = \begin{pmatrix} 0 \\ \rho_\mathrm{h}\, g\, \sin(\vartheta) - F_{\mathrm{w}\,\ell} - F_{\mathrm{w}\,g} \\ \rho_\mathrm{h} v_\mathrm{h}\, g\, \sin(\vartheta) - (F_{\mathrm{w}\,\ell} + F_{\mathrm{w}\,g}) v_\mathrm{h} + Q_{\mathrm{w}\,\ell} + Q_{\mathrm{w}\,g} \end{pmatrix}. \tag{2.142}$$

All solutions for $\mathbf{u}$ are in the set of admissible states,

$$\Omega = \{\mathbf{u} \in \mathbb{R}^n \mid \rho_\mathrm{h} > 0, u_\mathrm{h} > 0, c_\mathrm{h} > 0\} \tag{2.143}$$

with $n = 3$, where u_h is the specific internal energy, given by the physical law

$$u_\mathrm{h} = E_\mathrm{h} - \tfrac{1}{2} v_\mathrm{h}^2. \tag{2.144}$$

To transform the system in closed quasilinear form,

$$\partial_t \mathbf{u} + A(\mathbf{u})\, \partial_x \mathbf{u} = \mathbf{s}(\mathbf{u}), \tag{2.145}$$

with system matrix

$$A(\mathbf{u}) = \partial_\mathbf{u} \mathbf{f}(\mathbf{u}), \tag{2.146}$$

we need to differentiate the flux vector $\mathbf{f}(\mathbf{u})$ with respect to our independent variables $\mathbf{u}$. Therefore we need the partial derivatives of the primitives,

$$\begin{aligned} \partial_x v_\mathrm{h} &= -\frac{v_\mathrm{h}}{\rho_\mathrm{h}}\, \partial_x \rho_\mathrm{h} + \frac{1}{\rho_\mathrm{h}}\, \partial_x (\rho_\mathrm{h} v_\mathrm{h}) \\ \partial_x u_\mathrm{h} &= -\frac{E_\mathrm{h} - v_\mathrm{h}^2}{\rho_\mathrm{h}}\, \partial_x \rho_\mathrm{h} - \frac{v_\mathrm{h}}{\rho_\mathrm{h}}\, \partial_x (\rho_\mathrm{h} v_\mathrm{h}) + \frac{1}{\rho_\mathrm{h}}\, \partial_x (\rho_\mathrm{h} E_\mathrm{h}) \end{aligned} \tag{2.147}$$

and the partial derivatives of the pressure, which can be derived from the pressure relation (2.55), such that

$$\partial_x p_\mathrm{h} \overset{(2.55)}{=} (p_\mathrm{h})_\rho\, \partial_x \rho_\mathrm{h} + (p_\mathrm{h})_u\, \partial_x u_\mathrm{h}$$

$$\overset{(2.147)}{=} \left((p_\mathrm{h})_\rho - (p_\mathrm{h})_u \cdot \frac{E_\mathrm{h} - v_\mathrm{h}^2}{\rho_\mathrm{h}} \right) \partial_x \rho_\mathrm{h} - \frac{(p_\mathrm{h})_u \cdot v_\mathrm{h}}{\rho_\mathrm{h}} \partial_x (\rho_\mathrm{h} v_\mathrm{h}) + \frac{(p_\mathrm{h})_u}{\rho_\mathrm{h}} \partial_x (\rho_\mathrm{h} E_\mathrm{h}), \tag{2.148}$$

$$\begin{aligned} \partial_x (v_\mathrm{h} p_\mathrm{h}) &= v_\mathrm{h} \partial_x p_\mathrm{h} + p_\mathrm{h} \partial_x v_\mathrm{h} \\ &\overset{(2.148)}{=} v_\mathrm{h} \left((p_\mathrm{h})_\rho - (p_\mathrm{h})_u \cdot \frac{E_\mathrm{h} - v_\mathrm{h}^2}{\rho_\mathrm{h}} - \frac{p_\mathrm{h}}{\rho_\mathrm{h}} \right) \partial_x \rho_\mathrm{h} \\ &\quad + \frac{p_\mathrm{h} - (p_\mathrm{h})_u \cdot v_\mathrm{h}^2}{\rho_\mathrm{h}} \partial_x (\rho_\mathrm{h} v_\mathrm{h}) + \frac{v_\mathrm{h} \cdot (p_\mathrm{h})_u}{\rho_\mathrm{h}} \partial_x (\rho_\mathrm{h} E_\mathrm{h}). \end{aligned} \tag{2.149}$$

Then we can compute the derivative of $\mathbf{f}(\mathbf{u})$ with respect to $\mathbf{u}$, such that the system matrix $A(\mathbf{u})$ is given by (2.146):

$$\begin{aligned} A(\mathbf{u}) &= \partial_\mathbf{u} \mathbf{f}(\mathbf{u}) + B(\mathbf{u}) \\ &= \begin{pmatrix} 0 & 1 & 0 \\ -v_\mathrm{h}^2 + (p_\mathrm{h})_\rho - (p_\mathrm{h})_u \cdot \frac{E_\mathrm{h} - v_\mathrm{h}^2}{\rho_\mathrm{h}} & 2v_\mathrm{h} + \frac{-(p_\mathrm{h})_u \cdot v_\mathrm{h}}{\rho_\mathrm{h}} & \frac{(p_\mathrm{h})_u}{\rho_\mathrm{h}} \\ v_\mathrm{h} \left((p_\mathrm{h})_\rho - (p_\mathrm{h})_u \cdot \frac{E_\mathrm{h} - v_\mathrm{h}^2}{\rho_\mathrm{h}} - \frac{p_\mathrm{h}}{\rho_\mathrm{h}} - E_\mathrm{h} \right) & E_\mathrm{h} + \frac{p_\mathrm{h} - (p_\mathrm{h})_u \cdot v_\mathrm{h}^2}{\rho_\mathrm{h}} & v_\mathrm{h} + \frac{v_\mathrm{h} \cdot (p_\mathrm{h})_u}{\rho_\mathrm{h}} \end{pmatrix}. \end{aligned} \tag{2.150}$$

For the *two-velocity two-pressure model* closures for some unknown parameters have been found by requesting thermodynamical and mathematical properties, see Section 2.5. Now in contrast, the *homogeneous equilibrium model* has no unknown parameters. Anyway, in the following it is shown that the model fulfilles the same thermodynamical and mathematical properties like the *two-velocity two-pressure model*, see Section 2.5.

2.6.1. Second law of thermodynamics

A candidate for an entropy-entropy flux pair for the Euler equations is the physical entropy of the fluid,

$$\eta(\mathbf{u}) = -\rho_\mathrm{h} s_\mathrm{h} \quad \text{and} \quad \psi(\mathbf{u}) = -\rho_\mathrm{h} v_\mathrm{h} s_\mathrm{h}. \tag{2.151}$$

The entropy variables are given by

$$\mathbf{v}(\mathbf{u}) := \eta'(\mathbf{u}) = \begin{pmatrix} \frac{\frac{p_\mathrm{h}}{\rho_\mathrm{h}} + u_\mathrm{h} - \frac{1}{2} v_\mathrm{h}^2}{T_\mathrm{h}} - s_\mathrm{h} \\ \frac{v_\mathrm{h}}{T_\mathrm{h}} \\ -\frac{1}{T_\mathrm{h}} \end{pmatrix} \overset{(2.88)}{=} \begin{pmatrix} \frac{g_\mathrm{h} - \frac{1}{2} v_\mathrm{h}^2}{T_\mathrm{h}} \\ \frac{v_\mathrm{h}}{T_\mathrm{h}} \\ -\frac{1}{T_\mathrm{h}} \end{pmatrix}, \tag{2.152}$$

such that the entropy compatibility condition (2.85) is fulfilled:

$$\partial_{\mathbf{u}}\psi(\mathbf{u})^{\mathsf{T}} = \begin{pmatrix} v_{\mathrm{h}} \cdot \dfrac{\frac{p_{\mathrm{h}}}{\rho_{\mathrm{h}}} + u_{\mathrm{h}} - \frac{1}{2}v_{\mathrm{h}}^2}{T_{\mathrm{h}}} \\ \dfrac{v_{\mathrm{h}}^2}{T_{\mathrm{h}}} - s_{\mathrm{h}} \\ -\dfrac{v_{\mathrm{h}}}{T_{\mathrm{h}}} \end{pmatrix} = \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}} \cdot A(\mathbf{u}). \tag{2.153}$$

The Hessian of the entropy function is given by

$$\eta''(\mathbf{u}) = \begin{pmatrix} \frac{T_{\mathrm{h}}(p_{\mathrm{h}})_\rho - \left(p_{\mathrm{h}} + 2\rho_{\mathrm{h}}(E_{\mathrm{h}} - v_{\mathrm{h}}^2)\right)(T_{\mathrm{h}})_\rho + \left(E_{\mathrm{h}} - v_{\mathrm{h}}^2\right)^2 (T_{\mathrm{h}})_u + v_{\mathrm{h}}^2 T_{\mathrm{h}}}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2} & -v_{\mathrm{h}}\left(\eta''_{13} + \frac{1}{\rho_{\mathrm{h}} T_{\mathrm{h}}}\right) & \eta''_{13} \\ -v_{\mathrm{h}}\left(\eta''_{13} + \frac{1}{\rho_{\mathrm{h}} T_{\mathrm{h}}}\right) & \frac{v_{\mathrm{h}}^2 (T_{\mathrm{h}})_u + T_{\mathrm{h}}}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2} & \frac{-v_{\mathrm{h}}(T_{\mathrm{h}})_u}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2} \\ \eta''_{13} & \frac{-v_{\mathrm{h}}(T_{\mathrm{h}})_u}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2} & \frac{(T_{\mathrm{h}})_u}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2} \end{pmatrix}, \tag{2.154}$$

with $\eta''_{13} := \frac{\rho_{\mathrm{h}}(T_{\mathrm{h}})_\rho - (E_{\mathrm{h}} - v_{\mathrm{h}}^2)(T_{\mathrm{h}})_u}{\rho_{\mathrm{h}} T_{\mathrm{h}}^2}$. The needed derivatives of the temperature are given in (2.90). In order to verify now that $\eta(\mathbf{u})$ is a convex function of $\mathbf{u}$, we follow again the proof of Godlewski and Raviart [99]. Above in (2.95), it was shown that the phasic entropy functions $S_k(\mathbf{u}_k) := -\rho_k s_k$ with $\mathbf{u}_k^{\mathsf{T}} = (\rho_k, \rho_k v_k, \rho_k E_k)$ are strictly convex functions. This proof can be adopted one-to-one for this entropy function.

In order to ensure the entropy inequality (2.84) it remains to verify the entropy dissipation (2.87). The entropy balance law is derived by (2.86),

$$\partial_t \eta(\mathbf{u}) + \partial_x \psi(\mathbf{u}) = \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}} \mathbf{s}_{\mathrm{i}}(\mathbf{u}) = 0. \tag{2.155}$$

Thus, the entropy production is zero.

2.6.2. Hyperbolicity

In order to characterize mathematical properties, the equation systems are transformed in quasi-conservative form (2.114)

$$\partial_t \tilde{\mathbf{u}} + \tilde{A}(\tilde{\mathbf{u}})\partial_x \tilde{\mathbf{u}} = \tilde{\mathbf{s}}(\tilde{\mathbf{u}}),$$

with primitive system vector

$$\tilde{\mathbf{u}} := \begin{pmatrix} \rho_{\mathrm{h}} & v_{\mathrm{h}} & p_{\mathrm{h}} \end{pmatrix}^{\mathsf{T}}. \tag{2.156}$$

and primitive system matrix

$$\tilde{A}(\tilde{\mathbf{u}}) = \begin{pmatrix} v_{\mathrm{h}} & \rho_{\mathrm{h}} & 0 \\ 0 & v_{\mathrm{h}} & \frac{1}{\rho_{\mathrm{h}}} \\ 0 & c_{\mathrm{h}}^2 \rho_{\mathrm{h}} & v_{\mathrm{h}} \end{pmatrix}. \tag{2.157}$$

The eigenvalue matrix Λ and the right and left eigenvectors R and L are given by

$$\Lambda = \begin{pmatrix} v_\mathrm{h} - c_\mathrm{h} & 0 & 0 \\ 0 & v_\mathrm{h} & 0 \\ 0 & 0 & v_\mathrm{h} + c_\mathrm{h} \end{pmatrix}, \tag{2.158}$$

$$R^\mathsf{T} = \begin{pmatrix} \frac{1}{c_\mathrm{h}^2} & 1 & \frac{1}{c_\mathrm{h}^2} \\ -\frac{1}{\rho_\mathrm{h} c_\mathrm{h}} & 0 & \frac{1}{\rho_\mathrm{h} c_\mathrm{h}} \\ 1 & 0 & 1 \end{pmatrix}, \tag{2.159}$$

and

$$L = \begin{pmatrix} 0 & -\frac{\rho_\mathrm{h} c_\mathrm{h}}{2} & \frac{1}{2} \\ 1 & 0 & -\frac{1}{c_\mathrm{h}^2} \\ 0 & \frac{\rho_\mathrm{h} c_\mathrm{h}}{2} & \frac{1}{2} \end{pmatrix}. \tag{2.160}$$

The system admits three real eigenvalues, v_h and $v_\mathrm{h} \pm c_\mathrm{h}$. Because the corresponding eigenvectors are linearly independent the quasilinear system is hyperbolic.

2.6.3. Characteristic fields

As known for the Euler equations, the field associated with the fluid velocity v_h is linearly degenerate, where the other fields associated with eigenvalues $v_\mathrm{h} \pm c_\mathrm{h}$ are genuinely nonlinear:

$$\partial_{\tilde{\mathbf{u}}} \lambda_1 \, R_1 = -\frac{1}{\rho_\mathrm{h} c_\mathrm{h}}, \qquad \partial_{\tilde{\mathbf{u}}} \lambda_2 \, R_2 = 0, \qquad \partial_{\tilde{\mathbf{u}}} \lambda_3 \, R_3 = \frac{1}{\rho_\mathrm{h} c_\mathrm{h}}. \tag{2.161}$$

2.6.4. Symmetrization of the hyperbolic system

As consequence of the conservative form and the existence of an entropy-entropy flux pair the system is symmetrizable, see Godlewski and Raviart [99]. A symmetric positiv definite matrix $P(\tilde{\mathbf{u}})$ is given by choosing [5]

$$P(\tilde{\mathbf{u}}) := R^{-1} L = \begin{pmatrix} 1 & 0 & -\frac{1}{c_\mathrm{h}^2} \\ 0 & \frac{\rho_\mathrm{h}^2 c_\mathrm{h}^2}{2} & 0 \\ -\frac{1}{c_\mathrm{h}^2} & 0 & \frac{1}{2} + \frac{1}{c_\mathrm{h}^4} \end{pmatrix}. \tag{2.162}$$

With this choice also $P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})$ is symmetric.

3. Finite volume methods

In one space dimension, our first-order system of partial differential equations in space x and time t has the form

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = \mathbf{s}(\mathbf{u}), \tag{3.1}$$

where $\mathbf{u} \in \mathbb{R}^n$ is the state vector of conservative variables, $\mathbf{f}(\mathbf{u}) \in \mathbb{R}^n$ is the corresponding flux vector, $B(\mathbf{u}) \in \mathbb{R}^{n\times n}$ is the non-conservative part of the coefficient matrix, and $\mathbf{s}(\mathbf{u}) \in \mathbb{R}^n$ is the source term vector. The system is called to be in *conservative form*, if $B(\mathbf{u}) \equiv 0$, otherwise it is in *non-conservative form*. If $\mathbf{s}(\mathbf{u}) \equiv 0$, the system is called *homogeneous*, otherwise *inhomogeneous*.

In one-dimensional space, the finite volume method partitions the domain in a finite number of grid cells. The conservation of the fluid must hold in every volume. In every time step, the grid cells are updated by solving the conservation laws of the fluid by integration.

The spatial and temporal space for $x \in [0, x_{\max}]$ and $t \in [0, t^{\max}]$ is covered uniformly by a grid of I times N cells. For simplicity we assume an equidistant grid in space and time, with a grid size of $\Delta x = x_{i+1} - x_i$ and time step of $\Delta t = t^{n+1} - t^n$. The i-th grid cell is given by

$$C_i := \left(x_{i-1/2},\, x_{i+1/2}\right), \tag{3.2}$$

with its border points $x_{i-1/2}$ and $x_{i+1/2}$ and grid center point $x_i := \frac{1}{2}(x_{i-1/2} + x_{i+1/2})$, see Figure 35. We suppose that the unknown function $\mathbf{u} = \mathbf{u}(x,t)$ of the first-order PDE

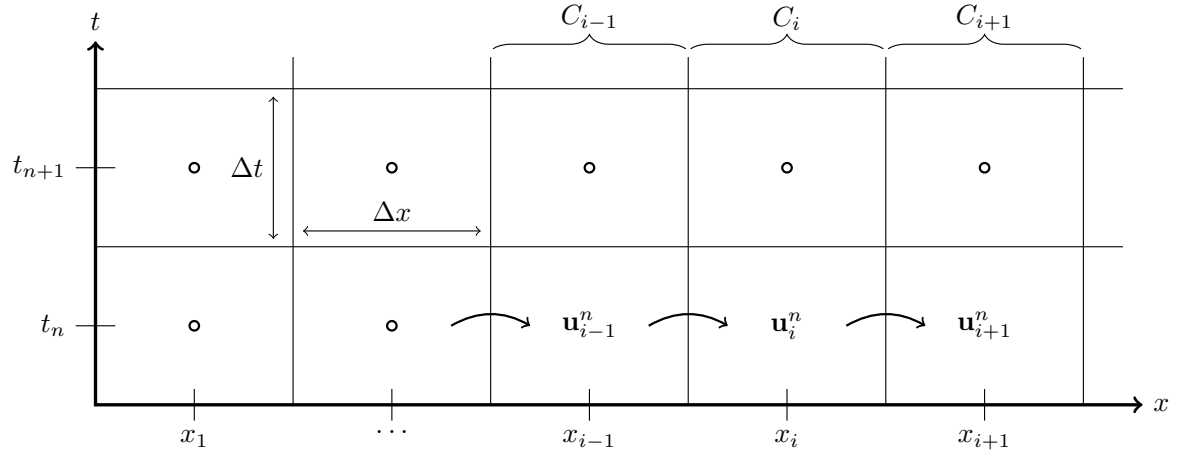


Figure 35: Schematic of grid cells in x-t space. The vector $\mathbf{u}_i^n$ is the collection of approximated mean values in grid cell C_i at time t^n.

system takes its values on $\Omega \subseteq \mathbb{R}^n$. In one space dimension, the Cauchy problem of a first-order system of partial differential equations in space x and time t has the form

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = \mathbf{s}(\mathbf{u}) \quad \text{with} \quad \mathbf{u}(x,t^n) = \mathbf{u}_i^n \quad \text{for} \quad x \in C_i,\ \ t \in (t^n, t^{n+1}). \tag{3.3}$$

3.1. Convergence

The effectiveness of a numerical method is described by its convergence. A method is called convergent, if its global error at a finite time t^N tends to zero for a grid refinement ($\Delta x \to 0$ with fix $\frac{\Delta t}{\Delta x} \equiv \lambda$),

$$\lim_{\Delta x \to 0} \Big\| \underbrace{\mathbf{u}_i^N - \frac{1}{\Delta x} \int_{C_i} \mathbf{u}(x, t^N)\, \mathrm{d}x}_{=:E_i^N} \Big\|_p \stackrel{!}{=} 0. \tag{3.4}$$

The p-norm is defined by

$$\|E_i^N\|_p := \left(\Delta x \cdot \sum_{i=1}^{I} |E_i^N|^p \right)^{1/p}. \tag{3.5}$$

The CFL condition, named after Courant, Friedrichs and Lewy [101], is a necessary condition for stability. This condition ensures that the physical information is just crossing one grid cell per time step. In a hyperbolic system, the eigenvalues of the system are the velocities of the information. Therefore, with $\lambda_{\max}$ as maximum eigenvalue over all cells at a time step i, it must hold

$$\frac{\Delta t}{\Delta x} \stackrel{!}{\leq} \lambda_{\max}. \tag{3.6}$$

3.2. Source terms with splitting techniques

A standard approach of considering the source terms is to use a *fractional-step* method, which complies with the entropy inequality [102]. The full problem is divided in two sub-problems: A first *evolution step* accounts for all convective effects and dissipation,

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u}) \partial_x \mathbf{u} = 0. \tag{3.7}$$

For given initial values $\mathbf{u}_i^n$ this step computes approximate solutions $\mathbf{u}_i^{n+1,-}$ of the hyperbolic homogeneous sub-problem through the time interval $[t^n, t^n + \Delta t]$. Based on these discrete cell values $\mathbf{u}_i^{n+1,-}$ the second *relaxation* step takes the source terms into account,

$$\partial_t \mathbf{u} = \mathbf{s}(\mathbf{u}). \tag{3.8}$$

This ordinary differential equation system is solved with a full time step of length Δt. Thus, the solution of the full system is approximated by alternating between solving the sub-problems, which is known as *Godunov splitting*. This approach allows us to couple high-order methods for the homogeneous system with standard ODE solvers for the source equations.

To solve the whole problem in higher order accuracy, it is necessary that besides the

numerical schemes for (3.7) and (3.8) also the splitting technique is of higher order. For second order, the *Strang splitting operator* [103] can be used which consists of three steps. At first and at last the homogeneous problem is solved with a half time step of length $\frac{\Delta t}{2}$, and in-between the relaxation problem is solved with a full time step of Δt.

If the source terms are stiff, then besides the CFL restriction we get here a much more restrictive time limitation. For this case it is recommended to use implicit ODE solvers.

It remains to find a solver for the homogeneous two-phase flow models in (3.7).

3.3. Homogeneous system in conservative form

We consider the Cauchy problem of the homogeneous system of conservation laws

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = 0 \quad \text{with} \quad \mathbf{u}(x, t^n) = \mathbf{u}_i^n \quad \text{for} \quad x \in C_i, \;\; t \in (t^n, t^{n+1}). \tag{3.9}$$

For its discretization the system is transformed in integral form by spatial integration over cell C_i and temporal integration from t^n to t^{n+1}. The exact integration of the flux derivative is evaluated at the cell interfaces $x_{i\pm 1/2}$, and the integral over the state vector is approximated by Δx times its mean value $\mathbf{u}_i(t)$:

$$\int_{t^n}^{t^{n+1}} \partial_t \mathbf{u}_i(t) \, \mathrm{d}t + \frac{1}{\Delta x} \left(\int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{i+1/2}, t)\big) \, \mathrm{d}t - \int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{i-1/2}, t)\big) \, \mathrm{d}t \right) = 0. \tag{3.10}$$

The exact integration of the state vector derivative is the state vector evaluated at the time intervall values,

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n - \frac{1}{\Delta x} \left(\int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{i+1/2}, t)\big) \, \mathrm{d}t - \int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{i-1/2}, t)\big) \, \mathrm{d}t \right). \tag{3.11}$$

With an approximation of the time integral by some *numerical flux function*

$$\mathbf{F}_{i-1/2}^n \approx \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{i-1/2}, t)\big) \, \mathrm{d}t, \tag{3.12}$$

the update formulation can be written in the general form of a finite volume method:

$$\boxed{\mathbf{u}_i^{n+1} \;=\; \mathbf{u}_i^n - \frac{\Delta t}{\Delta x} \Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n \Big),} \tag{3.13}$$

In the following, different finite volume schemes are presented, where a specific method depends on how to choose the flux function. We suppose that the numerical flux function just depends on the cell averages of both neighboring cells, $\mathbf{F}_{i-1/2}^n := \mathbf{F}(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n)$. This is due to the fact that in hyperbolic systems information propagate with finite speed. In the following, some well-known centered schemes or upwind schemes (by solving the Riemann problem) are presented. For the sake of simplicity, the numerical flux function between two states $\mathbf{u}_\mathrm{L}$ and $\mathbf{u}_\mathrm{R}$ is denoted as $\mathbf{F}(\mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R})$.

▶ Lax-Friedrichs

The classical Lax-Friedrichs method is a forward in time, centered in space scheme. Its numerical flux function is given by

$$\mathbf{F}(\mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) = \frac{1}{2}\left(\mathbf{f}(\mathbf{u}_\mathrm{L}) + \mathbf{f}(\mathbf{u}_\mathrm{R}) - \frac{\Delta x}{\Delta t} \cdot \big(\mathbf{u}_\mathrm{R} - \mathbf{u}_\mathrm{L}\big)\right). \tag{3.14}$$

▶ Rusanov

The Rusanov scheme is also called local Lax-Friedrichs method, because it exploits the maximum wave speed at every cell interface. The numerical flux function is given by

$$\mathbf{F}(\mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) = \frac{1}{2}\left(\mathbf{f}(\mathbf{u}_\mathrm{L}) + \mathbf{f}(\mathbf{u}_\mathrm{R}) - \lambda^{\max} \cdot \big(\mathbf{u}_\mathrm{R} - \mathbf{u}_\mathrm{L}\big)\right). \tag{3.15}$$

with maximum absolute eigenvalue in both neighboring cells

$$\lambda^{\max} = \max\left\{\lambda^{\max}\big(|A(\mathbf{u}_\mathrm{L})|\big),\ \lambda^{\max}\big(|A(\mathbf{u}_\mathrm{R})|\big)\right\} \tag{3.16}$$

with matrix $A(\mathbf{u})$ as Jacobian of the flux function $A(\mathbf{u}) = \mathbf{f}'(\mathbf{u})$.

▶ FORCE

The FORCE scheme is a first-order centered scheme. The numerical flux function is given by

$$\mathbf{F}(\mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) = \frac{1}{4}\left(\mathbf{f}(\mathbf{u}_\mathrm{L}) + 2 \cdot \mathbf{f}\big(\mathbf{u}_{1/2}\big) + \mathbf{f}(\mathbf{u}_\mathrm{R}) - \frac{\Delta x}{\Delta t} \cdot \Big(\mathbf{u}_\mathrm{R} - \mathbf{u}_\mathrm{L}\Big)\right), \tag{3.17}$$

with intermediate state

$$\mathbf{u}_{1/2} = \frac{1}{2}\left(\mathbf{u}_\mathrm{L} + \mathbf{u}_\mathrm{R} - \frac{\Delta t}{\Delta x} \cdot \big(\mathbf{f}(\mathbf{u}_\mathrm{R}) - \mathbf{f}(\mathbf{u}_\mathrm{L})\big)\right). \tag{3.18}$$

Compared to the centred Lax-Friedrichs flux, the FORCE flux has smaller numerical viscosity [104].

▶ Godunov

The Godunov method reconstructs the discrete cell averages by a simple piece-wise constant function. As reconstruction the exact solution $\mathbf{u}_r$ of the Riemann problem is used and plugged into the flux function in (3.12). Along the interface at $x_{1/2}$ the Riemann solution is constant in time, such that the time integral can be solved exactly,

$$\begin{aligned}\mathbf{F}(\mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) &= \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{f}\big(\mathbf{u}(x_{1/2}, t)\big)\,\mathrm{d}t \\ &= \mathbf{f}\big(\mathbf{u}_\mathrm{r}(0; \mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R})\big).\end{aligned} \tag{3.19}$$

Notice that the reconstruction $\mathbf{u}_\mathrm{r}(\frac{x}{t}; \mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R})$ only depends on the neighboring states $\mathbf{u}_\mathrm{L}$ and $\mathbf{u}_\mathrm{R}$.

▶ HLL

The HLL scheme is an approximate Riemann solver proposed by Harten Lax and van Leer. This method considers just the slowest $\lambda^{\min}$ and fastest $\lambda^{\max}$ wave speeds of the system, emerging from the initial discontinuity at the interface. By applying the Rankine-Hugoniot conditions across the left and right waves respectively, the following intermediate state is obtained [105]:

$$\mathbf{u}_{1/2} = \frac{S_{\mathrm{R}}\,\mathbf{u}_{\mathrm{R}} - S_{\mathrm{L}}\,\mathbf{u}_{\mathrm{L}} + \mathbf{f}(\mathbf{u}_{\mathrm{L}}) - \mathbf{f}(\mathbf{u}_{\mathrm{R}})}{S_{\mathrm{R}} - S_{\mathrm{L}}}. \tag{3.20}$$

Thus, the numerical HLL flux function is given by

$$\begin{aligned}\mathbf{F}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \mathbf{f}(\mathbf{u}_{\mathrm{L}}) + S_{\mathrm{L}}\,(\mathbf{u}_{1/2} - \mathbf{u}_{\mathrm{L}}) \\ &= \frac{S_{\mathrm{R}}\,\mathbf{f}(\mathbf{u}_{\mathrm{L}}) - S_{\mathrm{L}}\,\mathbf{f}(\mathbf{u}_{\mathrm{R}}) + S_{\mathrm{L}}\,S_{\mathrm{R}}\,(\mathbf{u}_{\mathrm{R}} - \mathbf{u}_{\mathrm{L}})}{S_{\mathrm{R}} - S_{\mathrm{L}}}.\end{aligned} \tag{3.21}$$

The speed of the signals S_{L} and S_{R} are related to the minimum and maximum eigenvalues of the Jacobians of both neighboring states,

$$S_{\mathrm{L}} = \min\big\{0,\ \lambda^{\min}(\mathbf{f}_{\mathbf{u}}(\mathbf{u}_{\mathrm{L}})),\ \lambda^{\min}(\mathbf{f}_{\mathbf{u}}(\mathbf{u}_{\mathrm{R}}))\big\}, \tag{3.22}$$

and

$$S_{\mathrm{R}} = \max\big\{0,\ \lambda^{\max}(\mathbf{f}_{\mathbf{u}}(\mathbf{u}_{\mathrm{L}})),\ \lambda^{\max}(\mathbf{f}_{\mathbf{u}}(\mathbf{u}_{\mathrm{R}}))\big\}. \tag{3.23}$$

3.4. Finite volume method for two-phase flow model

Because the *homogeneous equilibrium model* (2.139) is in conservative form, the standard finite volume methods introduced in Section 3.3 can be used. Unfortunately, the *two-velocity two-pressure model* (2.57) additionally has a non-conservative coefficient matrix $B(\mathbf{u})$. Thus, the Cauchy problem is given by

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = 0 \quad \text{with} \quad \mathbf{u}(x, t^n) = \mathbf{u}_i^n \quad \text{for} \quad x \in C_i,\ \ t \in (t^n, t^{n+1}). \tag{3.24}$$

Again (as already done in Section 3.3) the system is discretized by spatial integration over cell C_i and temporal integration from t^n to t^{n+1}, such that

$$\mathbf{u}_i^{n+1} \;=\; \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big) - \frac{1}{\Delta x}\int_{t^{n+1,-}}^{t^{n+1}} \int_{C_i} B(\mathbf{u})\partial_x \mathbf{u}\ \mathrm{d}x, \tag{3.25}$$

with some numerical flux function, see (3.12). With an approximation of the last term

$$\mathbf{N}_i^n \approx \frac{1}{\Delta t}\int_{t^{n+1,-}}^{t^{n+1}} \int_{C_i} B(\mathbf{u})\partial_x \mathbf{u}\ \mathrm{d}x, \tag{3.26}$$

the update formulation can be written as

$$\boxed{\mathbf{u}_i^{n+1} \;=\; \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\Big(\mathbf{F}^n_{i+1/2} - \mathbf{F}^n_{i-1/2}\Big) - \frac{\Delta t}{\Delta x}\mathbf{N}_i^n.} \tag{3.27}$$

The integral form of a partial differential equation system forms the basis for the mathematical theory of weak solutions, including the derivation of the Rankine-Hugoniot conditions that govern the form and speed of shock waves [14]. Thus, for a method in integral form it is guaranteed that the discrete solution satisfies *mass conservation*, if the grid is refined [99].

Non-conservative systems consist of parts which are based on differential equations, here it is $B(\mathbf{u})\partial_x\mathbf{u}$. Methods for these forms can fail, because *mass conservation* is not guaranteed anymore: The question is how to approximate the non-conservative flux vector $\mathbf{N}_i$ in (3.26)? As LeVeque reported, an upwind method on smooth solutions is first-order accurate. But when the solution contains a shock wave, an upwind step fails to converge to a weak solution of the corresponding conservation law, compare with Figure 36.

We use this disadvantage as motivation to develop alternative approaches. In Section 4 a path-conservative and entropy stable solver is developed.

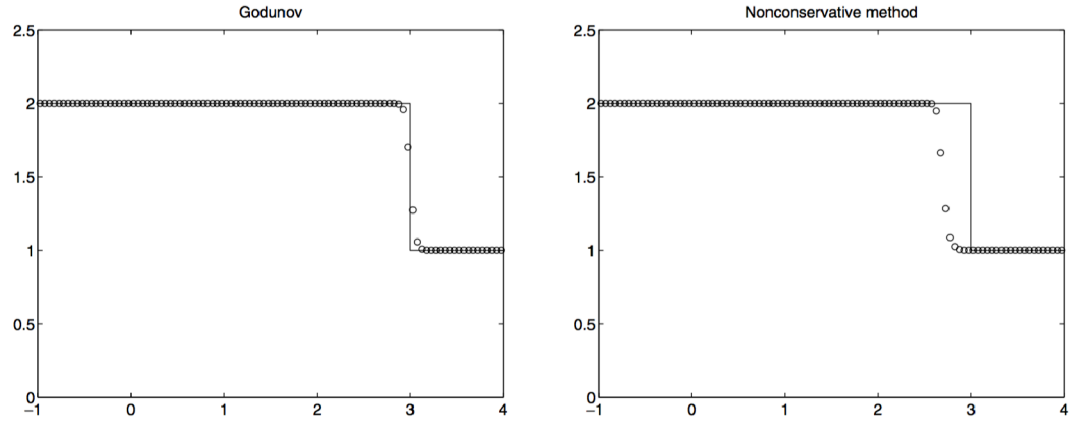


Figure 36: Burger's equation in conservative form $u_t + \frac{1}{2}(u^2)_x = 0$, and in non-conservative form $u_t + u\,u_x = 0$, solved by conservative and non-conservative upwind methods. On the domain $x \in [-5, 5]$ a Riemann problem is defined at $x = 0$ with $u_\mathrm{L} = 2$ and $u_\mathrm{R} = 1$. The result is shown at time $t = 2$. Image taken from LeVeque [14].

4. Path-conservative numerical solver

Because the integrand of a non-conservative term is not defined for discontinuous functions $\mathbf{u}$, the term is expected to produce a Dirac measure [106]. To define weak solutions of this integral equation we need to approximate the non-conservative term. We follow Dal Maso, LeFloch and Murat [107], who proposed the path-conservative method, where the non-conservative term is interpreted as Borel measures over a Lipschitz continuous path $\Phi : [0,1] \times \Omega \times \Omega \to \Omega$ with

$$\Phi(0;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) = \mathbf{u}_\mathrm{L}, \quad \Phi(1;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) = \mathbf{u}_\mathrm{R}, \quad \Phi(s;\mathbf{u},\mathbf{u}) = \mathbf{u}, \quad \forall\, \mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R},\mathbf{u} \in \Omega. \tag{4.1}$$

This approach is now used to solve the Cauchy problem of the homogeneous *two-velocity two-pressure model* (2.57),

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = 0 \quad \text{with} \quad \mathbf{u}(x,t^n) = \mathbf{u}_i^n \quad \text{for} \quad x \in C_i, \;\; t \in (t^n, t^{n+1}). \tag{4.2}$$

Now, the spatial derivative of the flux function is replaced by applying the chain rule. With $A(\mathbf{u}) := \partial_\mathbf{u}\mathbf{f}(\mathbf{u}) + B(\mathbf{u})$ we finally get the following Cauchy problem

$$\partial_t \mathbf{u} + A(\mathbf{u})\,\partial_x \mathbf{u} = 0 \quad \text{with} \quad \mathbf{u}(x,t^n) = \mathbf{u}_i^n \quad \text{for} \quad x \in C_i, \;\; t \in (t^n, t^{n+1}). \tag{4.3}$$

Notice that for the *two-velocity two-pressure model* this system is in non-conservative form, because $A(\mathbf{u})$ is not a Jacobian. Non-conservative systems are regarded in integral form

$$\partial_t \mathbf{u}_i(t) + \int_{C_i} A(\mathbf{u})\,\partial_x \mathbf{u}\; \mathrm{d}x\,\mathrm{d}t = \mathbf{0}. \tag{4.4}$$

The path defined in (4.1) connects two states $\mathbf{u}_\mathrm{L}$ and $\mathbf{u}_\mathrm{R}$ at its left and right limits across a discontinuity with $s \in [0,1]$. Hence, the matrix $A(\mathbf{u})$ is interpreted as $A(\Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}))$. The chosen path of a weak solution influences the speed of propagation σ_Φ of the discontinuity, due to the generalized Rankine-Hugoniot condition [108, 109],

$$\begin{aligned} \sigma_\Phi \cdot (\mathbf{u}_\mathrm{R} - \mathbf{u}_\mathrm{L}) &= \int_{\mathbf{u}_\mathrm{L}}^{\mathbf{u}_\mathrm{R}} A(\mathbf{u})\,\mathrm{d}\mathbf{u} \\ &= \int_0^1 A\big(\Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})\big)\frac{\partial \Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})}{\partial s}\,\mathrm{d}s. \end{aligned} \tag{4.5}$$

The weak solutions now depend on the arbitrary chosen path Φ, where different families of paths lead to different jump conditions [108]. Note, that contact discontinuities and rarefraction waves do not depend on the path. We are looking for the physically relevant solutions, which ensure that the jump conditions are consistent with the viscous profiles Notice that if $A(\mathbf{u})$ is the Jacobian matrix of some flux vector $f(\mathbf{u})$, then σ_Φ does not depend on the path, i.e. it does not depend on numerical viscosity, and (4.5) reduces to the standard Rankine-Hugoniot condition, $\sigma \cdot (\mathbf{u}_\mathrm{R} - \mathbf{u}_\mathrm{L}) = f(\mathbf{u}_\mathrm{R}) - f(\mathbf{u}_\mathrm{L})$.

By choosing a segment path, $\Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) := \mathbf{u}_\mathrm{L} + s\cdot(\mathbf{u}_\mathrm{R}-\mathbf{u}_\mathrm{L})$, the generalized Rankine-Hugoniot condition (4.5) simplifies to

$$\sigma_\Phi \cdot (\mathbf{u}_\mathrm{R}-\mathbf{u}_\mathrm{L}) = \underbrace{\int_0^1 A\big(\Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})\big)\,\mathrm{d}s}_{=:A_\Phi(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})}\ (\mathbf{u}_\mathrm{R}-\mathbf{u}_\mathrm{L}). \tag{4.6}$$

The matrix $A_\Phi(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})$ is a local linearization of the system matrix $A(\mathbf{u})$ and is called *Roe matrix*. For complicated hyperbolic systems it may become very tedious or even impossible to compute an analytical expression for the Roe matrix. Therefore, Canestrelli et al. [110] propose to compute the integral directly along the segment path Φ by using high order accurate Gaussian quadrature rules[13],

$$A_\Phi(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) \approx \sum_{j=1}^{J} w_j\, A\big(\Phi(s_j;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})\big). \tag{4.7}$$

The general idea is to decompose the total mass of this Dirac measure into two summands $\mathbf{D}^\pm_{i+1/2}$, such that the scheme can be written in the following *fluctuation* form [111, 108],

$$\partial_t \mathbf{u}_i(t) + \frac{1}{\Delta x}\left(\mathbf{D}^-_{i+1/2}(t) + \mathbf{D}^+_{i-1/2}(t)\right) = \mathbf{0}, \tag{4.8}$$

where the *fluctuations* $\mathbf{D}^\pm : \Omega\times\Omega\to\Omega$ are two Lipschitz continuous functions, defined by $\mathbf{D}^\pm_{i-1/2} := \mathbf{D}^\pm(\mathbf{u}_{i-1},\mathbf{u}_i)$. For an explicit Euler step in time, the update formulation for the cell average at the next time step can be written in the form

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \ - \ \frac{\Delta t}{\Delta x}\left(\mathbf{D}^-_{i+1/2} + \mathbf{D}^+_{i-1/2}\right). \tag{4.9}$$

In the following different path-conservative schemes are presented. For notational simplicity we drop the depency of the path Φ on two neighboring states $\mathbf{u}_\mathrm{L}$ and $\mathbf{u}_\mathrm{R}$ by using $\Phi(s) := \Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R})$.

4.1. Path-consistent schemes

A numerical scheme should be *consistent* in the usual sense. For path-conservative schemes Parés [111] introduced an analogue property: A numerical scheme is said to be *path-consistent* if the fluctuations are satisfying

$$\mathbf{D}^\pm(\mathbf{u},\mathbf{u}) = 0 \tag{4.10}$$

and

$$\mathbf{D}^-(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) + \mathbf{D}^+(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) = \int_0^1 A\big(\Phi(s)\big)\ \Phi'(s)\,\mathrm{d}s. \tag{4.11}$$

In the following, the non-conservative version of some well known finite volume schemes are presented.

[13] By using e.g. a three-point Gaussian quadrature rule ($J=3$), the re-scaled points are given by $s_1 = \frac{1}{2}$ and $s_{2,3} = \frac{1}{2} \pm \frac{\sqrt{15}}{10}$, with weights $w_1 = \frac{8}{18}$ and $w_{2,3} = \frac{5}{18}$.

▶ Path-conservative Lax-Friedrichs

The path-conservative version of the classical Lax-Friedrichs method is given by [106]

$$\mathbf{D}^{\pm}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) = \frac{1}{2}\left(\int_0^1 A\big(\Phi(s)\big)\ \Phi'(s)\,\mathrm{d}s\ \pm \frac{\Delta x}{\Delta t} \cdot \big(\mathbf{u}_{\mathrm{R}} - \mathbf{u}_{\mathrm{L}}\big) \right). \tag{4.12}$$

▶ Path-conservative Rusanov

The path-conservative version of the local Lax-Friedrichs method is given by [106]

$$\mathbf{D}^{\pm}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) = \frac{1}{2}\left(\int_0^1 A\big(\Phi(s)\big)\ \Phi'(s)\,\mathrm{d}s\ \pm \lambda^{\max} \cdot \big(\mathbf{u}_{\mathrm{R}} - \mathbf{u}_{\mathrm{L}}\big) \right). \tag{4.13}$$

with maximum absolute eigenvalue in both neighboring cells

$$\lambda^{\max} = \max\Big\{ \lambda^{\max}\big(|A(\mathbf{u}_{\mathrm{L}})|\big),\ \lambda^{\max}\big(|A(\mathbf{u}_{\mathrm{R}})|\big) \Big\}. \tag{4.14}$$

▶ Path-conservative FORCE (PRICE-C)

The PRICE-C scheme is a primitive centred scheme for non-conservative system that automatically reduces to a modified conservative FORCE scheme if the underlying PDE system is a conservation law [110]. The numerical flux function of the scheme is given by

$$\mathbf{D}^{\pm}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) = \frac{1}{4}\left(2 \cdot A_{\Phi}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) \pm \frac{\Delta x}{\Delta t}\mathcal{I} \pm \frac{\Delta t}{\Delta x}\Big(A_{\Phi}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})\Big)^2 \right), \tag{4.15}$$

with Roe matrix A_{Φ} (4.7) and identity matrix $\mathcal{I}$.

▶ Path-conservative Godunov

Again we denote with $\mathbf{u}_{\mathrm{r}}$ the exact solution of the Riemann problem, which is known to be constant along the interface at $x_{1/2}$ between two states $\mathbf{u}_{\mathrm{L}}$ and $\mathbf{u}_{\mathrm{R}}$. With the Riemann solution $\mathbf{u}_{\mathrm{r}}^0 := \mathbf{u}_{\mathrm{r}}(0; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})$, the fluctuations are given by [106]

$$\begin{aligned} \mathbf{D}^{-}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \int_0^1 A\Big(\Phi\big(s; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{r}}^0\big)\Big)\ \Phi'\big(s; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{r}}^0\big)\,\mathrm{d}s \\ \mathbf{D}^{+}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \int_0^1 A\Big(\Phi\big(s; \mathbf{u}_{\mathrm{r}}^0, \mathbf{u}_{\mathrm{R}}\big)\Big)\ \Phi'\big(s; \mathbf{u}_{\mathrm{r}}^0, \mathbf{u}_{\mathrm{R}}\big)\,\mathrm{d}s \end{aligned} \tag{4.16}$$

By choosing a segment path, $\Phi(s; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) := \mathbf{u}_{\mathrm{L}} + s \cdot (\mathbf{u}_{\mathrm{R}} - \mathbf{u}_{\mathrm{L}})$ the scheme can be written in the form (3.27),

$$\mathbf{u}_i^{n+1} \ = \ \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big) - \frac{\Delta t}{\Delta x}\mathbf{N}_i^n,$$

with numerical flux function $\mathbf{F}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})$ from (3.19), and with an approximation of the non-conservative part in (3.26),

$$\mathbf{N}_i^n = \frac{1}{2}\Big(\mathbf{B}(\mathbf{u}_{i-1/2}^n) + \mathbf{B}(\mathbf{u}_{i+1/2}^n)\Big)\left(\mathbf{u}_{i+1/2}^n - \mathbf{u}_{i-1/2}^n\right). \tag{4.17}$$

▶ Path-conservative HLL

The fluctuations of the non-conservative HLL scheme are given by [112]

$$\begin{aligned}\mathbf{D}^{-}(\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}) &= -\frac{S_{\mathrm{L}}}{S_{\mathrm{R}}-S_{\mathrm{L}}}\cdot\int_0^1 A\Big(\Phi\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\Big)\;\Phi'\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\,\mathrm{d}s + \frac{S_{\mathrm{L}}\,S_{\mathrm{R}}\,(\mathbf{u}_{\mathrm{R}}-\mathbf{u}_{\mathrm{L}})}{S_{\mathrm{R}}-S_{\mathrm{L}}},\\ \mathbf{D}^{+}(\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}) &= +\frac{S_{\mathrm{R}}}{S_{\mathrm{R}}-S_{\mathrm{L}}}\cdot\int_0^1 A\Big(\Phi\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\Big)\;\Phi'\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\,\mathrm{d}s - \frac{S_{\mathrm{L}}\,S_{\mathrm{R}}\,(\mathbf{u}_{\mathrm{R}}-\mathbf{u}_{\mathrm{L}})}{S_{\mathrm{R}}-S_{\mathrm{L}}},\end{aligned} \tag{4.18}$$

where the speed of the signals S_{L} and S_{R} are related to the minimum and maximum eigenvalues of the Jacobians of both neighboring states,

$$S_{\mathrm{L}} = \min\big\{0,\,\lambda^{\min}(A(\mathbf{u}_{\mathrm{L}})),\,\lambda^{\min}(A(\mathbf{u}_{\mathrm{R}}))\big\}, \tag{4.19}$$

and

$$S_{\mathrm{R}} = \max\big\{0,\,\lambda^{\max}(A(\mathbf{u}_{\mathrm{L}})),\,\lambda^{\max}(A(\mathbf{u}_{\mathrm{R}}))\big\}. \tag{4.20}$$

By applying the Rankine-Hugoniot conditions across the left and right waves respectively, the following approximated intermediate state is obtained [112],

$$\mathbf{u}_{1/2} = \frac{S_{\mathrm{R}}\,\mathbf{u}_{\mathrm{R}} - S_{\mathrm{L}}\,\mathbf{u}_{\mathrm{L}} + \mathbf{f}(\mathbf{u}_{\mathrm{L}}) - \mathbf{f}(\mathbf{u}_{\mathrm{R}})}{S_{\mathrm{R}}-S_{\mathrm{L}}} + \frac{1}{S_{\mathrm{R}}-S_{\mathrm{L}}}\cdot\int_0^1 A\Big(\Phi\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\Big)\;\Phi'\big(s;\mathbf{u}_{\mathrm{L}},\mathbf{u}_{\mathrm{R}}\big)\,\mathrm{d}s, \tag{4.21}$$

compare with (3.20). This intermediate state can be used to modify the integration path.

Castro et al. [108, 113], and Abgrall and Karni [109] observed that the numerical solution of a path-conservative scheme with a suitable consistent selected path along the viscous profile, will not necessarily converge to the correct and physically relevant solution. This lack of convergence has its origin in the numerical viscosity of the scheme. Thus, here we follow the approach of entropy-conservative schemes.

4.2. Entropy conservative path-consistent (ECPC) schemes

For a given entropy pair (η, ψ) with entropy variables $\mathbf{v}(\mathbf{u}) := \partial_{\mathbf{u}}\eta(\mathbf{u})$ the solutions $\mathbf{u}_i$ in cell C_i should satisfy the discrete entropy identity

$$\partial_t \eta(\mathbf{u}_i) + \frac{1}{\Delta x}\left(\tilde{\psi}_{i+1/2} - \tilde{\psi}_{i-1/2}\right) = 0, \tag{4.22}$$

with numerical entropy flux

$$\tilde{\psi}_{i+1/2} := \tilde{\psi}(\mathbf{u}_i, \mathbf{u}_{i+1}) = \psi(\mathbf{u}_i) + \mathbf{v}_i^{\mathsf{T}}\, \mathbf{D}^-(\mathbf{u}_i, \mathbf{u}_{i+1}). \tag{4.23}$$

Notice that with (4.10) the numerical entropy flux $\tilde{\psi}$ is consistent with the entropy flux ψ, meaning $\tilde{\psi}(\mathbf{u}, \mathbf{u}) \equiv \psi(\mathbf{u})$. Following Castro et al. [108], a path-consistent scheme is said to be *entropy conservative* if the fluctuations are satisfying

$$\begin{aligned}\mathbf{v}_{\mathrm{L}}^{\mathsf{T}}\, \mathbf{D}^-(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) + \mathbf{v}_{\mathrm{R}}^{\mathsf{T}}\, \mathbf{D}^+(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \int_0^1 \mathbf{v}\big(\Phi(s)\big)^{\mathsf{T}}\, A\big(\Phi(s)\big)\; \Phi'(s)\, \mathrm{d}s \\ &= \psi(\mathbf{u}_{\mathrm{R}}) - \psi(\mathbf{u}_{\mathrm{L}}).\end{aligned} \tag{4.24}$$

If the above condition holds the discrete entropy identity can directly be derived from our scheme in fluctuation form (4.8). Thus we get

$$\begin{aligned}\partial_t \mathbf{u}_i &= -\frac{1}{\Delta x}\big(\mathbf{D}^-_{i+1/2} + \mathbf{D}^+_{i-1/2}\big) \\ \Leftrightarrow \quad \mathbf{v}_i^{\mathsf{T}}\, \partial_t \mathbf{u}_i &= -\frac{1}{\Delta x}\left(\mathbf{v}_i^{\mathsf{T}}\, \mathbf{D}^-_{i+1/2} + \psi(\mathbf{u}_i) - \psi(\mathbf{u}_i) + \mathbf{v}_i^{\mathsf{T}}\, \mathbf{D}^+_{i-1/2}\right) \\ \overset{(4.23)}{\underset{(4.24)}{\Leftrightarrow}}\ \partial_t \eta(\mathbf{u}_i) &= -\frac{1}{\Delta x}\left(\tilde{\psi}_{i+1/2} - \tilde{\psi}_{i-1/2}\right).\end{aligned} \tag{4.25}$$

Here we present two different ways to create an ECPC scheme.

▶ ECPC (along state vector)

Castro et al. [108] assumed the existence of matrices $B^-\big(\Phi(s)\big)$ and $B^+\big(\Phi(s)\big)$ with $B^- + B^+ = \mathcal{I}$ and $B^-\mathbf{v}_{\mathrm{L}} + B^+\mathbf{v}_{\mathrm{R}} = \mathbf{v}(\Phi)$, such that

$$\begin{aligned}\mathbf{D}^+(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \int_0^1 B^+\big(\Phi(s)\big)^{\mathsf{T}}\; A\big(\Phi(s)\big)\; \Phi'(s)\, \mathrm{d}s, \\ \mathbf{D}^-(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) &= \int_0^1 B^-\big(\Phi(s)\big)^{\mathsf{T}}\; A\big(\Phi(s)\big)\; \Phi'(s)\, \mathrm{d}s.\end{aligned} \tag{4.26}$$

is an ECPC scheme. The integrals can be solved numerically with Gaussian quadrature rule. With $\xi \in [0, 1]$, the choice

$$B^+\big(\Phi(s)\big)^{\mathsf{T}} = \xi\, \mathcal{I} \qquad + \frac{\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}}{|\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}|^2}\left(\mathbf{v}\big(\Phi(s)\big) - \xi\mathbf{v}_{\mathrm{R}} - (1-\xi)\mathbf{v}_{\mathrm{L}}\right)^{\mathsf{T}},$$

$$B^-\big(\Phi(s)\big)^\mathsf{T} = (1-\xi)\,\mathcal{I} \quad - \frac{\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}}{|\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}|^2}\,\Big(\mathbf{v}\big(\Phi(s)\big) - \xi\mathbf{v}_\mathrm{R} - (1-\xi)\mathbf{v}_\mathrm{L}\Big)^\mathsf{T}, \tag{4.27}$$

shows the existence of infinitely many ECPC schemes for a given entropy pair (η,ψ) and familiy of paths Φ [108]. For the case of $\mathbf{v}_\mathrm{L}\to\mathbf{v}_\mathrm{R}$ the fluctuations $D^\pm$ go to zero, due to Φ'. That's what we expect from the path consistency condition in (4.10). For the implementation we have to be careful with the denominator $|\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}|^2$ in $B^\pm$, which is zero for the case of $\mathbf{v}_\mathrm{L}\equiv\mathbf{v}_\mathrm{R}$.

▶ ECPCs (along entropy variables)

An equivalent condition for an ECPC scheme can be given by inserting the path consistency condition (4.11) into the entropy conservative condition (4.24), such that it must hold

$$\begin{aligned}
\big(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}\big)^\mathsf{T}\,\mathbf{D}^+(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &= \int_0^1 \big(\mathbf{v}\big(\Phi(s)\big)-\mathbf{v}_\mathrm{L}\big)^\mathsf{T}\,A\big(\Phi(s)\big)\,\Phi'(s)\,\mathrm{d}s,\\
\big(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}\big)^\mathsf{T}\,\mathbf{D}^+(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &= \int_0^1 \big(\mathbf{v}_\mathrm{R}-\mathbf{v}\big(\Phi(s)\big)\big)^\mathsf{T}\,A\big(\Phi(s)\big)\,\Phi'(s)\,\mathrm{d}s.
\end{aligned} \tag{4.28}$$

Now we choose the path Φ as a segment along the entropy variables,

$$\begin{aligned}
\Phi(s;\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &:= \mathbf{u}\big(\Psi(s;\mathbf{v}_\mathrm{L},\mathbf{v}_\mathrm{R})\big)\\
&:= \mathbf{u}\big(\mathbf{v}_\mathrm{L}+s\cdot(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L})\big)\\
&= \mathbf{u}\big(\mathbf{v}_\mathrm{R}-(1-s)\cdot(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L})\big),
\end{aligned} \tag{4.29}$$

with $\mathbf{v}_\mathrm{R}:=\mathbf{v}(\mathbf{u}_\mathrm{R})$ and $\mathbf{v}_\mathrm{L}:=\mathbf{v}(\mathbf{u}_\mathrm{L})$ as entropy variables and $\mathbf{u}(\mathbf{v})$ as the inverse mapping. With the above chosen path, the factor $\big(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}\big)^\mathsf{T}$ shows up also at the right-hand side of (4.28). So, finally we get

$$\begin{aligned}
\mathbf{D}^+(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &= \left(\int_0^1 s\cdot A\big(\Phi(s)\big)\,\mathbf{u}'\big(\mathbf{v}_\mathrm{L}+s\cdot(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L})\big)\,\mathrm{d}s\right)\;(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}),\\
\mathbf{D}^-(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &= \left(\int_0^1 (1-s)\cdot A\big(\Phi(s)\big)\,\mathbf{u}'\big(\mathbf{v}_\mathrm{L}+s\cdot(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L})\big)\,\mathrm{d}s\right)\;(\mathbf{v}_\mathrm{R}-\mathbf{v}_\mathrm{L}).
\end{aligned} \tag{4.30}$$

The integrals can be solved numerically applying a Gaussian quadrature rule. For this scheme we need the inverse mapping of the entropy variables $\mathbf{u}(\mathbf{v})$ and its derivative $\mathbf{u}':=\partial_\mathbf{v}\mathbf{u}$. Both quantities are derived now for each model.

Because the entropy function $\eta(\mathbf{u})$ is strictly convex, the mapping $\mathbf{u}\mapsto\mathbf{v}$ of the state vector to the entropy variables is bijective, such that an inverse mapping $\mathbf{u}(\mathbf{v})$ exists. Tadmor and Zhong [114] computed the inverse mapping for the Euler equations (one-phase flow) under the assumption of ideal gas laws. In our case we have to find the inverse mapping for the *two-velocity two-pressure model* paired with more complicated (because more accurate) equations of state[14]. We call this type of solver ESPCs, where 's' stands for the used path along the entropy variables.

[14] The data-based equation of state bases on a fundamental equation for the specific Gibbs free energy [78].

▶ Inverse mapping of entropy variables

- **Homogeneous equilibrium model**
  Because the entropy variables consist of non-linear derivatives of the specific entropy, such that an explicit expression for the state vector $\mathbf{u}(\mathbf{v})$ cannot be given. Therefore we need another approach. We use the fact that the density and the specific inner energy can be expressed by the temperature and specific Gibbs free energy[15].

  So, for given entropy variables $\mathbf{v} \in \mathbb{R}^3$ with $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)^\mathsf{T}$, we can invert the entropy variables from (2.152), such that we get:

$$T_\mathrm{h} = -\frac{1}{\mathbf{v}_3}, \qquad v_\mathrm{h} = \mathbf{v}_2 T_\mathrm{h} = -\frac{\mathbf{v}_2}{\mathbf{v}_3}, \qquad g_\mathrm{h} = \mathbf{v}_1 T_\mathrm{h} + \tfrac{1}{2} v_\mathrm{h}^2 = -\frac{\mathbf{v}_1}{\mathbf{v}_3} + \frac{\mathbf{v}_2^2}{2\,\mathbf{v}_3^2}. \tag{4.31}$$

  Because the density $\rho_\mathrm{h} := \rho_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big)$ and the specific inner energy $u_\mathrm{h} := u_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big)$ are given by an equation of state in dependency of the temperature and the Gibbs free energy, the inverse mapping of the entropy variables is given by

$$\mathbf{u}\big(s; \mathbf{v}\big) := \begin{pmatrix} \rho_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h} \\ \rho_\mathrm{h} E_\mathrm{h} \end{pmatrix} = \begin{pmatrix} \rho_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big) \\ \rho_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big)\; v_\mathrm{h} \\ \rho_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big)\;\; \big(u_\mathrm{h}\big(T_\mathrm{h},\, g_\mathrm{h}\big) + \tfrac{1}{2} v_\mathrm{h}^2\big) \end{pmatrix} \tag{4.32}$$

- **Two-velocity two-pressure model**
  For the two-velocity two-pressure model we cannot find an exact inverting of the entropy variables. We follow the above proposed approach: For given entropy variables $\mathbf{v} \in \mathbb{R}^7$ with $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_7)^\mathsf{T}$, we can invert the entropy variables from (2.92), such that we get:

$$\begin{aligned} T_\ell &= -\frac{1}{\mathbf{v}_4}, & T_g &= -\frac{1}{\mathbf{v}_7}, \\ v_\ell &= \mathbf{v}_3 T_\ell = -\frac{\mathbf{v}_3}{\mathbf{v}_4}, & v_g &= \mathbf{v}_6 T_g = -\frac{\mathbf{v}_6}{\mathbf{v}_7}, \\ g_\ell &= \mathbf{v}_2 T_\ell + \tfrac{1}{2} v_\ell^2 = -\frac{\mathbf{v}_2}{\mathbf{v}_4} + \frac{\mathbf{v}_3^2}{2\,\mathbf{v}_4^2}, & g_g &= \mathbf{v}_5 T_g + \tfrac{1}{2} v_g^2 = -\frac{\mathbf{v}_5}{\mathbf{v}_7} + \frac{\mathbf{v}_6^2}{2\,\mathbf{v}_7^2}, \end{aligned} \tag{4.33}$$

  As stated above, the density $\rho_k := \rho_k\big(T_k,\, g_k\big)$ and the specific inner energy $u_k := u_k\big(T_k,\, g_k\big)$ are given by an equation of state in dependency of the temperature and the Gibbs free energy.

  It remains to compute the void fraction α from the entropy variables. Remember that the scheme just needs the inverse mapping for computing the path

$$\Phi(s; \mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) := \mathbf{u}\big(\Psi(s; \mathbf{v}_\mathrm{L}, \mathbf{v}_\mathrm{R})\big) := \mathbf{u}(s; \mathbf{v})$$

[15] From [78] the relation $g(p, T)$ can be inverted to get the pressure p. With known Gibbs free energy, pressure and temperature the other properties can be derived.

between two given states $\mathbf{v}_\mathrm{L}$ and $\mathbf{v}_\mathrm{R}$ with known $\mathbf{u}_\mathrm{L}$ and $\mathbf{u}_\mathrm{R}$, with $s \in [0,1]$. Because the left and right volume fraction α_L and α_R are known, we have the possibility to give an approximate expression for α. We simply approximate α by the mean value

$$\alpha(s) \equiv \frac{1}{2}(\alpha_\mathrm{L} + \alpha_\mathrm{R}). \tag{4.34}$$

So finally, the approximate inverse mapping of the entropy variables is given by

$$\mathbf{u}(\mathbf{v}) \approx \mathbf{u}_{\text{approx}}\big(s;\mathbf{v}\big) := \begin{pmatrix} \alpha_s \\ (1-\alpha_s)\rho_\ell \\ (1-\alpha_s)\rho_\ell v_\ell \\ (1-\alpha_s)\rho_\ell E_\ell \\ \alpha_s\rho_g \\ \alpha_s\rho_g v_g \\ \alpha_s\rho_g E_g \end{pmatrix} = \begin{pmatrix} \alpha_s \\ (1-\alpha_s)\ \rho_\ell\big(T_\ell,\, g_\ell\big) \\ (1-\alpha_s)\ \rho_\ell\big(T_\ell,\, g_\ell\big)\ v_\ell \\ (1-\alpha_s)\ \rho_\ell\big(T_\ell,\, g_\ell\big)\ \big(u_\ell\big(T_\ell,\, g_\ell\big) + \frac{1}{2}v_\ell^2\big) \\ \alpha_s\ \rho_g\big(T_g,\, g_g\big) \\ \alpha_s\ \rho_g\big(T_g,\, g_g\big)\ v_g \\ \alpha_s\ \rho_g\big(T_g,\, g_g\big)\ \big(u_g\big(T_g,\, g_g\big) + \frac{1}{2}v_g^2\big) \end{pmatrix}. \tag{4.35}$$

Note that because of the approximation the entropy variables are not exact. In the following, the derivative $\mathbf{u}'(\mathbf{v})$ of the inverse mapping of the entropy variables for both models are developed. This is needed for the ESPCs scheme, see (4.30), and later for the regularization (4.42) of ESPC and ESPCs.

▶ Derivative of inverse mapping of entropy variables

- **Homogeneous equilibrium model**
  The derivative of inverse mapping of entropy variables can directly be derived from the Hessian $\eta''(\mathbf{u})$ (2.154) by the equality

$$\mathbf{u}'(\mathbf{v}) := \partial_\mathbf{v}\mathbf{u} = \big(\partial_\mathbf{u}\mathbf{v}\big)^{-1} = \big(\eta''(\mathbf{u})\big)^{-1}. \tag{4.36}$$

  Thus, by computing the inverse of the Hessian we get

$$\mathbf{u}'(\mathbf{v}) = \begin{pmatrix} \mathbf{u}'_{11} & v_\mathrm{h}\mathbf{u}'_{11} & \mathbf{u}'_{13} \\ v_\mathrm{h}\mathbf{u}'_{11} & v_\mathrm{h}\mathbf{u}'_{11} + \rho_\mathrm{h}^2 & \mathbf{u}'_{23} \\ \mathbf{u}'_{13} & \mathbf{u}'_{23} & \mathbf{u}'_{33} \end{pmatrix}, \tag{4.37}$$

  with

$$\mathbf{u}'_{11} := \frac{(T_\mathrm{h})_u {\rho_\mathrm{h}}^3}{(p_\mathrm{h})_\rho (T_\mathrm{h})_u \rho_\mathrm{h} - (T_\mathrm{h})_\rho \left((T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right)},$$

$$\mathbf{u}'_{13} := \frac{{\rho_\mathrm{h}}^3 \left(2(T_\mathrm{h})_\rho \rho_\mathrm{h} - (T_\mathrm{h})_u \left({v_\mathrm{h}}^2 + 2u_\mathrm{h}\right)\right)}{2(T_\mathrm{h})_\rho \left((T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right) - 2(p_\mathrm{h})_\rho (T_\mathrm{h})_u \rho_\mathrm{h}},$$

$$
\begin{aligned}
\mathbf{u}'_{23} :=& \frac{v_\mathrm{h}{\rho_\mathrm{h}}^2 \left(2(T_\mathrm{h})_\rho^2 {\rho_\mathrm{h}}^2 - (T_\mathrm{h})_u \left({v_\mathrm{h}}^2 + 2((p_\mathrm{h})_\rho + u_\mathrm{h})\right)\rho_\mathrm{h} + 2(T_\mathrm{h})_\rho \left({\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right)\right)}{2(T_\mathrm{h})_\rho \left((T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right) - 2(p_\mathrm{h})_\rho (T_\mathrm{h})_u \rho_\mathrm{h}}, \\
\mathbf{u}'_{33} :=& \frac{4(T_\mathrm{h})_\rho^2 {v_\mathrm{h}}^2 {\rho_\mathrm{h}}^4 - \left((T_\mathrm{h})_u \left({v_\mathrm{h}}^2 + 2u_\mathrm{h}\right)^2 + 4(p_\mathrm{h})_\rho \left((T_\mathrm{h})_u {v_\mathrm{h}}^2 + \rho_\mathrm{h}\right)\right)\rho_\mathrm{h}^2}{4(T_\mathrm{h})_\rho \left((T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right) - 4(p_\mathrm{h})_\rho (T_\mathrm{h})_u \rho_\mathrm{h}} \\
&+ \frac{+4(T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 \left(\left({v_\mathrm{h}}^2 + 2u_\mathrm{h}\right)\rho_\mathrm{h}^2 + p_\mathrm{h}\left((T_\mathrm{h})_u {v_\mathrm{h}}^2 + \rho_\mathrm{h}\right)\right)}{4(T_\mathrm{h})_\rho \left((T_\mathrm{h})_\rho {\rho_\mathrm{h}}^2 + (T_\mathrm{h})_u p_\mathrm{h}\right) - 4(p_\mathrm{h})_\rho (T_\mathrm{h})_u \rho_\mathrm{h}}.
\end{aligned} \tag{4.38}
$$

- **Two-velocity two-pressure model**
  In Section 2.5.1 it was proven that there exists an entropy pair (η, ψ) with convex entropy η which is not necessarily strictly convex.. Thus, η'' might be not strictly positive definite and might be locally singular such that it might not be possible to invert η'' as we did above for the homogeneous equilibrium model.

  As ad hoc fix we propose to compute this matrix numerically by finite differences,

$$
\mathbf{u}'(\mathbf{v}) := \partial_\mathbf{v}\mathbf{u} \approx \begin{pmatrix} \frac{\mathbf{u}_1(\mathbf{v}+\varepsilon_1)-\mathbf{u}_1(\mathbf{v}-\varepsilon_1)}{2\varepsilon} & \frac{\mathbf{u}_1(\mathbf{v}+\varepsilon_2)-\mathbf{u}_1(\mathbf{v}-\varepsilon_2)}{2\varepsilon} & \cdots & \frac{\mathbf{u}_1(\mathbf{v}+\varepsilon_7)-\mathbf{u}_1(\mathbf{v}-\varepsilon_7)}{2\varepsilon} \\ \vdots & \vdots & & \vdots \\ \frac{\mathbf{u}_7(\mathbf{v}+\varepsilon_1)-\mathbf{u}_7(\mathbf{v}-\varepsilon_1)}{2\varepsilon} & \frac{\mathbf{u}_7(\mathbf{v}+\varepsilon_2)-\mathbf{u}_7(\mathbf{v}-\varepsilon_2)}{2\varepsilon} & \cdots & \frac{\mathbf{u}_7(\mathbf{v}+\varepsilon_7)-\mathbf{u}_7(\mathbf{v}-\varepsilon_7)}{2\varepsilon} \end{pmatrix}, \tag{4.39}
$$

  with zero-vectors $\varepsilon_{j=1,\dots,7} \in \mathbb{R}^7$, which just have an $\varepsilon > 0$ value at the j-th position. But here we need to be carefully with the discretization ε_j, because for $\varepsilon_j \to 0$ the approximate matrix must become singular as well.

  The influence on the solution of this ad hoc fix must be investigated and a reasonable choice for the discretization step size ε_j needs to be found. In Section 7.2 the solution for Sod's shock tube problem for different finite differences approximations (4.39) are considered. Because this problem bases on the homogeneous equilibrium model, the solutions can be compared with the results using the exact Hessian (4.37). It can be seen, that with a value of $\varepsilon_j = 0.1$ or lower the approximated solution is similar to the solution with exact entropy viscosity, see Figure 47 and Table 2.

4.3. Entropy stable path-consistent (ESPC) schemes

With the above postulated condition of an entropy identity (4.22), it is ensured that entropy is never dissipated. But this is just what we expect in the case for smooth solutions. Especially for the case of shocks, we would assume that entropy is dissipated. Therefore, we can conclude that an ECPC scheme just covers the smooth solutions, where no shocks appear. Fjordholm and Mishra [115] observed that for ECPC schemes large oscillations appear behind a shock. Therefore we need a stronger property for our scheme to model the dissipation at shocks.

4.3.1. Regularization

Castro et al. [108] considered the regularized equation,

$$\partial_t \mathbf{u} + A(\mathbf{u})\,\partial_x \mathbf{u} = \varepsilon_{\mathrm{R}}\,\partial_x\big(\mathcal{R}(\mathbf{u})\,\partial_x \mathbf{u}\big), \tag{4.40}$$

with numerical viscosity coefficient $\varepsilon_{\mathrm{R}} > 0$ and viscosity matrix $\mathcal{R}(\mathbf{u})$. This equation extends the quasi-linear system (3.7) by adding numerical diffusion on the right-hand side. The numerical viscosity $\mathcal{R}(\mathbf{u})$ is responsible for the observed lack of convergence to the physically relevant solutions in the ECPC scheme.

A path-consistent scheme is said to be *entropy stable* if the fluctuations are satisfying the following condition:

$$\mathbf{D}^{\pm}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) = \mathbf{D}^{\pm}_{\mathrm{EC}}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) \pm \frac{\varepsilon_{\mathrm{R}}}{\Delta x}\widehat{\mathcal{R}}(\mathbf{v}_{\mathrm{L}}, \mathbf{v}_{\mathrm{R}})\,(\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}). \tag{4.41}$$

$\mathbf{D}^{\pm}_{\mathrm{EC}}$ are the fluctuations of the ECPC or ECPSc scheme and $\widehat{\mathcal{R}}(\mathbf{v}_{\mathrm{L}}, \mathbf{v}_{\mathrm{R}})$ is the numerical viscosity operator, which must be a symmetric positive definite matrix. This property is used below in (4.47) to show the entropy stability of the scheme. Fjordholm and Mishra [115] stated that the limit of viscous regularizations for the non-conservative system (3.7) is sensitive to the choice of the diffusion operator $\widehat{\mathcal{R}}$. Note, that for conservative systems this is well known. To obtain convergence to a correct solution it is essential to choose a suitable numerical diffusion operator that matches the underlying physical viscosity.

▶ Entropy viscosity

One possibility is to choose the *entropy viscosity*, which is given by the derivative of the inverse mapping of the entropy variables,

$$\widehat{\mathcal{R}}(\mathbf{v}_{\mathrm{L}}, \mathbf{v}_{\mathrm{R}})\,(\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}) := \mathbf{u}'(\mathbf{v}_{\mathrm{L}}, \mathbf{v}_{\mathrm{R}})\,(\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}). \tag{4.42}$$

The needed derivatives are given in (4.37) and (4.39).

▶ Uniform viscosity

Besides, Fjordholm and Mishra [115] propose to use *uniform viscosity*, by using

$$\widehat{\mathcal{R}}(\mathbf{v}_{\mathrm{L}}, \mathbf{v}_{\mathrm{R}})\,(\mathbf{v}_{\mathrm{R}} - \mathbf{v}_{\mathrm{L}}) := (\mathbf{u}_{\mathrm{R}} - \mathbf{u}_{\mathrm{L}}). \tag{4.43}$$

▶ Navier-Stokes viscosity

Additionally, Fjordholm and Mishra propose to use the *Navier-Stokes viscosity*. But Guermond and Popov [116] encountered two difficulties: Because there is no regularization in the continuity equation the viscosity operator might be inconsistent with most numerical discretizations. And additionally the Navier-Stokes equations violate the minimum entropy principle if the thermal diffusivity is nonzero, see e.g., Serre [117], Theorem 8.2.3.

With the above proposed choice it is ensured that $\widehat{\mathcal{R}}$ is symmetric positive definite, because it can be derived from the Hessian $\eta''(\mathbf{u})$, compare its derivation in (4.36). The diffusion operator is evaluated at the cell interface. Because we assume a linear path along the entropy variables, we use the inverse mapping at the center of the mean entropy variables, $\frac{1}{2}(\mathbf{v}_\mathrm{L} + \mathbf{v}_\mathrm{R})$.

4.3.2. Path consistency

It can easily be shown that the path consistency of the extended scheme follows from the path consistency of the ECPC or ECPCs scheme:

$$\begin{aligned} \mathbf{D}^{\pm}(\mathbf{u},\mathbf{u}) &= \mathbf{D}^{\pm}_{\mathrm{EC}}(\mathbf{u},\mathbf{u}) = 0, \\ \mathbf{D}^{-}(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) + \mathbf{D}^{+}(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) &= \mathbf{D}^{-}_{\mathrm{EC}}(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}) + \mathbf{D}^{+}_{\mathrm{EC}}(\mathbf{u}_\mathrm{L},\mathbf{u}_\mathrm{R}), \end{aligned} \tag{4.44}$$

see (4.10) and (4.11).

4.3.3. Entropy stability

We need to show that the ESPC or ESPCs scheme satisfies the discrete entropy inequality,

$$\partial_t \eta(\mathbf{u}_i) + \frac{1}{\Delta x}\left(\widehat{\psi}_{i+1/2} - \widehat{\psi}_{i-1/2}\right) \leq 0, \tag{4.45}$$

with numerical entropy flux

$$\begin{aligned} \widehat{\psi}_{i+1/2} := \widehat{\psi}(\mathbf{u}_i,\mathbf{u}_{i+1}) \quad &= \quad \psi(\mathbf{u}_i) + \mathbf{v}_i^\mathsf{T}\,\mathbf{D}^{-}_{\mathrm{EC}}(\mathbf{u}_i,\mathbf{u}_{i+1}) - \frac{\varepsilon_\mathrm{R}}{2\Delta x}(\mathbf{v}_{i+1}+\mathbf{v}_i)^\mathsf{T}\,\widehat{\mathcal{R}}\,(\mathbf{v}_{i+1}-\mathbf{v}_i) \\ &\overset{(4.23)}{=} \tilde{\psi}_{i+1/2} - \frac{\varepsilon_\mathrm{R}}{2\Delta x}(\mathbf{v}_{i+1}+\mathbf{v}_i)^\mathsf{T}\,\widehat{\mathcal{R}}\,(\mathbf{v}_{i+1}-\mathbf{v}_i). \end{aligned} \tag{4.46}$$

We notice that with (4.44) the numerical entropy flux $\widehat{\psi}$ is consistent with the entropy flux ψ, meaning $\widehat{\psi}(\mathbf{u},\mathbf{u}) \equiv \psi(\mathbf{u})$. To show the entropy inequality, we start from our scheme in fluctuation form (4.8):

$$\begin{aligned} \partial_t \mathbf{u}_i \; &+ \frac{1}{\Delta x}\left(\mathbf{D}^{-}_{i+1/2} + \mathbf{D}^{+}_{i-1/2}\right) = 0 \\ \overset{(4.41)}{\Leftrightarrow} \quad \mathbf{v}_i^\mathsf{T}\,\partial_t \mathbf{u}_i \; &+ \frac{1}{\Delta x}\left(\mathbf{v}_i^\mathsf{T}\,\mathbf{D}^{-}_{\mathrm{EC},\,i+1/2} + \mathbf{v}_i^\mathsf{T}\,\mathbf{D}^{+}_{\mathrm{EC},\,i-1/2}\right) = \frac{\varepsilon_\mathrm{R}}{\Delta x^2}\,\mathbf{v}_i^\mathsf{T}\,\widehat{\mathcal{R}}\,(\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}) \end{aligned}$$

$$
\begin{aligned}
\Leftrightarrow \quad & \partial_t \eta(\mathbf{u}_i) \; + \frac{1}{\Delta x}\left(\mathbf{v}_i^\mathsf{T}\, \mathbf{D}^-_{\mathrm{EC},\, i+1/2} + \psi(\mathbf{u}_i) - \psi(\mathbf{u}_i) + \mathbf{v}_i^\mathsf{T}\, \mathbf{D}^+_{\mathrm{EC},\, i-1/2}\right) \\
& \qquad = \frac{\varepsilon_\mathrm{R}}{\Delta x^2}\, \mathbf{v}_i^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}\right) \\
\overset{(4.23)}{\underset{(4.24)}{\Leftrightarrow}} \quad & \partial_t \eta(\mathbf{u}_i) \; + \frac{1}{\Delta x}\left(\tilde{\psi}_{i+1/2} - \tilde{\psi}_{i-1/2}\right) = \frac{\varepsilon_\mathrm{R}}{\Delta x^2}\, \mathbf{v}_i^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}\right) \\
\overset{(4.46)}{\Leftrightarrow} \quad & \partial_t \eta(\mathbf{u}_i) \; + \frac{1}{\Delta x}\left(\widehat{\psi}_{i+1/2} - \widehat{\psi}_{i-1/2}\right) = \frac{\varepsilon_\mathrm{R}}{\Delta x^2}\, \mathbf{v}_i^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}\right) \\
& \qquad - \frac{\varepsilon_\mathrm{R}}{2\Delta x^2}(\mathbf{v}_{i+1} + \mathbf{v}_i)^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_{i+1} - \mathbf{v}_i\right) + \frac{\varepsilon_\mathrm{R}}{2\Delta x^2}(\mathbf{v}_i + \mathbf{v}_{i-1})^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_i - \mathbf{v}_{i-1}\right) \\
\Leftrightarrow \quad & \partial_t \eta(\mathbf{u}_i) \; + \frac{1}{\Delta x}\left(\widehat{\psi}_{i+1/2} - \widehat{\psi}_{i-1/2}\right) = \\
& \qquad - \frac{\varepsilon_\mathrm{R}}{2\Delta x^2}(\mathbf{v}_{i+1} - \mathbf{v}_i)^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_{i+1} - \mathbf{v}_i\right) - \frac{\varepsilon_\mathrm{R}}{2\Delta x^2}(\mathbf{v}_i - \mathbf{v}_{i-1})^\mathsf{T}\, \widehat{\mathcal{R}}\left(\mathbf{v}_i - \mathbf{v}_{i-1}\right)
\end{aligned}
\tag{4.47}
$$

4.3.4. CFL condition

Because $\widehat{\mathcal{R}}$ is assumed as symmetric positive definite the entropy inequality (4.45) holds. The additional numerical viscosity changes the standard CFL condition:

$$
\frac{\Delta t}{\Delta x}\left(|\lambda_{\max}| + \frac{2\varepsilon_\mathrm{R}}{\Delta x}\right) \leq 1, \tag{4.48}
$$

with $\lambda_{\max}$ as the maximal absolute eigenvalue over all cells at some time t. The numerical viscosity coefficient is chosen as

$$
\varepsilon_\mathrm{R} = \frac{|\lambda_{\max}|}{2}\Delta x, \tag{4.49}
$$

such that ε_R vanishes as the mesh is refined. This agrees with the choice of Fjordholm and Mishra [115].

5. Relaxation approach

In Section 4 we presented path-conservative schemes to solve the equations of the two-phase flow model. The motivation for this Section is to find Riemann solutions of the models, such that the Godunov solver introduced in (4.16) and (4.17) can be used. Because the pressure is non-linear, the conventional model provides genuinely nonlinear waves, such that the exact Riemann invariants cannot be found. Here we follow the relaxation approach: The original two-phase flow model is extended, such that the enlarged model is linearly degenerate. Thus, a Riemann solution is easier to find. So, the general idea is to approximate the exact solution of the original two-phase flow model by the Riemann solution of the relaxation system.

Suliciu [118, 9] proposed to modify only the pressure law by making the pressure a new dependent variable. The advantage is that a quasilinear system of first order is obtained which is consistent with the original system. Additionally, the relaxation system is consistent with the entropy inequality in the regime of a relaxation time of zero [119]. See also Liu [120] and Chen, Levermore and Liu [121].

In general, a Suliciu relaxation scheme is obtained by adding new balance laws which depend on a constant relaxation parameter such that all eigenvalues of the relaxation model are associated with a linearly degenerate field. Thus, the system of conservation laws in non-conservative form is given by

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = \mathbf{s}_{\mathrm{r}}(\mathbf{u}) + \mathbf{s}(\mathbf{u}). \tag{5.1}$$

The full problem is solved using the two-step splitting strategy of Jin and Xin [122]: At first the *evolution step* (which corresponds to an infinite relaxation time) accounts for all convective effects, where the second *relaxation* step (which corresponds to a relaxation time of zero) takes the relaxation source terms into account. In the following, both steps are explained in more detail:

▶ Evolution in time ($t^n \to t^{n+1,-}$)

The homogeneous subproblem describing all convective effects is given by

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) + B(\mathbf{u})\partial_x \mathbf{u} = 0. \tag{5.2}$$

Because all eigenvalues of the model are associated with a linearly degenerate field, at each cell interface a sequence of non-interacting Riemann problems occur. Therefore we regard for the homogeneous quasilinear system (5.2) simply the Riemann problem satisfying the initial condition

$$\mathbf{u}_0(x) = \begin{cases} \mathbf{u}_{\mathrm{L}}, & x < 0, \\ \mathbf{u}_{\mathrm{R}}, & x > 0, \end{cases} \tag{5.3}$$

with two valid states $\mathbf{u}_{\mathrm{L}}$ and $\mathbf{u}_{\mathrm{R}}$ in Ω. A solution of the Riemann problem is a self-similar mapping $\mathbf{u}(x,t) = \mathbf{u}_{\mathrm{r}}(\frac{x}{t}; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})$, which is made of constant intermediate states

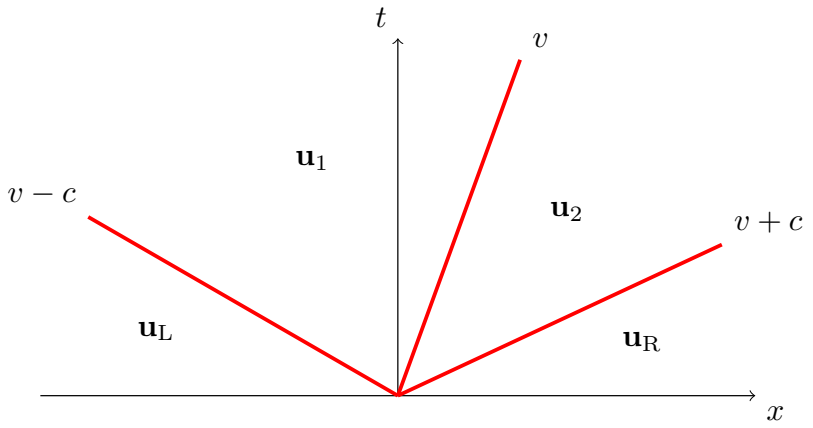


Figure 37: Sketch of the solution of the Riemann problem.

separated by waves whose constant speeds are distinct eigenvalues of the system [123], a, see Figure 37.
So, for $\tilde{n}$ distinct eigenvalues $\lambda_{j=1,\dots\tilde{n}}$, the solution consists of the left state $\mathbf{u}_\mathrm{L}$, $\tilde{n}-1$ unknown intermediate states $\mathbf{u}_{j=1,\dots\tilde{n}-1}$, and the right state $\mathbf{u}_\mathrm{R}$, ordered in ascending order in space. Two neighboring states $\mathbf{u}_{j-1}$ and $\mathbf{u}_j$ are separated by a contact discontinuity which propagates with the corresponding characteristic speed λ_j, where $\lambda_j(\mathbf{u}_{j-1}) = \lambda_j(\mathbf{u}_j)$. Because the corresponding j-Riemann invariants $\mathbf{w}^{(j)}$ are constant across the $\tilde{n}$ contact discontinuities they are equal for both neighboring states, $\mathbf{w}^{(j)}(\mathbf{u}_{j-1}) \overset{!}{=} \mathbf{w}^{(j)}(\mathbf{u}_j)$. This constraint is used to find the unknown intermediate states.

Usually the natural order of the waves is known, such that the waves can be sorted in ascending order, with speed $\lambda_j < \lambda_{j+1}$. The constant relaxation parameter should be chosen large enough, such that this natural order of the waves also holds for the intermediate states, i.e.

$$\begin{aligned} \lambda_1(\mathbf{u}_\mathrm{L}) = \lambda_1(\mathbf{u}_1) &< \lambda_2(\mathbf{u}_1) = \lambda_2(\mathbf{u}_2) \\ &< \dots \\ &< \lambda_{\tilde{n}-1}(\mathbf{u}_{\tilde{n}-2}) = \lambda_{\tilde{n}-1}(\mathbf{u}_{\tilde{n}-1}) \\ &< \lambda_{\tilde{n}}(\mathbf{u}_{\tilde{n}-1}) = \lambda_{\tilde{n}}(\mathbf{u}_\mathrm{R}). \end{aligned} \tag{5.4}$$

The solution of the Riemann problem is then given by

$$\mathbf{u}(x,t) = \mathbf{u}_{\mathrm{r}}(\tfrac{x}{t}; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) = \begin{cases} \mathbf{u}_{\mathrm{L}}, & \tfrac{x}{t} < \lambda_1(\mathbf{u}_{\mathrm{L}}) \\ \mathbf{u}_1, & \lambda_1(\mathbf{u}_1) < \tfrac{x}{t} < \lambda_2(\mathbf{u}_1) \\ \mathbf{u}_2, & \lambda_2(\mathbf{u}_2) < \tfrac{x}{t} < \lambda_3(\mathbf{u}_2) \\ \vdots & \vdots \\ \mathbf{u}_{\tilde{n}-1}, & \lambda_{\tilde{n}-1}(\mathbf{u}_{\tilde{n}-1}) < \tfrac{x}{t} < \lambda_{\tilde{n}}(\mathbf{u}_{\tilde{n}-1}) \\ \mathbf{u}_{\mathrm{R}}, & \lambda_{\tilde{n}}(\mathbf{u}_{\mathrm{R}}) < \tfrac{x}{t} \end{cases} . \tag{5.5}$$

The intermediate states $\mathbf{u}_{j=1,\dots\tilde{n}-1}$ are obtained using the Rankine-Hugoniot jump relations across the $\tilde{n}$ contact discontinuities [119].

So for given initial values $\mathbf{u}_i^n$ this step computes approximate solutions $\mathbf{u}_i^{n+1,-}$ through the time interval $[t^n, t^n + \Delta t]$. As numerical solver the Godunov method can be used, see Section 3. This method is called *Godunov-Suliciu* solver. We need to ensure that the local solutions of neighboring cells do not interact during the time period Δt. This gives us the classical CFL condition

$$\frac{\Delta t}{\Delta x} \max_i |\lambda_j(\mathbf{u}_i^n)| < \frac{1}{2}, \quad j \in \{0, \dots, \tilde{n}\}. \tag{5.6}$$

Observe that under this CFL condition the constant relaxation parameter can be chosen locally at each cell [123].

▶ Source terms at instantaneous relaxation ($t^{n+1,-} \to t^{n+1}$)

In the second step the relaxation source terms are considered for an infinite relaxation time,

$$\partial_t \mathbf{u} = \mathbf{s}_{\mathrm{r}}(\mathbf{u}), \tag{5.7}$$

with $\mathbf{u}_i^{n+1,-}$ as initial data. This step corresponds to a projection on the equilibrium manifold. The ordinary differential equation system can be solved exactly with a full time step of length Δt which finally provides the solution $\mathbf{u}_i^{n+1}$.

In the following, relaxation models for the *homogeneous equilibrium model* and the *two-velocity two-pressure model* are derived, see Sections 5.1 and 5.3. For better readability, the same notation as introduced in 2.3 is used, despite that the system dimension n is now 4 or 9 instead of 7. In Section 5.2 and 5.4 a numerical solution of the relaxation model is derived.

5.1. Relaxation homogeneous equilibrium four-equations model

For the relaxation of the homogenous model, a pressure balance law is needed. It can be derived from the pressure relation (2.55), multiplying it ρ_k times, and adding p_k times the mass balance equation, such that we get

$$\partial_t(\rho_\mathrm{h} p_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} p_\mathrm{h}) + \rho_\mathrm{h}^2 c_\mathrm{h}^2 \, \partial_x v_\mathrm{h} = \rho_\mathrm{h} \, \mathbf{s}_{p_\mathrm{h}}, \tag{5.8}$$

with the source term

$$\mathbf{s}_{p_\mathrm{h}} = -\frac{v_\mathrm{h}(p_k)_u}{\rho_\mathrm{h}} \, \mathbf{s}_{\rho_\mathrm{h} v_\mathrm{h}} + \frac{(p_k)_u}{\rho_\mathrm{h}} \, \mathbf{s}_{\rho_\mathrm{h} E_\mathrm{h}}. \tag{5.9}$$

Suliciu introduced a new parameter π_h which is defined by a Chapman-Enskog expansion of first order,

$$\pi_\mathrm{h} := \underbrace{\pi_\mathrm{h}^{(0)}}_{:=p_\mathrm{h}} + \varepsilon_\mathrm{h} \pi_\mathrm{h}^{(1)}. \tag{5.10}$$

The zeroth-order approximation is taken to be the pressure such that for the relaxation time $\varepsilon_\mathrm{h} \to 0$ it holds $\pi_\mathrm{h} \to p_\mathrm{h}$. The system is at equilibrium whenever $\pi_\mathrm{h} = p_\mathrm{h}$ [124].

With this expansion, the above approximate pressure balance law (5.8) will be relaxated by replacing the pressure p_h by π_h. The pre-factor of the velocity derivative, $\rho_\mathrm{h}^2 c_\mathrm{h}^2$, is chosen to be locally (in time and space) constant, thus it is replaced by a constant parameter a_h^2. Additionally, the source terms are neglected and instead a procedure of relaxation to equilibrium is added at the right hand side. So finally, the relaxation balance law is given by

$$\partial_t(\rho_\mathrm{h} \pi_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} \pi_\mathrm{h}) + a_\mathrm{h}^2 \, \partial_x v_\mathrm{h} = \rho_\mathrm{h} \underbrace{\frac{1}{\varepsilon_\mathrm{h}}(p_\mathrm{h} - \pi_\mathrm{h})}_{=-\pi_\mathrm{h}^{(1)}}. \tag{5.11}$$

Now, we can give an expression of the first-order corrector of π_h by studying the asymptotic behavior of the relaxation pressure balance law for $\varepsilon_\mathrm{h} \to 0$. It holds:

$$\begin{aligned}
&\partial_t(\rho_\mathrm{h} \pi_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} \pi_\mathrm{h}) + a_\mathrm{h}^2 \, \partial_x v_\mathrm{h} + \rho_\mathrm{h} \pi_\mathrm{h}^{(1)} \\
&\quad = \partial_t \left(\rho_\mathrm{h} (p_\mathrm{h} + \varepsilon_\mathrm{h} \pi_\mathrm{h}^{(1)}) \right) + \partial_x \left(\rho_\mathrm{h} v_\mathrm{h} (p_\mathrm{h} + \varepsilon_\mathrm{h} \pi_\mathrm{h}^{(1)}) \right) + a_\mathrm{h}^2 \, \partial_x v_\mathrm{h} + \rho_\mathrm{h} \pi_\mathrm{h}^{(1)} \\
&\quad = \partial_t(\rho_\mathrm{h} p_\mathrm{h}) + \underbrace{\varepsilon_\mathrm{h} \, \partial_t(\rho_\mathrm{h} \pi_\mathrm{h}^{(1)})}_{\to 0} + \partial_x(\rho_\mathrm{h} v_\mathrm{h} p_\mathrm{h}) + \underbrace{\varepsilon_\mathrm{h} \, \partial_x(\rho_\mathrm{h} v_\mathrm{h} \pi_\mathrm{h}^{(1)})}_{\to 0} + a_\mathrm{h}^2 \, \partial_x v_\mathrm{h} + \rho_\mathrm{h} \pi_\mathrm{h}^{(1)} \\
&\quad = \partial_t(\rho_\mathrm{h} p_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} p_\mathrm{h}) + a_\mathrm{h}^2 \, \partial_x v_\mathrm{h} + \rho_\mathrm{h} \pi_\mathrm{h}^{(1)} \\
&\quad \overset{!}{=} \partial_t(\rho_\mathrm{h} p_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} p_\mathrm{h}) + \rho_\mathrm{h}^2 c_\mathrm{h}^2 \, \partial_x v_\mathrm{h},
\end{aligned} \tag{5.12}$$

such that we get

$$-\pi_\mathrm{h}^{(1)} = \frac{a_\mathrm{h}^2 - \rho_\mathrm{h}^2 c_\mathrm{h}^2}{\rho_\mathrm{h}} \, \partial_x v_\mathrm{h}. \tag{5.13}$$

The constant parameter a_h is chosen below in Section 5.1.4, such that the subcharacteristic condition holds.

Now, the *relaxation homogeneous equilibrium model* is obtained by replacing all pressure terms p_h in the mass, momentum and energy balance laws by π_h. To express the system in the general conservative form,

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = \mathbf{s}_\mathrm{r}(\mathbf{u}) + \mathbf{s}(\mathbf{u}),$$

the independent variables $\mathbf{u}$, the flux vector $\mathbf{f}(\mathbf{u})$ and the relaxation source vector $\mathbf{s}_\mathrm{r}(\mathbf{u})$ are given by

$$\mathbf{u} := \begin{pmatrix} \rho_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h} \\ \rho_\mathrm{h} E_\mathrm{h} \\ \rho_\mathrm{h} \pi_\mathrm{h} \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} \rho_\mathrm{h} v_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h}^2 + \pi_\mathrm{h} \\ (\rho_\mathrm{h} E_\mathrm{h} + \pi_\mathrm{h}) v_\mathrm{h} \\ \rho_\mathrm{h} v_\mathrm{h} \pi_\mathrm{h} + a_\mathrm{h}^2 v_\mathrm{h} \end{pmatrix}, \quad \mathbf{s}_\mathrm{r}(\mathbf{u}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \rho_\mathrm{h} \frac{1}{\varepsilon_\mathrm{h}} (p_\mathrm{h} - \pi_\mathrm{h}) \end{pmatrix}, \tag{5.14}$$

where the source vector $\mathbf{s}(\mathbf{u})$ is given by

$$\mathbf{s}(\mathbf{u}) = \begin{pmatrix} \mathbf{s}_{\rho_\mathrm{h}} \\ \mathbf{s}_{\rho_\mathrm{h} v_\mathrm{h}} \\ \mathbf{s}_{\rho_\mathrm{h} E_\mathrm{h}} \\ \mathbf{s}_{\rho_\mathrm{h} \pi_\mathrm{h}} \end{pmatrix} = \begin{pmatrix} 0 \\ \rho_\mathrm{h}\, g\, \sin(\vartheta) + F_{\mathrm{w}\,\ell} + F_{\mathrm{w}\,g} \\ \rho_\mathrm{h} v_\mathrm{h}\, g\, \sin(\vartheta) + (F_{\mathrm{w}\,\ell} + F_{\mathrm{w}\,g}) v_\mathrm{h} + Q_{\mathrm{w}\,\ell} + Q_{\mathrm{w}\,g} \\ 0 \end{pmatrix}. \tag{5.15}$$

All solutions for $\mathbf{u}$ are in the set of admissible states,

$$\Omega = \{\mathbf{u} \in \mathbb{R}^n \mid \rho_\mathrm{h} > 0, u_\mathrm{h} > 0, c_\mathrm{h} > 0\} \tag{5.16}$$

with $n = 3$, where u_h is the specific internal energy, given by the physical law

$$u_\mathrm{h} = E_\mathrm{h} - \tfrac{1}{2} v_\mathrm{h}^2. \tag{5.17}$$

To transform the system in closed quasilinear form, we need to differentiate the flux vector $\mathbf{f}(\mathbf{u})$ with respect to our independent variables $\mathbf{u}$, such that the system matrix $A(\mathbf{u})$ is given by:

$$A(\mathbf{u}) = \partial_{\mathbf{u}} \mathbf{f}(\mathbf{u}) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\frac{\pi_\mathrm{h} + \rho_\mathrm{h} v_\mathrm{h}^2}{\rho_\mathrm{h}} & 2 v_\mathrm{h} & 0 & \frac{1}{\rho_\mathrm{h}} \\ -\frac{(2\pi_\mathrm{h} + E_\mathrm{h} \rho_\mathrm{h}) v_\mathrm{h}}{\rho_\mathrm{h}} & E_\mathrm{h} + \frac{\pi_\mathrm{h}}{\rho_\mathrm{h}} & v_\mathrm{h} & \frac{v_\mathrm{h}}{\rho_\mathrm{h}} \\ -\pi_\mathrm{h} v_\mathrm{h} - \frac{a_\mathrm{h}^2 v_\mathrm{h}}{\rho_\mathrm{h}} & \pi_\mathrm{h} + \frac{a_\mathrm{h}^2}{\rho_\mathrm{h}} & 0 & v_\mathrm{h} \end{pmatrix}. \tag{5.18}$$

By construction of the Suliciu relaxation, the relaxation model should have the same thermodynamical and mathematical properties as the original two-phase flow model (see Section 2.5).

5.1.1. Second law of thermodynamics

For the *relaxation homogeneous equilibrium model* the same entropy-entropy flux pair is chosen as for the *homogeneous equilibrium model* (2.151):

$$\eta(\mathbf{u}) = -\rho_\mathrm{h} s_\mathrm{h} \quad \text{and} \quad \psi(\mathbf{u}) = -\rho_\mathrm{h} v_\mathrm{h} s_\mathrm{h}, \tag{5.19}$$

such that the following entropy variables arise

$$\eta'(\mathbf{u}) := \mathbf{v}(\mathbf{u}) = \begin{pmatrix} \frac{\frac{p_\mathrm{h}}{\rho_\mathrm{h}} + u_\mathrm{h} - \frac{1}{2}v_\mathrm{h}^2}{T_\mathrm{h}} - s_\mathrm{h} \\ \frac{v_\mathrm{h}}{T_\mathrm{h}} \\ -\frac{1}{T_\mathrm{h}} \\ 0 \end{pmatrix} \overset{(2.88)}{=} \begin{pmatrix} \frac{g_\mathrm{h} - \frac{1}{2}v_\mathrm{h}^2}{T_\mathrm{h}} \\ \frac{v_\mathrm{h}}{T_\mathrm{h}} \\ -\frac{1}{T_\mathrm{h}} \\ 0 \end{pmatrix}, \tag{5.20}$$

The entropy compatibility condition in (2.85) is fulfilled:

$$\partial_\mathbf{u}\psi(\mathbf{u})^\mathsf{T} = \begin{pmatrix} v_\mathrm{h} \cdot \frac{\frac{p_\mathrm{h}}{\rho_\mathrm{h}} + u_\mathrm{h} - \frac{1}{2}v_\mathrm{h}^2}{T_\mathrm{h}} \\ \frac{v_\mathrm{h}^2}{T_\mathrm{h}} - s_\mathrm{h} \\ -\frac{v_\mathrm{h}}{T_\mathrm{h}} \\ 0 \end{pmatrix} = \partial_\mathbf{u}\eta(\mathbf{u})^\mathsf{T} \cdot A(\mathbf{u}). \tag{5.21}$$

and the entropy production stays zero. The Hessian of the entropy function is given by

$$\eta''(\mathbf{u}) = \begin{pmatrix} \frac{T_\mathrm{h}(p_\mathrm{h})_\rho - \left(p_\mathrm{h} + 2\rho_\mathrm{h}(E_\mathrm{h} - v_\mathrm{h}^2)\right)(T_\mathrm{h})_\rho + \left(E_\mathrm{h} - v_\mathrm{h}^2\right)^2 (T_\mathrm{h})_u + v_\mathrm{h}^2 T_\mathrm{h}}{\rho_\mathrm{h} T_\mathrm{h}^2} & -v_\mathrm{h}\left(\eta_{13} + \frac{T_\mathrm{h}}{\rho_\mathrm{h} T_\mathrm{h}^2}\right) & \eta_{13} & 0 \\ \frac{-\rho_\mathrm{h} v_\mathrm{h}(T_\mathrm{h})_\rho + v_\mathrm{h}(E_\mathrm{h} - v_\mathrm{h}^2)(T_\mathrm{h})_u - v_\mathrm{h} T_\mathrm{h}}{\rho_\mathrm{h} T_\mathrm{h}^2} & \frac{v_\mathrm{h}^2 (T_\mathrm{h})_u + T_\mathrm{h}}{\rho_\mathrm{h} T_\mathrm{h}^2} & \frac{-v_\mathrm{h}(T_\mathrm{h})_u}{\rho_\mathrm{h} T_\mathrm{h}^2} & 0 \\ \frac{\rho_\mathrm{h}(T_\mathrm{h})_\rho - (E_\mathrm{h} - v_\mathrm{h}^2)(T_\mathrm{h})_u}{\rho_\mathrm{h} T_\mathrm{h}^2} & \frac{-v_\mathrm{h}(T_\mathrm{h})_u}{\rho_\mathrm{h} T_\mathrm{h}^2} & \frac{(T_\mathrm{h})_u}{\rho_\mathrm{h} T_\mathrm{h}^2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{5.22}$$

with $\eta_{13} := \frac{\rho_\mathrm{h}(T_\mathrm{h})_\rho - (E_\mathrm{h} - v_\mathrm{h}^2)(T_\mathrm{h})_u}{\rho_\mathrm{h} T_\mathrm{h}^2}$. This Hessian is similar to the Hessian of the *homogeneous equilibrium model* (2.154), it has just one additional row and column with zeros. It is obvious that $\eta(\mathbf{u})$ is convex as it is known from the *homogeneous equilibrium model*, but $\eta(\mathbf{u})$ is not strictly convex. Therefore we need to show that the system is still hyperbolic.

5.1.2. Hyperbolicity

In order to characterize mathematical properties, the equation systems are transformed in quasi-conservative form (2.114)

$$\partial_t \tilde{\mathbf{u}} + \tilde{A}(\tilde{\mathbf{u}})\partial_x \tilde{\mathbf{u}} = \tilde{\mathbf{s}}(\tilde{\mathbf{u}}),$$

with primitive system vector

$$\tilde{\mathbf{u}} := \begin{pmatrix} \rho_\mathrm{h} & v_\mathrm{h} & u_\mathrm{h} & p_\mathrm{h} \end{pmatrix}^\mathsf{T}. \tag{5.23}$$

and primitive system matrix

$$\tilde{A}(\tilde{\mathbf{u}}) = \begin{pmatrix} v_\mathrm{h} & \rho_\mathrm{h} & 0 & 0 \\ 0 & v_\mathrm{h} & 0 & \frac{1}{\rho_\mathrm{h}} \\ 0 & \frac{\pi_\mathrm{h}}{\rho_\mathrm{h}} & v_\mathrm{h} & 0 \\ 0 & \frac{a_\mathrm{h}^2}{\rho_\mathrm{h}} & 0 & v_\mathrm{h} \end{pmatrix}. \tag{5.24}$$

The eigenvalue matrix Λ and the right and left eigenvectors R and L are given by

$$\Lambda = \begin{pmatrix} v_\mathrm{h} - \frac{a_\mathrm{h}}{\rho_\mathrm{h}} & 0 & 0 & 0 \\ 0 & v_\mathrm{h} & 0 & 0 \\ 0 & 0 & v_\mathrm{h} & 0 \\ 0 & 0 & 0 & v_\mathrm{h} + \frac{a_\mathrm{h}}{\rho_\mathrm{h}} \end{pmatrix}, \tag{5.25}$$

$$R^\mathsf{T} = \begin{pmatrix} \frac{\rho_\mathrm{h}^2}{a_\mathrm{h}^2} & 1 & 0 & \frac{\rho_\mathrm{h}^2}{a_\mathrm{h}^2} \\ -\frac{1}{a_\mathrm{h}} & 0 & 0 & \frac{1}{a_\mathrm{h}} \\ \frac{p_\mathrm{h}}{a_\mathrm{h}^2} & 0 & 1 & \frac{p_\mathrm{h}}{a_\mathrm{h}^2} \\ 1 & 0 & 0 & 1 \end{pmatrix}, \tag{5.26}$$

and

$$L = \begin{pmatrix} 0 & -\frac{a_\mathrm{h}}{2} & 0 & \frac{1}{2} \\ 1 & 0 & 0 & -\frac{\rho_\mathrm{h}^2}{a_\mathrm{h}^2} \\ 0 & 0 & 1 & -\frac{p_\mathrm{h}}{a_\mathrm{h}^2} \\ 0 & \frac{a_\mathrm{h}}{2} & 0 & \frac{1}{2} \end{pmatrix}. \tag{5.27}$$

The system admits four real eigenvalues, v_h (double) and $v_\mathrm{h} \pm \frac{a_\mathrm{h}}{\rho_\mathrm{h}}$. Because the corresponding eigenvectors are linearly independent the quasilinear system is hyperbolic.

5.1.3. Characteristic fields

The motivation to construct a relaxation model was to obtain a linearly degenerate system, such that $\partial_{\tilde{\mathbf{u}}} \lambda_j \, R_j = 0$ for all $j = 1, \ldots, n$. The j-Riemann invariants $\mathbf{w}^{(j)}$ of the linearly degenerate j-characteristic fields are given by

$$\begin{aligned} \mathbf{w}^{(1)} &= \left\{ v_\mathrm{h} - \frac{a_\mathrm{h}}{\rho_\mathrm{h}}, p_\mathrm{h} + v_\mathrm{h} a_\mathrm{h}, -\tfrac{1}{2} p_\mathrm{h}^2 + u_\mathrm{h} a_\mathrm{h}^2 \right\}, \\ \mathbf{w}^{(2)} = \mathbf{w}^{(3)} &= \left\{ v_\mathrm{h}, p_\mathrm{h} \right\}, \\ \mathbf{w}^{(4)} &= \left\{ v_\mathrm{h} + \frac{a_\mathrm{h}}{\rho_\mathrm{h}}, p_\mathrm{h} - v_\mathrm{h} a_\mathrm{h}, -\tfrac{1}{2} p_\mathrm{h}^2 + u_\mathrm{h} a_\mathrm{h}^2 \right\}. \end{aligned} \tag{5.28}$$

5.1.4. Sub-characteristic condition

The subcharacteristic condition is fulfilled, if the eigenvalues of the *homogeneous equilibrium model* are interlaced with the eigenvalues of the *relaxation homogeneous equilibrium model*. It's obvious that with

$$c_{\mathrm{h}} \leq \frac{a_{\mathrm{h}}}{\rho_{\mathrm{h}}} \tag{5.29}$$

the sub-characteristic condition is fulfilled. This gives us a condition for the constant parameter a_{h}, such that we choose

$$a_{\mathrm{h}} \geq \rho_{\mathrm{h}} c_{\mathrm{h}}. \tag{5.30}$$

5.1.5. Symmetrisation of the hyperbolic system

A symmetric positiv definite matrix $P(\tilde{\mathbf{u}})$ is given by choosing

$$P(\tilde{\mathbf{u}}) := R^{-1}L = \begin{pmatrix} 1 & 0 & 0 & -\frac{\rho_{\mathrm{h}}^2}{a_{\mathrm{h}}^2} \\ 0 & \frac{a_{\mathrm{h}}^2}{2} & 0 & 0 \\ 0 & 0 & 1 & -\frac{p_{\mathrm{h}}}{a_{\mathrm{h}}^2} \\ -\frac{\rho_{\mathrm{h}}^2}{a_{\mathrm{h}}^2} & 0 & -\frac{p_{\mathrm{h}}}{a_{\mathrm{h}}^2} & \frac{1}{2} + \frac{p_{\mathrm{h}}^2 + \rho_{\mathrm{h}}^4}{a_{\mathrm{h}}^4} \end{pmatrix}, \tag{5.31}$$

With this choice also $P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})$ is symmetric.

5.2. Riemann solver for the relaxation homogeneous equilibrium model

As introduced at the beginning of this Section, the Riemann solution is provided in two steps: Starting with $\mathbf{u}_i^n$ as initial data, at first the convective part is solved, such that the solution $\mathbf{u}_i^{n+1,-}$ is provided. Then as second step, the projection on the equilibrium manifold is solved, such that the solution $\mathbf{u}_i^{n+1}$ at the next time step is provided.

▶ Evolution in time

The *relaxation homogeneous equilibrium model* has $\tilde{n} = 3$ distinct eigenvalues v_h and $v_\mathrm{h} \pm \frac{a_\mathrm{h}}{\rho_\mathrm{h}}$, where the term $\frac{a_\mathrm{h}}{\rho_\mathrm{h}}$ is dominating the magnitude. The natural order of the waves is given by

$$\lambda_1 := v_\mathrm{h} - \frac{a_\mathrm{h}}{\rho_\mathrm{h}} \quad < \quad \lambda_2 := v_\mathrm{h} \quad < \quad \lambda_3 := v_\mathrm{h} + \frac{a_\mathrm{h}}{\rho_\mathrm{h}}. \tag{5.32}$$

Using now the Rankine-Hugoniot jump relations across the discontinuities[16] (see Section 5.1.3) we get the equation system:

$$\begin{aligned}
\left(v_\mathrm{h} - \frac{a_\mathrm{h}}{\rho_\mathrm{h}}\right)_\mathrm{L} &\overset{!}{=} \left(v_\mathrm{h} - \frac{a_\mathrm{h}}{\rho_\mathrm{h}}\right)_1, \\
\left(\pi_\mathrm{h} + v_\mathrm{h} a_\mathrm{h}\right)_\mathrm{L} &\overset{!}{=} \left(\pi_\mathrm{h} + v_\mathrm{h} a_\mathrm{h}\right)_1, \\
\left(-\tfrac{1}{2}\left(\pi_\mathrm{h}\right)^2 + u_\mathrm{h} a_\mathrm{h}^2\right)_\mathrm{L} &\overset{!}{=} \left(-\tfrac{1}{2}\left(\pi_\mathrm{h}\right)^2 + u_\mathrm{h} a_\mathrm{h}^2\right)_1,
\end{aligned}$$

$$\begin{aligned}
\left(v_\mathrm{h}\right)_1 &\overset{!}{=} \left(v_\mathrm{h}\right)_2, \\
\left(\pi_\mathrm{h}\right)_1 &\overset{!}{=} \left(\pi_\mathrm{h}\right)_2,
\end{aligned}$$

$$\begin{aligned}
\left(v_\mathrm{h} + \frac{a_\mathrm{h}}{\rho_\mathrm{h}}\right)_2 &\overset{!}{=} \left(v_\mathrm{h} + \frac{a_\mathrm{h}}{\rho_\mathrm{h}}\right)_\mathrm{R}, \\
\left(\pi_\mathrm{h} - v_\mathrm{h} a_\mathrm{h}\right)_2 &\overset{!}{=} \left(\pi_\mathrm{h} - v_\mathrm{h} a_\mathrm{h}\right)_\mathrm{R}, \\
\left(-\tfrac{1}{2}\left(\pi_\mathrm{h}\right)^2 + u_\mathrm{h} a_\mathrm{h}^2\right)_2 &\overset{!}{=} \left(-\tfrac{1}{2}\left(\pi_\mathrm{h}\right)^2 + u_\mathrm{h} a_\mathrm{h}^2\right)_\mathrm{R}.
\end{aligned} \tag{5.33}$$

Solving for the parameters of the unknown intermediate states $\mathbf{u}_1$ and $\mathbf{u}_2$, a solution in dependency of the known left $\mathbf{u}_\mathrm{L}$ and right state $\mathbf{u}_\mathrm{R}$ is obtained:

$$\tilde{\mathbf{u}}_1 = \begin{pmatrix} \left(\rho_\mathrm{h}\right)_1 \\ \left(v_\mathrm{h}\right)_\star \\ \left(u_\mathrm{h}\right)_1 \\ \left(\pi_\mathrm{h}\right)_\star \end{pmatrix}, \quad \tilde{\mathbf{u}}_2 = \begin{pmatrix} \left(\rho_\mathrm{h}\right)_2 \\ \left(v_\mathrm{h}\right)_\star \\ \left(u_\mathrm{h}\right)_2 \\ \left(\pi_\mathrm{h}\right)_\star \end{pmatrix}, \tag{5.34}$$

[16] Notice that due to the in (5.32) introduced order of the waves, the Riemann invariants in Section 5.1.3 have a different numbering than used in this Section.

with

$$
\begin{aligned}
(v_\mathrm{h})_\star &:= \frac{(v_\mathrm{h})_\mathrm{R} + (v_\mathrm{h})_\mathrm{L}}{2} - \frac{(\pi_\mathrm{h})_\mathrm{R} - (\pi_\mathrm{h})_\mathrm{L}}{2a_\mathrm{h}},\\
(\pi_\mathrm{h})_\star &:= \frac{(\pi_\mathrm{h})_\mathrm{R} + (\pi_\mathrm{h})_\mathrm{L}}{2} - a_\mathrm{h}\frac{(v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}}{2},\\
(\rho_\mathrm{h})_1 &= \frac{a_\mathrm{h}\,(\rho_\mathrm{h})_\mathrm{L}}{a_\mathrm{h} + (\rho_\mathrm{h})_\mathrm{L}\left((v_\mathrm{h})_\star - (v_\mathrm{h})_\mathrm{L}\right)},\\
(\rho_\mathrm{h})_2 &= \frac{a_\mathrm{h}\,(\rho_\mathrm{h})_\mathrm{R}}{a_\mathrm{h} + (\rho_\mathrm{h})_\mathrm{R}\left((v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\star\right)},\\
(u_\mathrm{h})_1 &= (u_\mathrm{h})_\mathrm{L} + \frac{(\pi_\mathrm{h})_\star^2 - (\pi_\mathrm{h})_\mathrm{L}^2}{2a_\mathrm{h}^2},\\
(u_\mathrm{h})_2 &= (u_\mathrm{h})_\mathrm{R} + \frac{(\pi_\mathrm{h})_\star^2 - (\pi_\mathrm{h})_\mathrm{R}^2}{2a_\mathrm{h}^2}.
\end{aligned}
\tag{5.35}
$$

Now, we need to ensure that the in (5.32) assumed order of the contact waves also holds for the determined intermediate states:

$$
\lambda_1(\mathbf{u}_1) = (v_\mathrm{h})_\mathrm{L} - \frac{a_\mathrm{h}}{(\rho_\mathrm{h})_\mathrm{L}} \quad \overset{!}{<} \quad \lambda_2(\mathbf{u}_2) = (v_\mathrm{h})_\star \quad \overset{!}{<} \quad \lambda_3(\mathbf{u}_\mathrm{R}) = (v_\mathrm{h})_\mathrm{R} + \frac{a_\mathrm{h}}{(\rho_\mathrm{h})_\mathrm{R}}. \tag{5.36}
$$

The validity of the two inequalities can be controlled by the constant parameter a_h. The inequalities deliver two quadratic conditions for a_h and they hold, if a_h is large enough with

$$
a_\mathrm{h} > \max\Big\{ (\rho_\mathrm{h} c_\mathrm{h})_\mathrm{L}\,,\ \ (\rho_\mathrm{h} c_\mathrm{h})_\mathrm{R}\,,\ \ a_\mathrm{h}^\star,\ \ a_\mathrm{h}^{\star\star}\Big\}, \tag{5.37}
$$

where the first condition comes from the sub-characteristic condition (5.30), and

$$
\begin{aligned}
a_\mathrm{h}^\star &= \frac{1}{4}\left(-(\rho_\mathrm{h})_\mathrm{L}\left((v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}\right) \pm \sqrt{\left((\rho_\mathrm{h})_\mathrm{L}\right)^2\left((v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}\right)^2 + 8\,(\rho_\mathrm{h})_\mathrm{L}\left((\pi_\mathrm{h})_\mathrm{R} - (\pi_\mathrm{h})_\mathrm{L}\right)}\right),\\
a_\mathrm{h}^{\star\star} &= \frac{1}{4}\left(-(\rho_\mathrm{h})_\mathrm{R}\left((v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}\right) \pm \sqrt{\left((\rho_\mathrm{h})_\mathrm{R}\right)^2\left((v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}\right)^2 - 8\,(\rho_\mathrm{h})_\mathrm{R}\left((\pi_\mathrm{h})_\mathrm{R} - (\pi_\mathrm{h})_\mathrm{L}\right)}\right),
\end{aligned}
\tag{5.38}
$$

if the radicands are positive, otherwise $a_\mathrm{h}^\star$ and $a_\mathrm{h}^{\star\star}$ are zero, respectively. With this choice of a_h it is also ensured that the intermediate densities are positive. This comes directly from the order of the eigenvalues (5.36), and the fact that at each discontinuity it holds $\lambda_j(\mathbf{u}_{j-1}) = \lambda_j(\mathbf{u}_j)$. So, for the intermediate density $(\rho_\mathrm{h})_1$ we know

$$
(v_\mathrm{h})_1 - \frac{a_\mathrm{h}}{(\rho_\mathrm{h})_1} < (v_\mathrm{h})_1\,, \tag{5.39}
$$

such that with a positive constant parameter a_h in (5.37) the positivity for $(\rho_\mathrm{h})_1$ is given. In the same way this can be shown for $(\rho_\mathrm{h})_2$.

Overall this means that for the *relaxation homogeneous equilibrium model* the solution of the Riemann problem is known. For given initial values $\mathbf{u}_i^n$ the obtained result of this step is denoted with $\mathbf{u}_i^{n+1,-}$.

▶ Source terms at instantaneous relaxation ($t^{n+1,-} \to t^{n+1}$)

In the second step the relaxation source terms are considered for an infinite relaxation time, see (5.7), with $\mathbf{u}_i^{n+1,-}$ as initial data. This ordinary differential equation system can be solved exactly. By the form of the relaxation source terms $\mathbf{s}_\mathrm{r}(\mathbf{u})$, just $\rho_\mathrm{h}\pi_\mathrm{h}$ evolves according to the ordinary differential equation.

For the *relaxation homogeneous equilibrium model* we get

$$\mathbf{u}_i^{n+1} = \begin{pmatrix} (\rho_\mathrm{h})_i^{n+1} \\ (\rho_\mathrm{h} v_\mathrm{h})_i^{n+1} \\ (\rho_\mathrm{h} E_\mathrm{h})_i^{n+1} \\ (\rho_\mathrm{h} \pi_\mathrm{h})_i^{n+1} \end{pmatrix} = \begin{pmatrix} (\rho_\mathrm{h})_i^{n+1,-} \\ (\rho_\mathrm{h} v_\mathrm{h})_i^{n+1,-} \\ (\rho_\mathrm{h} E_\mathrm{h})_i^{n+1,-} \\ \left(\rho_\mathrm{h} p_\mathrm{h} + \exp\left(-\frac{\Delta t}{\varepsilon_\mathrm{h}}\right)(\rho_\mathrm{h}\pi_\mathrm{h} - \rho_\mathrm{h} p_\mathrm{h})\right)_i^{n+1,-} \end{pmatrix}. \tag{5.40}$$

5.3. Relaxation two-velocity two-pressure nine-equations model

The *two-velocity two-pressure seven-equations model* is extended by appending the relaxation pressure balance laws. A balance law for the pressure can be derived from the pressure relation (2.55), multiplying it $\alpha_k\rho_k$ times, and adding p_k times the mass balance equation, such that we get

$$\partial_t(\alpha_k\rho_k p_k) + \partial_x(\alpha_k\rho_k v_k p_k) + \alpha_k\rho_k^2 c_k^2\,\partial_x v_k + (v_k - v_\mathrm{i})\rho_k^2\left(\frac{p_\mathrm{i}}{\rho_k^2}(p_k)_u + (p_k)_\rho\right)\partial_x\alpha_k = \alpha_k\rho_k\,\mathbf{s}_{p_k} + p_k\,\mathbf{s}_{\alpha_k\rho_k}, \quad (5.41)$$

with source term [5]

$$\mathbf{s}_{p_k} = -\frac{\rho_k^2(p_k)_\rho}{\alpha_k\rho_k}\,\mathbf{s}_{\alpha_k} + \frac{\rho_k(p_k)_\rho - \left(E_k - v_k^2\right)(p_k)_u}{\alpha_k\rho_k}\,\mathbf{s}_{\alpha_k\rho_k} - \frac{v_k(p_k)_u}{\alpha_k\rho_k}\,\mathbf{s}_{\alpha_k\rho_k v_k} + \frac{(p_k)_u}{\alpha_k\rho_k}\,\mathbf{s}_{\alpha_k\rho_k E_k}. \quad (5.42)$$

Additionally the term with the spatial derivative of α is neglected, such that we use the balance law

$$\partial_t(\alpha_k\rho_k p_k) + \partial_x(\alpha_k\rho_k v_k p_k) + \alpha_k\rho_k^2 c_k^2\,\partial_x v_k = 0. \quad (5.43)$$

Suliciu intoduced a new parameter π_k which is defined by a Chapman-Enskog expansion of first order,

$$\pi_k := \underbrace{\pi_k^{(0)}}_{:=p_k} + \varepsilon_k\pi_k^{(1)}. \quad (5.44)$$

The zeroth-order approximation is taken to be the pressure such that for the relaxation time $\varepsilon_k \to 0$ it holds $\pi_k \to p_k$. The system is at equilibrium whenever $\pi_k = p_k$ [124].

With this expansion, the above approximate pressure balance law (5.43) will be relaxated by replacing the pressure p_k by π_k and by replacing the pre-factor of the velocity derivative, $\alpha_k\rho_k^2c_k^2$, by a constant parameter a_k^2. Additionally, a procedure of relaxation to equilibrium is added at the right-hand side, such that the relaxation balance law is given by

$$\partial_t(\alpha_k\rho_k\pi_k) + \partial_x(\alpha_k\rho_k v_k\pi_k) = -a_k^2\,\partial_x v_k + \alpha_k\rho_k\underbrace{\frac{1}{\varepsilon_k}(p_k - \pi_k)}_{=-\pi_k^{(1)}}. \quad (5.45)$$

Now, we can give an expression of the first-order corrector of π_k by studying the asymptotic behavior of the relaxation pressure balance law for $\varepsilon_k \to 0$. It holds:

$$\begin{aligned}&\partial_t(\alpha_k\rho_k\pi_k) + \partial_x(\alpha_k\rho_k v_k\pi_k) + a_k^2\,\partial_x v_k + \alpha_k\rho_k\pi_k^{(1)}\\ &\quad= \partial_t\left(\alpha_k\rho_k(p_k + \varepsilon_k\pi_k^{(1)})\right) + \partial_x\left(\alpha_k\rho_k v_k(p_k + \varepsilon_k\pi_k^{(1)}\right) + a_k^2\,\partial_x v_k + \alpha_k\rho_k\pi_k^{(1)}\end{aligned}$$

$$
\begin{aligned}
&= \partial_t(\alpha_k\rho_k p_k) + \underbrace{\varepsilon_k\,\partial_t(\alpha_k\rho_k\pi_k^{(1)})}_{\to 0} + \partial_x(\alpha_k\rho_k v_k p_k) + \underbrace{\varepsilon_k\,\partial_x(\alpha_k\rho_k v_k\pi_k^{(1)})}_{\to 0} + a_k^2\,\partial_x v_k + \alpha_k\rho_k\pi_k^{(1)} \\
&= \partial_t(\alpha_k\rho_k p_k) + \partial_x(\alpha_k\rho_k v_k p_k) + a_k^2\,\partial_x v_k + \alpha_k\rho_k\pi_k^{(1)} \\
&\overset{!}{=} \partial_t(\alpha_k\rho_k p_k) + \partial_x(\alpha_k\rho_k v_k p_k) + \alpha_k\rho_k^2 c_k^2\,\partial_x v_k, \qquad (5.46)
\end{aligned}
$$

such that we get

$$
-\pi_k^{(1)} = \frac{a_k^2 - \alpha_k\rho_k^2 c_k^2}{\alpha_k\rho_k}\,\partial_x v_k. \qquad (5.47)
$$

The constant parameter a_k is chosen below in Section 5.3.4, such that the subcharacteristic condition holds.

The *two-velocity two-pressure seven-equations model* is extended by appending the relaxation pressure balance laws (5.45). Additionally, all pressure terms p_k in the mass, momentum and energy balance laws are replaced by π_k, such that a nine-equation two-phase flow model is reached. For the general non-conservative form,

$$
\partial_t\mathbf{u} + \partial_x\mathbf{f}(\mathbf{u}) + B(\mathbf{u})\,\partial_x\mathbf{u} = \mathbf{s}_{\mathrm{r}}(\mathbf{u}) + \mathbf{s}(\mathbf{u}), \qquad (5.48)
$$

the independent variables $\mathbf{u}$ and the flux vector $\mathbf{f}(\mathbf{u})$ are given by

$$
\mathbf{u} := \begin{pmatrix} \alpha \\ (1-\alpha)\rho_\ell \\ (1-\alpha)\rho_\ell v_\ell \\ (1-\alpha)\rho_\ell E_\ell \\ (1-\alpha)\rho_\ell\pi_\ell \\ \alpha\rho_g \\ \alpha\rho_g v_g \\ \alpha\rho_g E_g \\ \alpha\rho_g\pi_g \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} 0 \\ (1-\alpha)\rho_\ell v_\ell \\ (1-\alpha)(\rho_\ell v_\ell^2 + \pi_\ell) \\ (1-\alpha)(\rho_\ell E_\ell + \pi_\ell)v_\ell \\ (1-\alpha)\rho_\ell v_\ell\pi_\ell + a_\ell^2 v_\ell \\ \alpha\rho_g v_g \\ \alpha(\rho_g v_g^2 + \pi_g) \\ \alpha(\rho_g E_g + \pi_g)v_g \\ \alpha\rho_g v_g\pi_g + a_g^2 v_g \end{pmatrix}, \qquad (5.49)
$$

where the non-conservative system matrix $B(\mathbf{u})$, the relaxation source vector $\mathbf{s}_{\mathrm{r}}(\mathbf{u})$ and the source vector $\mathbf{s}(\mathbf{u})$ are given by

$$B(\mathbf{u}) = \begin{pmatrix} v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \pi_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \pi_{\mathrm{i}} v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\pi_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\pi_{\mathrm{i}} v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{s}_{\mathrm{r}}(\mathbf{u}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \alpha_\ell \rho_\ell \frac{1}{\varepsilon_\ell}(p_\ell - \pi_\ell) \\ 0 \\ 0 \\ 0 \\ \alpha_g \rho_g \frac{1}{\varepsilon_g}(p_g - \pi_g) \end{pmatrix}, \tag{5.50}$$

and

$$\mathbf{s}(\mathbf{u}) = \begin{pmatrix} \mathbf{s}_{\alpha} \\ \mathbf{s}_{\alpha_\ell \rho_\ell} \\ \mathbf{s}_{\alpha_\ell \rho_\ell v_\ell} \\ \mathbf{s}_{\alpha_\ell \rho_\ell E_\ell} \\ \mathbf{s}_{\alpha_\ell \rho_\ell \pi_\ell} \\ \mathbf{s}_{\alpha_g \rho_g} \\ \mathbf{s}_{\alpha_g \rho_g v_g} \\ \mathbf{s}_{\alpha_g \rho_g E_g} \\ \mathbf{s}_{\alpha_g \rho_g \pi_g} \end{pmatrix} = \begin{pmatrix} \frac{\Gamma_{\mathrm{i}}}{\widehat{\rho}_{\mathrm{i}}} \\ -\Gamma_{\mathrm{i}} \\ F_{\mathrm{i}} + \alpha_\ell \rho_\ell \, g \, \sin(\vartheta) - \widehat{v}_{\mathrm{i}} \Gamma_{\mathrm{i}} + F_{\mathrm{w}\,\ell} \\ F_{\mathrm{i}} v_\ell + Q_{\mathrm{i}\,\ell} + \alpha_\ell \rho_\ell v_\ell \, g \, \sin(\vartheta) - \widehat{E}_{\mathrm{i}\,\ell} \Gamma_{\mathrm{i}} + F_{\mathrm{w}\,\ell} v_\ell + Q_{\mathrm{w}\,\ell} \\ 0 \\ \Gamma_{\mathrm{i}} \\ -F_{\mathrm{i}} + \alpha_g \rho_g \, g \, \sin(\vartheta) + \widehat{v}_{\mathrm{i}} \Gamma_{\mathrm{i}} + F_{\mathrm{w}\,g} \\ -F_{\mathrm{i}} v_g + Q_{\mathrm{i}\,g} + \alpha_g \rho_g v_g \, g \, \sin(\vartheta) + \widehat{E}_{\mathrm{i}\,g} \Gamma_{\mathrm{i}} + F_{\mathrm{w}\,g} v_g + Q_{\mathrm{w}\,g} \\ 0 \end{pmatrix}. \tag{5.51}$$

All solutions for $\mathbf{u}$ are in the set of admissible states,

$$\Omega = \{\mathbf{u} \in \mathbb{R}^n \mid \rho_k > 0, u_k > 0, c_k > 0\} \tag{5.52}$$

with $n = 9$, where u_k is the specific internal energy, given by the physical law

$$u_k = E_k - \tfrac{1}{2} v_k^2. \tag{5.53}$$

For the closed quasilinear system $\partial_t \mathbf{u} + A(\mathbf{u}) \, \partial_x \mathbf{u} = \mathbf{s}_{\mathrm{r}}(\mathbf{u}) + \mathbf{s}(\mathbf{u})$, we need to differentiate the flux vector $\mathbf{f}(\mathbf{u})$ with respect to our independent variables $\mathbf{u}$, such that the system matrix $A(\mathbf{u})$ is given by

$$A(\mathbf{u}) = \partial_{\mathbf{u}} \mathbf{f}(\mathbf{u}) + B(\mathbf{u}) =$$

$$\begin{pmatrix}
v_\mathrm{i} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\pi_\mathrm{i} - \pi_\ell & -\frac{\pi_\ell + \rho_\ell v_\ell^2}{\rho_\ell} & 2v_\ell & 0 & \frac{1}{\rho_\ell} & 0 & 0 & 0 & 0 \\
\pi_\mathrm{i} v_\mathrm{i} - \pi_\ell v_\ell & -\frac{(2\pi_\ell + E_\ell \rho_\ell) v_\ell}{\rho_\ell} & E_\ell + \frac{\pi_\ell}{\rho_\ell} & v_\ell & \frac{v_\ell}{\rho_\ell} & 0 & 0 & 0 & 0 \\
0 & -\pi_\ell v_\ell - \frac{a_\ell^2 v_\ell}{\alpha_\ell \rho_\ell} & \pi_\ell + \frac{a_\ell^2}{\alpha_\ell \rho_\ell} & 0 & v_\ell & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\pi_g - \pi_\mathrm{i} & 0 & 0 & 0 & 0 & -\frac{\pi_g + \rho_g v_g^2}{\rho_g} & 2v_g & 0 & \frac{1}{\rho_g} \\
\pi_g v_g - \pi_\mathrm{i} v_\mathrm{i} & 0 & 0 & 0 & 0 & -\frac{(2\pi_g + E_g \rho_g) v_g}{\rho_g} & E_g + \frac{\pi_g}{\rho_g} & v_g & \frac{v_g}{\rho_g} \\
0 & 0 & 0 & 0 & 0 & -\pi_g v_g - \frac{a_g^2 v_g}{\alpha_g \rho_g} & \pi_g + \frac{a_g^2}{\alpha_g \rho_g} & 0 & v_g
\end{pmatrix}. \tag{5.54}$$

Again we expect by construction of the Suliciu relaxation that the relaxation model has the same thermodynamical and mathematical properties as the original two-phase flow model (see Section 2.5). That this expectation holds is shown in the following.

5.3.1. Second law of thermodynamics

For the *relaxation two-velocity two-pressure nine-equations model* the same entropy-entropy flux pair is chosen as for the *two-velocity two-pressure seven-equations model* (2.91),

$$\eta(\mathbf{u}) = -((1-\alpha)\rho_\ell s_\ell + \alpha \rho_g s_g) \quad \text{and} \quad \psi(\mathbf{u}) = -((1-\alpha)\rho_\ell v_\ell s_\ell + \alpha \rho_g v_g s_g), \tag{5.55}$$

such that the following entropy variables arise

$$\mathbf{v}(\mathbf{u}) := \begin{pmatrix}
\frac{p_\ell}{T_\ell} - \frac{p_g}{T_g} \\
\frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} - s_\ell \\
\frac{v_\ell}{T_\ell} \\
-\frac{1}{T_\ell} \\
0 \\
\frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} - s_g \\
\frac{v_g}{T_g} \\
-\frac{1}{T_g} \\
0
\end{pmatrix} \overset{(2.88)}{=} \begin{pmatrix}
\frac{p_\ell}{T_\ell} - \frac{p_g}{T_g} \\
\frac{g_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\
\frac{v_\ell}{T_\ell} \\
-\frac{1}{T_\ell} \\
0 \\
\frac{g_g - \frac{1}{2}v_g^2}{T_g} \\
\frac{v_g}{T_g} \\
-\frac{1}{T_g} \\
0
\end{pmatrix}. \tag{5.56}$$

The entropy compatibility condition in (2.85) gives the same condition on the interfacial pressure and velocity as, for the *two-velocity two-pressure seven-equations model*:

$$\begin{aligned}
\partial_{\mathbf{u}}\psi(\mathbf{u})^{\mathsf{T}} &\overset{!}{=} \partial_{\mathbf{u}}\eta(\mathbf{u})^{\mathsf{T}}\cdot A(\mathbf{u}) \\
\Leftrightarrow \begin{pmatrix}
\frac{p_\ell v_\ell}{T_\ell} - \frac{p_g v_g}{T_g} \\
v_\ell \cdot \frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\
\frac{v_\ell^2}{T_\ell} - s_\ell \\
-\frac{v_\ell}{T_\ell} \\
0 \\
v_g \cdot \frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} \\
\frac{v_g^2}{T_g} - s_g \\
-\frac{v_g}{T_g} \\
0
\end{pmatrix} &\overset{!}{=} \begin{pmatrix}
\frac{p_\ell v_\ell}{T_\ell} - \frac{p_g v_g}{T_g} + \frac{(p_g - p_\mathrm{i})(v_g - v_\mathrm{i})}{T_g} - \frac{(p_\ell - p_\mathrm{i})(v_\ell - v_\mathrm{i})}{T_\ell} \\
v_\ell \cdot \frac{\frac{p_\ell}{\rho_\ell} + u_\ell - \frac{1}{2}v_\ell^2}{T_\ell} \\
\frac{v_\ell^2}{T_\ell} - s_\ell \\
-\frac{v_\ell}{T_\ell} \\
0 \\
v_g \cdot \frac{\frac{p_g}{\rho_g} + u_g - \frac{1}{2}v_g^2}{T_g} \\
\frac{v_g^2}{T_g} - s_g \\
-\frac{v_g}{T_g} \\
0
\end{pmatrix}.
\end{aligned} \tag{5.57}$$

Therefore the interfacial pressure is defined as in (2.94). The Hessian of the entropy function is given by

$$\eta''(\mathbf{u}) = \begin{pmatrix}
\eta_{\alpha,\alpha} & \eta_{(1-\alpha)\mathbf{u}_\ell,\alpha}^{\mathsf{T}} & \eta_{\alpha\mathbf{u}_g,\alpha}^{\mathsf{T}} \\
\eta_{(1-\alpha)\mathbf{u}_\ell,\alpha} & \frac{1}{1-\alpha}S_\ell''(\mathbf{u}_\ell) & 0 \\
\eta_{\alpha\mathbf{u}_g,\alpha} & 0 & \frac{1}{\alpha}S_g''(\mathbf{u}_g)
\end{pmatrix} \tag{5.58}$$

with phasic entropy function

$$S_k(\mathbf{u}_k) := -\rho_k s_k \quad \text{with} \quad \mathbf{u}_k^{\mathsf{T}} = (\rho_k, \rho_k v_k, \rho_k E_k, \pi_k E_k), \tag{5.59}$$

and derivatives

$$S_k'(\mathbf{u}_k) = \begin{pmatrix}
\frac{p_k + \rho_k(E_k - v_k^2)}{\rho_k T_k} - s_k \\
\frac{v_k}{T_k} \\
-\frac{1}{T_k} \\
0
\end{pmatrix}, \tag{5.60}$$

and

$$S_k''(\mathbf{u}_k) = \begin{pmatrix} \frac{T_k(p_k)_\rho - \left(p_k + 2\rho_k(E_k - v_k^2)\right)(T_k)_\rho + \left(E_k - v_k^2\right)^2 (T_k)_u + v_k^2 T_k}{\rho_k T_k^2} & -v_k\left(S_{13} + \frac{T_k}{\rho_k T_k^2}\right) & S_{13} & 0 \\ \frac{-\rho_k v_k (T_k)_\rho + v_k(E_k - v_k^2)(T_k)_u - v_k T_k}{\rho_k T_k^2} & \frac{v_k^2 (T_k)_u + T_k}{\rho_k T_k^2} & \frac{-v_k (T_k)_u}{\rho_k T_k^2} & 0 \\ \frac{\rho_k (T_k)_\rho - (E_k - v_k^2)(T_k)_u}{\rho_k T_k^2} & \frac{-v_k (T_k)_u}{\rho_k T_k^2} & \frac{(T_k)_u}{\rho_k T_k^2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{5.61}$$

with $S_{13} := \frac{\rho_k (T_k)_\rho - (E_k - v_k^2)(T_k)_u}{\rho_k T_k^2}$. From the *relaxation homogeneous equilibrium model* we already know that the phasic entropy functions $S_k(\mathbf{u}_k)$ are strictly convex, because the equation of state satisfies the same constraints. In order to verify now that $\eta(\mathbf{u})$ is a convex function of $\mathbf{u}$, we follow the same proof as shown for the *two-velocity two-pressure seven-equations model*.

The entropy production terms are the same as those from the *two-velocity two-pressure seven-equations model* (2.105), because the additional source entries for the pressure balance laws cancel out. This means that we get the same closures for

- source specific total energy (2.106),
- source specific enthalpy (2.107),
- source velocity (2.110), and
- source saturated temperature (2.112).

5.3.2. Hyperbolicity

In order to characterize mathematical properties, the equation systems are transformed in quasi-conservative form (2.114)

$$\partial_t \tilde{\mathbf{u}} + \tilde{A}(\tilde{\mathbf{u}}) \partial_x \tilde{\mathbf{u}} = \tilde{\mathbf{s}}(\tilde{\mathbf{u}}),$$

with primitive system vector

$$\tilde{\mathbf{u}} := \begin{pmatrix} \alpha & \rho_\ell & v_\ell & u_\ell & \pi_\ell & \rho_g & v_g & u_g & \pi_g \end{pmatrix}^\mathsf{T}. \tag{5.62}$$

and primitive system matrix

$$\tilde{A}(\tilde{\mathbf{u}}) = \begin{pmatrix} v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\rho_\ell(v_\ell - v_{\mathrm{i}})}{1-\alpha} & v_\ell & \rho_\ell & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\pi_\ell - \pi_{\mathrm{i}}}{(1-\alpha)\rho_\ell} & 0 & v_\ell & 0 & \frac{1}{\rho_\ell} & 0 & 0 & 0 & 0 \\ -\frac{\pi_{\mathrm{i}}(v_\ell - v_{\mathrm{i}})}{(1-\alpha)\rho_\ell} & 0 & \frac{\pi_\ell}{\rho_\ell} & v_\ell & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{a_\ell^2}{(1-\alpha)\rho_\ell} & 0 & v_\ell & 0 & 0 & 0 & 0 \\ \frac{\rho_g(v_g - v_{\mathrm{i}})}{\alpha} & 0 & 0 & 0 & 0 & v_g & \rho_g & 0 & 0 \\ \frac{\pi_g - \pi_{\mathrm{i}}}{\alpha\rho_g} & 0 & 0 & 0 & 0 & 0 & v_g & 0 & \frac{1}{\rho_g} \\ \frac{\pi_{\mathrm{i}}(v_g - v_{\mathrm{i}})}{\alpha\rho_g} & 0 & 0 & 0 & 0 & 0 & \frac{\pi_g}{\rho_g} & v_g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{a_g^2}{\alpha\rho_g} & 0 & v_g \end{pmatrix}. \tag{5.63}$$

The eigenvalue matrix Λ and the right and left eigenvectors R and L are given by

$$\Lambda = \begin{pmatrix} v_{\mathrm{i}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & v_\ell - \frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & v_\ell & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & v_\ell & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & v_\ell + \frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & v_g - \frac{a_g}{\rho_g\sqrt{\alpha_g}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & v_g & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & v_g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & v_g + \frac{a_g}{\rho_g\sqrt{\alpha_g}} \end{pmatrix}, \tag{5.64}$$

$$R^{\mathsf{T}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\pi_\ell - \pi_{\mathrm{i}} + \rho_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)}{\alpha_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)} & \frac{\alpha_\ell \rho_\ell^2}{a_\ell^2} & 1 & 0 & \frac{\alpha_\ell \rho_\ell^2}{a_\ell^2} & 0 & 0 & 0 & 0 \\ -\frac{(\pi_\ell - \pi_{\mathrm{i}})(v_\ell - v_{\mathrm{i}})}{\alpha_\ell \rho_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)} & -\frac{\sqrt{\alpha_\ell}}{a_\ell} & 0 & 0 & \frac{\sqrt{\alpha_\ell}}{a_\ell} & 0 & 0 & 0 & 0 \\ \frac{\pi_\ell^2 - \pi_\ell \pi_{\mathrm{i}} + \pi_{\mathrm{i}} \rho_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)}{\alpha_\ell \rho_\ell^2 \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)} & \frac{\alpha_\ell \pi_\ell}{a_\ell^2} & 0 & 1 & \frac{\alpha_\ell \pi_\ell}{a_\ell^2} & 0 & 0 & 0 & 0 \\ \frac{(\pi_\ell - \pi_{\mathrm{i}}) \frac{a_\ell^2}{\alpha_\ell \rho_\ell^2}}{\alpha_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_{\mathrm{i}})^2 \right)} & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{\pi_g - \pi_{\mathrm{i}} + \rho_g \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)}{\alpha \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)} & 0 & 0 & 0 & 0 & \frac{\alpha_g \rho_g^2}{a_g^2} & 1 & 0 & \frac{\alpha_g \rho_g^2}{a_g^2} \\ \frac{(\pi_g - \pi_{\mathrm{i}})(v_g - v_{\mathrm{i}})}{\alpha \rho_g \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)} & 0 & 0 & 0 & 0 & -\frac{\sqrt{\alpha_g}}{a_g} & 0 & 0 & \frac{\sqrt{\alpha_g}}{a_g} \\ -\frac{\pi_g^2 - \pi_g \pi_{\mathrm{i}} + \pi_{\mathrm{i}} \rho_g \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)}{\alpha \rho_g^2 \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)} & 0 & 0 & 0 & 0 & \frac{\alpha_g \pi_g}{a_g^2} & 0 & 1 & \frac{\alpha_g \pi_g}{a_g^2} \\ -\frac{(\pi_g - \pi_{\mathrm{i}}) \frac{a_g^2}{\alpha_g \rho_g^2}}{\alpha \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_{\mathrm{i}})^2 \right)} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad (5.65)$$

and

$$L = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{(\pi_\ell-\pi_\mathrm{i})\frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}}}{2\,\alpha_\ell\left(\frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}}-v_\ell+v_\mathrm{i}\right)} & 0 & -\frac{a_\ell}{2\sqrt{\alpha_\ell}} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\
-\frac{\rho_\ell}{1-\alpha} & 1 & 0 & 0 & -\frac{\alpha_\ell\rho_\ell^2}{a_\ell^2} & 0 & 0 & 0 & 0 \\
-\frac{\pi_\mathrm{i}}{\alpha_\ell\rho_\ell} & 0 & 0 & 1 & -\frac{\alpha_\ell\pi_\ell}{a_\ell^2} & 0 & 0 & 0 & 0 \\
-\frac{(\pi_\ell-\pi_\mathrm{i})\frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}}}{2\,\alpha_\ell\left(\frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}}+v_\ell-v_\mathrm{i}\right)} & 0 & \frac{a_\ell}{2\sqrt{\alpha_\ell}} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\
\frac{(\pi_g-\pi_\mathrm{i})\frac{a_g}{\rho_g\sqrt{\alpha_g}}}{2\,\alpha\left(\frac{a_g}{\rho_g\sqrt{\alpha_g}}-v_g+v_\mathrm{i}\right)} & 0 & 0 & 0 & 0 & 0 & -\frac{a_g}{2\sqrt{\alpha_g}} & 0 & \frac{1}{2} \\
\frac{\rho_g}{\alpha} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -\frac{\alpha_g\rho_g^2}{a_g^2} \\
\frac{\pi_\mathrm{i}}{\alpha\rho_g} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\frac{\alpha_g\pi_g}{a_g^2} \\
\frac{(\pi_g-\pi_\mathrm{i})\frac{a_g}{\rho_g\sqrt{\alpha_g}}}{2\,\alpha\left(\frac{a_g}{\rho_g\sqrt{\alpha_g}}+v_g-v_\mathrm{i}\right)} & 0 & 0 & 0 & 0 & 0 & \frac{a_g}{2\sqrt{\alpha_g}} & 0 & \frac{1}{2}
\end{pmatrix}. \tag{5.66}$$

The system admits nine real eigenvalues which are all real but not necessarily distinct: v_i, v_k and $v_k \pm \frac{a_k}{\rho_k\sqrt{\alpha_k}}$. Also here the corresponding eigenvectors are linearly independent, as soon as the non-resonance condition is fulfilled:

$$v_\mathrm{i} \neq v_k \pm \frac{a_k}{\rho_k\sqrt{\alpha_k}}. \tag{5.67}$$

Below in (5.71) the parameter a_k is chosen, such that the term $\frac{a_k}{\rho_k\sqrt{\alpha_k}}$ is close to the speed of sound. Thus, for subsonic flow the non-resonance condition is fulfilled.

5.3.3. Characteristic fields

The motivation to construct a relaxation model was to obtain a linearly degenerate system, such that $\partial_{\tilde{\mathbf{u}}}\lambda_j\, R_j = 0$ for all $j = 1, \ldots, n$. Note that the first eigenvalue v_i was already chosen in equation (2.122), such that its associated field is linearly degenerated.

Each j-characteristic field has several j-Riemann invariants $\mathbf{w}_p^{(j)}$, which are constant along the trajectories of the vector field R_j:

$$\partial_{\tilde{\mathbf{u}}}\mathbf{w}_p^{(j)}\, R_j \stackrel{!}{=} 0, \tag{5.68}$$

From this definition it is obvious to see that λ_j is also a j-Riemann invariant if the j-characteristic field is linearly degenerated [99]. A linearly degenerated j-characteristic

field of an eigenvalue λ_j with multiplicity m_j has exactly $(n-m_j)$ j-Riemann invariants, whose gradients are all linearly independent[17]. The j-Riemann invariants are given by

$$
\begin{aligned}
\mathbf{w}^{(1)} = \Big\{ & v_\mathrm{i}, \\
& \alpha_\ell \rho_\ell (v_\ell - v_\mathrm{i}), \\
& u_\ell + \frac{\pi_\ell}{\rho_\ell} + \tfrac{1}{2}(v_\ell - v_\mathrm{i})^2, \\
& u_g + \frac{\pi_g}{\rho_g} + \tfrac{1}{2}(v_g - v_\mathrm{i})^2, \\
& \alpha_\ell \rho_\ell \pi_\ell (v_\ell - v_\mathrm{i}) + a_\ell^2 v_\ell, \\
& \alpha_g \rho_g \pi_g (v_g - v_\mathrm{i}) + a_g^2 v_g, \\
& \alpha_\ell \big(\pi_\ell + \rho_\ell v_\ell (v_\ell - v_\mathrm{i})\big) + \alpha_g \big(\pi_g + \rho_g v_g (v_g - v_\mathrm{i})\big), \\
& \alpha_\ell \rho_\ell s_\ell (v_\ell - v_\mathrm{i}) + \alpha_g \rho_g s_g (v_g - v_\mathrm{i}) \Big\}, \\
\mathbf{w}^{(2)} = \Big\{ & \alpha, v_\ell - \frac{a_\ell}{\rho_\ell \sqrt{\alpha_\ell}}, \pi_\ell + v_\ell \frac{a_\ell}{\sqrt{\alpha_\ell}}, -\tfrac{1}{2}\pi_\ell^2 + u_\ell \frac{a_\ell^2}{\alpha_\ell}, \rho_g, v_g, u_g, \pi_g \Big\}, \\
\mathbf{w}^{(3)} = \mathbf{w}^{(4)} = \Big\{ & \alpha, v_\ell, \pi_\ell, \rho_g, v_g, u_g, \pi_g \Big\}, \\
\mathbf{w}^{(5)} = \Big\{ & \alpha, v_\ell + \frac{a_\ell}{\rho_\ell \sqrt{\alpha_\ell}}, \pi_\ell - v_\ell \frac{a_\ell}{\sqrt{\alpha_\ell}}, -\tfrac{1}{2}\pi_\ell^2 + u_\ell \frac{a_\ell^2}{\alpha_\ell}, \rho_g, v_g, u_g, \pi_g \Big\}, \\
\mathbf{w}^{(6)} = \Big\{ & \alpha, \rho_\ell, v_\ell, u_\ell, \pi_\ell, v_g - \frac{a_g}{\rho_g \sqrt{\alpha_g}}, \pi_g + v_g \frac{a_g}{\sqrt{\alpha_g}}, -\tfrac{1}{2}\pi_g^2 + u_g \frac{a_g^2}{\alpha_g} \Big\}, \\
\mathbf{w}^{(7)} = \mathbf{w}^{(8)} = \Big\{ & \alpha, \rho_\ell, v_\ell, u_\ell, \pi_\ell, v_g, \pi_g \Big\}, \\
\mathbf{w}^{(9)} = \Big\{ & \alpha, \rho_\ell, v_\ell, u_\ell, \pi_\ell, v_g + \frac{a_g}{\rho_g \sqrt{\alpha_g}}, \pi_g - v_g \frac{a_g}{\sqrt{\alpha_g}}, -\tfrac{1}{2}\pi_g^2 + u_g \frac{a_g^2}{\alpha_g} \Big\}.
\end{aligned}
\tag{5.69}
$$

An interesting point is that the last 1-Riemann invariant can be written in dependency of the entropy-entropy flux pair as $\mathbf{w}_8^{(1)} = -v_\mathrm{i}\,\eta(\mathbf{u}) + \psi(\mathbf{u})$. This is due to the fact that the interfacial pressure in (2.94) was chosen such that the entropy compatibility condition holds. The special choice of p_i can be seen as applying an additional conservation law to the system. Further information about computing the Riemann invariants can be found in [8].

5.3.4. Subcharacteristic condition

The sub-characteristic condition can be seen as a measure for the stability of an extended system compared to the original system. This condition states that the wave velocities of the origin system should never exceed the corresponding wave velocities in the extended system [126], see also Whitham [127] and Liu [120].

[17] This was proven by Serre [125], Theorem 3.3.3.

The sub-characteristic condition is fulfilled, if the eigenvalues of the *two-velocity two-pressure model* are interlaced with the eigenvalues of the *relaxation two-velocity two-pressure model*. It's obvious that with

$$c_\ell \leq \frac{a_\ell}{\rho_\ell \sqrt{\alpha_\ell}} \quad \text{and} \quad c_g \leq \frac{a_g}{\rho_g \sqrt{\alpha_g}} \tag{5.70}$$

the sub-characteristic condition is fulfilled. This gives us a condition for the constant parameter a_k, such that we choose

$$a_k \geq \sqrt{\alpha_k}\rho_k c_k. \tag{5.71}$$

5.3.5. Symmetrisation of the hyperbolic system

We need to find a symmetric positive definite matrix $P(\tilde{\mathbf{u}})$ such that also the matrix $P(\tilde{\mathbf{u}})\tilde{A}(\tilde{\mathbf{u}})$ is symmetric, Here we follow the idea from the *two-velocity two-pressure seven-equations model* by choosing $P(\tilde{\mathbf{u}})$ as introduced in (2.127) with the formulas (2.128) and (2.129).

Please note that for the *relaxation two-velocity two-pressure nine-equations model* Λ_k, R_k and L_k describe the 4×4 sub-matrices from L, R and Λ, with rows/columns 2–5 for liquid, and 6–9 for steam phase. $\tilde{A}_{k\,\alpha}$ describes the 4×1 sub-matrix from the the first column of $\tilde{A}$, rows 2–5 for liquid, and 6–9 for steam phase. So, finally we get

$$P_\ell = \begin{pmatrix} 1 & 0 & 0 & -\frac{\alpha_\ell \rho_\ell^2}{a_\ell^2} \\ 0 & \frac{a_\ell^2}{2\alpha_\ell} & 0 & 0 \\ 0 & 0 & 1 & -\frac{\alpha_\ell \pi_\ell}{a_\ell^2} \\ -\frac{\alpha_\ell \rho_\ell^2}{a_\ell^2} & 0 & -\frac{\alpha_\ell \pi_\ell}{a_\ell^2} & \frac{1}{2} + \frac{\alpha_\ell^2(\rho_\ell^4 + \pi_\ell^2)}{a_\ell^4} \end{pmatrix}, \quad P_g = \begin{pmatrix} 1 & 0 & 0 & -\frac{\alpha_g \rho_g^2}{a_g^2} \\ 0 & \frac{a_g^2}{2\alpha_g} & 0 & 0 \\ 0 & 0 & 1 & -\frac{\alpha_g \pi_g}{a_g^2} \\ -\frac{\alpha_g \rho_g^2}{a_g^2} & 0 & -\frac{\alpha_g \pi_g}{a_g^2} & \frac{1}{2} + \frac{\alpha_g^2(\rho_g^4 + \pi_g^2)}{a_g^4} \end{pmatrix}, \tag{5.72}$$

and

$$P_{\ell\,\alpha} = \begin{pmatrix} -\frac{\rho_\ell}{\alpha_\ell} \\ \frac{a_\ell^2(\pi_\ell - \pi_\mathrm{i})(v_\ell - v_\mathrm{i})}{2\alpha_\ell^2 \rho_\ell \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_\mathrm{i})^2 \right)} \\ -\frac{\pi_\mathrm{i}}{\alpha_\ell \rho_\ell} \\ \frac{\pi_\ell \pi_\mathrm{i} + \rho_\ell^4}{\rho_\ell a_\ell^2} - \frac{a_\ell^2(\pi_\ell - \pi_\mathrm{i})}{2\alpha_\ell^2 \rho_\ell^2 \left(\frac{a_\ell^2}{\alpha_\ell \rho_\ell^2} - (v_\ell - v_\mathrm{i})^2 \right)} \end{pmatrix}, \quad P_{g\,\alpha} = \begin{pmatrix} \frac{\rho_g}{\alpha_g} \\ -\frac{a_g^2(\pi_g - \pi_\mathrm{i})(v_g - v_\mathrm{i})}{2\alpha_g^2 \rho_g \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_\mathrm{i})^2 \right)} \\ \frac{\pi_\mathrm{i}}{\alpha_g \rho_g} \\ -\frac{\pi_g \pi_\mathrm{i} + \rho_g^4}{\rho_g a_g^2} + \frac{a_g^2(\pi_g - \pi_\mathrm{i})}{2\alpha_g^2 \rho_g^2 \left(\frac{a_g^2}{\alpha_g \rho_g^2} - (v_g - v_\mathrm{i})^2 \right)} \end{pmatrix}. \tag{5.73}$$

Also here it holds that the choice for $P(\tilde{\mathbf{u}})$ is realizable if the non-resonance condition (5.67) holds. P_α is chosen as introduced in (2.134).

5.4. Riemann solver for the relaxation two-velocity two-pressure model

As introduced at the begin of this Section, the Riemann solution is provided in two steps: Starting with $\mathbf{u}_i^n$ as initial data, at first the convective part is solved, such that the solution $\mathbf{u}_i^{n+1,-}$ is provided. Then as second step, the projection on the equilibrium manifold is solved, such that the solution $\mathbf{u}_i^{n+1}$ at the next time step is provided.

▶ Evolution in time ($t^n \to t^{n+1,-}$)

For the *relaxation two-velocity two-pressure model* we would proceed in the same way as for the *homogeneous model* above. Thus, we would order the eigenvalues and formulate an equation system from the Rankine-Hugoniot jump relations across the discontinuities. Regarding now the Riemann invariants (see Section 5.3.3) it is easy to detect that the volume fractions α_ℓ and α_g stay constant on the left-hand and on the right-hand side of the v_{i}-contact discontinuity. The other six waves act like two independent Euler systems: they just affect their own phase without having influence on the other phase nor the volume fraction. Just the v_{i}-wave enforces an exchange of the liquid and steam phase corresponding to the difference of the left and right volume fraction. That is why the v_{i}-discontinuity is often called the coupling wave between liquid and vapor phase [128].

For the case of equal volume fractions $\alpha_{\mathrm{L}} = \alpha_{\mathrm{R}}$ at the left-hand and right-hand side of the Riemann problem, the spatial derivative of the volume fraction is zero, such that the Riemann invariants of the v_{i}-wave reduce to $\mathbf{w}^{(1)}_{\alpha_{\mathrm{L}}:=\alpha_{\mathrm{R}}} = \{\rho_\ell, v_\ell, u_\ell, p_\ell, \rho_g, v_g, u_g, p_g\}$. This leads to a decoupled system with two separated Euler equations such that the Riemann solution can be computed exactly, compare with the solution of the *homogeneous model*.

For the general case of different volume fractions $\alpha_{\mathrm{L}} \neq \alpha_{\mathrm{R}}$ at the left and right hand side, the Riemann problem becomes more complicated. Due to the non-linearities of the v_{i}-Riemann invariants, the intermediate states of the Riemann problem are difficult to determine. Since already for a simpler problem with an isentropic two-phase flow model, Ambroso, Chalons, Coquel, and Galié [128] did not find an explicit solution, we not either expect to find some explicit Riemann solution for the underlying nine-equations model.

An approximate solution can be obtained by weaken the nonlinear Rankine-Hugoniot conditions at the v_{i}-contact discontinuity. The approximation should retain the property that for the case of $\alpha_{\mathrm{L}} = \alpha_{\mathrm{R}}$ the approximated Riemann invariants $\widetilde{\mathbf{w}}^{(1)}$ again reduce to $\mathbf{w}^{(1)}_{\alpha_{\mathrm{L}}=\alpha_{\mathrm{R}}}$. Here we propose a simplification by linearization of some Riemann invariants:

$$\widetilde{\mathbf{w}}^{(1)} := \left\{ \alpha_\ell \rho_\ell, v_\ell, u_\ell + \frac{p_\ell}{\rho_\ell}, p_\ell, \alpha_g \rho_g, v_g, u_g + \frac{p_g}{\rho_g}, p_g \right\} \tag{5.74}$$

These simplified Riemann invariants do not affect the pressure and velocity of the liquid and steam phase, as we already know from the v_ℓ–wave and v_g–wave. Additionally, with

$\widetilde{\mathbf{w}}_3^{(1)} := h_\ell$, $\widetilde{\mathbf{w}}_7^{(1)} := h_g$, and the phasic pressures we have two extensive quantities for each phase, such that the specific entropy s_ℓ and s_g and the speed of sound c_ℓ and c_g do not affect the liquid and steam phase. Thus, with the choice of these simplified Riemann invariants $\widetilde{\mathbf{w}}^{(1)}$ it is ensured that at least six out of eight original Rankine-Hugoniot conditions are fulfilled. Just the fifth $\mathbf{w}_5^{(1)}$ and sixth $\mathbf{w}_6^{(1)}$ Riemann invariant can be unfilled because of the hidden α in a_k. Later, these quantities can be used as a *measure* for the quality of the solution.

This simplification allows now to compute an explicit solution of the Riemann problem. The underlying model has $\tilde{n} = 7$ distinct eigenvalues v_i, v_ℓ, v_g, $v_\ell \pm \frac{a_\ell}{\rho_\ell}$, and $v_g \pm \frac{a_g}{\rho_g}$, see 5.71). For subsonic flow of water we can assume that the speed of sound in liquid water is greater than the speed of sound in steam[18] [78], $c_\ell \gg c_g$, see also Figure 38. Therefore we assume $\frac{a_\ell}{\rho_\ell} \gg \frac{a_g}{\rho_g}$, Additionally, we know that v_i was chosen as convex combination of v_ℓ and v_g, see (2.122), such that v_i is ordered in-between the phasic velocities. From a physical point of view, we would assume that the liquid phase is slower than the vapor phase, but this is not guaranteed. Therefore here we need to consider two cases. This gives us the following order of the eigenvalues:

$$\begin{aligned}
\lambda_1 := v_\ell - \frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}} \quad &< \quad \lambda_2 := v_g - \frac{a_g}{\rho_g\sqrt{\alpha_g}} \\
&< \quad \lambda_3 := \min\left\{v_\ell, v_g\right\} \\
&< \quad \lambda_4 := v_\mathrm{i} \\
&< \quad \lambda_5 := \max\left\{v_\ell, v_g\right\} \\
&< \quad \lambda_6 := v_g + \frac{a_g}{\rho_g\sqrt{\alpha_g}} \qquad < \quad \lambda_7 := v_\ell + \frac{a_\ell}{\rho_\ell\sqrt{\alpha_\ell}}.
\end{aligned} \tag{5.75}$$

Now an equation system from the Rankine-Hugoniot jump relations across the discontinuities[19] (see Section 5.3.3) can be formulated. Solving for the parameters of the unknown intermediate states in primitive variables, $\tilde{\mathbf{u}}_{j=1,\dots,6}$, a solution in dependency

[18] Notice that in our application we deal with a pressure lower than 140 bar and a temperature lower than 700 °C.

[19] Notice that due to the in (5.75) introduced order of the waves, the Riemann invariants in Section 5.1.3 have a different numbering than used in this Section.

of the known left $\mathbf{u}_{\mathrm{L}}$ and right state $\mathbf{u}_{\mathrm{R}}$ is obtained:

$$\tilde{\mathbf{u}}_1 = \begin{pmatrix} (\alpha)_{\mathrm{L}} \\ (\rho_\ell)_1 \\ (v_\ell)_\star \\ (u_\ell)_1 \\ (\pi_\ell)_\star \\ (\rho_g)_{\mathrm{L}} \\ (v_g)_{\mathrm{L}} \\ (u_g)_{\mathrm{L}} \\ (\pi_g)_{\mathrm{L}} \end{pmatrix}, \quad \tilde{\mathbf{u}}_2 = \begin{pmatrix} (\alpha)_{\mathrm{L}} \\ (\rho_\ell)_2 \\ (v_\ell)_\star \\ (u_\ell)_2 \\ (\pi_\ell)_\star \\ (\rho_g)_2 \\ (v_g)_\star \\ (u_g)_2 \\ (\pi_g)_\star \end{pmatrix}, \quad \tilde{\mathbf{u}}_3 = \begin{pmatrix} (\alpha)_{\mathrm{L}} \\ (\rho_\ell)_3 \\ (v_\ell)_\star \\ (u_\ell)_3 \\ (\pi_\ell)_\star \\ (\rho_g)_3 \\ (v_g)_\star \\ (u_g)_3 \\ (\pi_g)_\star \end{pmatrix}, \quad \tilde{\mathbf{u}}_4 = \begin{pmatrix} (\alpha)_{\mathrm{R}} \\ (\rho_\ell)_4 \\ (v_\ell)_\star \\ (u_\ell)_4 \\ (\pi_\ell)_\star \\ (\rho_g)_4 \\ (v_g)_\star \\ (u_g)_4 \\ (\pi_g)_\star \end{pmatrix}, \quad \tilde{\mathbf{u}}_5 = \begin{pmatrix} (\alpha)_{\mathrm{R}} \\ (\rho_\ell)_5 \\ (v_\ell)_\star \\ (u_\ell)_5 \\ (\pi_\ell)_\star \\ (\rho_g)_5 \\ (v_g)_\star \\ (u_g)_5 \\ (\pi_g)_\star \end{pmatrix}, \quad \tilde{\mathbf{u}}_6 = \begin{pmatrix} (\alpha)_{\mathrm{R}} \\ (\rho_\ell)_6 \\ (v_\ell)_\star \\ (u_\ell)_6 \\ (\pi_\ell)_\star \\ (\rho_g)_{\mathrm{R}} \\ (v_g)_{\mathrm{R}} \\ (u_g)_{\mathrm{R}} \\ (\pi_g)_{\mathrm{R}} \end{pmatrix}, \tag{5.76}$$

with velocity

$$(v_k)_\star = \frac{(v_k)_{\mathrm{R}} \sqrt{(\alpha_k)_{\mathrm{L}}} + (v_k)_{\mathrm{L}} \sqrt{(\alpha_k)_{\mathrm{R}}}}{\sqrt{(\alpha_k)_{\mathrm{L}}} + \sqrt{(\alpha_k)_{\mathrm{R}}}} + \frac{\sqrt{(\alpha_k)_{\mathrm{L}}}\sqrt{(\alpha_k)_{\mathrm{R}}}\,((\pi_k)_{\mathrm{L}} - (\pi_k)_{\mathrm{R}})}{a_k\left(\sqrt{(\alpha_k)_{\mathrm{L}}} + \sqrt{(\alpha_k)_{\mathrm{R}}}\right)}, \tag{5.77}$$

relaxed pressure

$$(\pi_k)_\star = \frac{a_k\,(v_k)_{\mathrm{L}} - a_k\,(v_k)_{\mathrm{R}} + (\pi_k)_{\mathrm{L}} \sqrt{(\alpha_k)_{\mathrm{L}}} + (\pi_k)_{\mathrm{R}} \sqrt{(\alpha_k)_{\mathrm{R}}}}{\sqrt{(\alpha_k)_{\mathrm{L}}} + \sqrt{(\alpha_k)_{\mathrm{R}}}}, \tag{5.78}$$

density

$$(\rho_\ell)_1 = (\rho_\ell)_2 = \left(\frac{1}{(\rho_\ell)_{\mathrm{L}}} + \Big((v_\ell)_\star - (v_\ell)_{\mathrm{L}} \Big) \frac{\sqrt{(\alpha_\ell)_{\mathrm{L}}}}{a_\ell} \right)^{-1},$$

$$(\rho_\ell)_3 = \begin{cases} (\rho_\ell)_5 \dfrac{(\alpha_\ell)_{\mathrm{R}}}{(\alpha_\ell)_{\mathrm{L}}} & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (\rho_\ell)_2 & , \text{ if } (v_\ell)_\star > (v_g)_\star \end{cases}$$

$$(\rho_\ell)_4 = \begin{cases} (\rho_\ell)_5 & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (\rho_\ell)_2 \dfrac{(\alpha_\ell)_{\mathrm{L}}}{(\alpha_\ell)_{\mathrm{R}}} & , \text{ if } (v_\ell)_\star > (v_g)_\star \end{cases}$$

$$(\rho_\ell)_5 = (\rho_\ell)_6 = \left(\frac{1}{(\rho_\ell)_{\mathrm{R}}} - \Big((v_\ell)_\star - (v_\ell)_{\mathrm{R}} \Big) \frac{\sqrt{(\alpha_\ell)_{\mathrm{R}}}}{a_\ell} \right)^{-1},$$

$$(\rho_g)_2 = \left(\frac{1}{(\rho_g)_{\mathrm{L}}} + \Big((v_g)_\star - (v_g)_{\mathrm{L}} \Big) \frac{\sqrt{(\alpha_g)_{\mathrm{L}}}}{a_g} \right)^{-1},$$

$$(\rho_g)_3 = \begin{cases} (\rho_g)_2 & , \text{ if } (v_g)_\star < (v_g)_\star \\ (\rho_g)_5 \, \dfrac{(\alpha_g)_{\mathrm{R}}}{(\alpha_g)_{\mathrm{L}}} & , \text{ if } (v_g)_\star > (v_g)_\star \end{cases}$$

$$(\rho_g)_4 = \begin{cases} (\rho_g)_2 \, \dfrac{(\alpha_g)_{\mathrm{L}}}{(\alpha_g)_{\mathrm{R}}} & , \text{ if } (v_g)_\star < (v_g)_\star \\ (\rho_g)_5 & , \text{ if } (v_g)_\star > (v_g)_\star \end{cases}$$

$$(\rho_g)_5 = \left(\frac{1}{(\rho_g)_{\mathrm{R}}} - \Big((v_g)_\star - (v_g)_{\mathrm{R}} \Big) \frac{\sqrt{(\alpha_g)_{\mathrm{R}}}}{a_g} \right)^{-1}, \tag{5.79}$$

and specific inner energy

$$(u_\ell)_1 = (u_\ell)_2 = (u_\ell)_{\mathrm{L}} + \Big((\pi_\ell)_\star^2 - (\pi_\ell)_{\mathrm{L}}^2 \Big) \frac{(\alpha_\ell)_{\mathrm{L}}}{2a_\ell^2},$$

$$(u_\ell)_3 = \begin{cases} (u_\ell)_5 + \frac{(\alpha_\ell)_{\mathrm{R}} - (\alpha_\ell)_{\mathrm{L}}}{(\alpha_\ell)_{\mathrm{R}}} \cdot \frac{(\pi_\ell)_\star}{(\rho_\ell)_5} & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (u_\ell)_2 & , \text{ if } (v_\ell)_\star > (v_g)_\star \end{cases}$$

$$(u_\ell)_4 = \begin{cases} (u_\ell)_5 & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (u_\ell)_2 + \frac{(\alpha_\ell)_{\mathrm{L}} - (\alpha_\ell)_{\mathrm{R}}}{(\alpha_\ell)_{\mathrm{L}}} \cdot \frac{(\pi_\ell)_\star}{(\rho_\ell)_2} & , \text{ if } (v_\ell)_\star > (v_g)_\star \end{cases}$$

$$(u_\ell)_5 = (u_\ell)_6 = (u_\ell)_{\mathrm{R}} + \Big((\pi_\ell)_\star^2 - (\pi_\ell)_{\mathrm{R}}^2 \Big) \frac{(\alpha_\ell)_{\mathrm{R}}}{2a_\ell^2},$$

$$(u_g)_2 = (u_g)_{\mathrm{L}} + \Big((\pi_g)_\star^2 - (\pi_g)_{\mathrm{L}}^2 \Big) \frac{(\alpha_g)_{\mathrm{L}}}{2a_g^2},$$

$$(u_g)_3 = \begin{cases} (u_g)_2 & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (u_g)_5 + \frac{(\alpha_g)_{\mathrm{R}} - (\alpha_g)_{\mathrm{L}}}{(\alpha_g)_{\mathrm{R}}} \cdot \frac{(\pi_g)_\star}{(\rho_g)_5} & , \text{ if } (v_\ell)_\star > (v_g)_\star \end{cases}$$

$$(u_g)_4 = \begin{cases} (u_g)_2 + \frac{(\alpha_g)_{\mathrm{L}} - (\alpha_g)_{\mathrm{R}}}{(\alpha_g)_{\mathrm{L}}} \cdot \frac{(\pi_g)_\star}{(\rho_g)_2} & , \text{ if } (v_\ell)_\star < (v_g)_\star \\ (u_g)_5 & , \text{ if } (v_\ell)_\star > (v_g)_\star , \end{cases}$$

$$(u_g)_5 = (u_g)_{\mathrm{R}} + \Big((\pi_g)_\star^2 - (\pi_g)_{\mathrm{R}}^2 \Big) \frac{(\alpha_g)_{\mathrm{R}}}{2a_g^2}. \tag{5.80}$$

The order of the fluid velocities determines the order of the inner waves, which influences the solution of the third and fourth intermediate state. Therefore a case analysis is made, whether $(v_\ell)_\star$ is smaller or larger than $(v_g)_\star$. Now, we need to ensure that the in (5.75)

assumed order of the contact waves also hold for the determined intermediate states:

$$\begin{aligned}
\lambda_1(\mathbf{u}_1) := (v_\ell)_{\mathrm{L}} - \frac{a_\ell}{(\rho_\ell)_{\mathrm{L}}\sqrt{(\alpha_\ell)_{\mathrm{L}}}} \quad &\overset{!}{<} \quad \lambda_2(\mathbf{u}_2) := (v_g)_{\mathrm{L}} - \frac{a_g}{(\rho_g)_{\mathrm{L}}\sqrt{(\alpha_g)_{\mathrm{L}}}} \\
&\overset{!}{<} \quad \lambda_3(\mathbf{u}_3) := \min\left\{ (v_\ell)_\star, (v_g)_\star \right\} \\
&\overset{!}{<} \quad \lambda_4(\mathbf{u}_4) := (v_{\mathrm{i}})_\star \\
&\overset{!}{<} \quad \lambda_5(\mathbf{u}_5) := \max\left\{ (v_\ell)_\star, (v_g)_\star \right\} \\
&\overset{!}{<} \quad \lambda_6(\mathbf{u}_6) := (v_g)_{\mathrm{R}} + \frac{a_g}{(\rho_g)_{\mathrm{R}}\sqrt{(\alpha_g)_{\mathrm{R}}}} \\
&\overset{!}{<} \quad \lambda_7(\mathbf{u}_{\mathrm{R}}) := (v_\ell)_{\mathrm{R}} + \frac{a_\ell}{(\rho_\ell)_{\mathrm{R}}\sqrt{(\alpha_\ell)_{\mathrm{R}}}}.
\end{aligned} \tag{5.81}$$

First of all it is clear that the inequalities $\lambda_3(\mathbf{u}_3) < \lambda_4(\mathbf{u}_4) < \lambda_5(\mathbf{u}_5)$ hold, because the interfacial velocity at intermediate state can be written as convex combination of $(v_\ell)_\star$ and $(v_g)_\star$:

$$\begin{aligned}
(v_{\mathrm{i}})_\star &= (\beta)_{\mathrm{L}} (v_\ell)_\star + \left(1 - (\beta)_{\mathrm{L}}\right) (v_g)_\star \\
&= (\beta)_{\mathrm{R}} (v_\ell)_\star + \left(1 - (\beta)_{\mathrm{R}}\right) (v_g)_\star
\end{aligned} \tag{5.82}$$

with

$$(\beta)_{\mathrm{L}} = \frac{\xi\,(\alpha_\ell)_{\mathrm{L}}\,(\rho_\ell)_3}{\xi\,(\alpha_\ell)_{\mathrm{L}}\,(\rho_\ell)_3 + (1-\xi)\,(\alpha_g)_{\mathrm{L}}\,(\rho_g)_3} \tag{5.83}$$

and

$$(\beta)_{\mathrm{R}} = \frac{\xi\,(\alpha_\ell)_{\mathrm{R}}\,(\rho_\ell)_4}{\xi\,(\alpha_\ell)_{\mathrm{R}}\,(\rho_\ell)_4 + (1-\xi)\,(\alpha_g)_{\mathrm{R}}\,(\rho_g)_4}. \tag{5.84}$$

The validity of the other inequalities can be controlled by the constant parameters a_ℓ and a_g. We proceed a *predictor-corrector method* in three steps:

(1) At first, we can ensure that the phasic inequalities

$$(v_k)_{\mathrm{L}} - \frac{a_k}{(\rho_k)_{\mathrm{L}}\sqrt{(\alpha_k)_{\mathrm{L}}}} \overset{!}{<} (v_k)_\star \overset{!}{<} (v_k)_{\mathrm{R}} + \frac{a_k}{(\rho_k)_{\mathrm{R}}\sqrt{(\alpha_k)_{\mathrm{R}}}}$$

hold by choosing a_k large enough with

$$a_k > \max\left\{ \left(\sqrt{\alpha_k}\rho_k c_k\right)_{\mathrm{L}}, \ \left(\sqrt{\alpha_k}\rho_k c_k\right)_{\mathrm{R}}, \ a_k^\star, \ a_k^{\star\star} \right\}, \tag{5.85}$$

where the first condition comes from the sub-characteristic condition (5.71), and the last two from the inequalities (remember that $(v_k)_\star$ also depends on a_k). These bounds are given by

$$
\begin{aligned}
a_k^\star =& \frac{-\left(\alpha_k\right)_{\mathrm{L}}\left(\rho_k\right)_{\mathrm{L}}\left(\left(v_k\right)_{\mathrm{R}}-\left(v_k\right)_{\mathrm{L}}\right)}{2\left(\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}\right)} \\
&+\frac{\sqrt{\left(\alpha_k\right)_{\mathrm{L}}^2\left(\rho_k\right)_{\mathrm{L}}^2\left(\left(v_k\right)_{\mathrm{R}}-\left(v_k\right)_{\mathrm{L}}\right)^2+4\left(\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}\right)\left(\alpha_k\right)_{\mathrm{L}}\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}\left(\rho_k\right)_{\mathrm{L}}\left(\left(\pi_k\right)_{\mathrm{R}}-\left(\pi_k\right)_{\mathrm{L}}\right)}}{2\left(\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}\right)}, \\
a_k^{\star\star} =& \frac{-\left(\alpha_k\right)_{\mathrm{R}}\left(\rho_k\right)_{\mathrm{R}}\left(\left(v_k\right)_{\mathrm{R}}-\left(v_k\right)_{\mathrm{L}}\right)}{2\left(\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}\right)} \\
&+\frac{\sqrt{\left(\alpha_k\right)_{\mathrm{R}}^2\left(\rho_k\right)_{\mathrm{R}}^2\left(\left(v_k\right)_{\mathrm{R}}-\left(v_k\right)_{\mathrm{L}}\right)^2-4\left(\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}\right)\left(\alpha_k\right)_{\mathrm{R}}\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}\left(\rho_k\right)_{\mathrm{R}}\left(\left(\pi_k\right)_{\mathrm{R}}-\left(\pi_k\right)_{\mathrm{L}}\right)}}{2\left(\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}\right)},
\end{aligned}
\tag{5.86}
$$

if the radicands are positive, otherwise $a_k^\star$ and $a_k^{\star\star}$ are zero, respectively. The predicted quantities are used now to bound the intermediate velocity by

$$
\left(v_k\right)_{\mathrm{L}}-\frac{\max\left\{\sqrt{\alpha_k}\rho_k c_k,\ a_k^\star,\ a_k^{\star\star}\right\}}{\left(\rho_k\right)_{\mathrm{L}}\sqrt{\left(\alpha_k\right)_{\mathrm{L}}}}<\left(v_k\right)_\star<\left(v_k\right)_{\mathrm{R}}+\frac{\max\left\{\sqrt{\alpha_k}\rho_k c_k,\ a_k^\star,\ a_k^{\star\star}\right\}}{\left(\rho_k\right)_{\mathrm{R}}\sqrt{\left(\alpha_k\right)_{\mathrm{R}}}}.
$$

(2) Now, we need to regard the coupling of the phasic inequalities. To ensure that the intermediate liquid velocity is in-between the slowest and fastest steam wave, $\left(v_g\right)_{\mathrm{L}}-\frac{a_g}{\left(\rho_g\right)_{\mathrm{L}}\sqrt{\left(\alpha_g\right)_{\mathrm{L}}}}<\left(v_\ell\right)_\star<\left(v_g\right)_{\mathrm{R}}+\frac{a_g}{\left(\rho_g\right)_{\mathrm{R}}\sqrt{\left(\alpha_g\right)_{\mathrm{R}}}}$, we bound $\left(v_\ell\right)_\star$ from (5.77) by

$$
\left(v_\ell\right)_\star^-<\left(v_\ell\right)_\star<\left(v_\ell\right)_\star^+ \tag{5.87}
$$

with

$$
\left(v_\ell\right)_\star^\pm=\frac{\left(v_\ell\right)_{\mathrm{R}}\sqrt{\left(\alpha_\ell\right)_{\mathrm{L}}}+\left(v_\ell\right)_{\mathrm{L}}\sqrt{\left(\alpha_\ell\right)_{\mathrm{R}}}}{\sqrt{\left(\alpha_\ell\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_\ell\right)_{\mathrm{R}}}}\pm\left|\frac{\sqrt{\left(\alpha_\ell\right)_{\mathrm{L}}}\sqrt{\left(\alpha_\ell\right)_{\mathrm{R}}}\left(\left(\pi_\ell\right)_{\mathrm{L}}-\left(\pi_\ell\right)_{\mathrm{R}}\right)}{\max\left\{\sqrt{\alpha_\ell}\rho_\ell c_\ell,\ a_\ell^\star,\ a_\ell^{\star\star}\right\}\left(\sqrt{\left(\alpha_\ell\right)_{\mathrm{L}}}+\sqrt{\left(\alpha_\ell\right)_{\mathrm{R}}}\right)}\right|. \tag{5.88}
$$

These bounds also hold if a_ℓ is again increasing.

$$
\begin{aligned}
a_g>a_g^{\star\star\star}&:=\left(\left(v_g\right)_{\mathrm{L}}-\left(v_\ell\right)_\star^-\right)\left(\rho_g\right)_{\mathrm{L}}\sqrt{\left(\alpha_g\right)_{\mathrm{L}}}, \\
a_g>a_g^{\star\star\star\star}&:=\left(\left(v_\ell\right)_\star^+-\left(v_g\right)_{\mathrm{R}}\right)\left(\rho_g\right)_{\mathrm{R}}\sqrt{\left(\alpha_g\right)_{\mathrm{R}}}.
\end{aligned}
\tag{5.89}
$$

(3) Finally we adjust a_ℓ again, such that the overall slowest and fastest wave is given by the the liquid phase. This means, we require

$$
\left(v_\ell\right)_{\mathrm{L}}-\frac{a_\ell}{\left(\rho_\ell\right)_{\mathrm{L}}\sqrt{\left(\alpha_\ell\right)_{\mathrm{L}}}}<\left(v_g\right)_{\mathrm{L}}-\frac{a_g}{\left(\rho_g\right)_{\mathrm{L}}\sqrt{\left(\alpha_g\right)_{\mathrm{L}}}}
$$

$$(v_g)_{\mathrm{R}} + \frac{a_g}{(\rho_g)_{\mathrm{R}}\sqrt{(\alpha_g)_{\mathrm{R}}}} < (v_\ell)_{\mathrm{R}} + \frac{a_\ell}{(\rho_\ell)_{\mathrm{R}}\sqrt{(\alpha_\ell)_{\mathrm{R}}}}. \tag{5.90}$$

such that

$$\begin{aligned} a_\ell > a_\ell^{\star\star\star} &:= \sqrt{(\alpha_\ell)_{\mathrm{L}}}\,(\rho_\ell)_{\mathrm{L}}\left(\frac{a_g}{\sqrt{(\alpha_g)_{\mathrm{L}}}\,(\rho_g)_{\mathrm{L}}} - (v_g)_{\mathrm{L}} + (v_\ell)_{\mathrm{L}}\right), \\ a_\ell > a_\ell^{\star\star\star\star} &:= \sqrt{(\alpha_\ell)_{\mathrm{R}}}\,(\rho_\ell)_{\mathrm{R}}\left(\frac{a_g}{\sqrt{(\alpha_g)_{\mathrm{R}}}\,(\rho_g)_{\mathrm{R}}} + (v_g)_{\mathrm{R}} - (v_\ell)_{\mathrm{R}}\right), \end{aligned} \tag{5.91}$$

Because both parameters a_ℓ and a_g are known, finally the order of the intermediate velocities $(v_\ell)_\star$ and $(v_g)_\star$ can be decided.

It remains to show the positivity of the intermediate densities $(\rho_k)_2, (\rho_k)_3, (\rho_k)_4$ and $(\rho_k)_5$, which can be derived from the order of the eigenvalues (5.81):

- From the inequalities

$$\lambda_1(\mathbf{u}_1) < \lambda_3(\mathbf{u}_3) < \lambda_5(\mathbf{u}_4) < \lambda_7(\mathbf{u}_6) \quad \text{and} \quad \lambda_2(\mathbf{u}_2) < \lambda_3(\mathbf{u}_3) < \lambda_5(\mathbf{u}_4) < \lambda_6(\mathbf{u}_5) \tag{5.92}$$

  we get the relation

$$(v_k)_\star - \frac{a_k}{(\rho_k)_2\sqrt{(\alpha_k)_{\mathrm{L}}}} < (v_k)_\star < (v_k)_\star + \frac{a_k}{(\rho_k)_5\sqrt{(\alpha_k)_{\mathrm{L}}}} \qquad a \tag{5.93}$$

  such that with a positive constant parameter a_k in (5.85) the positivity for $(\rho_k)_2$ and $(\rho_k)_5$ is given.

- To show the positivity of $(\rho_k)_3$ and $(\rho_k)_4$, we regard at first the case of $(v_\ell)_\star < (v_g)_\star$. Then the positivity of $(\rho_g)_3$ and $(\rho_\ell)_4$ is directly given by (5.79), because $(\rho_g)_3 = (\rho_g)_2$ and $(\rho_\ell)_4 = (\rho_\ell)_5$. From the inequalities

$$\begin{aligned} \lambda_3(\mathbf{u}_3) < \lambda_4(\mathbf{u}_3) < \lambda_5(\mathbf{u}_4) \\ \lambda_3(\mathbf{u}_3) < \lambda_4(\mathbf{u}_4) < \lambda_5(\mathbf{u}_4) \end{aligned} \tag{5.94}$$

  we get with (5.82) the relations

$$\begin{aligned} (v_\ell)_\star < (\beta)_{\mathrm{L}}\,(v_\ell)_\star + \big(1-(\beta)_{\mathrm{L}}\big)\,(v_g)_\star < (v_g)_\star \\ (v_\ell)_\star < (\beta)_{\mathrm{R}}\,(v_\ell)_\star + \big(1-(\beta)_{\mathrm{R}}\big)\,(v_g)_\star < (v_g)_\star \end{aligned} \tag{5.95}$$

  which holds for $(\beta)_{\mathrm{L}}, (\beta)_{\mathrm{R}} \in (0,1)$. For some $w, \tilde{w} \in (0,1)$ the intermediate densities can be derived from the relation $(\beta)_{\mathrm{L}} \overset{!}{=} w$ and $(\beta)_{\mathrm{R}} \overset{!}{=} \tilde{w}$,

$$(\rho_\ell)_3 = \frac{w(1-\xi)\,(\alpha_g)_{\mathrm{L}}}{(1-w)\xi\,(\alpha_\ell)_{\mathrm{L}}}\,(\rho_g)_3\,,$$

$$(\rho_g)_4 = \frac{(1-\tilde{w})\xi\,(\alpha_\ell)_{\mathrm{R}}}{\tilde{w}(1-\xi)\,(\alpha_g)_{\mathrm{R}}}\,(\rho_\ell)_4\,, \tag{5.96}$$

such that also the positivity of these densities is shown.

For the case of $(v_g)_\star < (v_\ell)_\star$, the positivity of $(\rho_k)_3$ and $(\rho_k)_4$ can be shown in the same way.

To solve now the convective part (5.2) by evolution of time, the Godunov method can be used, see Section 3. For low-Mach applications (as the underlying application) the first two eigenvalues λ_1 and λ_2 are always negative and the last two eigenvalues λ_6 and λ_7 are always positive, such that the intermediate states $\mathbf{u}_1$ and $\mathbf{u}_6$ are never reached. This means that related to (5.5) the Riemann solution is given by

$$\mathbf{u}_{\mathrm{r}}(0) = \begin{cases} \mathbf{u}_2 & , \text{ if } \quad 0 < \min\left((v_\ell)_\star\,,(v_g)_\star\right), \\ \mathbf{u}_3 & , \text{ if } \quad \min\left((v_\ell)_\star\,,(v_g)_\star\right) < 0 < (v_{\mathrm{i}})_\star\,, \\ \mathbf{u}_4 & , \text{ if } \quad (v_{\mathrm{i}})_\star < 0 < \max\left((v_\ell)_\star\,,(v_g)_\star\right), \\ \mathbf{u}_5 & , \text{ if } \quad \max\left((v_\ell)_\star\,,(v_g)_\star\right) < 0. \end{cases} \tag{5.97}$$

We need to ensure that the local solutions of neighboring cells do not interact during the time period Δt. This gives us the classical CFL condition

$$\frac{\Delta t}{\Delta x}\,\max_i\,|\lambda_j(\mathbf{u}_i^n)| < \frac{1}{2}, \quad j \in \{0,\ldots,\tilde{n}\}. \tag{5.98}$$

Observe that under this CFL condition the constant parameters a_ℓ and a_g can be chosen locally at each cell [123].

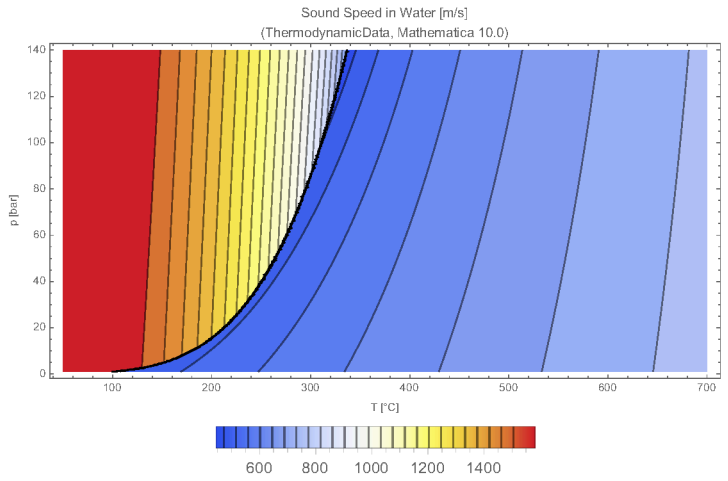


Figure 38: Speed of sound in water, using the `ThermodynamicData` function in Mathematica 10.3. The black boiling curve separates liquid water (above) and steam (below).

The solution of the Riemann problem is then given by (5.5). To solve now the convective part (5.2) by evolution of time, the Godunov method can be used, see Section 3. The arising solution of this step is denoted with $\mathbf{u}_i^{n+1,-}$.

▶ Source terms at instantaneous relaxation ($t^{n+1,-} \to t^{n+1}$)

In the second step the relaxation source terms are considered for an infinite relaxation time, see (5.7), with $\mathbf{u}_i^{n+1,-}$ as initial data. This ordinary differential equation system can be solved exactly. By the form of the relaxation source terms $\mathbf{s}_{\mathrm{r}}(\mathbf{u})$, just $\alpha\rho_k\pi_k$ evolves according to the ordinary differential equation.

For the *relaxation two-velocity two-pressure model* we get

$$\mathbf{u}_i^{n+1} = \begin{pmatrix} (\alpha_g)_i^{n+1} \\ (\alpha_\ell\rho_\ell)_i^{n+1} \\ (\alpha_\ell\rho_\ell v_\ell)_i^{n+1} \\ (\alpha_\ell\rho_\ell E_\ell)_i^{n+1} \\ (\alpha_\ell\rho_\ell\pi_\ell)_i^{n+1} \\ (\alpha_g\rho_g)_i^{n+1} \\ (\alpha_g\rho_g v_g)_i^{n+1} \\ (\alpha_g\rho_g E_g)_i^{n+1} \\ (\alpha_g\rho_g\pi_g)_i^{n+1} \end{pmatrix} = \begin{pmatrix} (\alpha_g)_i^{n+1,-} \\ (\alpha_\ell\rho_\ell)_i^{n+1,-} \\ (\alpha_\ell\rho_\ell v_\ell)_i^{n+1,-} \\ (\alpha_\ell\rho_\ell E_\ell)_i^{n+1,-} \\ \left(\alpha_\ell\rho_\ell p_\ell + \exp\left(-\frac{\Delta t}{\varepsilon_\ell}\right)(\alpha_\ell\rho_\ell\pi_\ell - \alpha_\ell\rho_\ell p_\ell)\right)_i^{n+1,-} \\ (\alpha_g\rho_g)_i^{n+1,-} \\ (\alpha_g\rho_g v_g)_i^{n+1,-} \\ (\alpha_g\rho_g E_g)_i^{n+1,-} \\ \left(\alpha_g\rho_g p_g + \exp\left(-\frac{\Delta t}{\varepsilon_g}\right)(\alpha_g\rho_g\pi_g - \alpha_g\rho_g p_g)\right)_i^{n+1,-} \end{pmatrix}. \quad (5.99)$$

6. Semi-Implicit solver

In Section 3 and 5 path-conservative schemes and a relaxation scheme have been presented for the two-phase flow model. Both approaches have the drawback that the acoustic waves determine the step size of the simulation time due to the CFL condition. To avoid this disadvantage, implicit schemes can be used for time integration. For a fully implicit scheme the nonlinear closures of large models are hard to solve. Iterative procedures like the Newton method are usually required such that high computational costs arise. A compromise between explicit and implicit solvers is the class of semi-implicit solvers, as proposed by Coquel et al. [129]: A time implicit-explicit Lagrange projection strategy is used to decouple the fast acoustic waves and the slow material waves. With a time-implicit treatment of the fast acoustic waves, we remove a too restrictive CFL condition. Additionally, an explicit treatment of the slow contact waves is provided in order to preserve accuracy.

6.1. Semi-Implicit solver for the homogeneous equilibrium model

The *relaxation homogeneous equilibrium model* can be understood as quasi-classical gas dynamics system and is given in (5.14),

$$\begin{aligned}
\partial_t \rho_\mathrm{h} + \partial_x(\rho_\mathrm{h} v_\mathrm{h}) &= 0,\\
\partial_t(\rho_\mathrm{h} v_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h}^2 + \pi_\mathrm{h}) &= 0,\\
\partial_t(\rho_\mathrm{h} E_\mathrm{h}) + \partial_x\big((\rho_\mathrm{h} E_\mathrm{h} + \pi_\mathrm{h}) v_\mathrm{h}\big) &= 0,\\
\partial_t(\rho_\mathrm{h} \pi_\mathrm{h}) + \partial_x(\rho_\mathrm{h} v_\mathrm{h} \pi_\mathrm{h}) + a_\mathrm{h}^2\, \partial_x v_\mathrm{h} &= \rho_\mathrm{h} \tfrac{1}{\varepsilon_\mathrm{h}}(p_\mathrm{h} - \pi_\mathrm{h}).
\end{aligned}$$

The constant parameter a_h is chosen in (5.30), such that the sub-characteristic condition is fulfilled, $a_\mathrm{h} \geq \rho_\mathrm{h} c_\mathrm{h}$. This system can be split into two subsystems describing acoustic effects,

$$\begin{aligned}
\partial_t \rho_\mathrm{h} + \rho_\mathrm{h}\, \partial_x v_\mathrm{h} &= 0,\\
\partial_t(\rho_\mathrm{h} v_\mathrm{h}) + \rho_\mathrm{h} v_\mathrm{h}\, \partial_x v_\mathrm{h} + \partial_x\, \pi_\mathrm{h} &= 0,\\
\partial_t(\rho_\mathrm{h} E_\mathrm{h}) + \rho_\mathrm{h} E_\mathrm{h}\, \partial_x v_\mathrm{h} + \partial_x(v_\mathrm{h} \pi_\mathrm{h}) &= 0,\\
\partial_t(\rho_\mathrm{h} \pi_\mathrm{h}) + \rho_\mathrm{h} \pi_\mathrm{h}\, \partial_x v_\mathrm{h} + a_\mathrm{h}^2\, \partial_x v_\mathrm{h} &= \rho_\mathrm{h} \tfrac{1}{\varepsilon_\mathrm{h}}(p_\mathrm{h} - \pi_\mathrm{h}).
\end{aligned} \tag{6.1}$$

and transport processes,

$$\begin{aligned}
\partial_t \rho_\mathrm{h} + v_\mathrm{h}\, \partial_x \rho_\mathrm{h} &= 0,\\
\partial_t(\rho_\mathrm{h} v_\mathrm{h}) + v_\mathrm{h}\, \partial_x(\rho_\mathrm{h} v_\mathrm{h}) &= 0,\\
\partial_t(\rho_\mathrm{h} E_\mathrm{h}) + v_\mathrm{h}\, \partial_x(\rho_\mathrm{h} E_\mathrm{h}) &= 0,\\
\partial_t(\rho_\mathrm{h} \pi_\mathrm{h}) + v_\mathrm{h}\, \partial_x(\rho_\mathrm{h} \pi_\mathrm{h}) &= 0.
\end{aligned} \tag{6.2}$$

We note that the conservative variables in the transport subsystem are transported with the velocity v_h, where the characteristic speeds of the acoustic subsystem appears at

sound speed. The whole system is now solved in a two step splitting strategy, where the acoustic subsystem (6.1) is solved implicitly in a first step and the transport subsystem (6.2) explicitly in the second step. Both steps will be solved over the same time interval Δt. In the following, numerical solvers for both subsystems are developed.

6.1.1. Properties of the acoustic subsystem

This system can be transformed into a Lagrangian system by using the specific volume $\tau_\mathrm{h} := \frac{1}{\rho_\mathrm{h}}$ [130], such that

$$\begin{aligned}
\partial_t \tau_\mathrm{h} - \tau_\mathrm{h}\, \partial_x v_\mathrm{h} &= 0,\\
\partial_t v_\mathrm{h} + \tau_\mathrm{h}\, \partial_x \pi_\mathrm{h} &= 0,\\
\partial_t E_\mathrm{h} + \tau_\mathrm{h}\, \partial_x (v_\mathrm{h}\pi_\mathrm{h}) &= 0,\\
\partial_t \pi_\mathrm{h} + a_\mathrm{h}^2 \tau_\mathrm{h}\, \partial_x v_\mathrm{h} &= \tfrac{1}{\varepsilon_\mathrm{h}}(p_\mathrm{h} - \pi_\mathrm{h}).
\end{aligned}$$

Now, a mass variable formulation $\mathrm{d}m = \frac{1}{\tau(x,t)}\mathrm{d}x$ is used. With this Lagrangian reformulation we obtain

$$\begin{aligned}
\partial_t \tau_\mathrm{h} - \partial_m v_\mathrm{h} &= 0,\\
\partial_t v_\mathrm{h} + \partial_m \pi_\mathrm{h} &= 0,\\
\partial_t E_\mathrm{h} + \partial_m (v_\mathrm{h}\pi_\mathrm{h}) &= 0,\\
\partial_t \pi_\mathrm{h} + a_\mathrm{h}^2\, \partial_m v_\mathrm{h} &= \tfrac{1}{\varepsilon_\mathrm{h}}(p_\mathrm{h} - \pi_\mathrm{h}).
\end{aligned} \tag{6.3}$$

The acoustic subsystem (6.3) can be expressed in the following fully conservative form

$$\partial_t \mathbf{u} + \partial_m \mathbf{f}(\mathbf{u}) = \mathbf{s}_\mathrm{r}(\mathbf{u}), \tag{6.4}$$

where $\mathbf{u} = (\tau_\mathrm{h}, v_\mathrm{h}, E_\mathrm{h}, \pi_\mathrm{h})^\mathsf{T}$ and $\mathbf{f}(\mathbf{u}) = (-v_\mathrm{h}, \pi_\mathrm{h}, v_\mathrm{h}\pi_\mathrm{h}, a_\mathrm{h}^2 v_\mathrm{h})^\mathsf{T}$. This subsystem is hyperbolic with $\tilde{n} = 3$ distinct eigenvalues $\pm a_\mathrm{h}$ and 0 (double) and right eigenvectors (as column vectors)

$$R^\mathsf{T} = \begin{pmatrix}
-\frac{1}{a_\mathrm{h}^2} & 1 & 0 & -\frac{1}{a_\mathrm{h}^2}\\
-\frac{1}{a_\mathrm{h}} & 0 & 0 & \frac{1}{a_\mathrm{h}}\\
\frac{\pi_\mathrm{h} - a_\mathrm{h} v_\mathrm{h}}{a_\mathrm{h}^2} & 0 & 1 & \frac{\pi_\mathrm{h} + a_\mathrm{h} v_\mathrm{h}}{a_\mathrm{h}^2}\\
1 & 0 & 0 & 1
\end{pmatrix}. \tag{6.5}$$

The natural order of the waves is given by

$$\lambda_1 := -a_\mathrm{h} \quad < \quad \lambda_{2,3} := 0 \quad < \quad \lambda_4 := a_\mathrm{h}, \tag{6.6}$$

which is always fulfilled for a positive a_h. It is obvious that the characteristic fields of the subsystem are linearly degenerate. The j-Riemann invariants $\mathbf{w}^{(j)}$ of the linearly degenerate j-characteristic fields are given by

$$\begin{aligned}
\mathbf{w}^{(1)} &= \left\{v_\mathrm{h} - a_\mathrm{h}\tau_\mathrm{h}, \pi_\mathrm{h} + v_\mathrm{h} a_\mathrm{h}, v_\mathrm{h}\pi_\mathrm{h} + E_\mathrm{h} a_\mathrm{h}\right\},\\
\mathbf{w}^{(2,3)} &= \Big\{v_\mathrm{h}, p_\mathrm{h}\Big\},\\
\mathbf{w}^{(4)} &= \left\{v_\mathrm{h} + a_\mathrm{h}\tau_\mathrm{h}, \pi_\mathrm{h} - v_\mathrm{h} a_\mathrm{h}, v_\mathrm{h}\pi_\mathrm{h} - E_\mathrm{h} a_\mathrm{h}\right\}.
\end{aligned} \tag{6.7}$$

6.1.2. Approximate Riemann solver of the acoustic subsystem ($t^n \to t^{n+1,-} \to t^{n+1,=}$)

To solve the acoustic subsystem, we proceed in the same way as we did for the *relaxation homogeneous equilibrium model* in Section 5.2. Related to the Cauchy problem for time $t \in (t^n, t^{n+1,=})$, again the two-step splitting strategy of Jin and Xin [122] is used to approximate the solution at time $t_{n+1,=}$: At first the convective part is solved (such that the solution at time $t_{n+1,-}$ is provided), and then the instantaneous relaxation as projection on the equilibrium manifold is solved (solution at time $t_{n+1,=}$).

For the evolution of time of the convective part the Godunov method can be used. With given left $\mathbf{u}_\mathrm{L}$ and right state $\mathbf{u}_\mathrm{R}$ the solution of the Riemann problem is defined as

$$\mathbf{u}(m,t) = \mathbf{u}_\mathrm{r}(\tfrac{m}{t}; \mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) = \begin{cases} \mathbf{u}_\mathrm{L}, & \frac{m}{t} < -a_\mathrm{h} \\ \mathbf{u}_1, & -a_\mathrm{h} < \frac{m}{t} < 0 \\ \mathbf{u}_2, & 0 < \frac{m}{t} < a_\mathrm{h} \\ \mathbf{u}_\mathrm{R}, & a_\mathrm{h} < \frac{m}{t} \end{cases} . \tag{6.8}$$

The Riemann solution consists of three contact discontinuities propagating with the speed of the eigenvalues, which separate two intermediate states $\mathbf{u}_1$ and $\mathbf{u}_2$. These states are obtained by solving the equation system of the Rankine-Hugoniot jump relations across the discontinuities:

$$\begin{aligned} (v_\mathrm{h} - a_\mathrm{h}\tau_\mathrm{h})_\mathrm{L} &\stackrel{!}{=} (v_\mathrm{h} - a_\mathrm{h}\tau_\mathrm{h})_1 , \\ (\pi_\mathrm{h} + v_\mathrm{h}a_\mathrm{h})_\mathrm{L} &\stackrel{!}{=} (\pi_\mathrm{h} + v_\mathrm{h}a_\mathrm{h})_1 , \\ (v_\mathrm{h}\pi_\mathrm{h} + E_\mathrm{h}a_\mathrm{h})_\mathrm{L} &\stackrel{!}{=} (v_\mathrm{h}\pi_\mathrm{h} + E_\mathrm{h}a_\mathrm{h})_1 , \\ \\ (v_\mathrm{h})_1 &\stackrel{!}{=} (v_\mathrm{h})_2 , \\ (\pi_\mathrm{h})_1 &\stackrel{!}{=} (\pi_\mathrm{h})_2 , \\ \\ (v_\mathrm{h} + a_\mathrm{h}\tau_\mathrm{h})_2 &\stackrel{!}{=} (v_\mathrm{h} + a_\mathrm{h}\tau_\mathrm{h})_\mathrm{R} , \\ (\pi_\mathrm{h} - v_\mathrm{h}a_\mathrm{h})_2 &\stackrel{!}{=} (\pi_\mathrm{h} - v_\mathrm{h}a_\mathrm{h})_\mathrm{R} , \\ (v_\mathrm{h}\pi_\mathrm{h} + E_\mathrm{h}a_\mathrm{h})_2 &\stackrel{!}{=} (v_\mathrm{h}\pi_\mathrm{h} + E_\mathrm{h}a_\mathrm{h})_\mathrm{R} . \end{aligned} \tag{6.9}$$

Solving for the parameters of the unknown intermediate states, a solution in dependency of the known left and right state is obtained:

$$(v_\mathrm{h})_\star := (v_\mathrm{h})_1 = (v_\mathrm{h})_2 = \frac{(v_\mathrm{h})_\mathrm{R} + (v_\mathrm{h})_\mathrm{L}}{2} - \frac{(\pi_\mathrm{h})_\mathrm{R} - (\pi_\mathrm{h})_\mathrm{L}}{2a_\mathrm{h}},$$

$$(\pi_\mathrm{h})_\star := (\pi_\mathrm{h})_1 = (\pi_\mathrm{h})_2 = \frac{(\pi_\mathrm{h})_\mathrm{R} + (\pi_\mathrm{h})_\mathrm{L}}{2} - a_\mathrm{h}\frac{(v_\mathrm{h})_\mathrm{R} - (v_\mathrm{h})_\mathrm{L}}{2},$$

$$
\begin{aligned}
(\tau_{\mathrm{h}})_1 &= (\tau_{\mathrm{h}})_{\mathrm{L}} + \frac{(v_{\mathrm{h}})_\star - (v_{\mathrm{h}})_{\mathrm{L}}}{a_{\mathrm{h}}},\\
(\tau_{\mathrm{h}})_2 &= (\tau_{\mathrm{h}})_{\mathrm{R}} + \frac{(v_{\mathrm{h}})_{\mathrm{R}} - (v_{\mathrm{h}})_\star}{a_{\mathrm{h}}},\\
(E_{\mathrm{h}})_1 &= (E_{\mathrm{h}})_{\mathrm{L}} + \frac{(v_{\mathrm{h}})_{\mathrm{L}}\,(\pi_{\mathrm{h}})_{\mathrm{L}} - (v_{\mathrm{h}})_\star\,(\pi_{\mathrm{h}})_\star}{a_{\mathrm{h}}},\\
(E_{\mathrm{h}})_2 &= (E_{\mathrm{h}})_{\mathrm{R}} - \frac{(v_{\mathrm{h}})_{\mathrm{R}}\,(\pi_{\mathrm{h}})_{\mathrm{R}} - (v_{\mathrm{h}})_\star\,(\pi_{\mathrm{h}})_\star}{a_{\mathrm{h}}}.
\end{aligned}
\tag{6.10}
$$

▶ Godunov method with explicit operator splitting ($t^n \to t^{n+1,-}$)

The finite volume method considers a time step Δt and a spatial step, which is expressed through the mass variable $\Delta m_i = \frac{1}{\tau_i^n}\Delta x$. So, in analogy to (3.13), the update formulation is given by

$$
\begin{aligned}
\mathbf{u}_i^{n+1,-} &= \mathbf{u}_i^n - \frac{\Delta t}{\Delta m_i}\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big)\\
&= \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big).
\end{aligned}
\tag{6.11}
$$

Following the Godunov method in Section 3, the flux vector $\mathbf{F}_{i-1/2}^n := \mathbf{F}(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n)$ is given in dependency of the Riemann solution $\mathbf{u}_{\mathrm{r}}^0 := \mathbf{u}_{\mathrm{r}}(0; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})$,

$$
\mathbf{F}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) \;=\; \mathbf{f}\big(\mathbf{u}_{\mathrm{r}}^0\big) \underset{(6.10)}{\overset{(6.8)}{=}} \begin{pmatrix} -\,(v_{\mathrm{h}})_\star \\ (\pi_{\mathrm{h}})_\star \\ (v_{\mathrm{h}})_\star\,(\pi_{\mathrm{h}})_\star \\ a_{\mathrm{h}}^2\,(v_{\mathrm{h}})_\star \end{pmatrix}.
\tag{6.12}
$$

We need to ensure that the local solutions of neighboring cells do not interact during the time period Δt. This gives us the classical CFL condition

$$
\frac{\Delta t}{\Delta x}\,\max_i |\lambda_j(\mathbf{u}_i^n)| < \frac{1}{2}, \quad j \in \{0, \dots, n\}.
\tag{6.13}
$$

$$
\max_i \left(\frac{\Delta t}{\Delta m_i}\,a_{\mathrm{h}}\right) = \max_i \left(\frac{\Delta t}{\Delta x}\,\tau_i^n\,a_{\mathrm{h}}\right) < \frac{1}{2},
\tag{6.14}
$$

where the constant parameter a_{h} can be chosen locally at each cell (such that locally the sub-characteristic condition is fulfilled).

▶ Godunov method with implicit operator splitting ($t^n \to t^{n+1,-}$)

For an implicit representation of the Godunov method, the update formulation is given by

$$
\mathbf{u}_i^{n+1,-} = \mathbf{u}_i^n - \frac{\Delta t}{\Delta m_i}\Big(\mathbf{F}_{i+1/2}^{n+1,-} - \mathbf{F}_{i-1/2}^{n+1,-}\Big)
$$

$$= \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left(\mathbf{F}_{i+1/2}^{n+1,-} - \mathbf{F}_{i-1/2}^{n+1,-}\right), \tag{6.15}$$

where the flux vectors at the new time step are used,

$$\mathbf{F}_{i-1/2}^{n+1,-} = \mathbf{F}(\mathbf{u}_{i-1}^{n+1,-}, \mathbf{u}_i^{n+1,-}) = \mathbf{f}\Big(\mathbf{u}_{\mathrm{r}}\left(0; \mathbf{u}_{i-1}^{n+1,-}, \mathbf{u}_i^{n+1,-}\right)\Big) \underset{(6.10)}{\overset{(6.8)}{=}} \begin{pmatrix} -(v_{\mathrm{h}})_\star \\ (\pi_{\mathrm{h}})_\star \\ (v_{\mathrm{h}})_\star\,(\pi_{\mathrm{h}})_\star \\ a_{\mathrm{h}}^2\,(v_{\mathrm{h}})_\star \end{pmatrix}_{i-1/2}^{n+1,-}. \tag{6.16}$$

From the implicit update formulation and the Riemann solution we can see that the two quantities v_{h} and π_{h} are independent from the other two quantities τ_{h} and E_{h}. Thus, at first the velocity and relaxation pressure at the next time step are computed from the relation (6.15) by solving a tridiagonal linear system:

$$\begin{aligned}
(v_{\mathrm{h}})_i^{n+1,-} &= (v_{\mathrm{h}})_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left((\pi_{\mathrm{h}})_{\star i+1/2}^{n+1,-} - (\pi_{\mathrm{h}})_{\star i-1/2}^{n+1,-}\right) \\
&\overset{(6.10)}{=} (v_{\mathrm{h}})_i^n - \frac{\Delta t}{\Delta x}\,\frac{\tau_i^n}{2} \Big((\pi_{\mathrm{h}})_{i+1}^{n+1,-} - (\pi_{\mathrm{h}})_{i-1}^{n+1,-} \\
&\qquad\qquad - a_{\mathrm{h}}(v_{\mathrm{h}})_{i+1}^{n+1,-} + 2a_{\mathrm{h}}(v_{\mathrm{h}})_i^{n+1,-} - a_{\mathrm{h}}(v_{\mathrm{h}})_{i-1}^{n+1,-}\Big),
\end{aligned}$$

$$\begin{aligned}
(\pi_{\mathrm{h}})_i^{n+1,-} &= (\pi_{\mathrm{h}})_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left(a_{\mathrm{h}}^2(v_{\mathrm{h}})_{\star i+1/2}^{n+1,-} - a_{\mathrm{h}}^2(v_{\mathrm{h}})_{\star i-1/2}^{n+1,-}\right) \\
&\overset{(6.10)}{=} (\pi_{\mathrm{h}})_i^n - \frac{\Delta t}{\Delta x}\,\frac{\tau_i^n}{2} \Big(a_{\mathrm{h}}^2(v_{\mathrm{h}})_{i+1}^{n+1,-} - a_{\mathrm{h}}^2(v_{\mathrm{h}})_{i-1}^{n+1,-} \\
&\qquad\qquad - a_{\mathrm{h}}(\pi_{\mathrm{h}})_{i+1}^{n+1,-} + 2a_{\mathrm{h}}(\pi_{\mathrm{h}})_i^{n+1,-} - a_{\mathrm{h}}(\pi_{\mathrm{h}})_{i-1}^{n+1,-}\Big).
\end{aligned} \tag{6.17}$$

This gives a linear system of two times the number of cells N. The matrix of the equation system is sparse with six entries around the diagonal.

With the knowledge of $(v_{\mathrm{h}})_i^{n+1,-}$ and $(\pi_{\mathrm{h}})_i^{n+1,-}$ also the intermediate states $(v_{\mathrm{h}})_{\star i\pm1/2}^{n+1,-}$ and $(\pi_{\mathrm{h}})_{\star i\pm1/2}^{n+1,-}$ at the next time step are known. Now, the specific volume τ_{h} and specific total energy E_{h} at the next time step can explicitly be computed from the implicit update formulation:

$$\begin{aligned}
(\tau_{\mathrm{h}})_i^{n+1,-} &= (\tau_{\mathrm{h}})_i^n + \frac{\Delta t}{\Delta x}\,\tau_i^n \left((v_{\mathrm{h}})_{\star i+1/2}^{n+1,-} - (v_{\mathrm{h}})_{\star i-1/2}^{n+1,-}\right) \\
(E_{\mathrm{h}})_i^{n+1,-} &= (E_{\mathrm{h}})_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left((v_{\mathrm{h}})_{\star i+1/2}^{n+1,-}(\pi_{\mathrm{h}})_{\star i+1/2}^{n+1,-} - (v_{\mathrm{h}})_{\star i-1/2}^{n+1,-}(\pi_{\mathrm{h}})_{\star i-1/2}^{n+1,-}\right).
\end{aligned} \tag{6.18}$$

▶ Instantaneous relaxation ($t^{n+1,-} \to t^{n+1,=}$)

In the second step the relaxation source terms are considered for an infinite relaxation time with $\mathbf{u}_i^{n+1,-}$ as initial data. This ordinary differential equation system can be solved

exactly. By the form of the relaxation source terms $\mathbf{s}_{\mathrm{r}}(\mathbf{u})$, just π evolves according to the ordinary differential equation, such that

$$\mathbf{u}_i^{n+1,=} = \begin{pmatrix} (\tau_{\mathrm{h}})_i^{n+1,=} \\ (v_{\mathrm{h}})_i^{n+1,=} \\ (E_{\mathrm{h}})_i^{n+1,=} \\ (\pi_{\mathrm{h}})_i^{n+1,=} \end{pmatrix} = \begin{pmatrix} (\tau_{\mathrm{h}})_i^{n+1,-} \\ (v_{\mathrm{h}})_i^{n+1,-} \\ (E_{\mathrm{h}})_i^{n+1,-} \\ \left(p_{\mathrm{h}} + \exp\left(-\frac{\Delta t}{\varepsilon_{\mathrm{h}}}\right)(\pi_{\mathrm{h}} - p_{\mathrm{h}})\right)_i^{n+1,-} \end{pmatrix}. \tag{6.19}$$

6.1.3. Upwind solver of the transport subsystem ($t^{n+1,=} \to t^{n+1}$)

In the last stage the transport subsystem (6.2) has to be solved. This equation system can be expressed in the form,

$$\partial_t \mathbf{u} + v_{\mathrm{h}}\, \partial_x \mathbf{u} = 0,$$

with the independent variables $\mathbf{u} := (\rho_{\mathrm{h}}, \rho_{\mathrm{h}} v_{\mathrm{h}}, \rho_{\mathrm{h}} E_{\mathrm{h}}, \rho_{\mathrm{h}} \pi_{\mathrm{h}})^{\mathsf{T}}$. This subsystem only involves the transport of the conservative variables with the velocity v_{h}, such that it is obvious that this subsystem is hyperbolic. To approximate the solution a standard upwind finite volume method can be used, [131],

$$\begin{aligned} \mathbf{u}_i^{n+1} = \mathbf{u}_i^{n+1,=} &- \frac{\Delta t}{\Delta x}\left((v_{\mathrm{h}})_{\star i+1/2}\mathbf{u}_{i+1/2}^{n+1,=} - (v_{\mathrm{h}})_{\star i-1/2}\mathbf{u}_{i-1/2}^{n+1,=}\right) \\ &+ \frac{\Delta t}{\Delta x}\mathbf{u}_i^{n+1,=}\left((v_{\mathrm{h}})_{\star i+1/2} - (v_{\mathrm{h}})_{\star i-1/2}\right), \end{aligned} \tag{6.20}$$

which depends on the direction of the flow at each interface,

$$\mathbf{u}_{i+1/2}^{n+1,=} = \begin{cases} \mathbf{u}_i^{n+1,=}, & \text{if } (v_{\mathrm{h}})_{\star i+1/2} \geq 0 \\ \mathbf{u}_{i+1}^{n+1,=}, & \text{if } (v_{\mathrm{h}})_{\star i+1/2} < 0, \end{cases} \tag{6.21}$$

where $(v_{\mathrm{h}})_{\star i+1/2}$ is the speed of the Riemann problem at the interface $x_{i+1/2}$, see (6.10).

6.2. Semi-Implicit solver for the two-velocity two-pressure model

The *relaxation two-phase flow model* is given in (5.49),

$$\begin{aligned}
\partial_t \alpha_k + v_\mathrm{i}\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k) + \partial_x(\alpha_k \rho_k v_k) &= 0,\\
\partial_t(\alpha_k \rho_k v_k) + \partial_x(\alpha_k \rho_k v_k^2 + \alpha_k \pi_k) - \pi_\mathrm{i}\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k E_k) + \partial_x\big(\alpha_k (\rho_k E_k + \pi_k) v_k\big) - \pi_\mathrm{i} v_\mathrm{i}\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k \pi_k) + \partial_x(\alpha_k \rho_k v_k \pi_k) + \alpha_k \rho_k^2 c_k^2\, \partial_x v_k &= \alpha_k \rho_k \tfrac{1}{\varepsilon_k}(p_k - \pi_k).
\end{aligned}$$

In the original equation system, the pressure balance law consists of constant parameters a_k^2 which replace the pre-factors $\alpha_k \rho_k^2 c_k^2$. This replacement is done in a later step. This system can be split into three subsystems describing acoustic effects,

$$\begin{aligned}
\partial_t \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k) + \alpha_k \rho_k\, \partial_x v_k &= 0,\\
\partial_t(\alpha_k \rho_k v_k) + \alpha_k \rho_k v_k\, \partial_x v_k + \alpha_k\, \partial_x \pi_k &= 0,\\
\partial_t(\alpha_k \rho_k E_k) + \alpha_k \rho_k E_k\, \partial_x v_k + \alpha_k\, \partial_x (v_k \pi_k) &= 0,\\
\partial_t(\alpha_k \rho_k \pi_k) + \alpha_k \rho_k \pi_k\, \partial_x v_k + \alpha_k \rho_k^2 c_k^2\, \partial_x v_k &= \alpha_k \rho_k \tfrac{1}{\varepsilon_k}(p_k - \pi_k).
\end{aligned} \tag{6.22}$$

transport processes,

$$\begin{aligned}
\partial_t \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k) + v_k\, \partial_x(\alpha_k \rho_k) &= 0,\\
\partial_t(\alpha_k \rho_k v_k) + v_k\, \partial_x(\alpha_k \rho_k v_k) &= 0,\\
\partial_t(\alpha_k \rho_k E_k) + v_k\, \partial_x(\alpha_k \rho_k E_k) &= 0,\\
\partial_t(\alpha_k \rho_k \pi_k) + v_k\, \partial_x(\alpha_k \rho_k \pi_k) &= 0.
\end{aligned} \tag{6.23}$$

and a genuinely nonconservative subsystem [132],

$$\begin{aligned}
\partial_t \alpha_k + v_\mathrm{i}\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k) &= 0,\\
\partial_t(\alpha_k \rho_k v_k) + (\pi_k - \pi_\mathrm{i})\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k E_k) + (\pi_k v_k - \pi_\mathrm{i} v_\mathrm{i})\, \partial_x \alpha_k &= 0,\\
\partial_t(\alpha_k \rho_k \pi_k) &= 0,
\end{aligned} \tag{6.24}$$

Similar to Section 6.1, the whole system is solved in a three step splitting strategy, where the acoustic subsystem (6.22) is solved implicitly in a first step, then the transport subsystem (6.23) explicitly, and in a last step the genuinely non-conservative subsystem. All steps will be solved over the same time interval Δt. In the following, numerical solvers for the three subsystems are developed.

6.2.1. Properties of the acoustic subsystem

The acoustic system (6.22) can be transformed into a Lagrangian system by using the specific volume $\tau_\mathrm{h} := \frac{1}{\rho_\mathrm{h}}$ [132], such that

$$\begin{aligned}
\partial_t \alpha_k &= 0,\\
\partial_t \tau_k - \tau_k\, \partial_x v_k &= 0,\\
\partial_t v_k + \tau_k\, \partial_x \pi_k &= 0,\\
\partial_t E_k + \tau_k\, \partial_x (v_k \pi_k) &= 0,\\
\partial_t \pi_k + a_k^2 \tau_k\, \partial_x v_k &= \tfrac{1}{\varepsilon_k}(p_k - \pi_k).
\end{aligned}$$

The constant parameters a_k is chosen such that the subcharacteristic condition is fulfilled, $a_k \geq \rho_k c_k$ [132]. Now, a mass variable formulation $\mathrm{d}m = \frac{1}{\tau(x,t^n)}\mathrm{d}x$ is used. With this reformulation we obtain

$$\begin{aligned}
\partial_t \alpha_k &= 0,\\
\partial_t \tau_k - \partial_m v_k &= 0,\\
\partial_t v_k + \partial_m \pi_k &= 0,\\
\partial_t E_k + \partial_m (v_k \pi_k) &= 0,\\
\partial_t \pi_k + a_k^2\, \partial_m v_k &= \tfrac{1}{\varepsilon_k}(p_k - \pi_k).
\end{aligned} \tag{6.25}$$

The acoustic subsystem (6.25) can be expressed in the following fully conservative form

$$\partial_t \mathbf{u} + \partial_m \mathbf{f}(\mathbf{u}) = \mathbf{s}_\mathrm{r}(\mathbf{u}), \tag{6.26}$$

with

$$\mathbf{u} = (\alpha, \tau_\ell, v_\ell, E_\ell, \pi_\ell, \tau_g, v_g, E_g, \pi_g)^\mathsf{T}$$

and

$$\mathbf{f}(\mathbf{u}) = (0, -v_\ell, \pi_\ell, v_\ell \pi_\ell, a_\ell^2 v_\ell, -v_g, \pi_g, v_g \pi_g, a_g^2 v_g)^\mathsf{T}.$$

This subsystem is hyperbolic with $\tilde{n} = 5$ distinct eigenvalues, $\pm a_k$ and 0 (fivefold). The natural order of the waves is given by

$$\lambda_1 := -a_\ell \quad < \quad \lambda_2 := -a_g \quad < \quad \lambda_3 := 0 \quad < \quad \lambda_4 := a_g \quad < \quad \lambda_5 := a_\ell, \tag{6.27}$$

which is always fulfilled for positive a_k. For the order of the liquid and steam eigenvalues, the reader is referred to (5.75). The corresponding right eigenvectors are given (as column vectors) by

$$R^\mathsf{T} = \begin{pmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
-\frac{1}{a_\ell^2} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -\frac{1}{a_\ell^2}\\
-\frac{1}{a_\ell} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{a_\ell}\\
\frac{\pi_\ell - a_\ell v_\ell}{a_\ell^2} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \frac{\pi_\ell + a_\ell v_\ell}{a_\ell^2}\\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\
0 & -\frac{1}{a_g^2} & 0 & 0 & 0 & 1 & 0 & -\frac{1}{a_g^2} & 0\\
0 & -\frac{1}{a_g} & 0 & 0 & 0 & 0 & 0 & \frac{1}{a_g} & 0\\
0 & \frac{\pi_g - a_g v_g}{a_g^2} & 0 & 0 & 0 & 0 & 1 & \frac{\pi_g + a_g v_g}{a_g^2} & 0\\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}, \tag{6.28}$$

The characteristic fields of the subsystem are linearly degenerate. The j-Riemann invariants $\mathbf{w}^{(j)}$ of the linearly degenerate j-characteristic fields are given by

$$\begin{aligned}
\mathbf{w}^{(1)} &= \{\alpha, v_\ell - a_\ell\tau_\ell, \pi_\ell + v_\ell a_\ell, v_\ell\pi_\ell + E_\ell a_\ell, \tau_g, v_g, E_g, \pi_g\}\,, \\
\mathbf{w}^{(2)} &= \{\alpha, \tau_\ell, v_\ell, E_\ell, \pi_\ell, v_g - a_g\tau_g, \pi_g + v_g a_g, v_g\pi_g + E_g a_g\}\,, \\
\mathbf{w}^{(3)} &= \Big\{v_\ell, \pi_\ell, v_g, \pi_g\Big\}, \\
\mathbf{w}^{(4)} &= \{\alpha, \tau_\ell, v_\ell, E_\ell, \pi_\ell, v_g + a_g\tau_g, \pi_g - v_g a_g, v_g\pi_g - E_g a_g\}\,, \\
\mathbf{w}^{(5)} &= \{\alpha, v_\ell + a_\ell\tau_\ell, \pi_\ell - v_\ell a_\ell, v_\ell\pi_\ell - E_\ell a_\ell, \tau_g, v_g, E_g, \pi_g\}\,.
\end{aligned} \tag{6.29}$$

6.2.2. Approximate Riemann solver of the acoustic subsystem ($t^n \to t^{n+1,-} \to t^{n+1,=}$)

To solve the acoustic subsystem, we proceed in the same way as we did for the *relaxation two-velocity two-pressure model* in Section 5.4. Related to the Cauchy problem for time $t \in (t^n, t^{n+1,=})$, again the two-step splitting strategy of Jin and Xin [122] is used to approximate the solution at time $t_{n+1,=}$: At first the convective part is solved (such that the solution at time $t_{n+1,-}$ is provided), and then the instantaneous relaxation as projection on the equilibrium manifold is solved (solution at time $t_{n+1,=}$).

For the evolution of time of the convective part the Godunov method can be used. With given left $\mathbf{u}_\mathrm{L}$ and right state $\mathbf{u}_\mathrm{R}$ the solution of the Riemann problem is defined as

$$\mathbf{u}(m,t) = \mathbf{u}_\mathrm{r}(\tfrac{m}{t}; \mathbf{u}_\mathrm{L}, \mathbf{u}_\mathrm{R}) = \begin{cases} \mathbf{u}_\mathrm{L}, & \frac{m}{t} < -a_\ell \\ \mathbf{u}_1, & -a_\ell < \frac{m}{t} < -a_g \\ \mathbf{u}_2, & -a_g < \frac{m}{t} < 0 \\ \mathbf{u}_3, & 0 < \frac{m}{t} < a_g \\ \mathbf{u}_4, & a_g < \frac{m}{t} < a_\ell \\ \mathbf{u}_\mathrm{R}, & a_\ell < \frac{m}{t} \end{cases}. \tag{6.30}$$

The Riemann solution consists of five contact discontinuities propagating with the speed of the eigenvalues, which separate four intermediate states $\mathbf{u}_{j=1,\dots,4}$. These states are obtained by solving the equation system of the Rankine-Hugoniot jump relations across the discontinuities, such that a solution in dependency of the known left $\mathbf{u}_\mathrm{L}$ and right

state $\mathbf{u}_{\mathrm{R}}$ is obtained:

$$\mathbf{u}_1 = \begin{pmatrix} (\alpha)_{\mathrm{L}} \\ (\tau_\ell)_1 \\ (v_\ell)_\star \\ (E_\ell)_1 \\ (\pi_\ell)_\star \\ (\tau_g)_{\mathrm{L}} \\ (v_g)_{\mathrm{L}} \\ (E_g)_{\mathrm{L}} \\ (\pi_g)_{\mathrm{L}} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} (\alpha)_{\mathrm{L}} \\ (\tau_\ell)_2 \\ (v_\ell)_\star \\ (E_\ell)_2 \\ (\pi_\ell)_\star \\ (\tau_g)_2 \\ (v_g)_\star \\ (E_g)_2 \\ (\pi_g)_\star \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} (\alpha)_{\mathrm{R}} \\ (\tau_\ell)_3 \\ (v_\ell)_\star \\ (E_\ell)_3 \\ (\pi_\ell)_\star \\ (\tau_g)_3 \\ (v_g)_\star \\ (E_g)_3 \\ (\pi_g)_\star \end{pmatrix}, \quad \mathbf{u}_4 = \begin{pmatrix} (\alpha)_{\mathrm{R}} \\ (\tau_\ell)_4 \\ (v_\ell)_\star \\ (E_\ell)_4 \\ (\pi_\ell)_\star \\ (\tau_g)_{\mathrm{R}} \\ (v_g)_{\mathrm{R}} \\ (E_g)_{\mathrm{R}} \\ (\pi_g)_{\mathrm{R}} \end{pmatrix}, \tag{6.31}$$

with velocity

$$(v_k)_\star = \frac{(v_k)_{\mathrm{R}} + (v_k)_{\mathrm{L}}}{2} - \frac{(\pi_k)_{\mathrm{R}} - (\pi_k)_{\mathrm{L}}}{2a_k}, \tag{6.32}$$

relaxed pressure

$$(\pi_k)_\star = \frac{(\pi_k)_{\mathrm{R}} + (\pi_k)_{\mathrm{L}}}{2} - a_k \frac{(v_k)_{\mathrm{R}} - (v_k)_{\mathrm{L}}}{2}, \tag{6.33}$$

specific volume

$$\begin{aligned} (\tau_\ell)_1 = (\tau_\ell)_2 &= (\tau_\ell)_{\mathrm{L}} + \frac{(v_\ell)_\star - (v_\ell)_{\mathrm{L}}}{a_\ell}, \\ (\tau_\ell)_3 = (\tau_\ell)_4 &= (\tau_\ell)_{\mathrm{R}} + \frac{(v_\ell)_{\mathrm{R}} - (v_\ell)_\star}{a_\ell}, \\ (\tau_g)_2 &= (\tau_g)_{\mathrm{L}} + \frac{(v_g)_\star - (v_g)_{\mathrm{L}}}{a_g}, \\ (\tau_g)_3 &= (\tau_g)_{\mathrm{R}} + \frac{(v_g)_{\mathrm{R}} - (v_g)_\star}{a_g}, \end{aligned} \tag{6.34}$$

and specific total energy

$$\begin{aligned} (E_\ell)_1 = (E_\ell)_2 &= (E_\ell)_{\mathrm{L}} + \frac{(v_\ell)_{\mathrm{L}} (\pi_\ell)_{\mathrm{L}} - (v_\ell)_\star (\pi_\ell)_\star}{a_\ell}, \\ (E_\ell)_3 = (E_\ell)_4 &= (E_\ell)_{\mathrm{R}} - \frac{(v_\ell)_{\mathrm{R}} (\pi_\ell)_{\mathrm{R}} - (v_\ell)_\star (\pi_\ell)_\star}{a_\ell} \\ (E_g)_2 &= (E_g)_{\mathrm{L}} + \frac{(v_g)_{\mathrm{L}} (\pi_g)_{\mathrm{L}} - (v_g)_\star (\pi_g)_\star}{a_g}, \\ (E_g)_3 &= (E_g)_{\mathrm{R}} - \frac{(v_g)_{\mathrm{R}} (\pi_g)_{\mathrm{R}} - (v_g)_\star (\pi_g)_\star}{a_g}. \end{aligned} \tag{6.35}$$

▶ Godunov method with explicit operator splitting ($t^n \to t^{n+1,-}$)

The finite volume method considers a time step Δt and a spatial step, which is expressed through the mass variable $\Delta m_i = \frac{1}{\tau_i^n}\Delta x$. So, in analogy to (3.13), the update formulation is given by

$$\begin{aligned}\mathbf{u}_i^{n+1,-} &= \mathbf{u}_i^n - \frac{\Delta t}{\Delta m_i}\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big) \\ &= \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n\Big(\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n\Big).\end{aligned} \tag{6.36}$$

Following the Godunov method in Section 3, the flux vector $\mathbf{F}_{i-1/2}^n := \mathbf{F}(\mathbf{u}_{i-1}^n, \mathbf{u}_i^n)$ is given in dependency of the Riemann solution $\mathbf{u}_{\mathrm{r}}^0 := \mathbf{u}_{\mathrm{r}}(0; \mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}})$,

$$\mathbf{f}(\mathbf{u}) = (0, -v_\ell, \pi_\ell, v_\ell\pi_\ell, a_\ell^2 v_\ell, -v_g, \pi_g, v_g\pi_g, a_g^2 v_g)^{\mathsf{T}}.$$

$$\mathbf{F}(\mathbf{u}_{\mathrm{L}}, \mathbf{u}_{\mathrm{R}}) \;=\; \mathbf{f}\big(\mathbf{u}_{\mathrm{r}}^0\big) \overset{(6.30)}{\underset{(6.31)}{=}} \begin{pmatrix} 0 \\ -(v_\ell)_\star \\ (\pi_\ell)_\star \\ (v_\ell)_\star\,(\pi_\ell)_\star \\ a_\ell^2\,(v_\ell)_\star \\ -(v_g)_\star \\ (\pi_g)_\star \\ (v_g)_\star\,(\pi_g)_\star \\ a_g^2\,(v_g)_\star \end{pmatrix}. \tag{6.37}$$

We need to ensure that the local solutions of neighboring cells do not interact during the time period Δt. This gives us the classical CFL condition

$$\frac{\Delta t}{\Delta x}\,\max_i |\lambda_j(\mathbf{u}_i^n)| < \frac{1}{2}, \quad j \in \{0, \ldots, \tilde{n}\}. \tag{6.38}$$

$$\max_i\left(\frac{\Delta t}{\Delta m_i}\,a_k\right) = \max_i\left(\frac{\Delta t}{\Delta x}\,\tau_i^n\,a_k\right) < \frac{1}{2}, \tag{6.39}$$

where the constant parameter a_k can be chosen locally at each cell (such that locally the sub-characteristic condition is fulfilled).

▶ Godunov method with implicit operator splitting ($t^n \to t^{n+1,-}$)

For an implicit representation of the Godunov method, the update formulation is given by

$$\begin{aligned}\mathbf{u}_i^{n+1,-} &= \mathbf{u}_i^n - \frac{\Delta t}{\Delta m_i}\Big(\mathbf{F}_{i+1/2}^{n+1,-} - \mathbf{F}_{i-1/2}^{n+1,-}\Big) \\ &= \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n\,\Big(\mathbf{F}_{i+1/2}^{n+1,-} - \mathbf{F}_{i-1/2}^{n+1,-}\Big),\end{aligned} \tag{6.40}$$

where the flux vectors at the new time step are used,

$$\mathbf{F}_{i-1/2}^{n+1,-} = \mathbf{F}(\mathbf{u}_{i-1}^{n+1,-}, \mathbf{u}_{i}^{n+1,-}) = \mathbf{f}\Big(\mathbf{u}_{\mathrm{r}}\left(0; \mathbf{u}_{i-1}^{n+1,-}, \mathbf{u}_{i}^{n+1,-}\right)\Big) \overset{(6.30)}{\underset{(6.31)}{=}} \begin{pmatrix} 0 \\ -(v_\ell)_\star \\ (\pi_\ell)_\star \\ (v_\ell)_\star (\pi_\ell)_\star \\ a_\ell^2 (v_\ell)_\star \\ -(v_g)_\star \\ (\pi_g)_\star \\ (v_g)_\star (\pi_g)_\star \\ a_g^2 (v_g)_\star \end{pmatrix}_{i-1/2}^{n+1,-} . \tag{6.41}$$

From the implicit update formulation and the Riemann solution we can see that the two quantities v_k and π_k are independent from the other two quantities τ_k and E_k. Thus, at first the velocity and relaxation pressure at the next time step are computed from the relation (6.40) by solving a doubled tridiagonal linear systems:

$$\begin{aligned}
(v_k)_i^{n+1,-} &= (v_k)_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left((\pi_k)_{\star i+1/2}^{n+1,-} - (\pi_k)_{\star i-1/2}^{n+1,-}\right) \\
&\overset{(6.33)}{=} (v_k)_i^n - \frac{\Delta t}{\Delta x}\,\frac{\tau_i^n}{2} \Big((\pi_k)_{i+1}^{n+1,-} - (\pi_k)_{i-1}^{n+1,-} \\
&\qquad\qquad - a_k (v_k)_{i+1}^{n+1,-} + 2a_k (v_k)_i^{n+1,-} - a_k (v_k)_{i-1}^{n+1,-}\Big), \\
(\pi_k)_i^{n+1,-} &= (\pi_k)_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left(a_k^2 (v_k)_{\star i+1/2}^{n+1,-} - a_k^2 (v_k)_{\star i-1/2}^{n+1,-}\right) \\
&\overset{(6.32)}{=} (\pi_k)_i^n - \frac{\Delta t}{\Delta x}\,\frac{\tau_i^n}{2} \Big(a_k^2 (v_k)_{i+1}^{n+1,-} - a_k^2 (v_k)_{i-1}^{n+1,-} \\
&\qquad\qquad - a_k (\pi_k)_{i+1}^{n+1,-} + 2a_k (\pi_k)_i^{n+1,-} - a_k (\pi_k)_{i-1}^{n+1,-}\Big).
\end{aligned} \tag{6.42}$$

With the knowlege of $(v_k)_i^{n+1,-}$ and $(\pi_k)_i^{n+1,-}$ also the intermediate states $(v_k)_{\star i\pm 1/2}^{n+1,-}$ and $(\pi_k)_{\star i\pm 1/2}^{n+1,-}$ at the next time step are known. Now, the specific volume τ_k and specific total energy E_k at the next time step can explicitly be computed from the implicit update formulation:

$$\begin{aligned}
(\tau_k)_i^{n+1,-} &= (\tau_k)_i^n + \frac{\Delta t}{\Delta x}\,\tau_i^n \left((v_k)_{\star i+1/2}^{n+1,-} - (v_k)_{\star i-1/2}^{n+1,-}\right) \\
(E_k)_i^{n+1,-} &= (E_k)_i^n - \frac{\Delta t}{\Delta x}\,\tau_i^n \left((v_k)_{\star i+1/2}^{n+1,-} (\pi_k)_{\star i+1/2}^{n+1,-} - (v_k)_{\star i-1/2}^{n+1,-} (\pi_k)_{\star i-1/2}^{n+1,-}\right).
\end{aligned} \tag{6.43}$$

▶ Instantaneous relaxation ($t^{n+1,-} \to t^{n+1,=}$)

In the second step the relaxation source terms are considered for an infinite relaxation time with $\mathbf{u}_i^{n+1,-}$ as initial data. This ordinary differential equation system can be solved

exactly. By the form of the relaxation source terms $\mathbf{s}_{\mathrm{r}}(\mathbf{u})$, just π evolves according to the ordinary differential equation, such that

$$\mathbf{u}_i^{n+1,=} = \begin{pmatrix} (\alpha)_i^{n+1,=} \\ (\tau_\ell)_i^{n+1,=} \\ (v_\ell)_i^{n+1,=} \\ (E_\ell)_i^{n+1,=} \\ (\pi_\ell)_i^{n+1,=} \\ (\tau_g)_i^{n+1,=} \\ (v_g)_i^{n+1,=} \\ (E_g)_i^{n+1,=} \\ (\pi_g)_i^{n+1,=} \end{pmatrix} = \begin{pmatrix} (\alpha)_i^{n+1,-} \\ (\tau_\ell)_i^{n+1,-} \\ (v_\ell)_i^{n+1,-} \\ (E_\ell)_i^{n+1,-} \\ \left(p_\ell + \exp\left(-\frac{\Delta t}{\varepsilon_\ell}\right)(\pi_\ell - p_\ell)\right)_i^{n+1,-} \\ (\tau_g)_i^{n+1,-} \\ (v_g)_i^{n+1,-} \\ (E_g)_i^{n+1,-} \\ \left(p_g + \exp\left(-\frac{\Delta t}{\varepsilon_g}\right)(\pi_g - p_g)\right)_i^{n+1,-} \end{pmatrix}. \tag{6.44}$$

6.2.3. Upwind solver of the transport subsystem ($t^{n+1,=} \to t^{n+1,\equiv}$)

The transport subsystem (6.23) can be expressed in the form,

$$\begin{cases} \partial_t \alpha = 0, \\ \partial_t \mathbf{u}_\ell + v_\ell\, \partial_x \mathbf{u}_\ell = 0, \\ \partial_t \mathbf{u}_g + v_g\, \partial_x \mathbf{u}_g = 0, \end{cases}$$

where

$$\mathbf{u}_k = (\alpha\rho_k, \alpha\rho_k v_k, \alpha\rho_k E_k, \alpha\rho_k \pi_k)^{\mathsf{T}}. \tag{6.45}$$

This subsystem only involves the transport of the conservative variables with the velocity v_k, such that it is obvious that this subsystem is hyperbolic. To approximate the solution a standard upwind finite volume method can be used, [131],

$$\begin{aligned} (\mathbf{u}_k)_i^{n+1,\equiv} = (\mathbf{u}_k)_i^{n+1,=} &- \frac{\Delta t}{\Delta x}\left((v_k)_{\star i+1/2}(\mathbf{u}_k)_{i+1/2}^{n+1,=} - (v_k)_{\star i-1/2}(\mathbf{u}_k)_{i-1/2}^{n+1,=}\right) \\ &+ \frac{\Delta t}{\Delta x}(\mathbf{u}_k)_i^{n+1,=}\left((v_k)_{\star i+1/2} - (v_k)_{\star i-1/2}\right), \end{aligned} \tag{6.46}$$

which depends on the direction of the flow at each interface,

$$(\mathbf{u}_k)_{i+1/2}^{n+1,=} = \begin{cases} (\mathbf{u}_k)_i^{n+1,=}, & \text{if } (v_k)_{\star i+1/2} \geq 0 \\ (\mathbf{u}_k)_{i+1}^{n+1,=}, & \text{if } (v_k)_{\star i+1/2} < 0, \end{cases} \tag{6.47}$$

where $(v_k)_{\star i+1/2}$ is the speed of the Riemann problem at the interface $x_{i+1/2}$, see (6.32).

6.2.4. Upwind solver of the genuinely nonconservative subsystem ($t^{n+1,\equiv} \to t^{n+1}$)

The genuinely non-conservative subsystem (6.24)

$$\begin{aligned}
\partial_t \alpha_k + v_\mathrm{i}\, \partial_x \alpha_k &= 0,\\
\partial_t (\alpha_k \rho_k) &= 0,\\
\partial_t (\alpha_k \rho_k v_k) + (\pi_k - \pi_\mathrm{i})\, \partial_x \alpha_k &= 0,\\
\partial_t (\alpha_k \rho_k E_k) + (\pi_k v_k - \pi_\mathrm{i} v_\mathrm{i})\, \partial_x \alpha_k &= 0,\\
\partial_t (\alpha_k \rho_k \pi_k) &= 0,
\end{aligned}$$

is solved by using an upwind solver. This subsystem only involves the transport of the volume fraction, each equation can be expressed in the form,

$$\partial_t \mathbf{u}_j + \mathbf{v}_j\, \partial_x \alpha = 0,$$

with

$$\mathbf{u} = (\alpha, \alpha_\ell \rho_\ell, \alpha_\ell \rho_\ell v_\ell, \alpha_\ell \rho_\ell E_\ell, \alpha_\ell \rho_\ell \pi_\ell, \alpha_g \rho_g, \alpha_g \rho_g v_g, \alpha_g \rho_g E_g, \alpha_g \rho_g \pi_g)^\mathsf{T} \tag{6.48}$$

and

$$\mathbf{v} = (v_\mathrm{i}, 0, -(\pi_\ell - \pi_\mathrm{i}), -(\pi_\ell v_\ell - \pi_\mathrm{i} v_\mathrm{i}), 0, 0, (\pi_g - \pi_\mathrm{i}), (\pi_g v_g - \pi_\mathrm{i} v_\mathrm{i}), 0)^\mathsf{T}. \tag{6.49}$$

To approximate the solution a standard upwind finite volume method can be used,

$$\begin{aligned}
(\mathbf{u}_j)_i^{n+1} = (\mathbf{u}_j)_i^{n+1,\equiv} &- \frac{\Delta t}{\Delta x} \left((\mathbf{v}_j)_{\star i+1/2} (\mathbf{u}_j)_{i+1/2}^{n+1,\equiv} - (\mathbf{v}_j)_{\star i-1/2} (\mathbf{u}_j)_{i-1/2}^{n+1,\equiv} \right)\\
&+ \frac{\Delta t}{\Delta x} (\mathbf{u}_j)_i^{n+1,\equiv} \left((\mathbf{v}_j)_{\star i+1/2} - (\mathbf{v}_j)_{\star i-1/2} \right),
\end{aligned} \tag{6.50}$$

which depends on the direction of the flow at each interface,

$$(\mathbf{u}_j)_{i+1/2}^{n+1,\equiv} = \begin{cases} (\mathbf{u}_j)_i^{n+1,\equiv}, & \text{if } (\mathbf{v}_j)_{\star i+1/2} \geq 0 \\ (\mathbf{u}_j)_{i+1}^{n+1,\equiv}, & \text{if } (\mathbf{v}_j)_{\star i+1/2} < 0, \end{cases} \tag{6.51}$$

where $(\mathbf{v}_j)_{\star i+1/2}$ is given by the solution of the Riemann problem at the interface $x_{i+1/2}$, see (6.32) and (6.33).

7. Case studies

Several test cases have been performed to investigate the behavior of the developed schemes. Because the path-conservative schemes ESPC and ESPCs need a CFL number of lower than 0.5, here we use for comparability for all schemes a CFL number of 0.45.

In Subsection 7.1 the *coupled Burgers' equation* is used to show the advantages of the ESPC scheme over the traditional path-conservative schemes. This test case was proposed by Castro, Fjordholm, Mishra, and Parés [108]. Subsection 7.2 deals with the investigation of the quality of path-conservative schemes in comparison to standard conservative schemes regarding *Sod's shock tube problem* [133]. Additionally, the entropy dissipation is investigated and for the ESPC and ESPCs scheme different regularizations are compared.

In Subsection 7.3, the *isolated coupling wave* is performed as a first test for the two-phase flow model. This problem was proposed by Gallouët, Hérard, and Seguin [8]. Furthermore, the convergence for the pressure relaxation time of the Godunov-Suliciu solver is shown. In Subsection 7.4 the *pressure disequilibrium* test case [8] was performed to investigate the Riemann invariants at the v_i-contact discontinuity. In Subsection 7.5 the impact of the parameter ξ (which defines the interfacial pressure p_i and v_i) is investigated regarding the *mixture at rest with increase in the volume fraction* test case. This test case was proposed by Schwendeman, Wahle and Kapila [134]. In Subsection 7.6 the influence of source terms in a vertical tube are investigated. This *water faucet problem* test case was proposed by Ransom [135].

In Section 7.7 we turn to the application and simulate the evaporation in a solar absorber tube is simulated. Here all source terms are used which has been developed in Section 2.4. Finally, in Section 7.8 all results are analyzed.

7.1. Coupled Burgers equation

As a first test for the path-conservative methods a non-conservative test case of the coupled Burgers equations is used,

$$\begin{aligned} \partial_t u + u\,\partial_x(u+v) &= 0, \\ \partial_t v + v\,\partial_x(u+v) &= 0, \end{aligned} \tag{7.1}$$

or equivalently

$$\begin{pmatrix} u \\ v \end{pmatrix}_t + \begin{pmatrix} u & u \\ v & v \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x = 0. \tag{7.2}$$

On the domain $x \in [-2, 10.5]$ a Riemann problem is defined at $x = 0$, where the initial values are given by [108]:

	u	v
Left	7.99	11.01
Right	0.25	0.75

Berthon [136] computed the exact viscous profile of the regularized system which gives in the limit the entropy solution of the Riemann problem. This solution is used as reference solution. For the ESPC scheme, the entropy function $\eta(u,v) = \frac{(u+v)^2}{2}$ with corresponding entropy flux $\psi(u,v) = \frac{(u+v)^3}{3}$ is used. The entropy variables are then given by $\mathbf{v}(u,v) = (u+v,\, u+v)^\mathsf{T}$. For the regularization, we either choose the entropy viscosity (4.42) with viscosity matrix

$$\widehat{\mathcal{R}} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix},$$

or the uniform viscosity (4.43).

Because with this choice of the entropy-entropy flux pair, the mapping $\mathbf{u} \mapsto \mathbf{v}$ of the state vector to the entropy variables is not bijective, such that an inverse mapping $\mathbf{u}(\mathbf{v})$ does not exist. Thus, here we cannot apply the ESPCs scheme.

The results for different path-conservative schemes 1500 cells at time $t = 0.5$ are shown in Figure 39. It can be seen, contrary to the simple path-conservative schemes that the entropy stable path-conservative (ESPC) scheme approximates the reference solution. Thus, the numerical viscosity operator of the ESPC scheme matches with the underlying viscosity.

It can be seen, that the ESPC scheme with entropy viscosity simulates the correct shock. The kink at about $x = 0$ appears because the chosen numerical viscosity is zero at this position. The kink can be avoided by choosing a different visocity operator, e.g. see [115]. That this problem highly depends on the viscosity operator can also be seen by the poor results of the ESPC scheme with uniform viscosity.

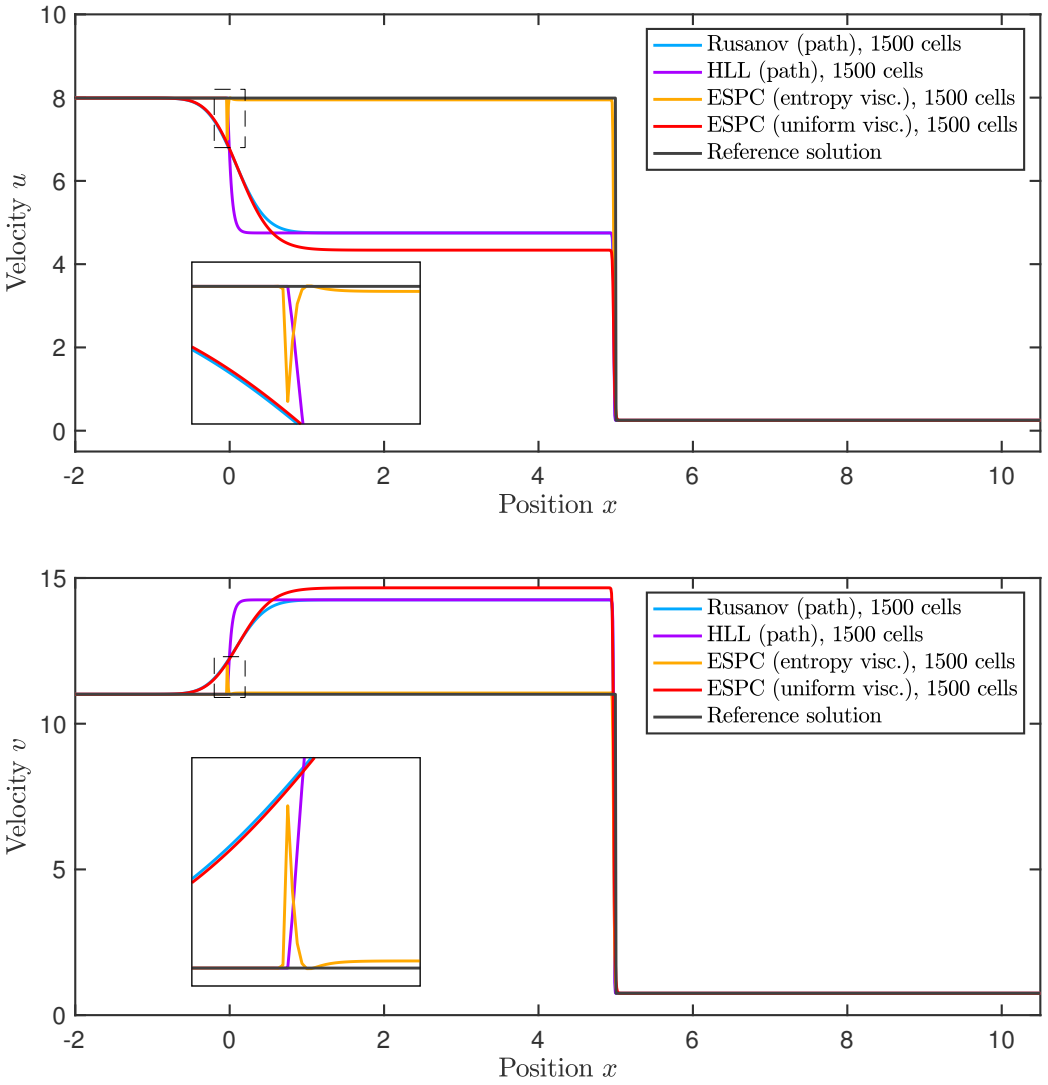


Figure 39: Coupled Burgers Equation at $t = 0.5$.

7.2. Sod's shock tube problem

The shock tube problem of Sod [133] can be used to test and compare the path-conservative schemes ESPC and ESPCs for the homogeneous equilibrium model, which is in conservative form. On the domain $x \in [0, 1]$ a Riemann problem is defined at $x = 0.5$, where the initial values are given by:

	ρ_h	$\rho_\mathrm{h} v_\mathrm{h}$	$\rho_\mathrm{h} E_\mathrm{h}$
Left	1	0	2.5
Right	0.125	0	0.25

As closure ideal gas with $\gamma = 1.4$ and $c_\mathrm{v} = 718\ \mathrm{J/kg\,K}$ is chosen.

For the ESPC and ESPCs scheme, the entropy/entropy flux pair is given by (2.151), with corresponding entropy variables (2.152). For the regularization, we either choose the entropy viscosity (4.42) with (4.37), or the uniform viscosity (4.43).

Because this problem is in conservative form, once again we can investigate the quality of path-conservative schemes in comparison to standard conservative schemes. In the following Figures 40, 41, 42, 43, 44, and 45 grid refinement results for the Rusanov, HLL, ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme are shown. It can be seen that all methods converge to the exact solution. But the Semi-Implicit scheme shows overshoots in the plot for the specific internal energy. From the plots of the ESPC and ESPCs results it can be seen that the different numerical viscosities result in a comparable solution.

In Figure 46 the total entropy dissipation $\eta(t_n) := \Delta x \cdot \sum_{i=1}^{N} \eta(\mathbf{u}_i^n)$ against simulation time for different meshes is plotted. It can be seen, that the total entropy dissipation of the ESPC and ESPCs scheme converges in direction of the reference solution. ESPC converges slightly faster than ESPCs. The regularization, whether uniform or entropy viscosity, has no influence on the solution. Also the Godunov-Suliciu solution converges against the reference solution, where the Semi-Implicit schemes does a poor job and in fact generates entropy for coarser meshes. Thus, at least one subsystem, which was used for the derivation of the Semi-Implicit scheme, does not satisfy the entropy inequality.

Additionally, for the ESPC and ESPCs scheme the influence on the solution is investigated by using the exact Hessian (4.37) and its finite differences approximation (4.39) for computing the derivative of the inverse mapping of entropy variables. In Figure 47 the results for different discretization steps ε for the ESPCs are shown. It can be seen, that for a coarser discretization jumps are generated at the contact discontinuity for velocity and pressure. These jumps are not present for the ESPC scheme with entropy viscosity. Furthermore, it can be seen, that a value of $\varepsilon = 0.1$ gives similar results to the ESPC scheme with entropy viscosity, see Table 2.

Discretization ε of ESPCs	$\|\cdot\|_2$ error to ESPC
100	1.2894
80	0.9828
60	0.6700
40	0.3520
20	0.0991
10	0.0258
1	0.0027
0.1	0.0027

Table 2: Comparison of the ESPC scheme (with entropy viscosity) with the ESPCs scheme using different discretization step sizes ε for computing the inverse mapping of entropy variables. As example the pressure of Sod's shock tube problem with 400 cells is used.

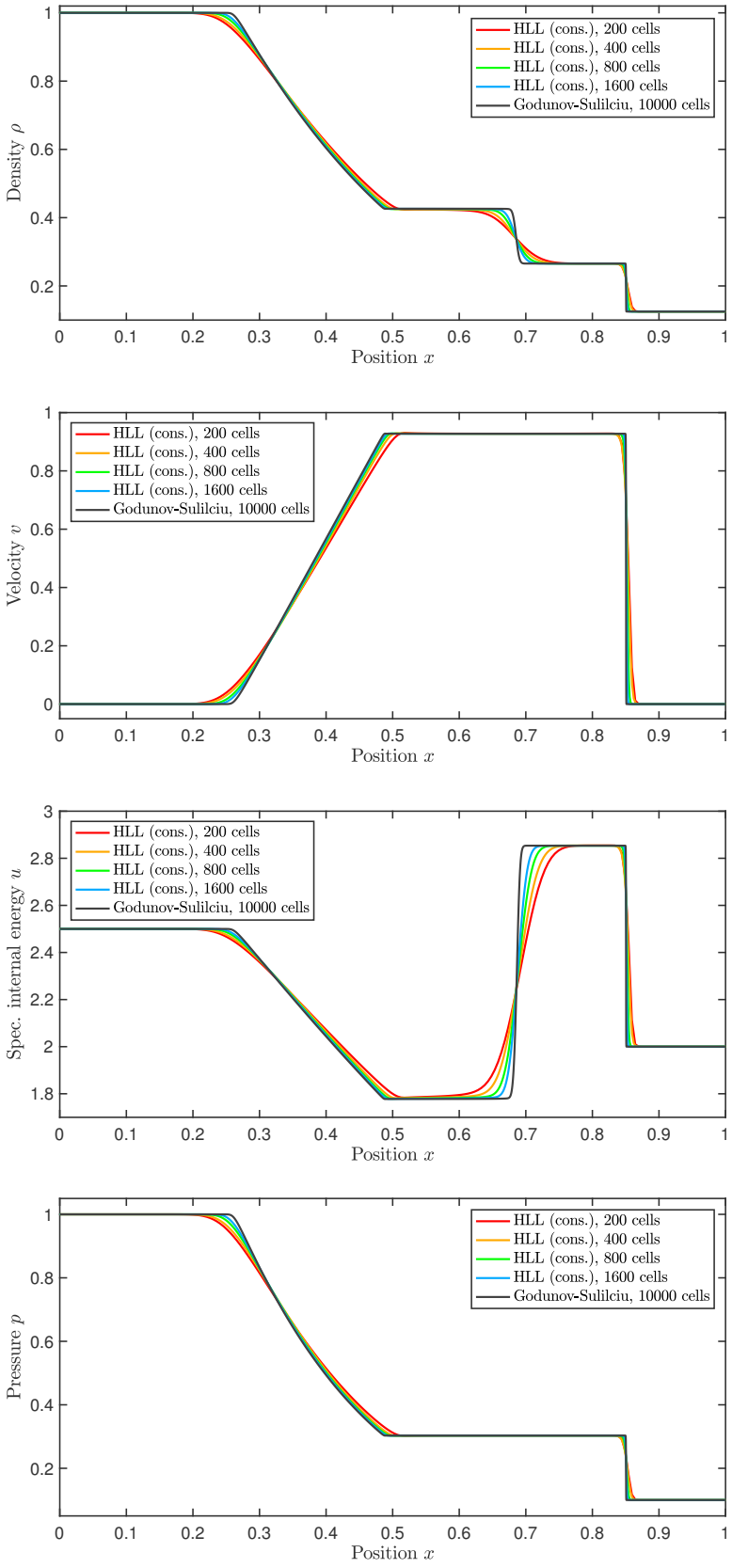


Figure 40: Grid refinement of the HLL solver for Sod's shock tube problem at $t = 0.2$.

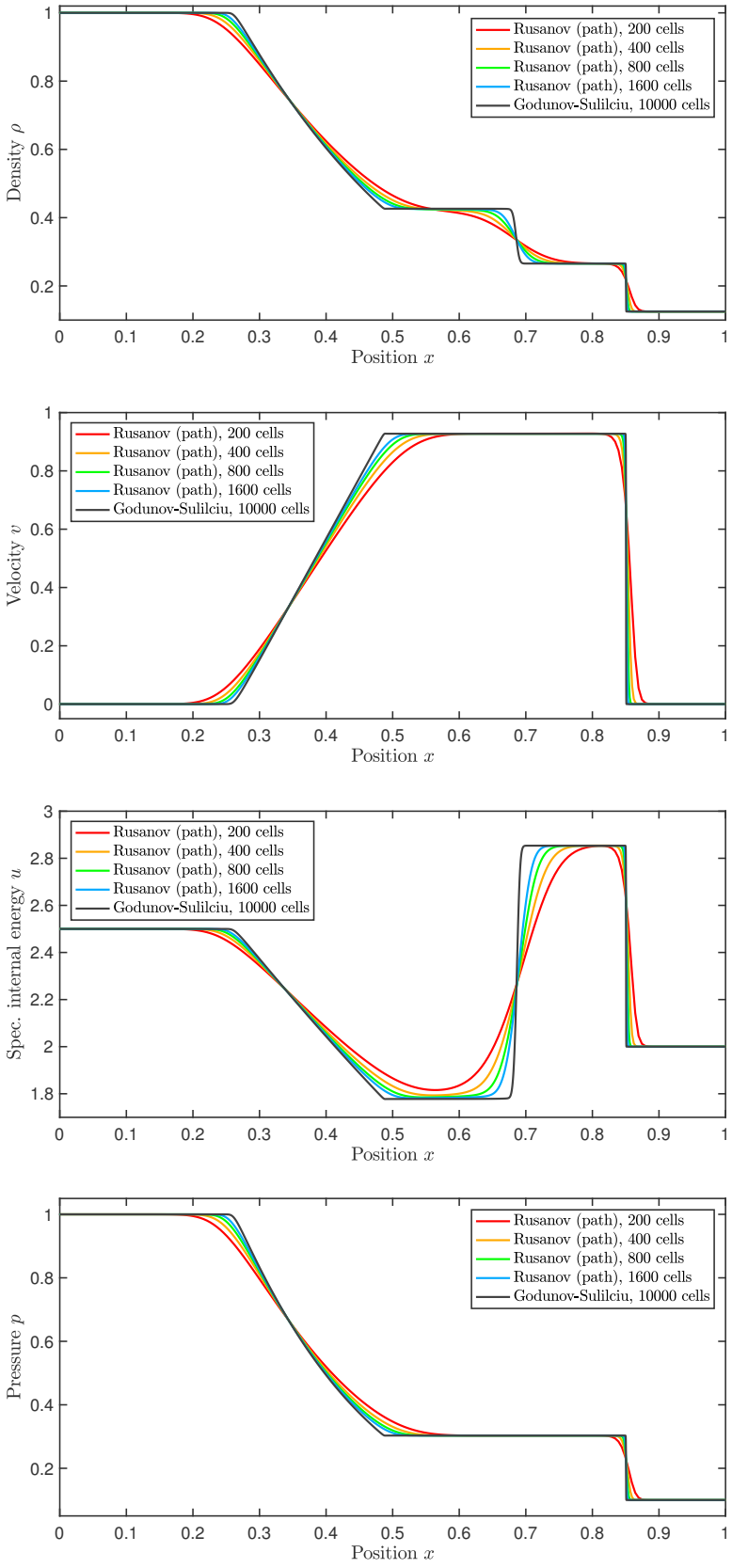


Figure 41: Grid refinement of the Rusanov solver for Sod's shock tube problem at $t = 0.2$.

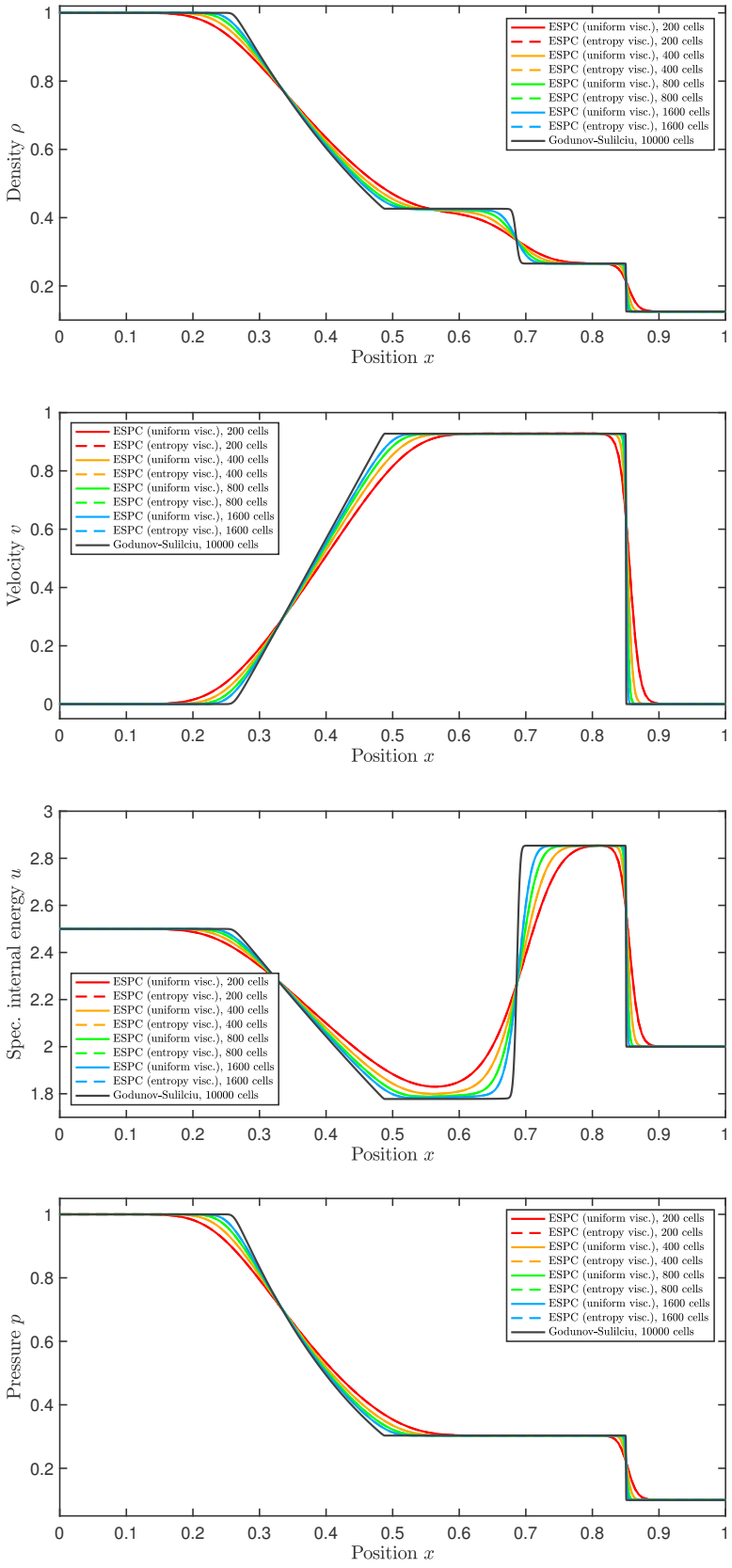


Figure 42: Grid refinement of the ESPC solver for Sod's shock tube problem at $t = 0.2$.

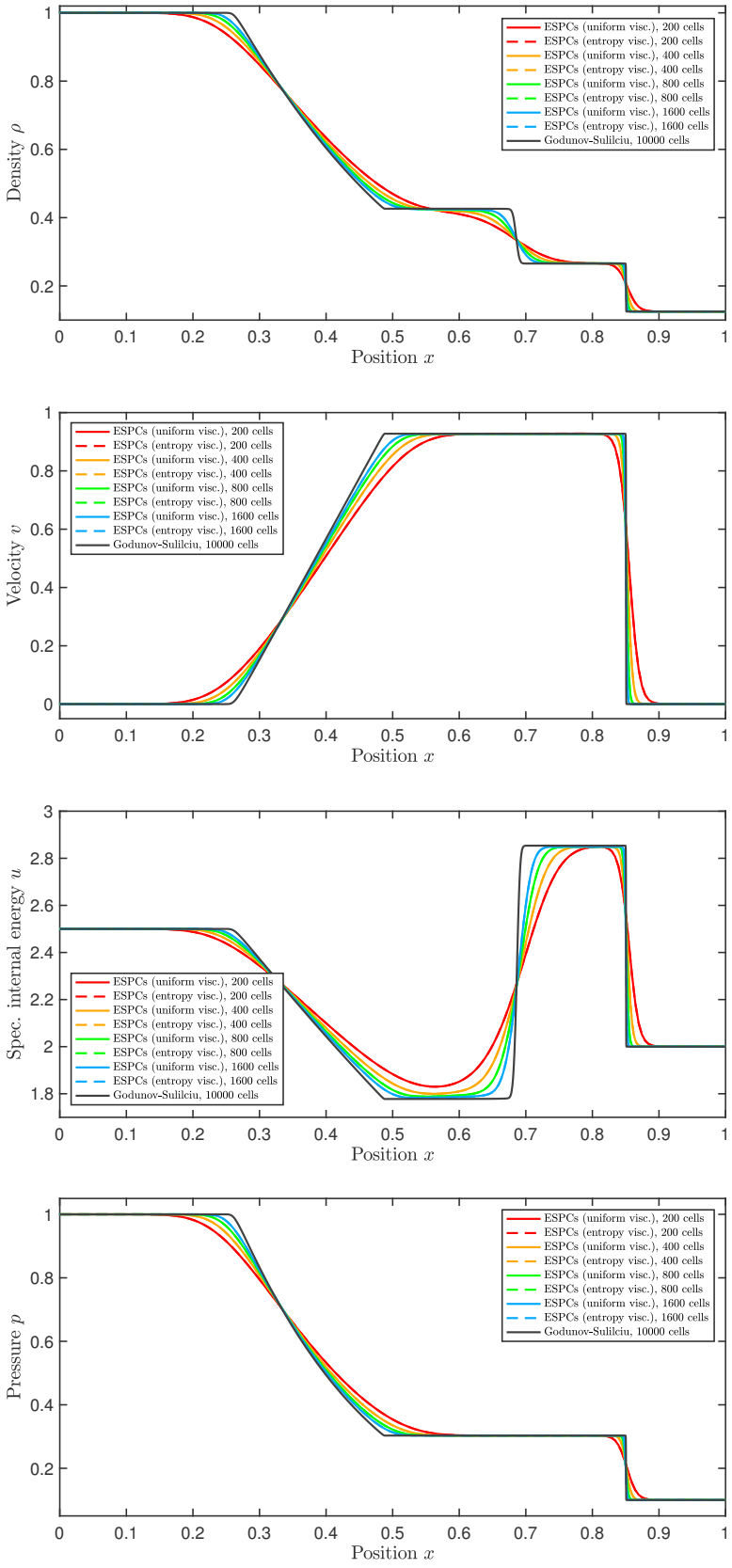


Figure 43: Grid refinement of the ESPCs solver for Sod's shock tube problem at $t = 0.2$.

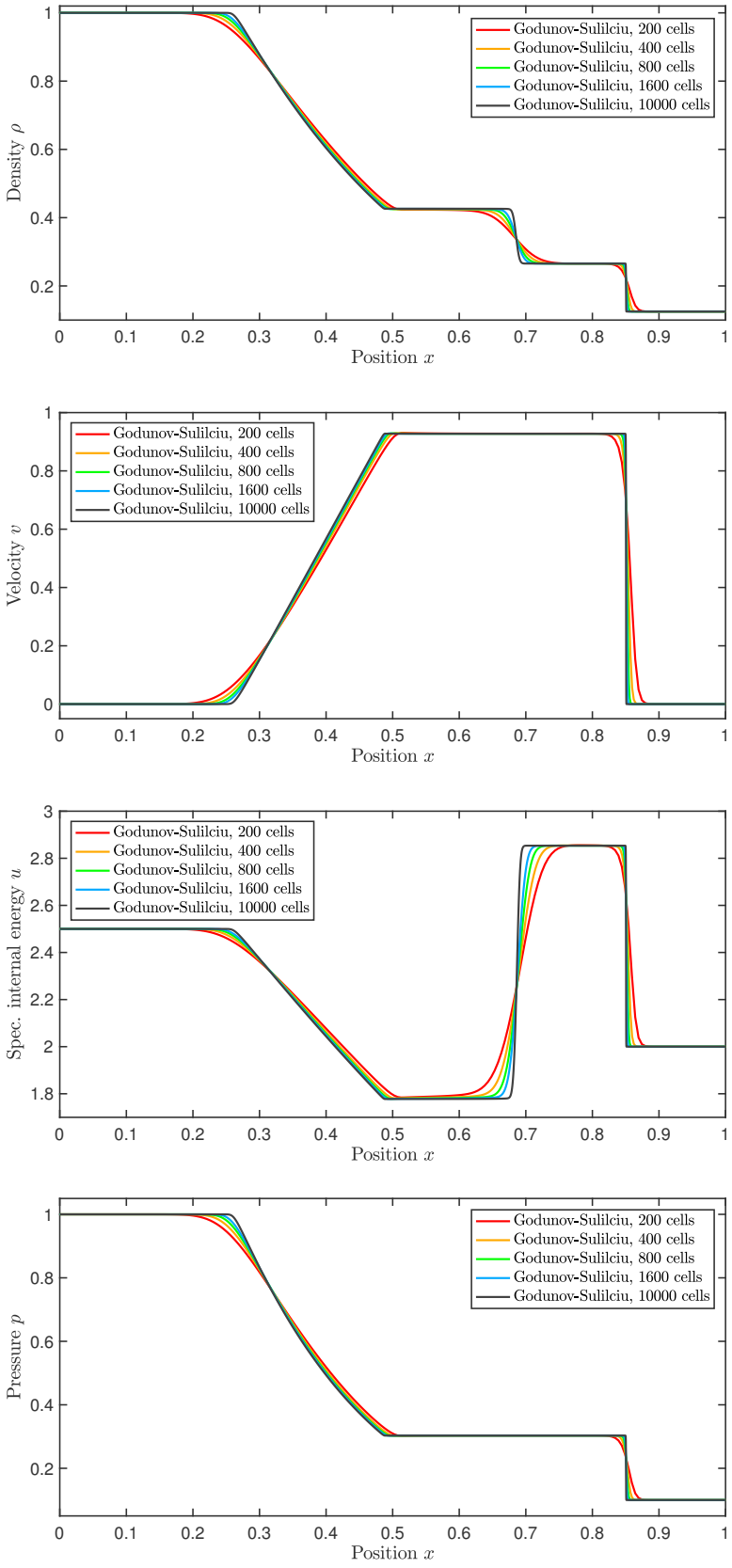


Figure 44: Grid refinement of the Godunov-Suliciu solver for Sod's shock tube problem at $t = 0.2$.

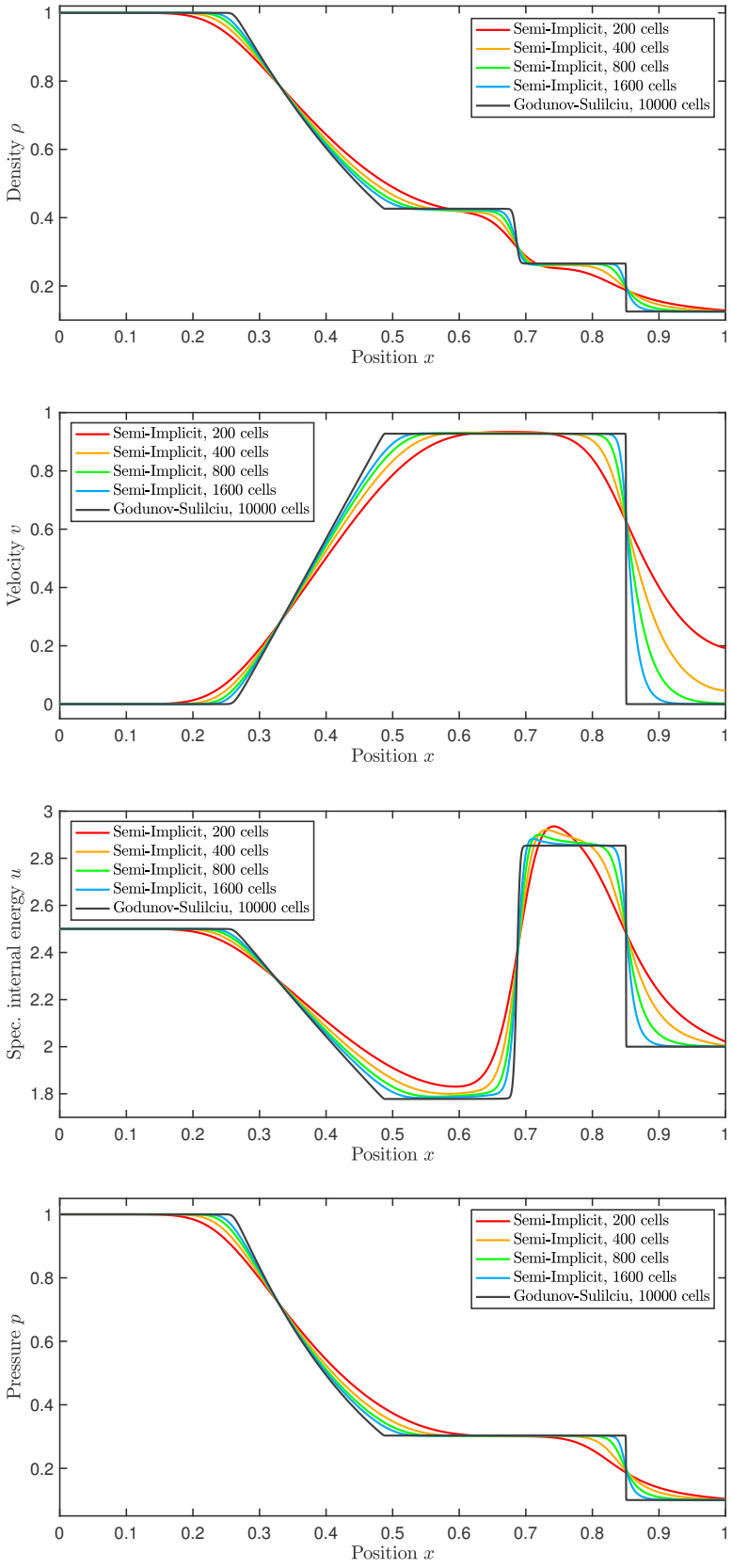


Figure 45: Grid refinement of the Semi-Implicit solver for Sod's shock tube problem at $t = 0.2$.

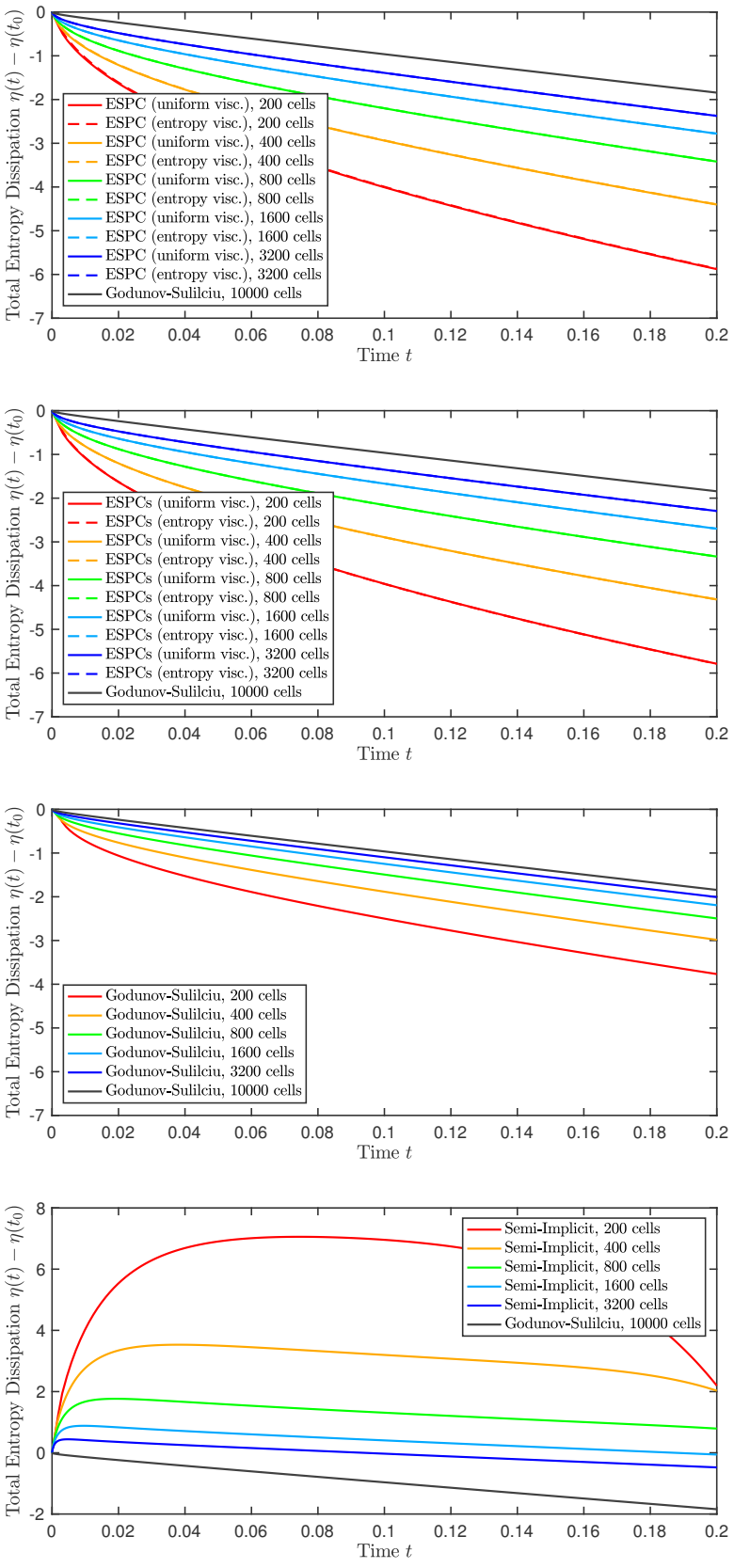


Figure 46: Total entropy dissipation of the ESPC, ESPCs, Godunov-Suliciu and Semi-Implicit scheme for different meshes for for Sod's shock tube problem.

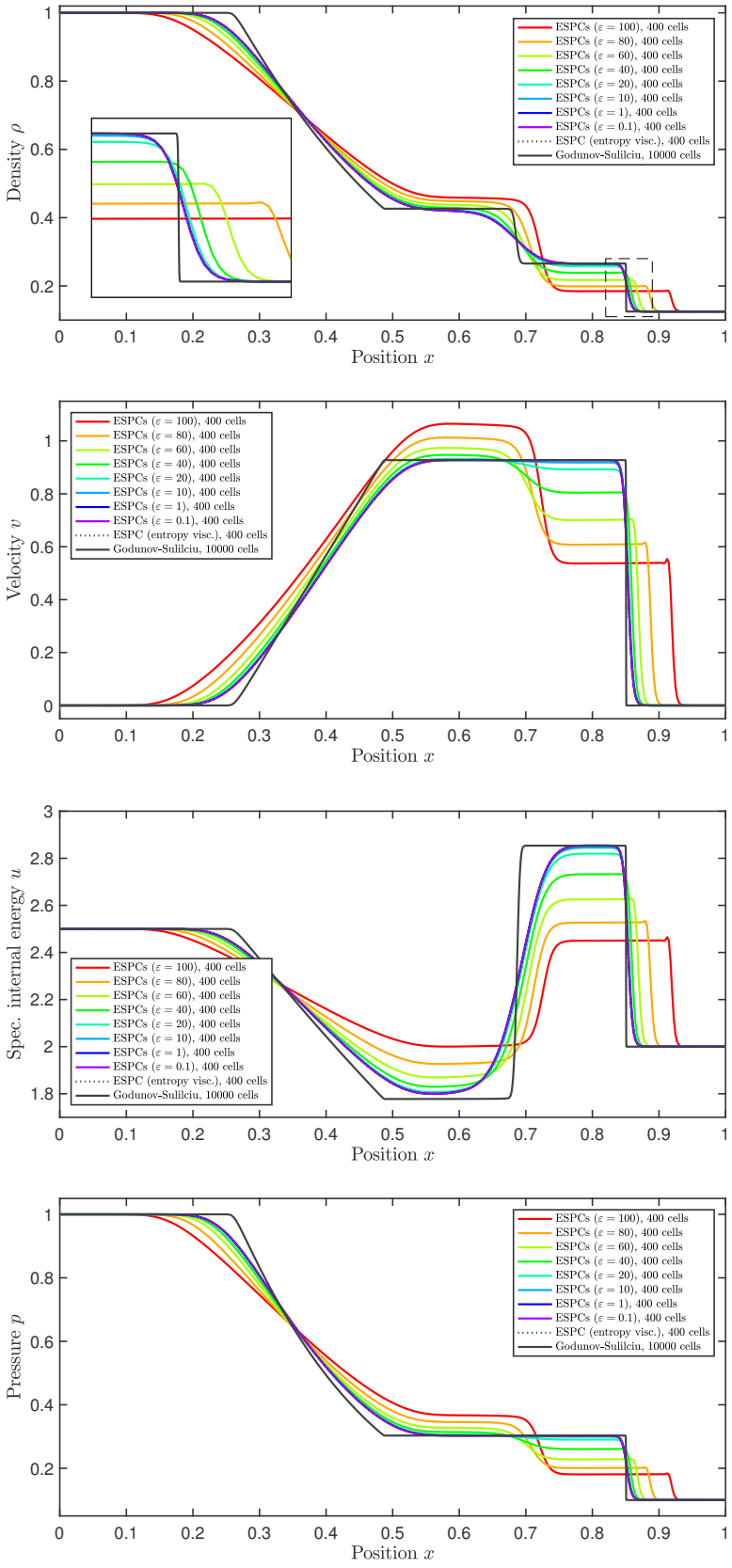


Figure 47: ESPCs scheme for Sod's shock tube problem. The derivative of the inverse mapping of entropy variables is approximated with finite differences at discretization step size ε.

7.3. Isolated coupling wave

As a first test for the two-phase flow model, a problem of Gallouët, Hérard, and Seguin [8] is used with the interfacial parameter $\xi = 0.5$, see (2.123). On the domain $x \in [0, 1000]$ a Riemann problem is defined at $x = 500$, where the initial values are given by:

	α_g	ρ_ℓ	v_ℓ	p_ℓ	ρ_g	v_g	p_g
Left	0.1	1	100	10^5	1	100	10^5
Right	0.5	0.125	100	10^5	0.125	100	10^5

As closure ideal gas with $\gamma_\ell = \gamma_g = 1.4$ and $c_{v\ell} = c_{vg} = 718$ J/kg K is chosen. Because all velocities are chosen equal, the volume fraction propagates with constant speed, such that we can give the exact solution for α [8]:

$$\alpha(x, t) = \alpha(0, x - 100t). \tag{7.3}$$

Thus, after a simulation time of $t = 3$ sec, the shock position moved from $x = 500$ to $x = 800$. For the ESPC and ESPCs scheme, the entropy/entropy flux pair is given by (2.91), with corresponding entropy variables (2.92). For the regularization uniform viscosity (4.43) is chosen.

Within this test case, the pressure relaxation time ε_k of the Godunov-Suliciu solver is investigated, compare with equation (5.50). In Figure 48 it can be seen that the solution converges for $\varepsilon_k \to 0$. In Figure 49 grid refinement results of the two path-conservative solvers ESPC and ESPCs, the Godunov-Suliciu solver from Section 5 and the Semi-Implicit solver from Section 6 is shown. It can be seen that all methods converge to the exact solution.

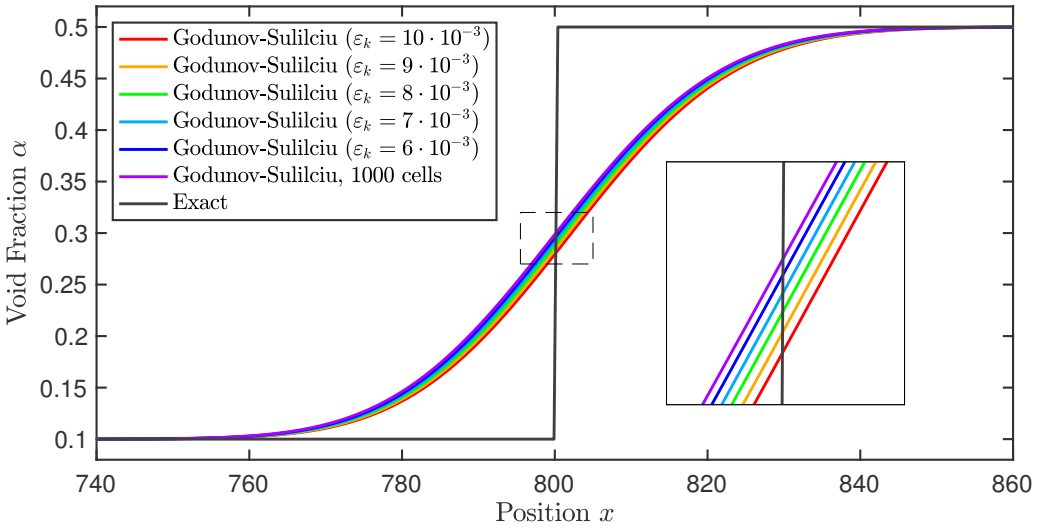


Figure 48: Isolated coupling wave problem at $t = 3$: Investigation of the pressure relaxation parameter ε of the Godunov-Suliciu solver with 2000 cells (zoomed around the shock position).

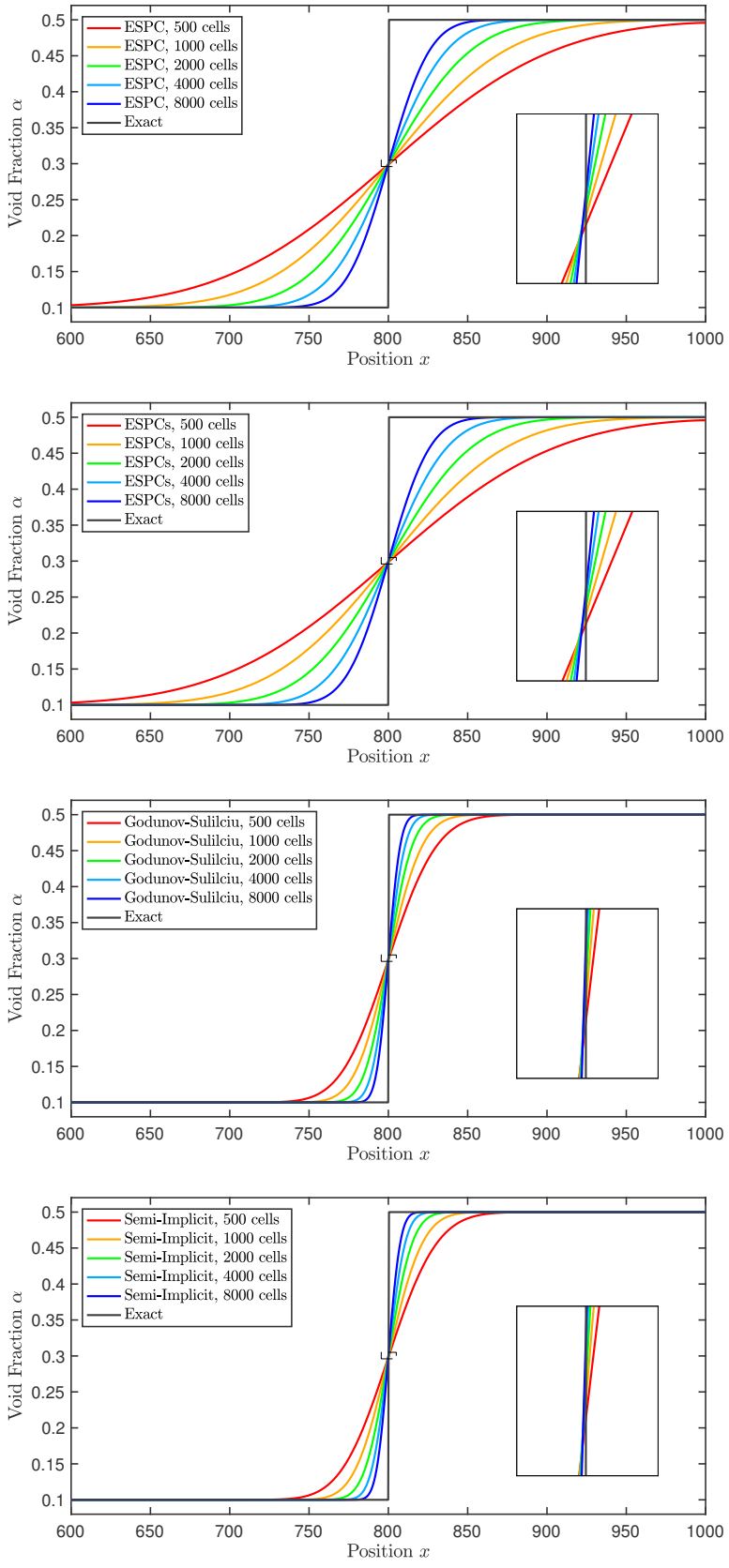


Figure 49: Grid refinement for the isolated coupling wave problem at $t = 3$ (zoomed around the shock position).

7.4. Pressure disequilibrium

In the above test case, a difference in quality can be seen between the path-conservative schemes (ESPC and ESPCs) and the relaxation schemes (Godunov-Suliciu and Semi-Implicit). Gallouët, Hérard, and Seguin [8] made this test case more difficult such that differences between all four solvers can be observed.

To investigate the behavior of a strong disequilibrium between both phases, the in 7.3 defined test case is modified by using the following initial values [77]:

	α_g	ρ_ℓ	v_ℓ	p_ℓ	ρ_g	v_g	p_g
Left	0.1	1	0	10^5	10	0	10^4
Right	0.5	0.125	0	10^4	1.25	0	10^3

Again, as closure ideal gas with $\gamma_\ell = \gamma_g = 1.4$ and $c_{v\ell} = c_{vg} = 718$ J/kg K is chosen. As reference solution, the Godunov-Suliciu solver with 100 000 cells is used. For the ESPC and ESPCs scheme, the entropy/entropy flux pair is given by (2.91), with corresponding entropy variables (2.92). For the regularization uniform viscosity (4.43) is chosen.

In the following Figures 50, 51, 52, and 53, grid refinement results for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme are shown. It can be seen that all methods converge to the reference solution. In the plot for the void fraction α the ESPC scheme creates a peak, where the ESPCs is smoother. But again, the path-conservative schemes converge slower than the relaxation schemes.

In Figure 54 the Riemann invariants at the v_i contact discontinuity defined in equations (5.69) are shown (just zoomed around the position of the corresponding wave) for all four schemes. At the contact discontinuity we expect a smooth behavior of the Riemann invariants. Due to the simplification we made in (5.74), we expect that the fifth $\mathbf{w}_5^{(1)}$ and sixth $\mathbf{w}_6^{(1)}$ Riemann invariant have a jump.

It can be seen that at the contact discontinuity (where x is around 605) the Godunov-Suliciu and the Semi-Implicit solver do not show jumps. For the fifth and sixth Riemann invariant we see some "delayed corrections" between $x = 605$ and $x = 615$, which may be caused by the simplified Riemann invariants.

It is interesting to see that the ESPC and the ESPCs have jumps in every Riemann invariant. It may be that the used uniform viscosity in (4.43) is not the best choice for this problem.

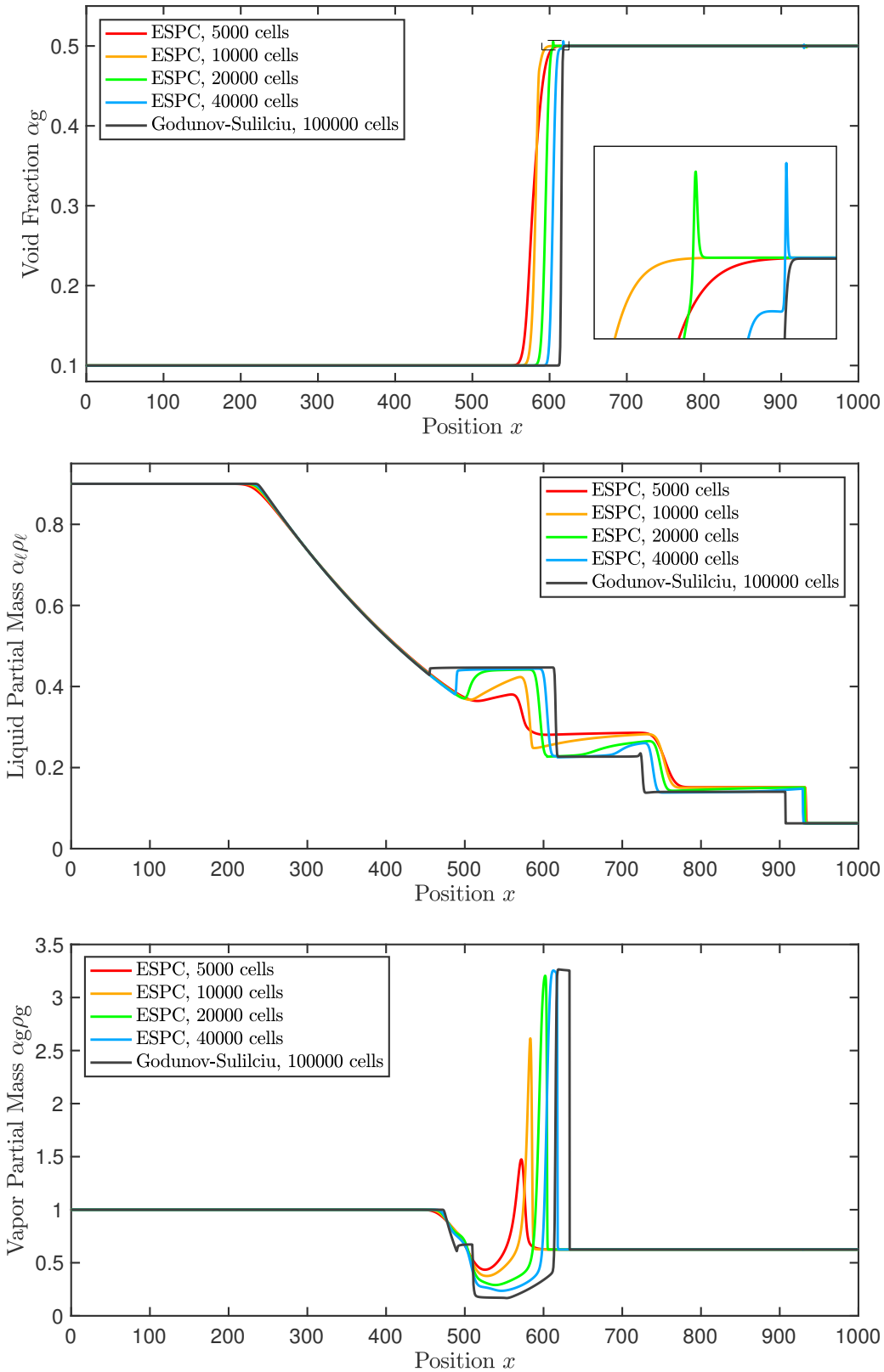


Figure 50: Grid refinement of the ESPC solver for the pressure disequilibrium problem at time $t = 0.7$.

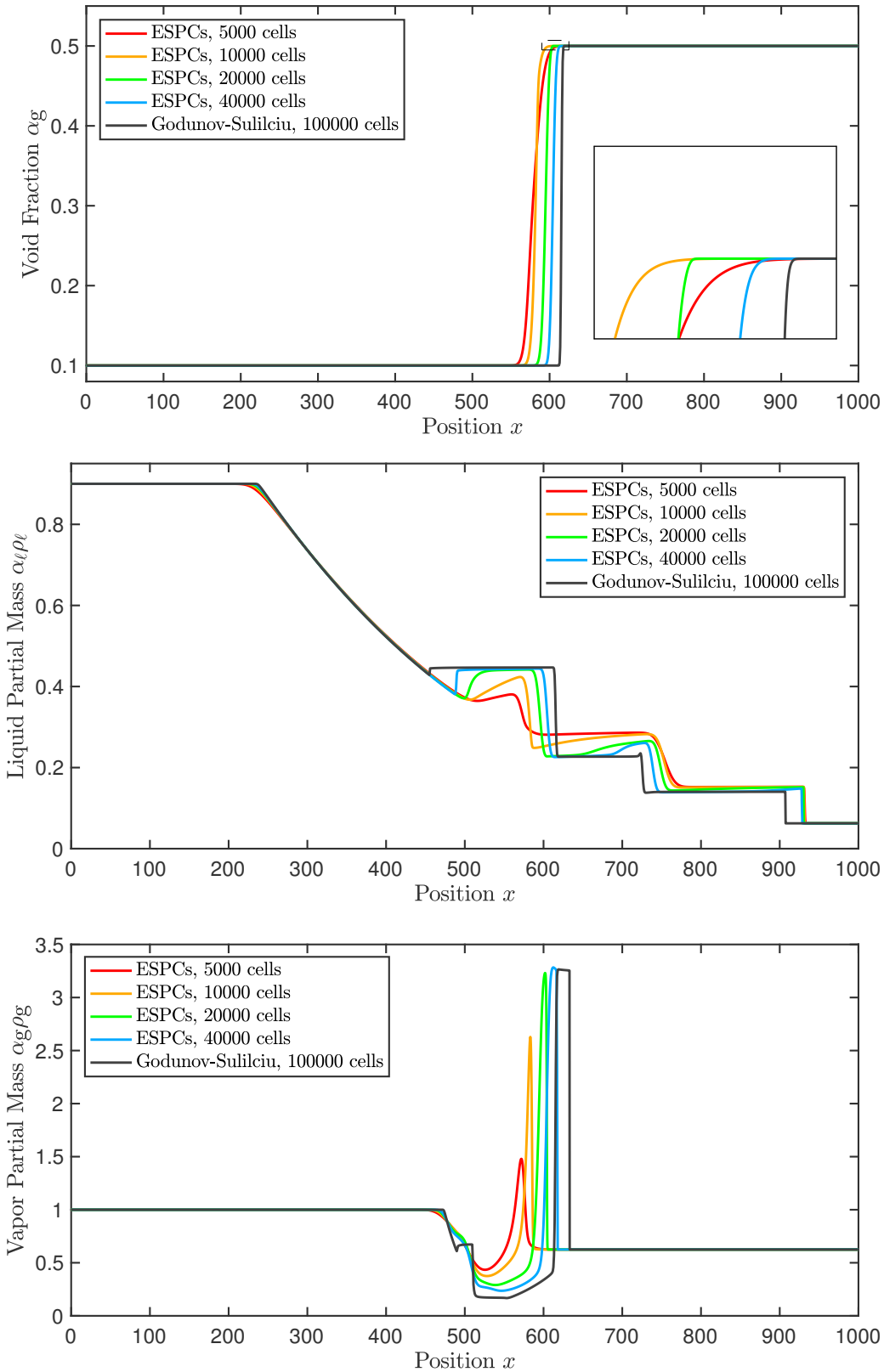


Figure 51: Grid refinement of the ESPCs solver for the pressure disequilibrium problem at time $t = 0.7$.

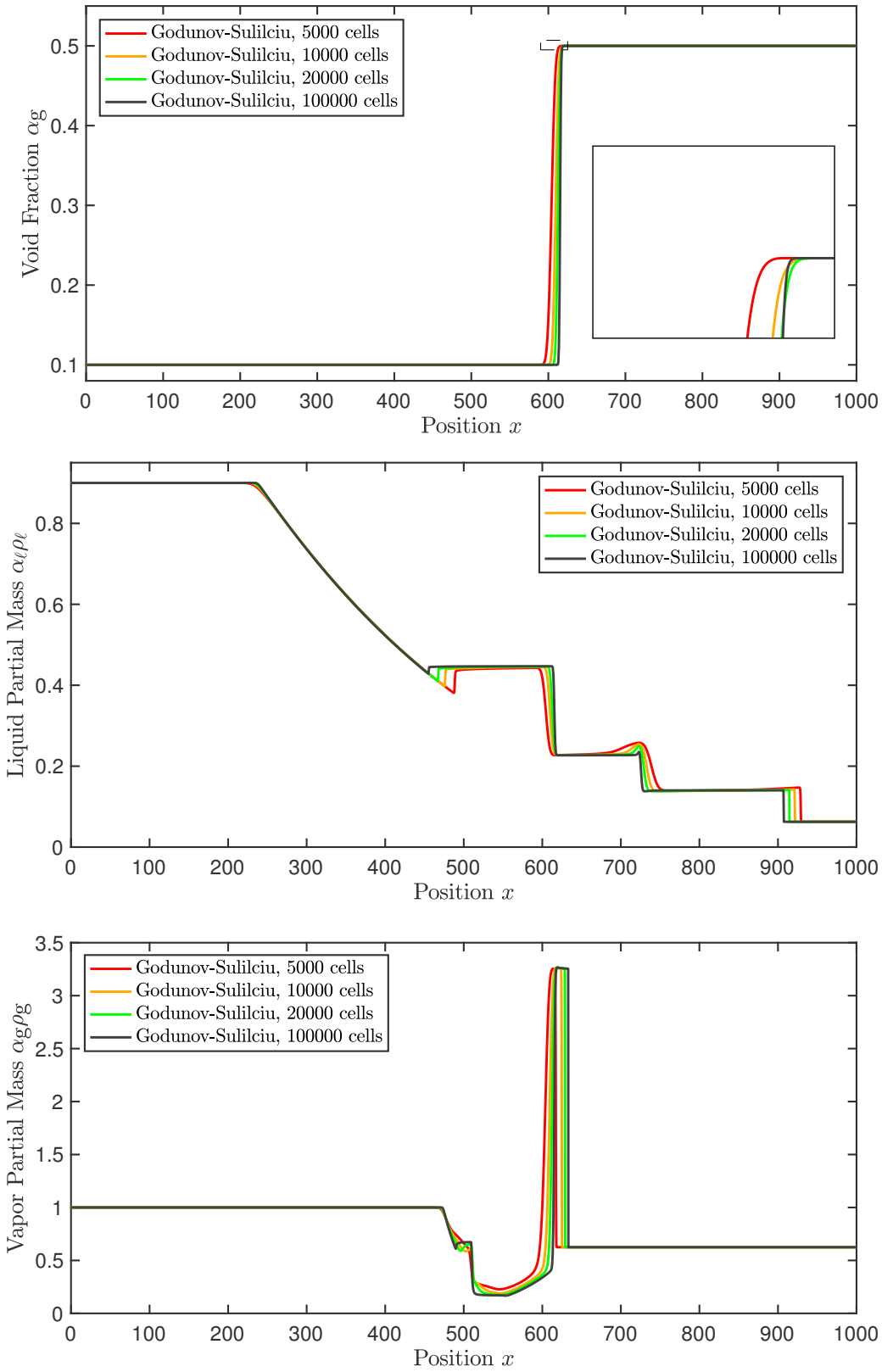


Figure 52: Grid refinement of the Godunov-Suliciu solver for the pressure disequilibrium problem at time $t = 0.7$.

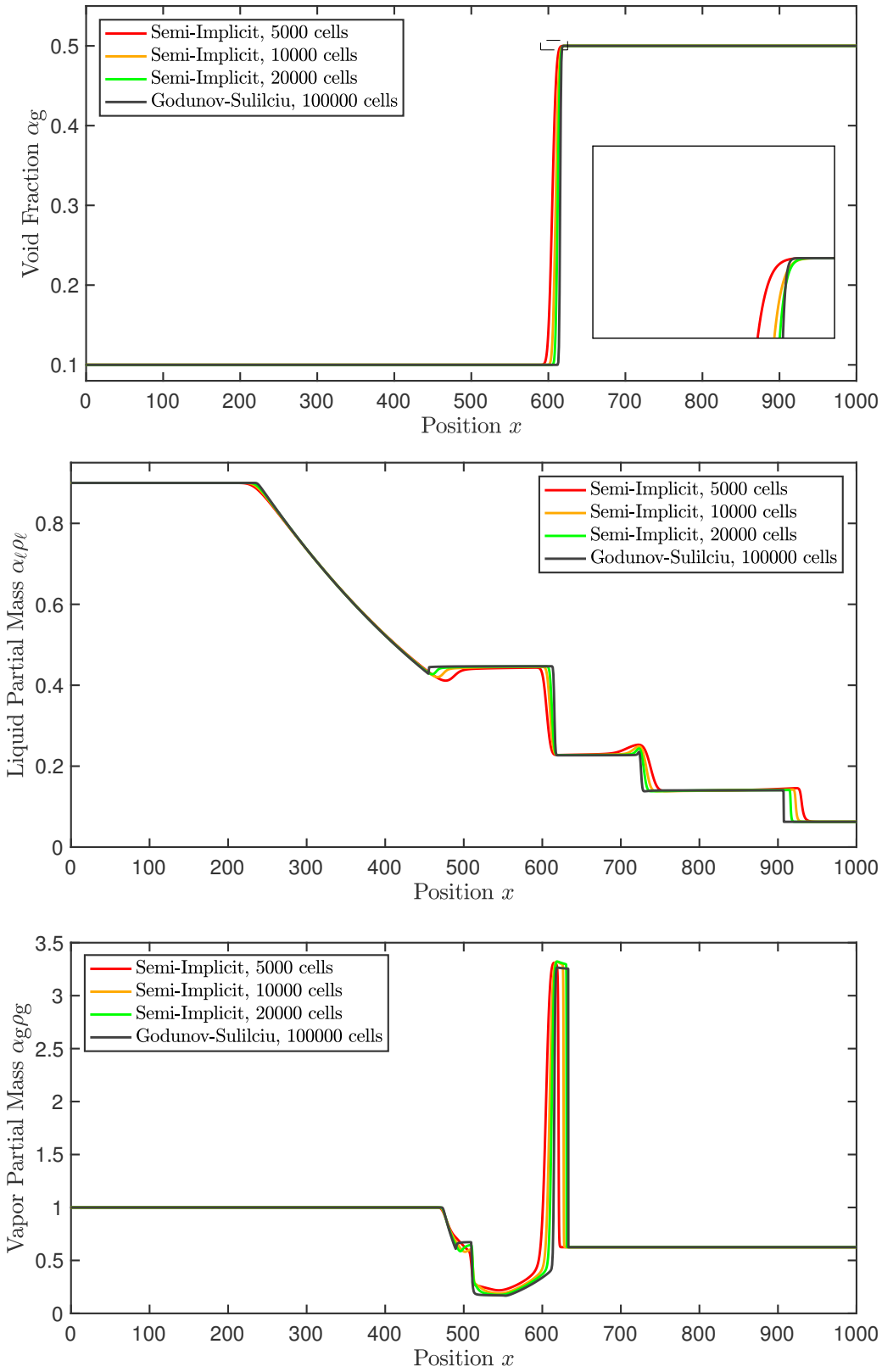


Figure 53: Grid refinement of the Semi-Implicit solver for the pressure disequilibrium problem at time $t = 0.7$.

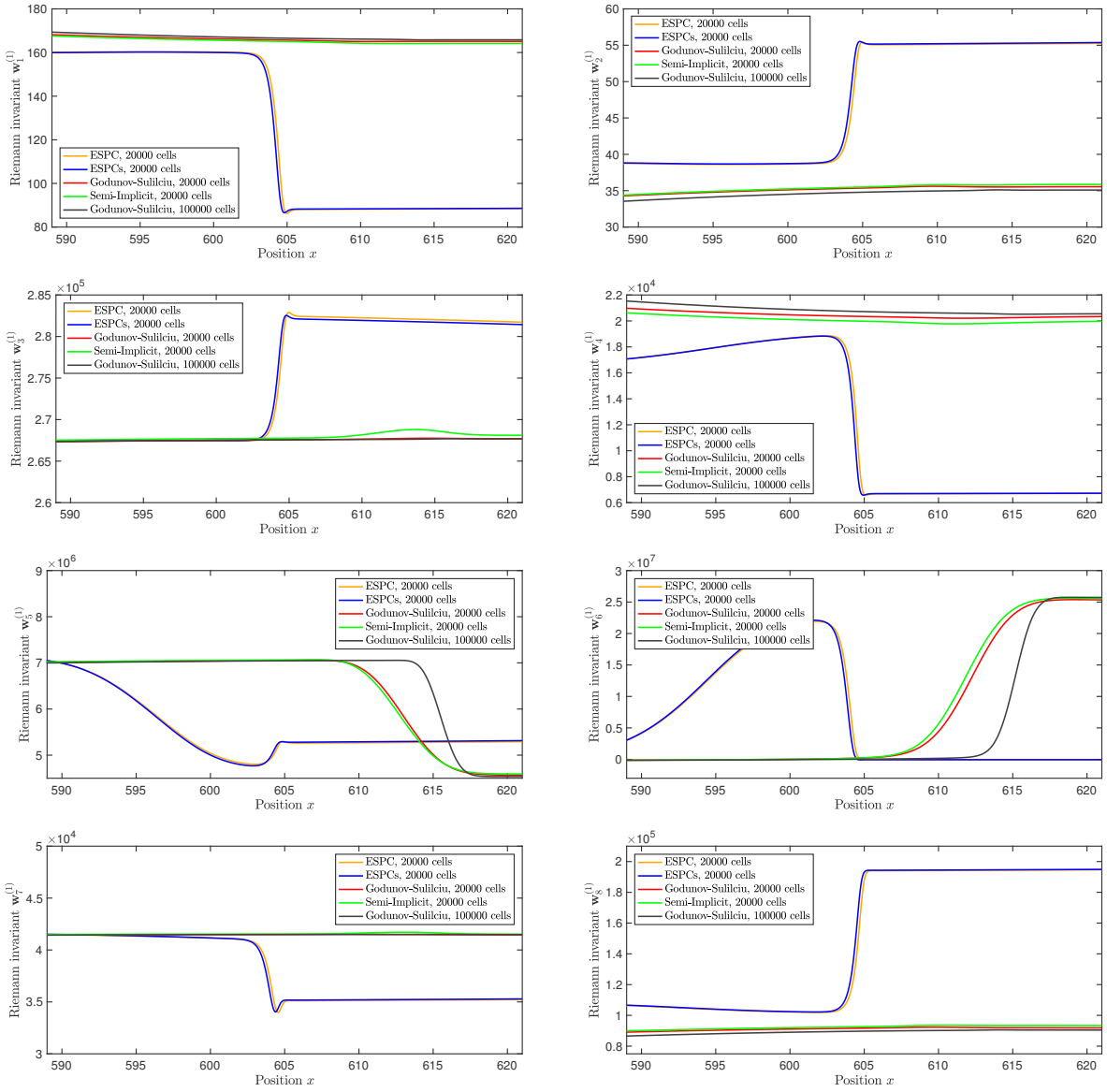


Figure 54: Riemann invariants at the v_{i}-contact discontinuity for pressure disequilibrium at $t = 0.7$ (just zoomed around the position of the corresponding wave).

7.5. Mixture at rest with increase in the volume fraction

The interfacial pressure and velocity was derived, such that the entropy compatibility condition is fulfilled, and that the v_{i}-contact discontinuity is linearly degenerate, see (2.94) and (2.123). This still gave us some freedom by choosing a parameter $\xi \in [0, 1]$, which can be seen as a weighting between the liquid phase and the vapor phase.

This test case by Schwendeman, Wahle and Kapila [134] is used to show the influence of the parameter ξ on the solution. On the domain $x \in [0, 1]$ a Riemann problem is defined at $x = 0.5$, where the initial two-phase mixture is at rest with an increase in the volume fraction:

	α_g	ρ_ℓ	v_ℓ	p_ℓ	ρ_g	v_g	p_g
Left	0.2	1	0	1	0.2	0	0.3
Right	0.7	1	0	1	1	0	1

As closure ideal gas with $\gamma_\ell = \gamma_g = 1.4$ and $c_{\mathrm{v}\ell} = c_{\mathrm{v}g} = 718$ J/kg K is chosen. For the ESPC and ESPCs scheme, the entropy/entropy flux pair is given by (2.91), with corresponding entropy variables (2.92). For the regularization uniform viscosity (4.43) is chosen.

In Figure 55 the spatial convergence for different parameters $\xi \in [0, 1]$ is shown. As expected, the limit depends on ξ because this directly influences the speed of propagation of the v_{i}-wave. In the following Figures 56, 57, 58, and 59, results with different values for ξ for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme are shown. The solution strongly depends on the choice of the parameter $\xi \in [0, 1]$. In the eyeball norm the solutions of the Semi-Implicit solver show a different behavior than the solutions of the ESPC, ESPCs, and Godunov-Suliciu solvers, which show a good agreements among themselves.

In Figure 60 the total entropy dissipation $\eta(t_n) := \Delta x \cdot \sum_{i=1}^{N} \eta(\mathbf{u}_i^n)$ against simulation time for different meshes is plotted, where ξ was chosen to be 0.5. As already observed for the homogeneous equilibrium model with Sod's shock tube problem, the entropy dissipation for the ESPC, ESPCs, and Godunov-Suliciu schemes converges against the reference solution. And again, the Semi-Implicit scheme does a poor job and in fact generates entropy for coarser meshes.

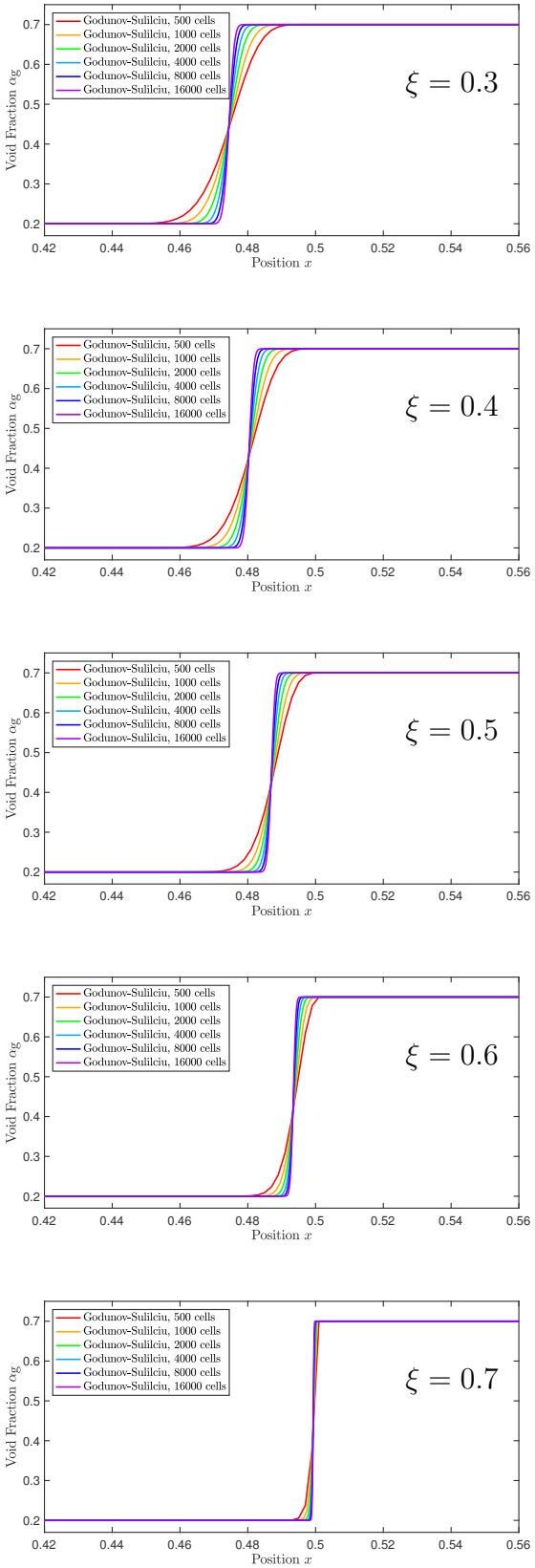


Figure 55: Mixture at rest with increase in the volume fraction using the Godunov-Suliciu scheme with different values for ξ at $t = 0.2$.

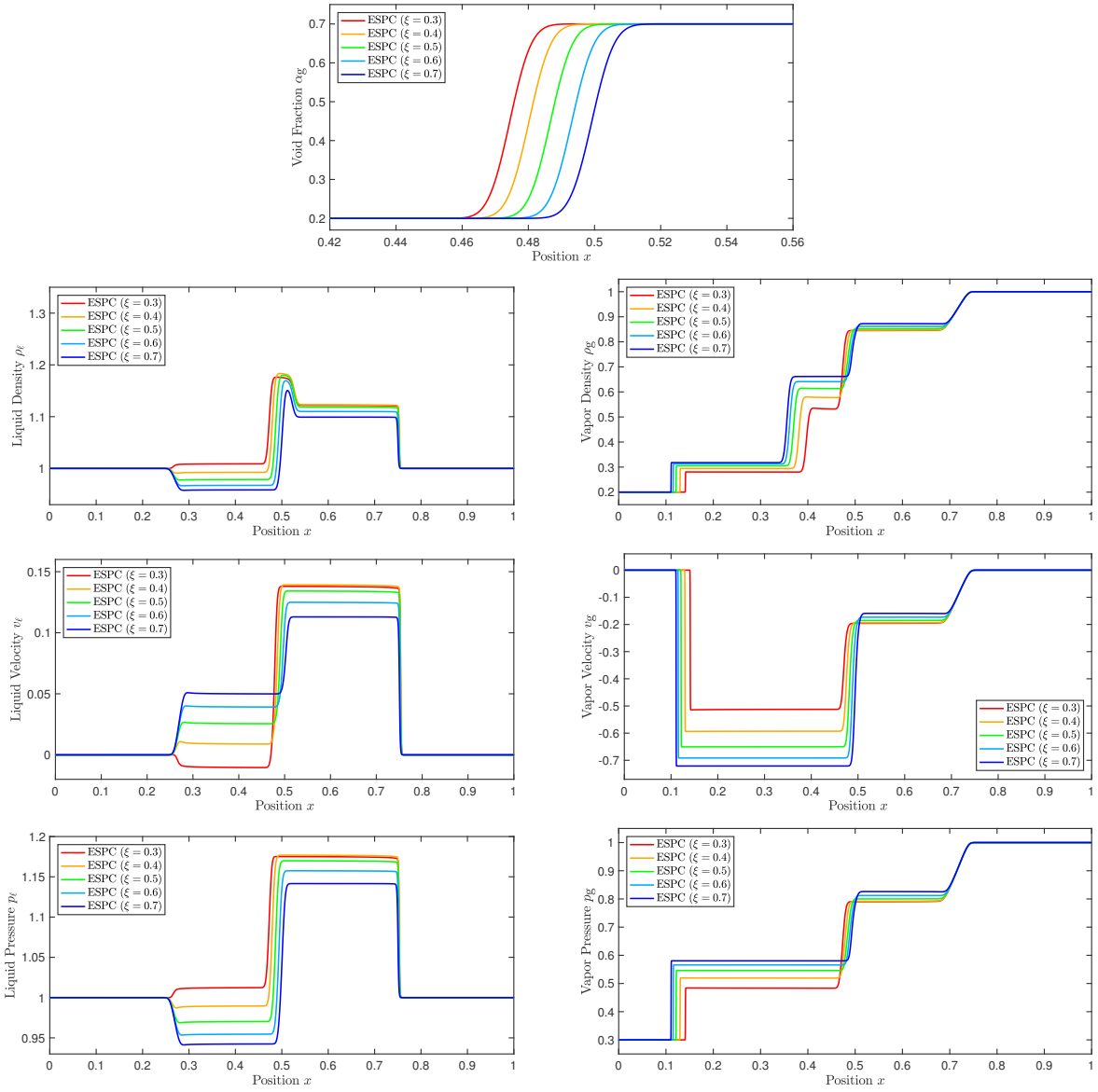


Figure 56: Mixture at rest with increase in the volume fraction using the ESPC scheme with 16 000 cells and different values for ξ at $t = 0.2$.

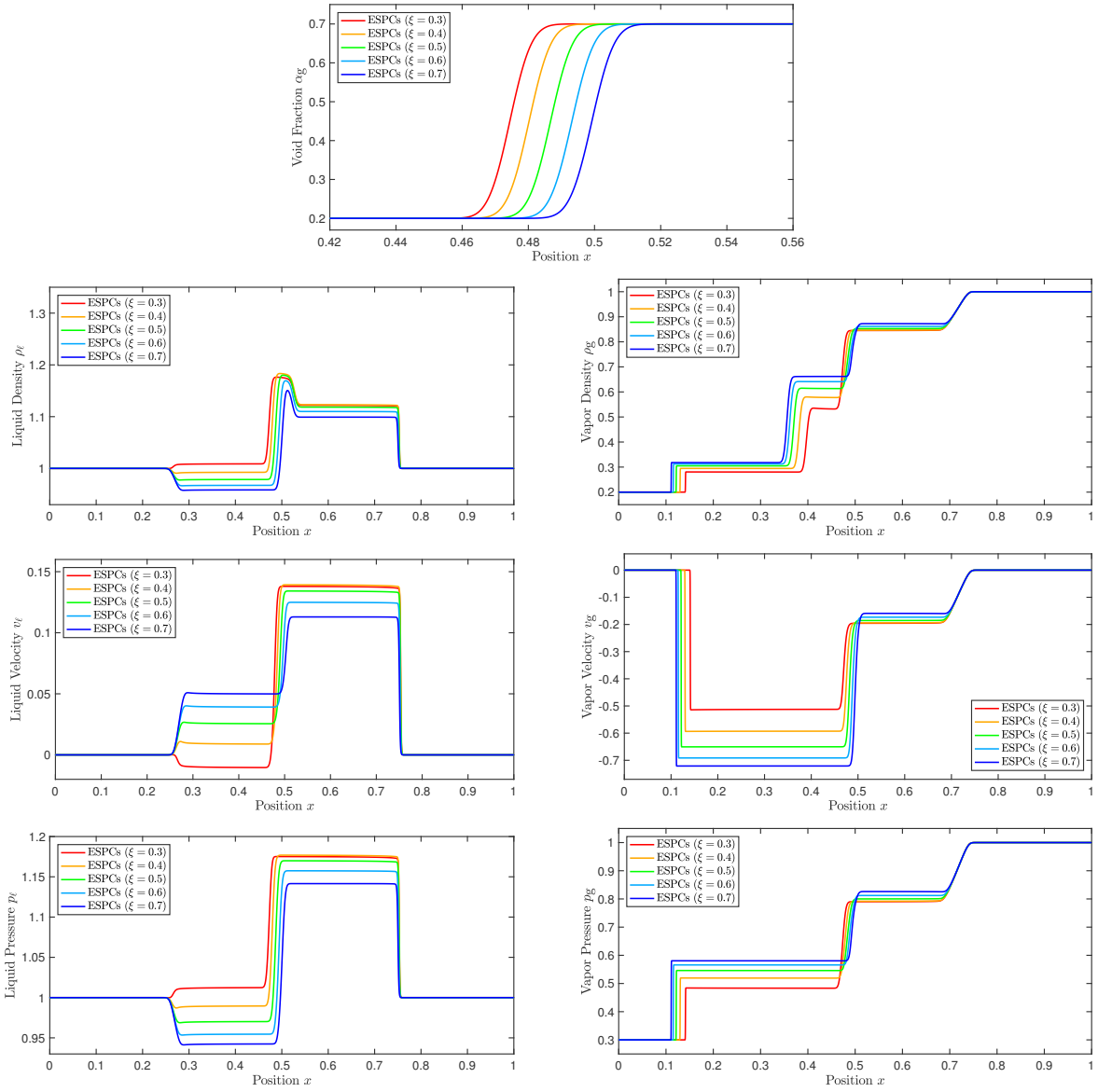


Figure 57: Mixture at rest with increase in the volume fraction using the ESPCs scheme with 16 000 cells and different values for ξ at $t = 0.2$.

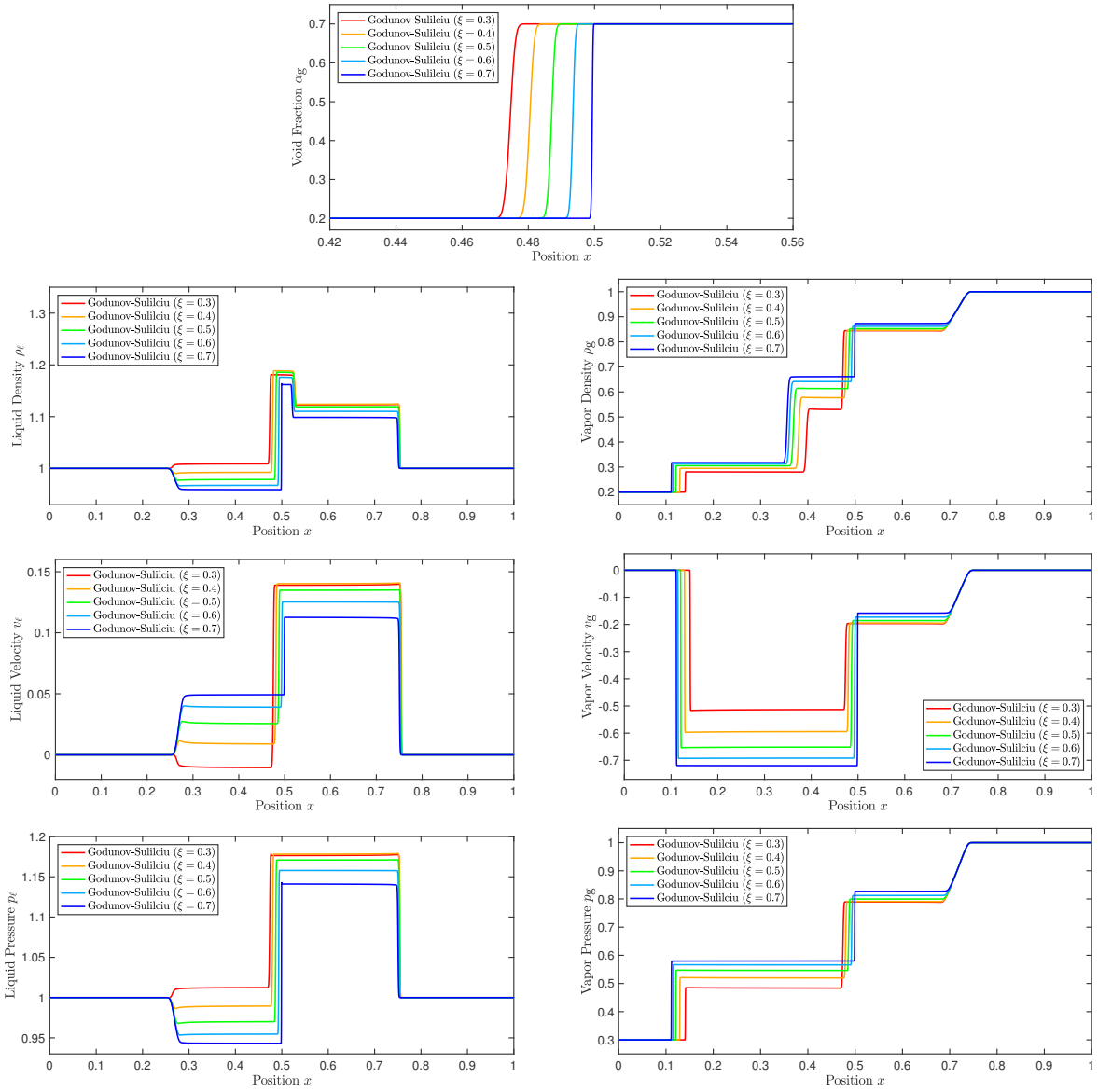


Figure 58: Mixture at rest with increase in the volume fraction using the Godunov-Suliciu scheme with 16 000 cells and different values for ξ at $t = 0.2$.

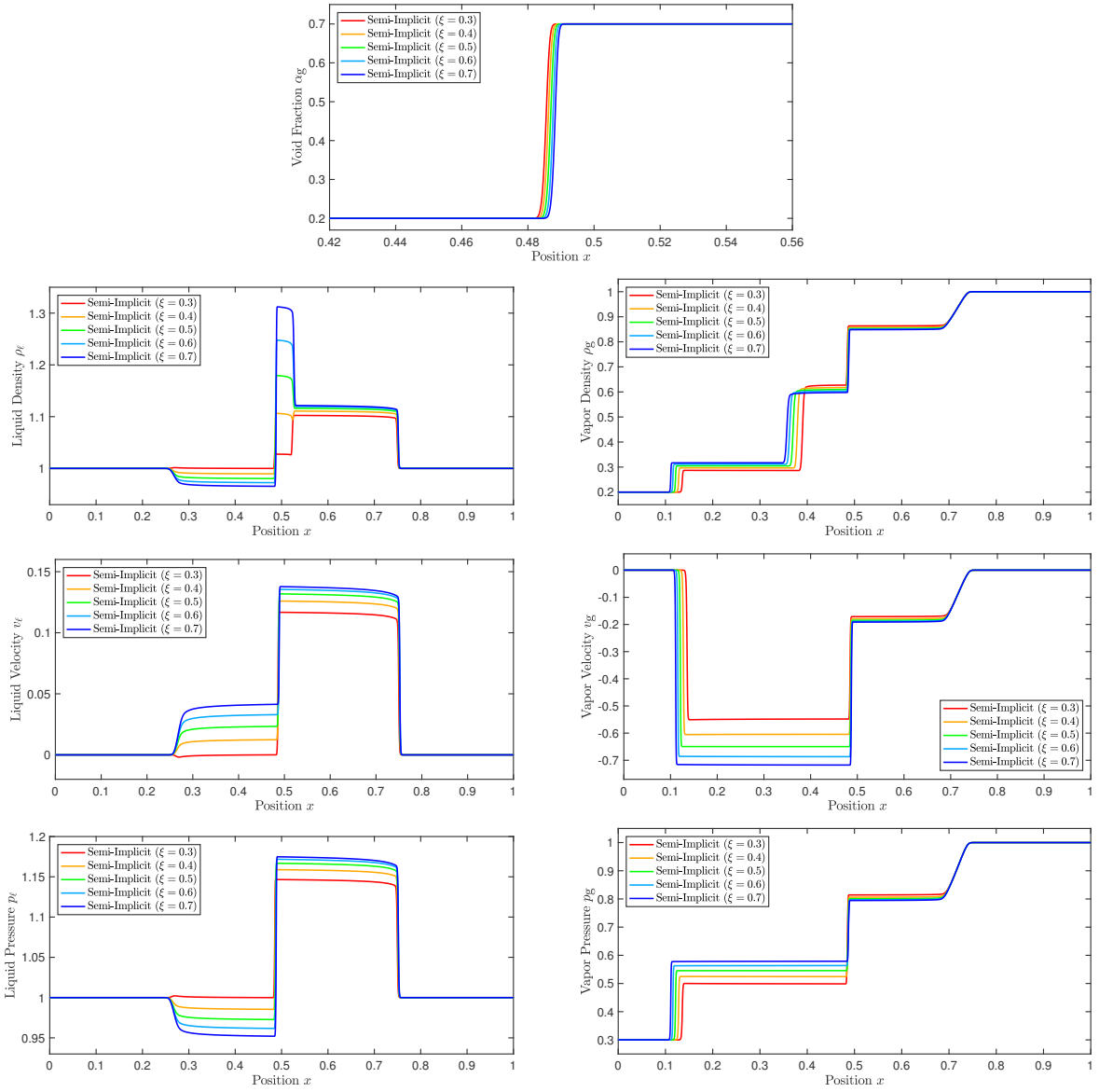


Figure 59: Mixture at rest with increase in the volume fraction using the Semi-Implicit scheme with 16 000 cells and different values for ξ at $t = 0.2$.

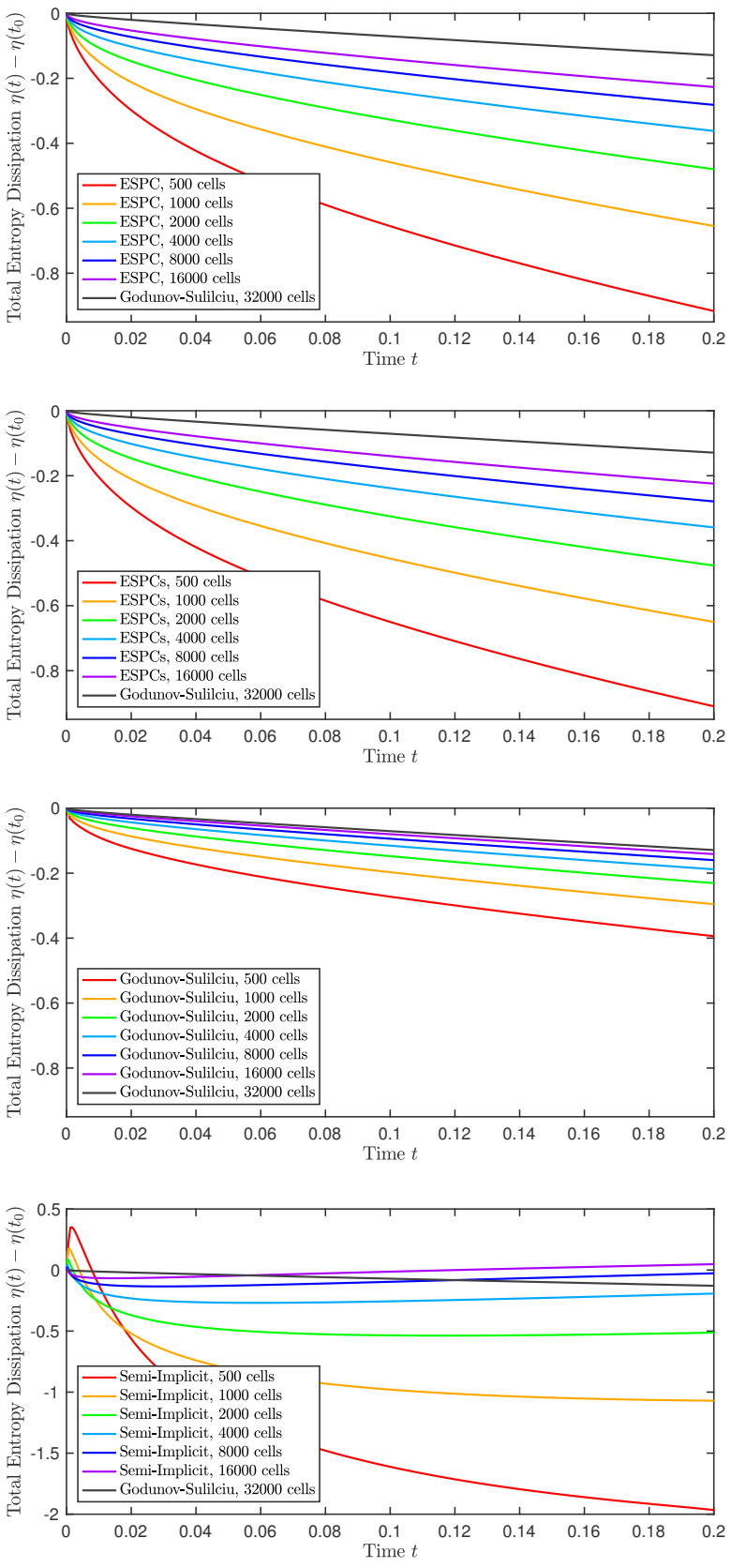


Figure 60: Total entropy dissipation of the ESPC, ESPCs, Godunov-Suliciu and Semi-Implicit scheme for different meshes.

7.6. Water faucet problem

So far, we just regarded homogeneous test cases. Because the goal is to simulate solar thermal power plants, here a test case is presented which considers mass transfer, momentum transfer and friction. The water faucet problem regards the flow through a vertical tube of 12 m in length [135], where the following source terms are considered,

$$\mathbf{s}(\mathbf{u}) = \begin{pmatrix} \frac{1}{\theta}\frac{\alpha_\ell \alpha_g}{p_\ell + p_g}(p_g - p_\ell) \\ 0 \\ \alpha_\ell \rho_\ell g \\ \alpha_\ell \rho_\ell v_\ell g + \frac{1}{\theta}\frac{\alpha_\ell \alpha_g}{p_\ell + p_g} p_\mathrm{i}(p_g - p_\ell) \\ 0 \\ \alpha_g \rho_g g \\ \alpha_g \rho_g v_g g - \frac{1}{\theta}\frac{\alpha_\ell \alpha_g}{p_\ell + p_g} p_\mathrm{i}(p_g - p_\ell) \end{pmatrix}, \tag{7.4}$$

with gravity $g = 9.81\mathrm{m/s}^2$, and pressure relaxation $\theta = 5 \cdot 10^{-4}\mathrm{s}$. The interfacial parameter is chosen as $\xi = 0.5$ which has an influence on the interfacial velocity v_i (2.123) and interfacial pressure p_i (2.94). Initially, the values of the domain $x \in [0, 12]$ are given by:

	α_g	ρ_ℓ	v_ℓ	p_ℓ	ρ_g	v_g	p_g
Initial	0.2	1000	10	10^5	1	0	10^5

This initial datum may be interpreted as a flow of water without gravity, where the gravity field is introduced with the start of the simulation. The top boundary ($x = 0$) has the same values as the initial values, where the bottom ($x = 12$) of the tube is open to atmospheric conditions [8]:

	α_g	ρ_ℓ	v_ℓ	p_ℓ	ρ_g	v_g	p_g
Top	0.2	1000	10	10^5	1	0	10^5
Bottom	–	–	–	10^5	–	–	10^5

The undefined boundary values at the bottom are taken from inside using extrapolation. As closure ideal gas with $\gamma_\ell = 1.0005$, $\gamma_g = 1.4$, and $c_{\mathrm{v}\ell} = c_{\mathrm{v}g} = 718\ \mathrm{J/kg\,K}$ is chosen [77]. For the ESPC and ESPCs scheme, the entropy/entropy flux pair is given by (2.91), with corresponding entropy variables (2.92). For the regularization uniform viscosity (4.43) is chosen.

As reference solution, the Semi-Implicit scheme with 10 000 cell is used. In Figure 64 can be seen that this reference solution is converged. In the following Figures 61 to 64 results with different resolutions for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme at time $t = 0.5$ are shown. It can be seen that all solutions converge to the reference solution, where the Godunov-Suliciu and the Semi-Implicit solver converge faster than the ESPC and ESPCs scheme. While the Godunov-Suliciu solver already shows good agreement with the reference solution by using just 300 grid cells, the ESPC and ESPCs scheme would need 8000 cells instead, compare the results for parameter α_g in Figure 62 and 63. This is due to the fact that ESPC and ESPCs are more dissipative schemes.

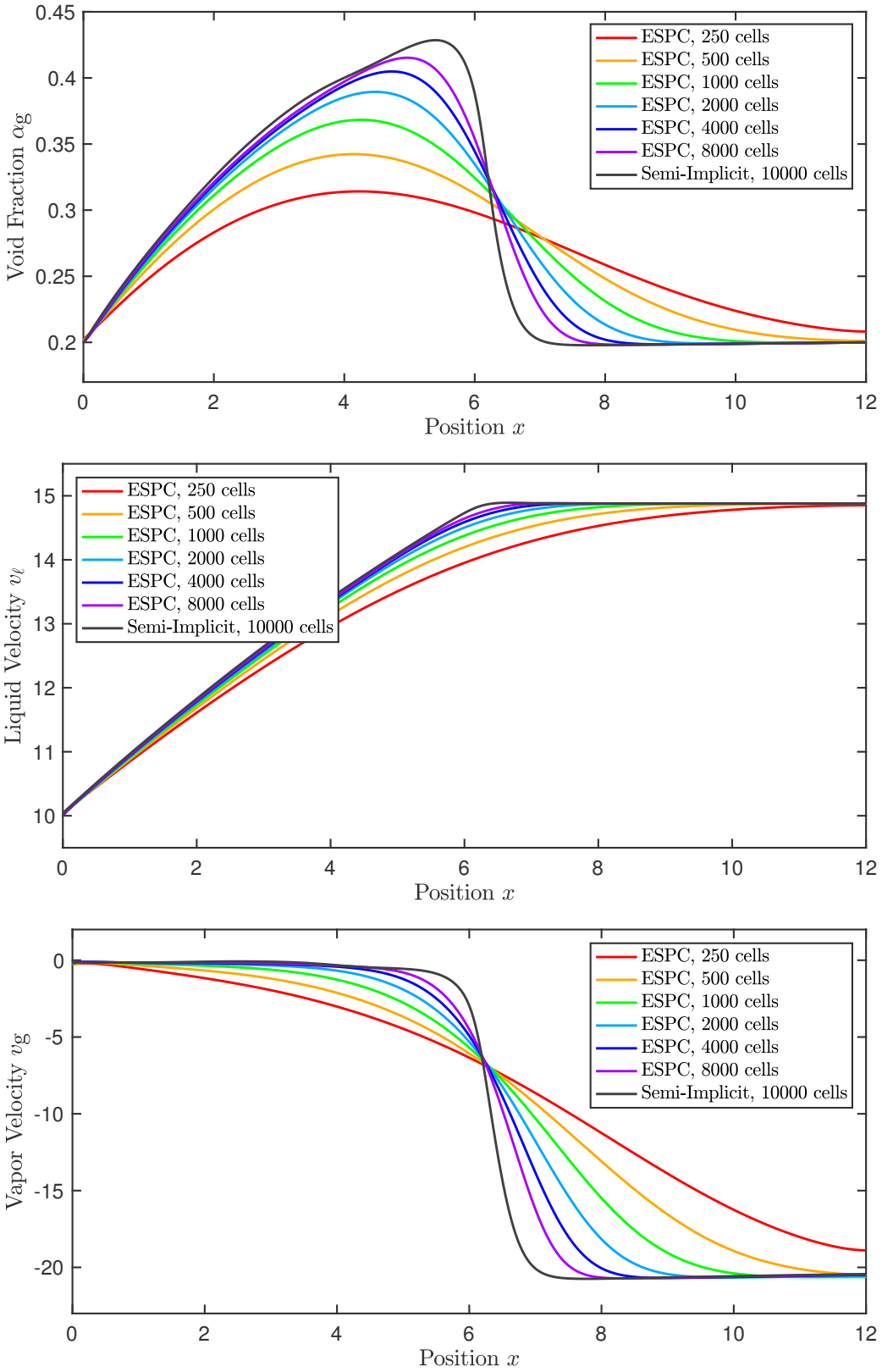


Figure 61: Water faucet problem at $t = 0.5$ using the ESPC scheme for different resolutions.

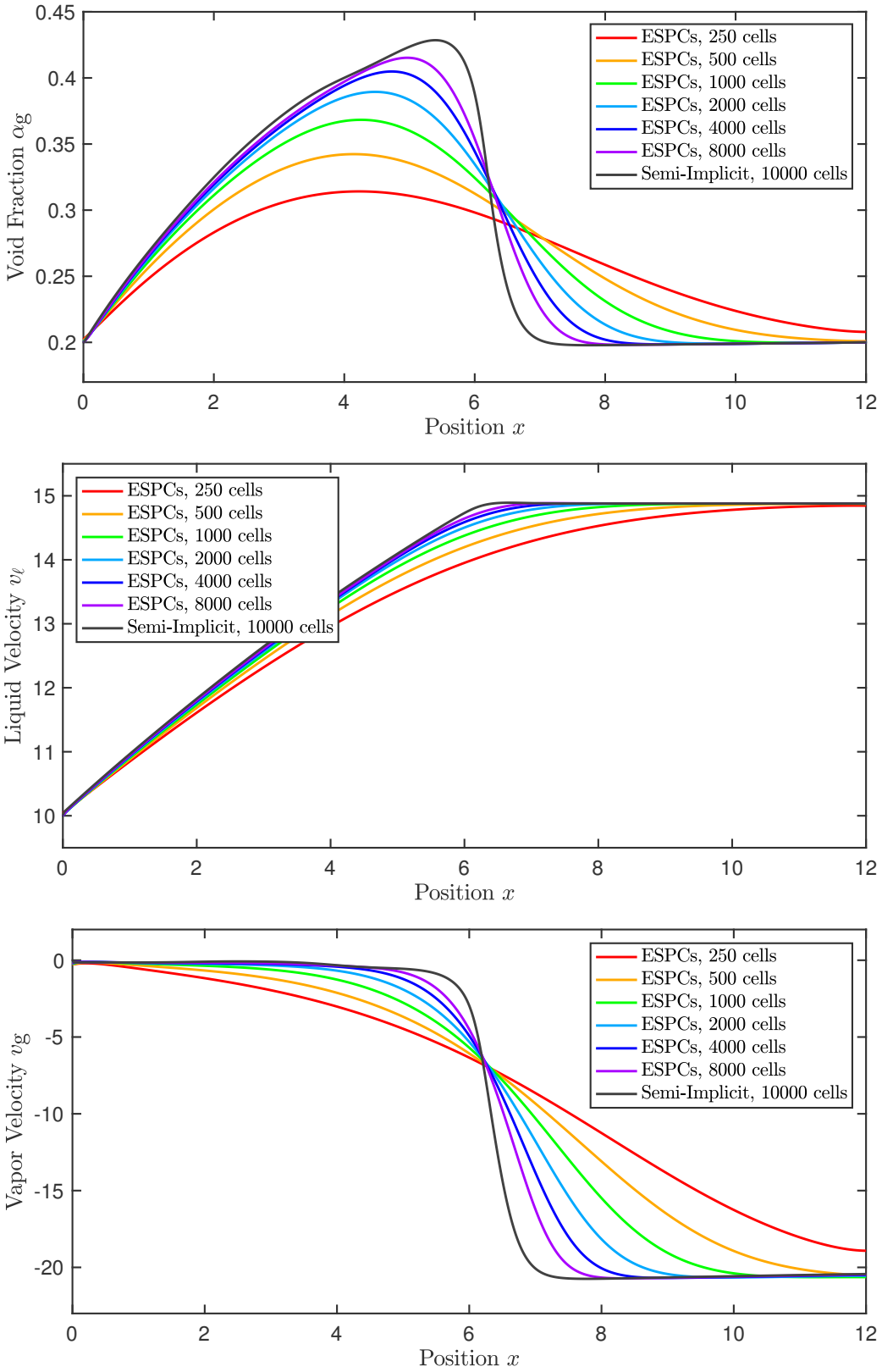


Figure 62: Water faucet problem at $t = 0.5$ using the ESPCs scheme for different resolutions.

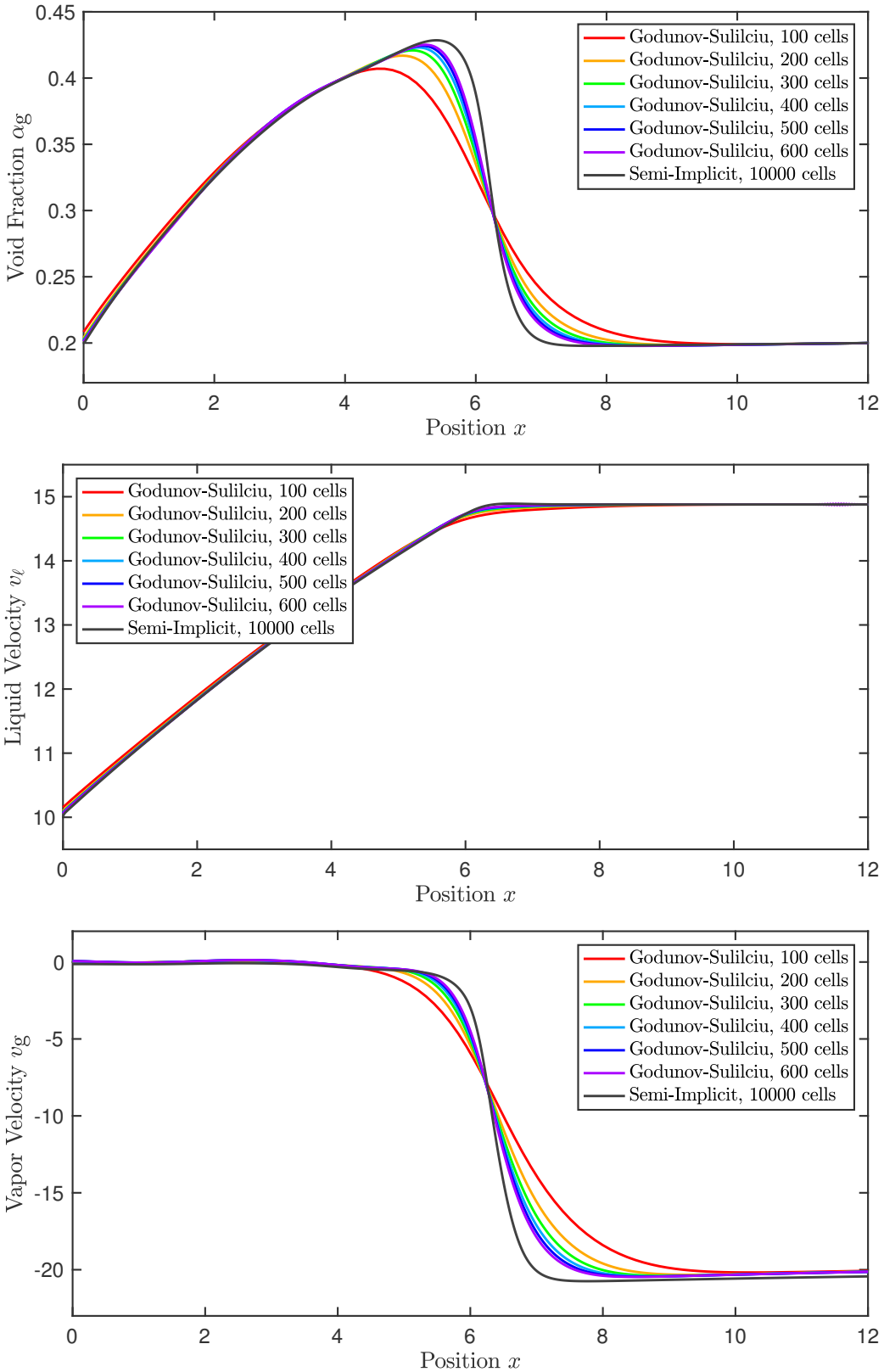


Figure 63: Water faucet problem at $t = 0.5$ using the Godunov-Suliciu scheme for different resolutions.

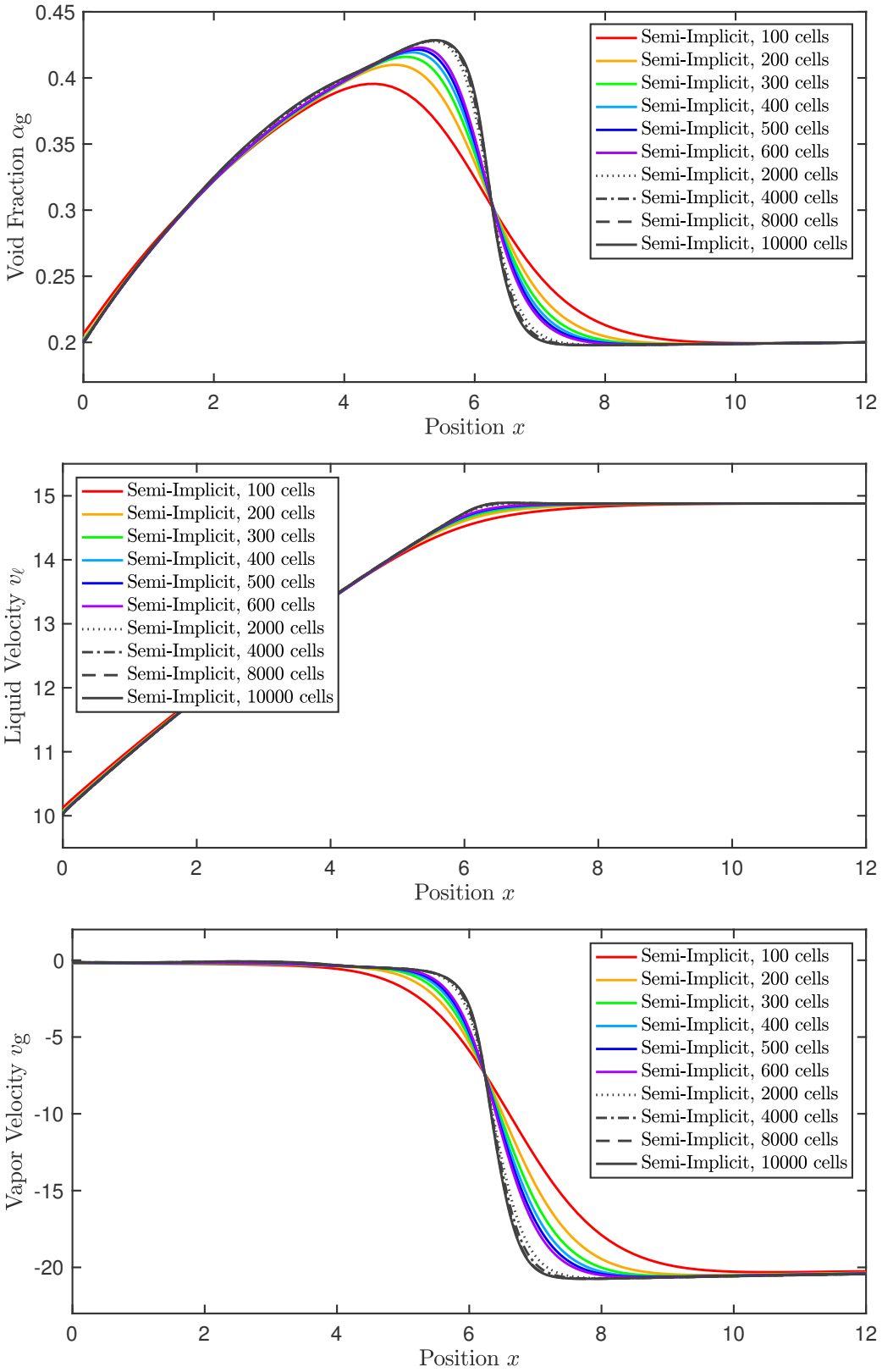


Figure 64: Water faucet problem at $t = 0.5$ using the Semi-Implicit scheme for different resolutions.

7.7. Evaporation in a solar absorber tube

To simulate the evaporation in a solar absorber tube, the source terms $\mathbf{s}(\mathbf{u})$ developed in Section 2.3 are used, which describe the mass transfer, friction and heat transfer in the tube. The single terms base on constitutive physical models, which have been developed in Section 2.4. For the underlying two-phase flow model the interfacial parameter is chosen as $\xi = 0.5$. In the following, the free parameters of the constitutive models are specified.

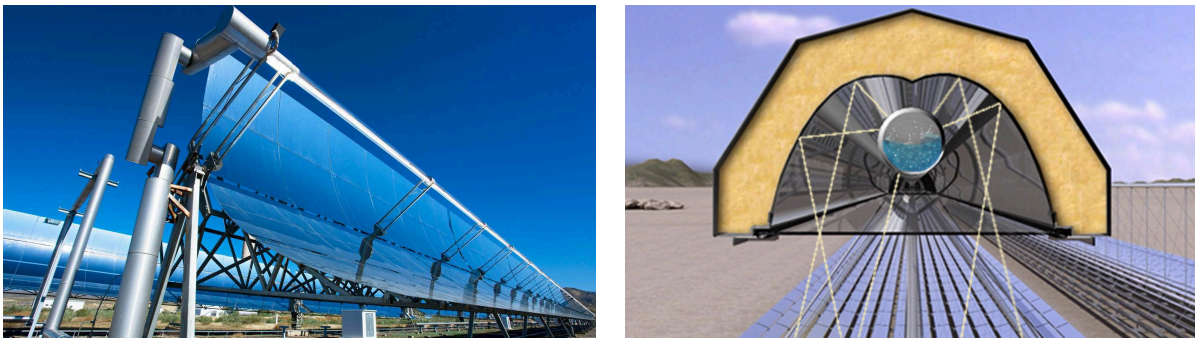


Figure 65: A parabolic trough solar power plant in Almería (left) and a Fresnel solar collector (right) using water in the absorber tubes for direct steam generation.
Sources: DLR/Ernsting and Novatec Solar GmbH

▶ Absorber tube

The water-steam fluid flows through a solar heated steel tube. The tube parameters are specified in the following Table 3:

Parameter	Value
Outer diameter D_{out}	0.070 [m]
Inner diameter D_{in}	0.050 [m]
Tube length ℓ_{w}	1000 [m]
Surface roughness ε_{w}	$0.1 \cdot 10^{-3}$ [m]
Thermal conductivity λ_{w}	38 [W/m K]
Density ρ_{w}	7500 [kg/m^3]
Specific heat capacity $c_{\text{p w}}$	540 [J/kg K]

Table 3: Absorber tube parameters.

▶ Equation of state

As equation of state stiffened gas (2.72) is used. The values for liquid water and vapor are given in Table 4. These parameter values appear to yield reasonable approximations over a temperature range from 298 to 473 Kelvin [95].
The thermal conductivity [W/m K] and dynamic viscosity [kg/m s] are given by $\lambda_\ell = 0.5$, $\lambda_g = 0.026$, $\mu_\ell = 281.8 \cdot 10^{-6}$, and $\mu_g = 134.4 \cdot 10^{-7}$. The phasic isobaric specific heats $c_{p\,k}$ are determined from the corresponding phasic isochoric specific heats as $c_{p\,k} = \gamma_k\, c_{v\,k}$.

	γ	q	q'	π	c_v
Liquid	2.35	$-1167 \cdot 10^3$	0	10^9	1816
Vapor	1.43	$2030 \cdot 10^3$	$-23 \cdot 10^3$	0	1040

Table 4: Stiffened gas parameters for liquid water and water vapor.

▶ Wall friction density

The constitutive model of the wall friction density is given by (2.76),

$$\boxed{F_{\mathrm{w}\,k} = \frac{1}{2}\rho_k v_k |v_k|\, f_{\mathrm{w}\,k}\, \frac{\alpha_{\mathrm{w}\,k}}{D_\mathrm{in}},}$$

where the wetted volume fraction in the wall film $\alpha_{\mathrm{w}\,k}$ and the wall friction factor $f_{\mathrm{w}\,k}$ need to be modeled. Usually, the wetted volume fraction in the wall film needs to be modeled in dependency of the flow pattern. Here we simply choose the volume fraction as a measure,

$$\alpha_{\mathrm{w}\,k} = \alpha_k. \tag{7.5}$$

The wall friction factor $f_{\mathrm{w}\,k}$ is modelled as a piecewise-defined function [74]:

$$f_{\mathrm{w}\,k} = \begin{cases} f_\mathrm{max}(\mathrm{Re}_{\mathrm{w}\,k}), & 0 \leq \mathrm{Re}_{\mathrm{w}\,k} < 64 \\ f_\mathrm{lam}(\mathrm{Re}_{\mathrm{w}\,k}), & 64 \leq \mathrm{Re}_{\mathrm{w}\,k} < 2200 \\ f_\mathrm{trans}(\mathrm{Re}_{\mathrm{w}\,k}), & 2200 \leq \mathrm{Re}_{\mathrm{w}\,k} < 3000 \\ f_\mathrm{turb}(\mathrm{Re}_{\mathrm{w}\,k}), & 3000 \leq \mathrm{Re}_{\mathrm{w}\,k} \end{cases} \tag{7.6}$$

with phasic Reynolds number

$$\mathrm{Re}_{\mathrm{w}\,k} = \frac{|v_k|\rho_k}{\mu_k} \cdot \frac{\alpha_k D_\mathrm{in}}{\alpha_{\mathrm{w}\,k}}, \tag{7.7}$$

where the last term is called hydraulic diameter. The laminar friction factor is calculated as

$$f_\mathrm{lam}(\mathrm{Re}_{\mathrm{w}\,k}) := \frac{64}{\mathrm{Re}_{\mathrm{w}\,k}}, \tag{7.8}$$

and is limited by the f_max friction factor

$$f_\mathrm{max}(\mathrm{Re}_{\mathrm{w}\,k}) := f_\mathrm{lam}(64) \equiv 1. \tag{7.9}$$

The turbulent friction factor is usually given by an approximation of the Colebrook-White equation [137]. We are using the approximation of Zigrang and Sylvester [138],

$$f_\mathrm{turb}(\mathrm{Re}_{\mathrm{w}\,k}) := \left(-2\log_{10}\left(\frac{\varepsilon_\mathrm{w}}{3.7 D_\mathrm{in}} + \frac{2.51}{\mathrm{Re}_{\mathrm{w}\,k}}\left(1.14 - 2\log_{10}\left(\frac{\varepsilon_\mathrm{w}}{D_\mathrm{in}} + \frac{21.25}{\mathrm{Re}_{\mathrm{w}\,k}^{0.9}}\right)\right)\right)\right)^{-2} \tag{7.10}$$

with surface roughness ε_{w} of the inner tube wall. To link the laminar and turbulent flow region an interpolation between the two states at the border of the laminar and turbulent region is used [74],

$$f_{\mathrm{trans}} = (1 - \xi_f) \cdot f_{\mathrm{lam}}(2200) + \xi_f \cdot f_{\mathrm{turb}}(3000) \tag{7.11}$$

with weight

$$\xi_f = \frac{3000\,(\mathrm{Re}_{\mathrm{w}\,k} - 2200)}{\mathrm{Re}_{\mathrm{w}\,k}\,(3000 - 2200)}. \tag{7.12}$$

▶ Interphase friction density

The constitutive model of the wall friction density is given by (2.77),

$$\boxed{F_{\mathrm{i}} \overset{(2.43)}{:=} -F_{\mathrm{i}\,g} = F_{\mathrm{i}\,\ell} = \frac{1}{2}\rho_{\mathrm{cont}}(v_g - v_\ell)|v_g - v_\ell| \cdot C_{\mathrm{D}} \cdot \frac{1}{4}a_{\mathrm{i}},}$$

with density $\rho_{\mathrm{cont}} = \alpha_g \rho_g$, drag coefficient $C_{\mathrm{D}} = 0.05$, and specific interfacial area as

$$a_{\mathrm{i}} = \frac{3.6 \cdot \alpha_g}{D_{\mathrm{in}}}, \tag{7.13}$$

▶ Wall temperature

Usually the wall temperature is computed by the concentrated external solar irradiation and its losses, see Section 2.4.6. For this test case the wall temperature is chosen to increase linearly from 500 to 550 Kelvin,

$$T_{\mathrm{w}}(x) = 500\,[\mathsf{K}] + 50\,[\mathsf{K}] \cdot \frac{x}{1000\,[\mathsf{m}]}. \tag{7.14}$$

▶ Wall heat transfer rate density

The constitutive model for the wall heat transfer rate density is given by (2.80)

$$\boxed{Q_{\mathrm{w}\,k} := \frac{\alpha_{\mathrm{w}\,k}}{D_{\mathrm{in}}} \cdot h_{\mathrm{T,w}\,k} \cdot \big(T_{\mathrm{w}} - T_{\mathrm{w}\,k}\big),}$$

with its partitioning (2.81)

$$\boxed{Q_{\mathrm{w}\,k}^{\mathrm{boil}} = (1 - \beta_k)\,Q_{\mathrm{w}\,k} \quad \text{and} \quad Q_{\mathrm{w}\,k}^{\mathrm{conv}} = \beta_k\,Q_{\mathrm{w}\,k}.} \tag{7.15}$$

The convective heat transfer coefficient $h_{\mathrm{T,w}\,k}$ from wall to phase k and the corresponding temperature $T_{\mathrm{w}\,k}$ in the wall film are chosen as

$$h_{\mathrm{T,w}\,k} = \frac{4.36 \cdot \lambda_k}{D_{\mathrm{in}}}, \tag{7.16}$$

and

$$T_{\mathrm{w}\,k} = T_k. \tag{7.17}$$

For the partitioning into boiling and convection heat transfer, the parameter β_k has to be modeled. For the vapor phase we choose $\beta_g \equiv 1$ which means, that the boiling part is always zero. For the liquid phase, a distinction is made, if a subcooled or superheated wall is there [74],

$$\beta_\ell = \min\Big(1,\ \exp\big(-\tfrac{1}{4}(T_{\mathrm{w}} - T_{\mathrm{sat}\,\ell})\big)\Big). \tag{7.18}$$

▶ Interphase heat transfer rate density (in the bulk)

The constitutive model for the interphase heat transfer rate density (in the bulk) is given by (2.82)

$$\boxed{Q_{\mathrm{i}\,k} = a_{\mathrm{i}}\, h_{\mathrm{T,i}\,k}\big(\widehat{T}_{\mathrm{i\,sat}} - T_k\big),}$$

The convective heat transfer coefficient is chosen as

$$h_{\mathrm{T,i}\,k} = \frac{\mathrm{Nu}_k \cdot \lambda_k}{D_{\mathrm{bub}}}. \tag{7.19}$$

with bubble diameter

$$D_{\mathrm{bub}} = \frac{6\alpha_g}{a_{\mathrm{i}}}. \tag{7.20}$$

While for the vapor phase the Nusselt number is chosen constant with $\mathrm{Nu}_\ell = 2.5$, it is given for the liquid phase by

$$\mathrm{Nu}_\ell = 2 + 0.6\ \mathrm{Re}_\ell^{0.5}\,\mathrm{Pr}_\ell^{0.33}, \tag{7.21}$$

with Reynolds and Prandtl number

$$\mathrm{Re}_\ell = \frac{\rho_\ell D_{\mathrm{bub}} |v_\ell - v_g|}{\mu_\ell} \quad \text{and} \quad \mathrm{Pr}_\ell = \frac{\mu_\ell\, c_{\mathrm{p}\,\ell}}{\lambda_\ell}. \tag{7.22}$$

▶ Interfacial mass transfer rate

The interfacial mass transfer rate is defined in Section 2.4.2 by a heat conduction limited model:

$$\boxed{\Gamma_{\mathrm{i}} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}}\big(F_{\mathrm{i}}(v_\ell - v_g) + Q_{\mathrm{i}\,\ell} + Q_{\mathrm{i}\,g}\big),}$$

and

$$\boxed{\Gamma_{\mathrm{w}} := \frac{1}{\widehat{E}_{\mathrm{i}\,\ell} - \widehat{E}_{\mathrm{i}\,g}}\big(-Q^{\mathrm{boil}}_{\mathrm{w}\,\ell} - Q^{\mathrm{boil}}_{\mathrm{w}\,g}\big).}$$

▶ Initial values

Initially, the values of the domain $x \in [0, 1000]$ are given by a pressure of 30 bar, a temperature of $T_\ell = 471\,[\mathsf{K}]$ and $T_g = 475\,[\mathsf{K}]$ and a flow velocity of $v_\ell = 1\,[\mathsf{m/s}]$ and $v_g = 18\,[\mathsf{m/s}]$:

	α_g	T_ℓ	v_ℓ	p_ℓ	T_g	v_g	p_g
Initial	0.02	471	1	$30 \cdot 10^5$	475	18	$30 \cdot 10^5$

The temperatures are chosen, such that they are close to the saturation temperature. This initial datum may be interpreted as a flow of water without mass transfer, friction and heat tranfer in the tube. With the start of the simulation, the source terms start to influence the solution.

▶ Boundary conditions

From the eigenspace we know, that the solution information propagates along seven wave directions. For the flow in a tube we can assume, that the fluid is flowing with a positive velocity, such that two characteristics are facing backwards and five forward in flow direction. For the case of a solar power plant it is the condenser which determines the outlet pressure. At the inlet the phasic temperatures and velocities are given by the cooling part of the power plant and the pump.

The boundary values on the left hand side ($x = 0$) and right hand side ($x = 1000$) are the same values as the initial values, where the undefined boundary values are taken from inside using extrapolation.

	α_g	T_ℓ	v_ℓ	p_ℓ	T_g	v_g	p_g
Left inflow boundary	0.02	471	1	–	475	18	–
Right outflow boundary	–	–	–	$30 \cdot 10^5$	–	–	$30 \cdot 10^5$

▶ Numerical results

In Figure 66 the temporal convergence to the steady state solution for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme are shown. The temporal residual for quantities of the vapor phase converges faster than the temporal residual for quantities of the liquid phase. But the temporal residuals of the vapor quantities do not reach a value of zero, they just stagnate. This is already known for explicit methods due to the time step restriction. Furthermore, the liquid quantities are still not stationary, which not necessarily means that there will be larger changes. A surprising observation is that the temporal residual of the ESPC and ESPCs schemes converges more than two times faster than the Godunov-Suliciu and the Semi-Implicit scheme. This can also be seen in Figure 67, where the temporal convergence to the steady state solution of the volume fraction α is shown.

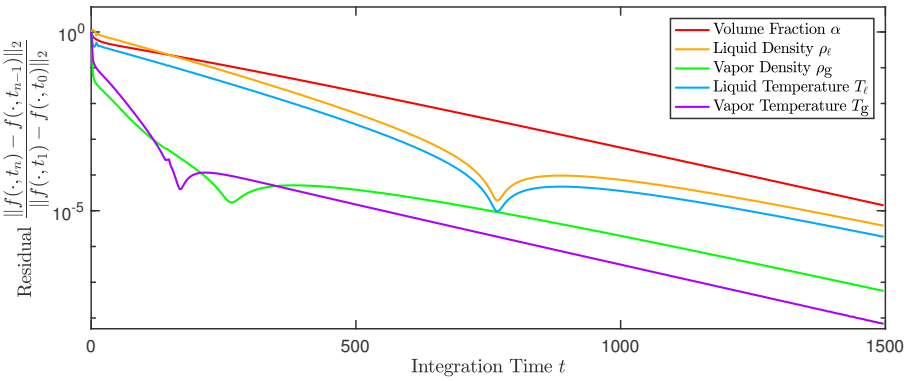
In Figure 68 the steady state results are shown for an integration time of $t = 500$ seconds. Due to low order and high numerical disspiation of the ESPC and ESPCs scheme we use here a higher resolution of 900 grid cells in comparison to 300 grid

cells for the Godunov-Suliciu and Semi-Implicit scheme. The liquid and vapor pressure decrease due to the influence of the friction, and the liquid temperature increases due to heat transfer from the hot tube wall and from the vapor phase. This directly influences the phasic densities which therefore decrease.

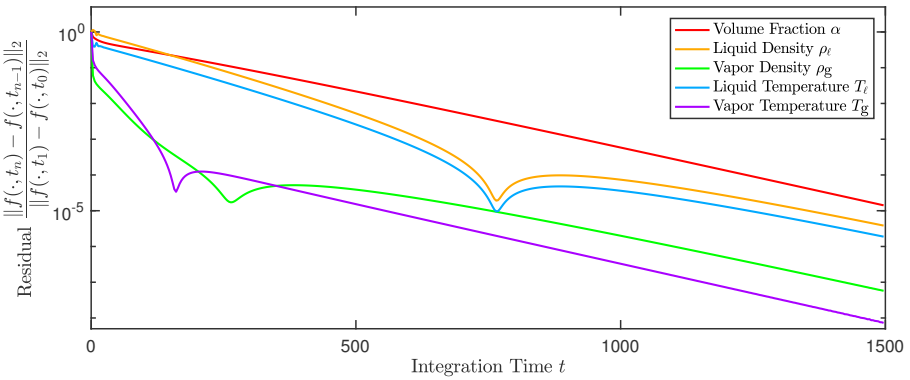
The following observations can be made: At first, it would be expected, that more vapor is generated inside the absorber tube. This can be understood as a hint for using more accurate closures of the constitutive models which influence the interfacial mass transfer rates Γ_{i} and Γ_{w}. There are more complex models, e.g. those used in the nuclear thermal-hydraulic code RELAP [74], which consider the flow regime for the interaction between the phases and Nukiyama's boiling curve for the wall heat transfer. Second, it is interesting that the path-conservative schemes (ESPC and ESPCs) generate once again similar results in the eyeball norm. The same holds for the Godunov-Suliciu and the Semi-Implicit scheme. But it is curious, that both different solution families show such different results. Only the results for the liquid pressure p_ℓ and the velocities v_ℓ and v_g show a good agreement. A reason for the uneven results may be the observation which we already did in the water faucet test case in Section 7.6. There we remarked, that the ESPC and ESPCs scheme need about 25 times more grid cells than the Godunov-Suliciu and Semi-Implicit schemes, to produce results of the similar quality (300 grid cells of Godunov-Suliciu vs. 8000 grid cells of ESPCs). Thus, the path-conservative results have not converged so far in space because they are more dissipative.

This can be seen for the mesh refinement for the ESPCs and Godunov-Suliciu scheme, see Figure 69 and 70. For the Godunov-Suliciu solver the solution with 300 grid cells already converged to the steady state solution. The solutions of the ESPCs scheme move into the direction of the Godunov-Suliciu solution, but some are still far away from convergence[20]. Taking a closer look at the solutions of e.g. the vapor density ρ_g and vapor temperature T_g, it can be seen that they are already close to the Godunov-Suliciu solution, see Figure 71.

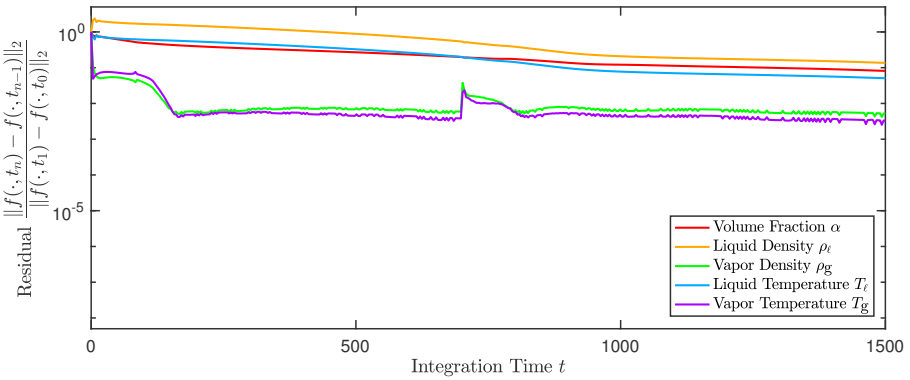
[20]Due to computation time restrictions the mesh refinement of the ESPCs scheme is done up to a factor of three (900 grid cells) in comparison to the converged mesh size (300 grid cells) of the Godunov-Suliciu solver.



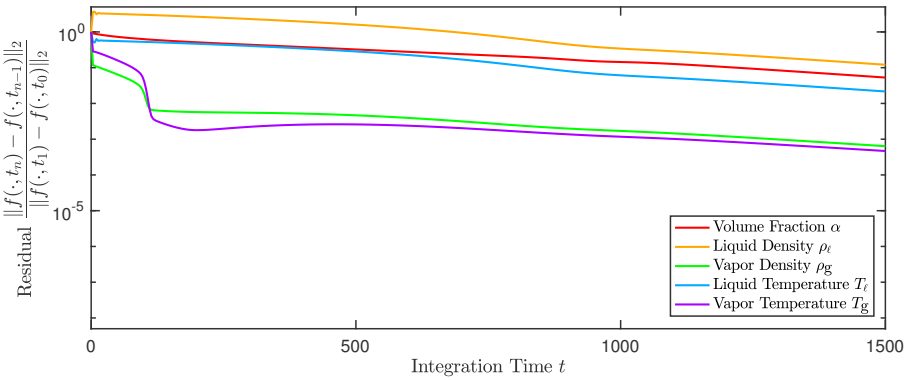
ESPC



ESPCs



Godunov-Suliciu



Semi-Implicit

Figure 66: Evaporation in a solar absorber tube. Temporal convergence to the steady state solution for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme with a maximum simulation time of $t_{\max} = 1500$ seconds for physical values $f(x,t)$ with $f = \alpha, \rho_\ell, \rho_g, T_\ell, T_g$.

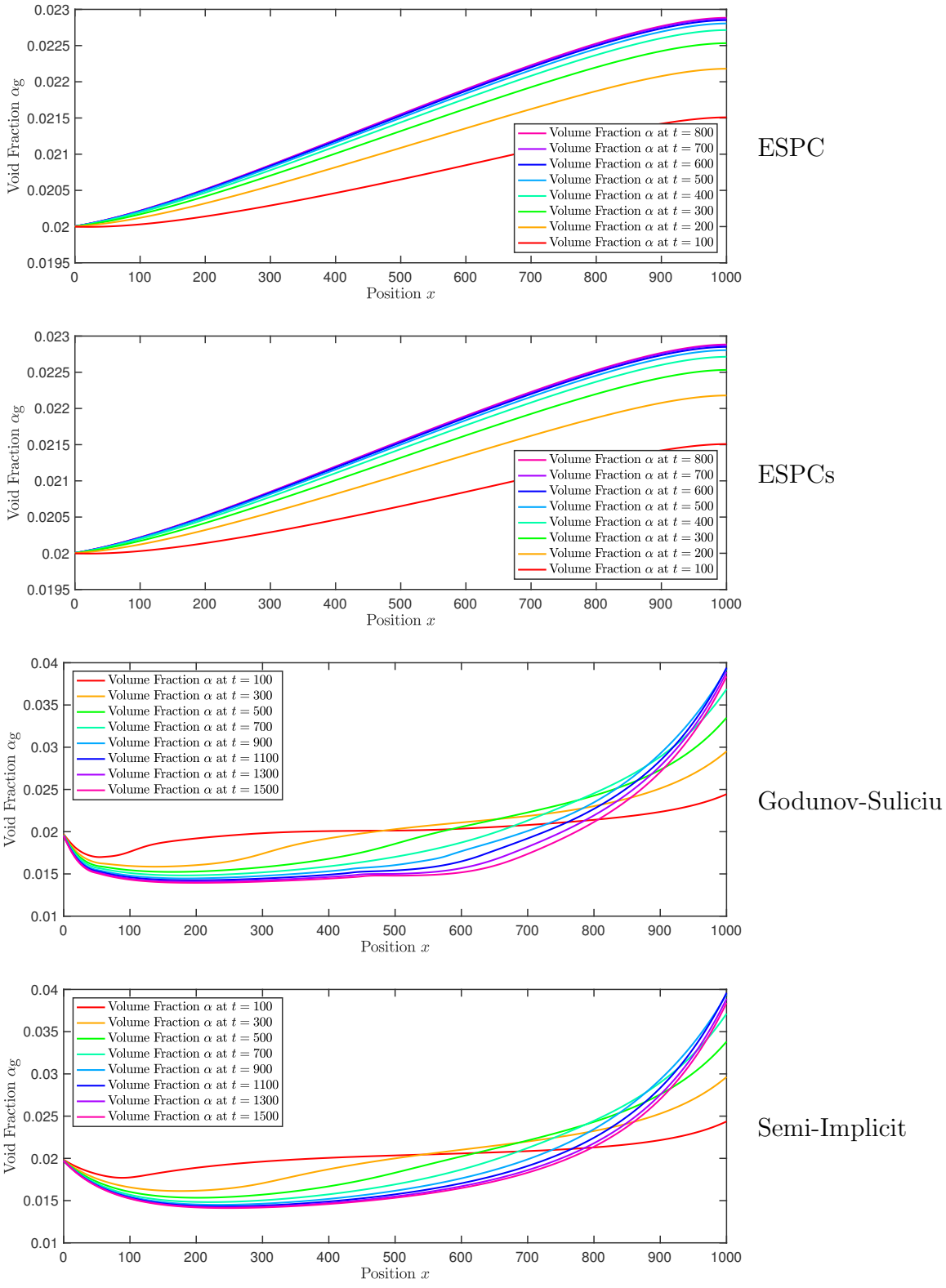


Figure 67: Evaporation in a solar absorber tube. Temporal convergence to the steady state solution of the volume fraction α for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme.

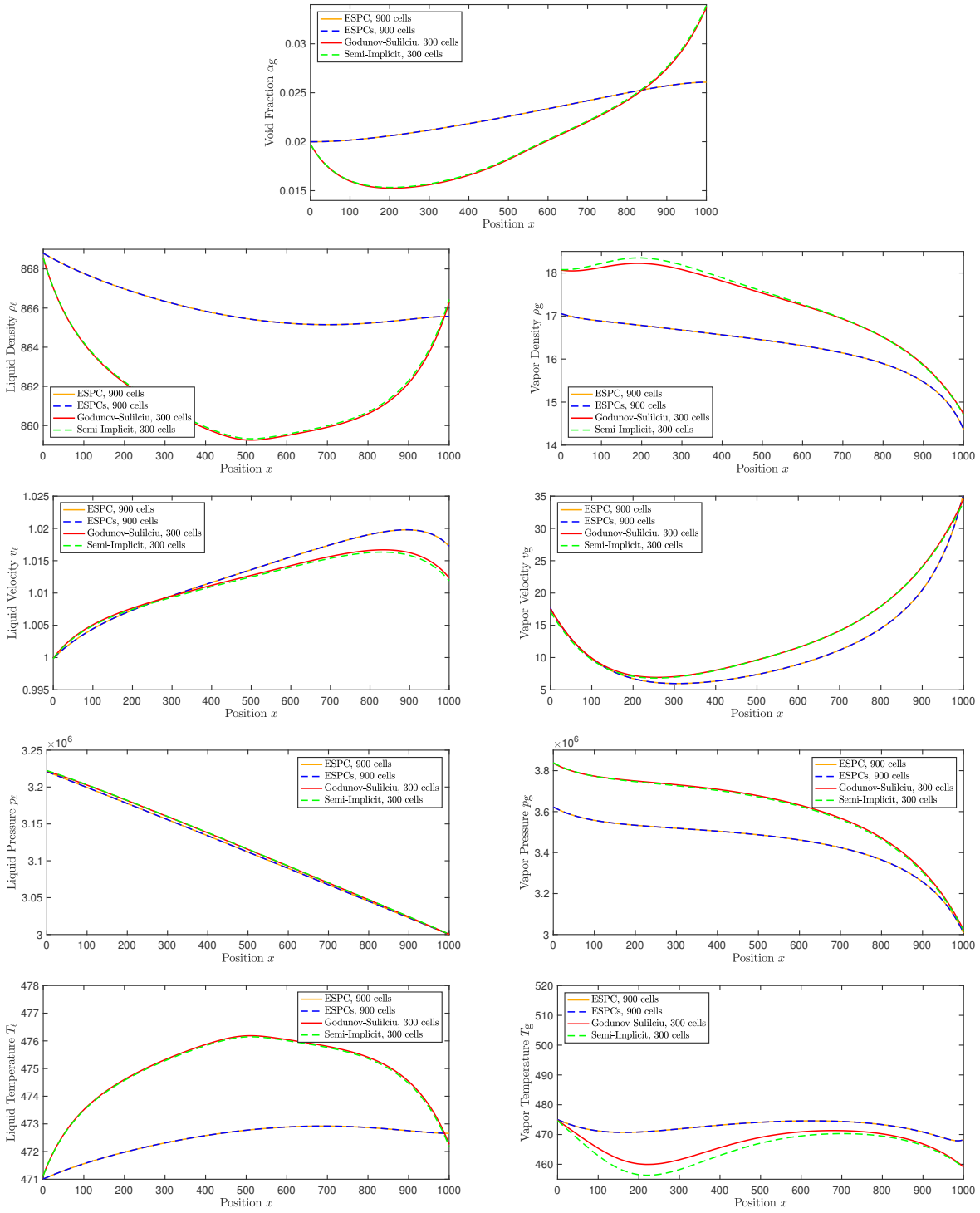


Figure 68: Evaporation in a solar absorber tube using the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme with 900 grid cells for $t = 500$ seconds.

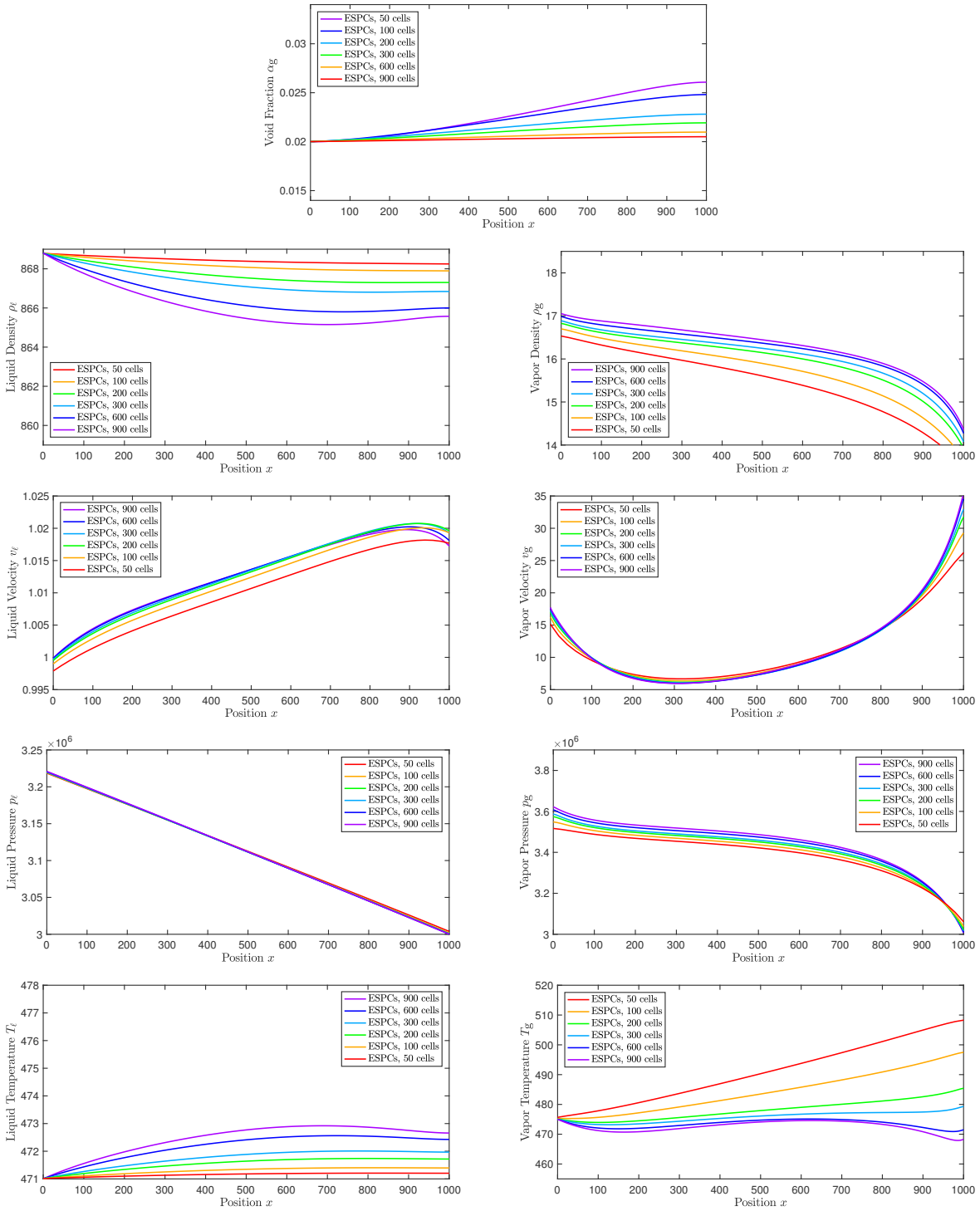


Figure 69: Grid refinement results of the ESPCs scheme for the evaporation in a solar absorber tube at $t = 500$.

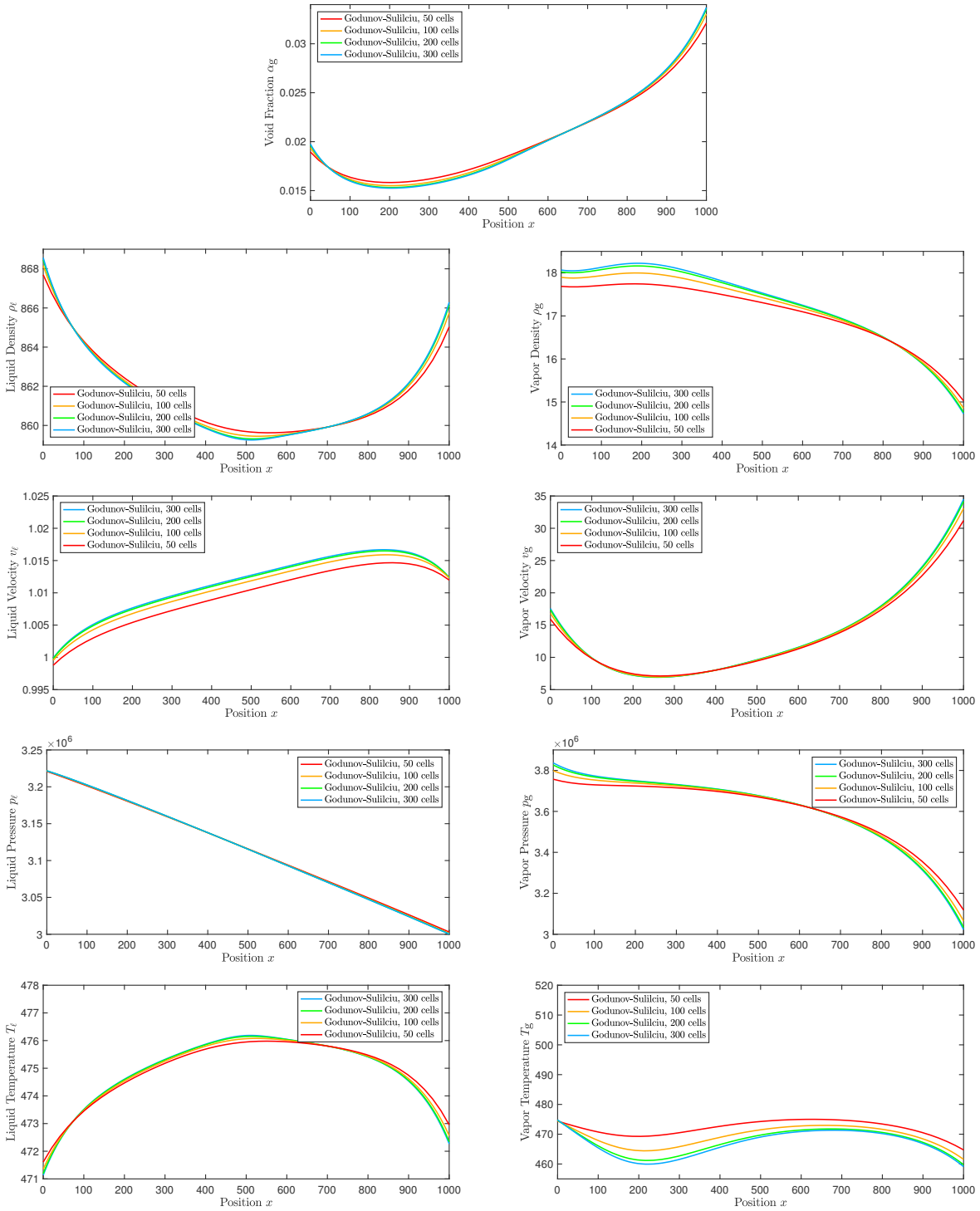


Figure 70: Grid refinement results of the Godunov-Suliciu scheme for the evaporation in a solar absorber tube at $t = 500$.

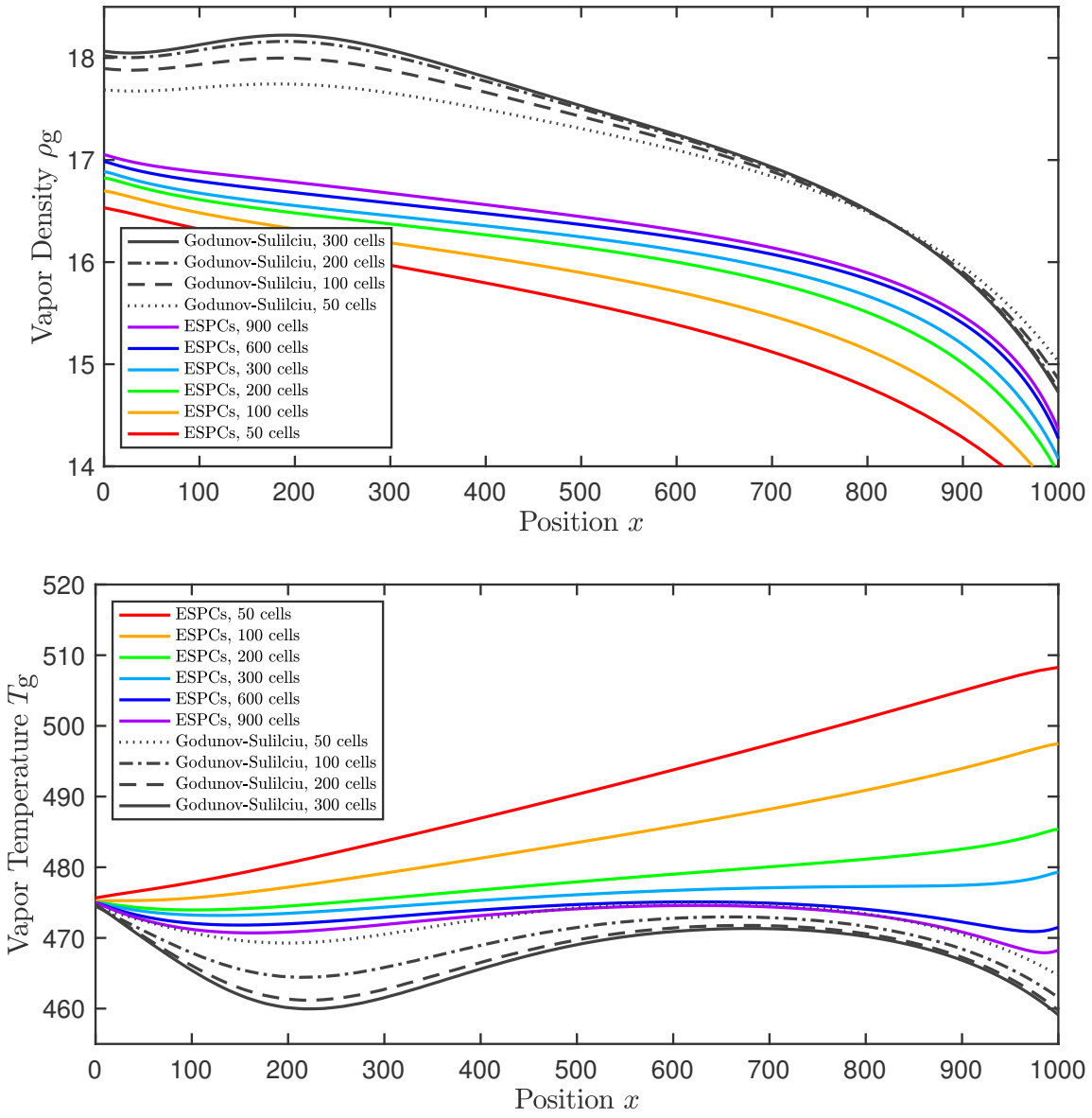


Figure 71: Spatial convergence of the vapor density and vapor temperature computed with the ESPCs scheme for the evaporation in a solar absorber tube at $t = 500$.

7.8. Analysis of the results

Within this Section, we compared the four schemes ESPC, ESPCs, Godunov-Suliciu and Semi-Implicit. In several test cases the qualitative behavior of the schemes has been investigated.

There are just minor differences between the ESPC and ESPCs scheme. The main difference has been found in the pressure disequilibrium test case in Section 7.4, where the ESPC scheme created a small kink at the shock position. In all other test cases both schemes created the same results (in the eyeball norm). Remember, that the only difference between both schemes is the path: while ESPC takes the straight path along the state variables, ESPCs takes the straight path along the entropy variables. This underlines once again the statement of Castro et al. [108], that for practical purposes it is often sufficient to use a simple path. Incidentally, the solution of a path-conservative scheme depends heavily on the numerical viscosity, e.g. compare with the results of the coupled Burgers equation in Section 7.1. For the two-phase flow test cases it can be seen that the path-conservative schemes not necessarily show a smooth behavior of the Riemann invariants, see Section 7.4. In contrast, the Godunov-Suliciu and Semi-Implicit scheme fulfill at least six of eight Rankine-Hugoniot conditions at the v_i contact discontinuity.

The convergence speed of the schemes is investigated for the isolated coupling wave test case from Section 7.3. Because the exact solution of the volume fraction α is known, this parameter is used as a measure for the accuracy in the L_1, L_2 and L_∞ norm. In Figure 72, the L_2-error is plotted against the number of grid cells. It can be seen that the empirical order of convergence is the same for all schemes, whereas the path-conservative schemes are more dissipative. The accuracy of the different schemes related to the CPU run-time is depicted in Figure 73. It can be seen, that the convergence speed of the Godunov-Suliciu and the Semi-Implicit scheme are higher than the convergence speed of the ESPC and ESPCs scheme. It is clear, that the run-time highly depends on the implementation. Here we want to point out, that no code optimization has been done so far, e.g. every grid cell evaluates both fluxes without re-using information from the neighboring cell. The path-conservative schemes approximate the integral along the path with a numerical quadrature rule. Thus, here the run-time depends on the number of chosen evaluation points. To speed-up this part, it is recommended to write the system in the form (3.1), where the conservative terms are stored in integral form in the flux vector $f(\mathbf{u})$, and the non-conservative terms are stored in matrix $B(\mathbf{u})$, which is less dense[21] than $A(\mathbf{u})$.

It is interesting to see, that the Godunov-Suliciu is faster than the Semi-Implicit scheme. This observation highly depends on the Mach number of the test case. For the isolated coupling wave stiffened gas was used in a region, where the speed of sound is just twice as fast as the flow velocity. Thus, the Semi-Implicit scheme needs just half of the time steps in comparison to the Godunov-Suliciu solver. But solving the implicit equation

[21] In the case of the two-velocity two-pressure model, $B(\mathbf{u}) \in \mathbb{R}^{7\times 7}$ has just 5 non-zero entries, see (2.60), where $A(\mathbf{u})$ has 17 entries which need to be computed (2.70).

system needs more than twice the time of one Godunov-Suliciu time step. Thus, for applications with lower Mach numbers than 0.5, we expect that the Semi-Implicit scheme will be faster than the Godunov-Suliciu solver. For the solar application with real water equations we expect a Mach number of $\mathcal{O}(10^{-2})$.

The total entropy dissipation for all schemes was investigated for Sod's shock tube test case in Section 7.2. While the ESPC and ESPCs schemes behave according to their their entropy stable property, the Semi-Implicit scheme surprisingly is very inaccurate and even generates entropy for coarser meshes. Fortunately, also the Godunov-Suliciu scheme dissipates entropy with a high convergence rate. This observation was once again confirmed for the two-phasic mixture at rest test case in Section 7.5.

For the two test cases with source terms, the water faucet problem in Section 7.6 and the evaporation in a solar absorber tube in Section 7.7, it was observed that the ESPC and ESPCs schemes need a finer mesh than the Godunov-Suliciu and Semi-Implicit scheme.

Number of	ESPC			ESPCs		
grid cells	L_1	L_2	L_∞	L_1	L_2	L_∞
500	0.026474	0.055776	0.0003962	0.026481	0.055784	0.00039638
1000	0.01878	0.046908	0.00019866	0.018784	0.046913	0.00019873
2000	0.013281	0.039445	9.9525e-05	0.013283	0.039448	9.9551e-05
4000	0.0093911	0.03317	4.9832e-05	0.0093921	0.033171	4.9842e-05
8000	0.0066405	0.027892	2.4941e-05	0.006641	0.027893	2.4944e-05

Number of	Godunov-Suliciu			Semi-Implicit		
grid cells	L_1	L_2	L_∞	L_1	L_2	L_∞
500	0.0076614	0.029954	0.0003908	0.0076614	0.029954	0.0003908
1000	0.005419	0.025194	0.00019675	0.005419	0.025194	0.00019675
2000	0.0038323	0.021188	9.885e-05	0.0038323	0.021188	9.885e-05
4000	0.00271	0.017818	4.9593e-05	0.00271	0.017818	4.9593e-05
8000	0.0019164	0.014984	2.4856e-05	0.0019164	0.014984	2.4856e-05

Table 5: Convergence analysis of the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme using the L_1, L_2, and L_∞ norm to measure the error. The L_p error of the ESPC and ESPCs scheme are similar. Also the Godunov-Suliciu and Semi-Implicit scheme have an similar L_p error.

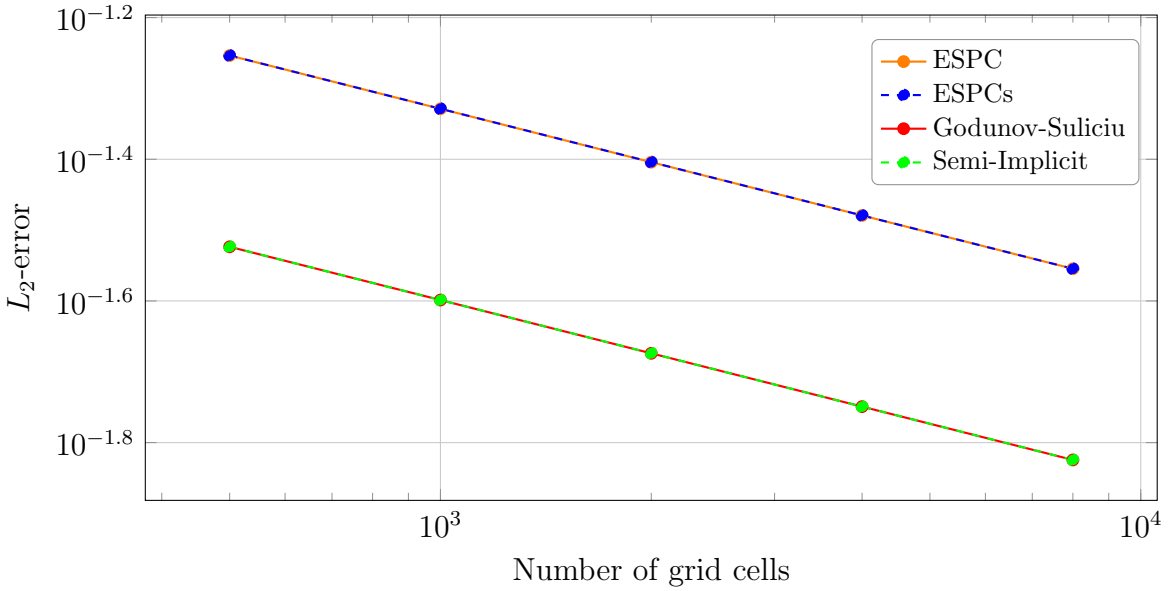


Figure 72: L_2-error against the number of grid cells for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme.

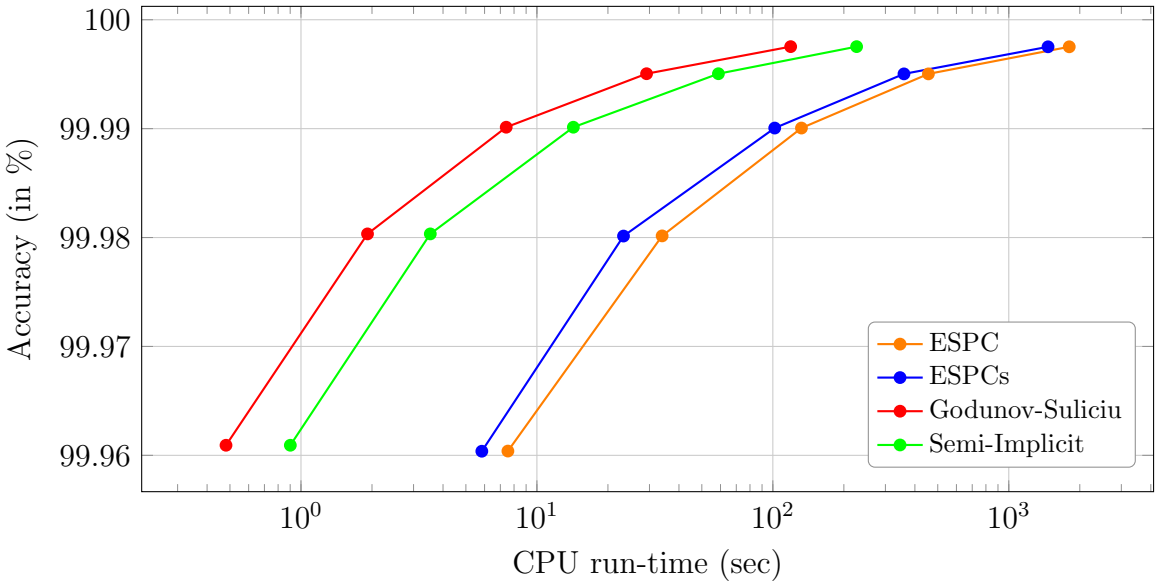


Figure 73: Run-time analysis for the ESPC, ESPCs, Godunov-Suliciu, and Semi-Implicit scheme. The accuracy is measured in the L_∞ error norm.

Conclusion and Outlook

Within this work, two problems out of the construction and operation of solar thermal power plants are regarded. A tool for the heliostat field layout optimization, and stable solvers for describing the two-phase flow of water/steam in the absorber tubes of a Fresnel solar collector system are developed. Both works can be used to help to develop more efficient and thus more competitive solar power plants.

▶ Heliostat field layout optimization of solar tower power plants

The heliostat layout optimization problem of solar tower power plants is a global, non-convex optimization problem with constraints. Usually this kind of problem is solved using a pattern-based optimizer. The drawback of these methods is the small search space by construction, as the solution is always a regular heliostat field. Within this work we used an evolutionary algorithm to improve the solution of the heliostat layout problem. Because the classical crossover operators lead to invalid layouts, and additionally are highly sensitive to the order of the heliostats, we introduced three new crossover operators. All operators are successfully tested and applied to two benchmarks, showing the applicability of our approach.

In the last section, the usage of a post-processing step was proposed, such that the complete optimization algorithm can be seen as a multi-step strategy. Even if with this Greedy approach the efficiency was improved, the achieved heliostat layouts still offer space for further improvements. One might investigate the usage of a gradient descent method where the gradients are computed using algorithmic differentiation.

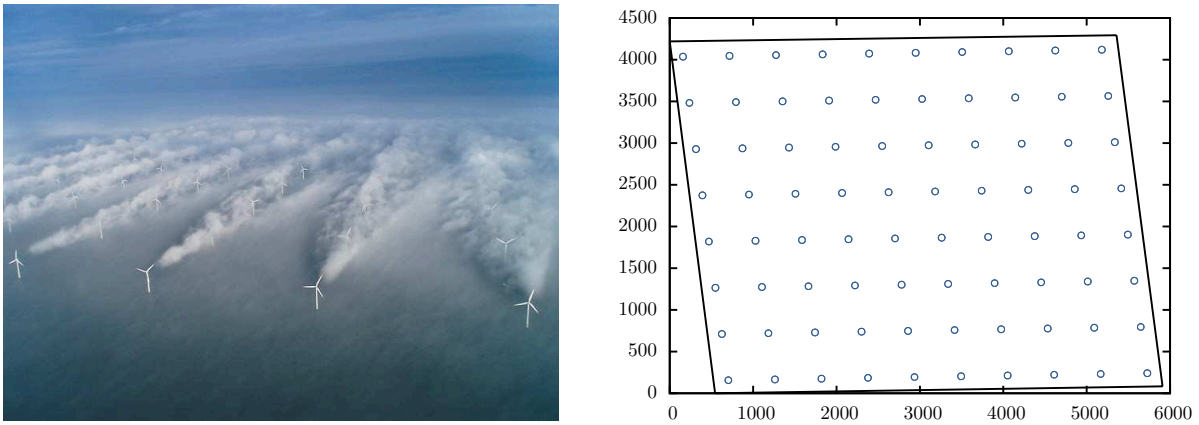


Figure 74: Layout of the Horns Rev offshore wind farm.

Sources: Aeolus, [15]

It is thinkable to use the optimizer also for other positioning problems, e.g. the layout optimization of an offshore wind farm [139]. Here we need to find for a given piece of land (under water) the best positions of N wind turbines, such that some objective function (e.g. efficiency) is optimized. Due to wind shading effects the wake of each wind turbine has to be considered. Actually, this approach was already checked in a Bachelor's thesis

by Gregor Heiming [15], which has been co-supervised by the author of this work. In future, this optimization can be applied and tested against existing offshore wind farms, where public data is available, e.g. Horns Rev see Figure 74.

▶ Direct steam generation in Fresnel solar collector systems

Within this work a two-velocity two-pressure seven-equations model is developed, such that important mathematical and thermodynamic properties hold. Different approaches are developed to solve the non-conservative parts of the PDE system. To increase the simulation time, a semi-implicit scheme is used which allows to control the time step. The new schemes are tested with multiphase flow problems from literature. As last test case, flow through a 1000 m long heated tube was simulated, where mass transfer, interfacial friction, wall friction, interfacial heat transfer, and wall heat transfer is considered. It is shown, that the entropy-stable path-conservative scheme and the developed Godunov-Suliciu are a good choice to solve this type of problem.

The path-conservative schemes ESPC and ESPCs show a slow convergence. To stabilize them, numerical dissipation for higher order has to be introduced. Furthermore, the reason for the entropy generation of the Semi-Implicit solver needs to be investigated in-depth. Then, the existing large time-step solver should be extended for low-Mach number flows and the source terms should be considered by a well-balanced scheme. This method should be parallelized for CPUs and/or GPUs. Depending on the length of the tube one could think about adaptive mesh refinement techniques (AMR). The schemes should be extended for higher order in time and space. For this purpose also the splitting operator needs to be solved at higher order (e.g. Strang splitting has 2nd order).

The motivation for these improvements are a desired increase of efficiency and accuracy, such that later this code can be used as a real-time simulation tool within a solar tower power plant, e.g. in EBL's Puerto Errado 2. Therefore, the tool needs to be extended to a network of tubes, which considers different tube types, non-homogeneous heat flux around the tube, pumps, and other power plant components. For this purpose more application tests with real data of a solar power plant are needed. It may be useful to use the tabulated water equations as equation of state. If so, here some work is needed to speed-up their computation, e.g. Spline interpolation of precomputed

Figure 75: Network of tubes in the Noor solar power station near Ouarzazate in Morocco. Source: ACWA Power

points. Additionally the source terms need to be developed more precisely related to the underlying flow pattern.

Altogether, within this work a well-posed thermal-hydraulic model for a real-world application was derived. The developed solvers allow to predict the ongoing processes in the absorber tubes. Thus, with this work the stage is set for the computer-based development of optimal control strategies for solar thermal power plants.

Bibliography

[1] P. Garcia, A. Ferriere, and J.-J. Bezian. Codes for solar flux calculation dedicated to central receiver system applications: a comparative review. *Solar Energy*, 82(3):189–197, 2008.

[2] P. Richter, M. Frank, and E. Abrahám. Multi-objective optimization of solar tower heliostat fields. In *Progress in Industrial Mathematics at ECMI 2014*, pages 357–363. Springer, 2016.

[3] C.-A. Domínguez-Bravo, S.-J. Bode, G. Heiming, P. Richter, E. Carrizosa, E. Fernández-Cara, M. Frank, and P. Gauché. Field-design optimization with triangular heliostat pods. In *Solarpaces 2015: International Conference on Concentrating Solar Power and Chemical Energy Systems*, volume 1734, pages 07006–1 – 8. AIP Publishing, 2016.

[4] P. Richter, D. Laukamp, L. Gerdes, M. Frank, and E. Ábrahám. Heliostat field layout optimization with evolutionary algorithms. In Christoph Benzmüller, Geoff Sutcliffe, and Raul Rojas, editors, *GCAI 2016. 2nd Global Conference on Artificial Intelligence*, volume 41 of *EPiC Series in Computing*, pages 240–252. EasyChair, 2016.

[5] S. Müller, M. Hantke, and P. Richter. Closure conditions for non-equilibrium multi-component models. *Continuum Mechanics and Thermodynamics*, 28(4):1157–1189, 2015.

[6] M.R. Baer and J.W. Nunziato. A two-phase mixture theory for the deflagration-to-detonation transition (ddt) in reactive granular materials. *International Journal of Multiphase Flow*, 12(6):861–889, 1986.

[7] N. Andrianov. *Analytical and numerical investigation of two-phase flows*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, Universitätsbibliothek, 2003.

[8] T. Gallouët, J.-M. Hérard, and N. Seguin. Numerical modeling of two-phase flows using the two-fluid two-pressure approach. *Mathematical Models and Methods in Applied Sciences*, 14(05):663–700, 2004.

[9] I. Suliciu. On the thermodynamics of fluids with relaxation and phase transitions. fluids with relaxation. *Int. J. Engag. Sci*, 36:921–947, 1998.

[10] J.G. Collier and J.R. Thome. *Convective boiling and condensation*, volume 38. Oxford University Press, USA, 1996.

[11] R. Leithner, B. Epple, W. Linzer, and H. Walter. *Simulation von Kraftwerken und wärmetechnischen Anlagen*. Springer, 2009.

[12] V.P. Carey. *Liquid-vapor phase-change phenomena*. Hemisphere Publishing Corporation, 1992.

[13] R. Chen, M.-C. Lu, V. Srinivasan, Z. Wang, H.H. Cho, and A. Majumdar. Nanowires for enhanced boiling heat transfer. *Nano letters*, 9(2):548–553, 2009.

[14] R.J. LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge University Press, 2002.

[15] Gregor Heiming. Modeling and simulation of offshore wind farms. Bachelor's thesis, MathCCES, RWTH Aachen University, 2015.

[16] S.-J. Bode and P. Gauché. Review of optical software for use in concentrating solar power systems. In *South African*

Solar Energy Conference, Stellenbosch, 2012.

[17] M. Geyer and W.B. Stine. Power from the sun. Website, 2001. Available online at `www.powerfromthesun.net/book.html`, visited on October 1st 2012.

[18] P.L. Leary and J.D. Hankins. User's guide for MIRVAL: a computer code for comparing designs of heliostat-receiver optics for central receiver solar power plants. Technical report, Sandia Laboratories, 1979.

[19] T. Wendelin. Soltrace: a new optical modeling tool for concentrating solar optics. In *ASME 2003 International Solar Energy Conference*, pages 253–260. American Society of Mechanical Engineers, 2003.

[20] M Blanco, J.M. Amieva, and A. Mancilla. The tonatiuh software development project: An open source approach to the simulation of solar concentrating systems. *Proceedings of the IMEC2005, Orlando, FL*, pages 157–164, 2005.

[21] B. Belhomme, R. Pitz-Paal, P. Schwarzbözl, and S. Ulmer. A new fast ray tracing tool for high-precision simulation of heliostat fields. *Journal of Solar Energy Engineering*, 131(3):031002, 2009.

[22] J.P. Roccia, B. Piaud, C. Coustet, C. Caliot, E. Guillot, G. Flamant, and J. Delatorre. Solfast, a ray-tracing monte-carlo software for solar concentrating facilities. In *Journal of Physics: Conference Series*, volume 369, page 012029. IOP Publishing, 2012.

[23] C.L. Pitman and L.L. Vant-Hull. The university of houston solar central receiver code system: Concepts, updates, and start-up kits. Technical Report SAND-88-7029, Houston Univ., TX (USA). Energy Lab., 3 1989.

[24] B.L. Kistler. A user's manual for DELSOL3: A computer code for calculating the optical performance and optimal system design for solar thermal central receiver plants. Technical report, Sandia National Labs., Livermore, CA (USA), 1986.

[25] P.K. Falcone. A handbook for solar central receiver design. Technical Report SAND-86-8009, Sandia National Labs., Livermore, CA (USA), 12 1986.

[26] M. Kiera. Description of the computing code system hflcal. *I. Report, GASTIAS-BT-200000-075, Interatom Report*, 1986.

[27] P. Schwarzbözl, R. Pitz-Paal, and M. Schmitz. Visual hflcal - a software tool for layout and optimisation of heliostat fields. In *SolarPACES Conference*, 2009.

[28] R. Monterreal. Software developments for system analysis and optimization. In *SolarPACES Tech. Report No. III-3/00, Proc. of the IEA SolarPACES Task III Workshop on Simulation of Solar Thermal Power Systems, Cologne, Germany*, 2000.

[29] D. Riveros-Rosas, M. Sánchez-González, and C.A. Estrada. Three-dimensional analysis of a concentrated solar flux. *Journal of Solar Energy Engineering*, 130(1):014503, 2008.

[30] X. Wei, Z. Lu, Z. Wang, W. Yu, H. Zhang, and Z. Yao. A new method for the design of the heliostat field layout for solar tower power plant. *Renewable Energy*, 35(9):1970–1975, 2010.

[31] E. Leonardi and B. DAguanno. Crs4-2: A numerical code for the calculation of

the solar power collected in a central receiver system. *Energy*, 36(8):4828–4837, 2011.

[32] S.L. Lutchman, A.A. Groenwold, P. Gauché, and S. Bode. On using a gradient-based method for heliostat field layout optimization. *Energy Procedia*, 49:1429–1438, 2014.

[33] M. Sanchez and M. Romero. Methodology for generation of heliostat field layout in central receiver systems based on yearly normalized energy surfaces. *Solar Energy*, 80(7):861–874, 2006.

[34] P. Schramek, D.R. Mills, W. Stein, and P. Le Lièvre. Design of the heliostat field of the csiro solar tower. *Journal of Solar Energy Engineering*, 131(2):024505, 2009.

[35] F.W. Lipps and L.L. Vant-Hull. A cellwise method for the optimization of large central receiver systems. *Solar Energy*, 20(6):505–516, 1978.

[36] C.J. Noone, M. Torrilhon, and A. Mitsos. Heliostat field optimization: A new computationally efficient model and biomimetic layout. *Solar Energy*, 2011.

[37] R. Buck. Heliostat field layout using non-restricted optimization. In *SolarPACES Conference*, 2012.

[38] M.M. Elsayed, H. Allah, and O.M. Al-Rabghi. Yearly-averaged daily usefulness efficiency of heliostat surfaces. *Solar energy*, 49(2):111–121, 1992.

[39] Z. Yao, Z. Wang, Z. Lu, and X. Wei. Modeling and simulation of the pioneer 1mw solar thermal central receiver system in china. *Renewable Energy*, 34(11):2437–2446, 2009.

[40] J.A. Duffie and W.A. Beckman. *Solar engineering of thermal processes*, volume 13. Wiley New York, 2006.

[41] I. Reda and A. Andreas. Solar position algorithm for solar radiation applications. *Solar energy*, 76(5):577–589, 2004.

[42] M. Blanco-Muriel, D.C. Alarcón-Padilla, T. López-Moratalla, and M. Lara-Coira. Computing the solar vector. *Solar Energy*, 70(5):431–441, 2001.

[43] J. Meeus. Astronomical algorithms. *Willmann-Bell Inc., Richmond, Virginia, USA*, 1998.

[44] P. Armstrong and M. Izygon. An innovative software for analysis of sun position algorithms. *Energy Procedia*, 49:2444–2453, 2014.

[45] V. Badescu. *Modeling solar radiation at the earth's surface: recent advances*. Springer Verlag, 2008.

[46] R. Pitz-Paal, N.B. Botero, and A. Steinfeld. Heliostat field layout optimization for high-temperature solar thermochemical processing. *Solar Energy*, 85(2):334–343, 2011.

[47] M. Schmitz, P. Schwarzbözl, R. Buck, and R. Pitz-Paal. Assessment of the potential improvement due to multiple apertures in central receiver systems with secondary concentrators. *Solar energy*, 80(1):111–120, 2006.

[48] C.L. Pitman and L.L. Vant-Hull. Atmospheric transmittance model for a solar beam propagating between a heliostat and a receiver. *ASES Progress in Solar Energy*, pages 1247–1251, 1982.

[49] A. Rabl. *Active solar collectors and their applications*. Oxford University Press, 1985.

[50] G. Augsburger. *Thermo-economic optimisation of large solar tower power*

plants. PhD thesis, École Polytechnique Fédérale de Lausanne, 2013.

[51] R. Buck, A. Pfahl, and T.H. Roos. Target aligned heliostat field layout for non-flat terrrain. In *SASEC Southern African Solar Energy Conference, 21-23 May 2012*, 2012.

[52] Clotilde C. Victor G. and Manuel B. Fourier sampling of sun path for applications in solar energy. In *SolarPACES Conference South Africa 13-16 October 2015*, 2015.

[53] G. Morin, P. Richter, and P. Nitz. New method and software for multi-variable techno-economic design optimisation of parabolic trough power plants. In *SolarPACES Conference*, 2010.

[54] P. Richter, E. Ábrahám, and G. Morin. Optimisation of concentrating solar thermal power plants with neural networks. In *International Conference on Adaptive and Natural Computing Algorithms*, pages 190–199. Springer, 2011.

[55] A. Baghernejad and M. Yaghoubi. Exergoeconomic analysis and optimization of an integrated solar combined cycle system (isccs) using genetic algorithm. *Energy conversion and Management*, 52(5):2193–2203, 2011.

[56] G.B. Wallis. *One-dimensional two-phase flow*. McGraw-Hill, 1969.

[57] N. Zuber and J.A. Findlay. Average volumetric concentration in two-phase flow systems. *Journal of Heat Transfer*, 87:453–464, 1965.

[58] P.J. Martínez Ferrer, T. Flåtten, and S.T. Munkejord. On the effect of temperature and velocity relaxation in two-phase flow models. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(02):411–442, 2012.

[59] J.G. Collier. *Convective boiling and condensation*. McGraw-Hill, 1981.

[60] R. Saurel and R. Abgrall. A multiphase godunov method for compressible multifluid and multiphase flows. *Journal of Computational Physics*, 150(2):425–467, 1999.

[61] D.A. Drew and S.L. Passman. *Theory of Multicomponent Fluids*. Applied mathematical sciences. Springer, 1998.

[62] S.S. Sahoo, S. Singh, and R. Banerjee. Steady state hydrothermal analysis of the absorber tubes used in linear fresnel reflector solar thermal system. *Solar Energy*, 87:84–95, 2013.

[63] J.F. Feldhoff. *Analysis of Once-Through Boiler Concepts in Parabolic Troughs*. PhD thesis, RWTH Aachen University, 2015.

[64] J.F. Feldhoff, T. Hirsch, and L. Pitz-Paal, R.and Valenzuela. Transient models and characteristics of once-through line focus systems. *Energy Procedia*, 69:626–637, 2015.

[65] A. Aurousseau, V. Vuillerme, and J.-J. Bezian. Modeling of linear concentrating solar power using direct steam generation with parabolic-trough. In *Proceedings of the 11th International Modelica Conference, September 21-23, 2015, Versailles, France*, 2015.

[66] J.J. Serrano-Aguilera and L. Valenzuela. Transient validation of relap5 model with the diss facility in once through operation mode. In *SOLARPACES 2015: International Conference on Concentrating Solar Power and Chemical Energy Systems*, volume 1734, page 040006. AIP Publishing, 2016.

[67] S.S. Sahoo, S. Singh, and R. Banerjee. Thermal hydraulic simulation of absorber tubes in linear fresnel reflector solar thermal system using relap. *Renewable Energy*, 86:507–516, 2016.

[68] H. Städtke, G. Franchello, and B. Worth. Numerical simulation of two-phase flows with stiff source terms. In *Computational Fluid Dynamics 2000*, pages 633–638. Springer, 2001.

[69] M. Ishii and T. Hibiki. *Thermo-fluid dynamics of two-phase flow*. Springer, 2010.

[70] D.A. Drew and R.T. Lahey. Analytical modeling of multiphase flow. *Particulate two-phase flow*, pages 509–566, 1993.

[71] R. Aris. Vectors, tensors and the basic equations of fluid dynamics. *Englewood Cliffs, J. Prentice–Hall*, 1962.

[72] A Zein. *Numerical methods for multiphase mixture conservation laws with phase transition*. PhD thesis, University of Magdeburg, 2010.

[73] Idaho National Laboratory. RELAP5-3D© Code Manual Volume I: Code Structure, System Models and Solution Methods. Technical report, INL Idaho National Laboratory (INL), INEEL Report EXT-98-00834, Revision 4.0, 2012.

[74] R.A. Berry, J. W. Peterson, H. Zhang, R. C. Martineau, H. Zhao, L. Zou, and D. Andrs. RELAP-7 Theory Manual. Technical report, INL Idaho National Laboratory (INL), EXT-14-31366, 2014.

[75] H. Herwig. *Strömungsmechanik A-Z: Eine systematische Einordnung von Begriffen und Konzepten der Strömungsmechanik*. Vieweg + Teubner Verlag, 2004.

[76] J.R. Cooper and R.B. Dooley. Revised Advisory Note No. 3 – Thermodynamic Derivatives from IAPWS Formulations. *The International Association for the Properties of Water and Steam*, pages 1–13, 2008.

[77] A. Ambroso, C. Chalons, and P.-A. Raviart. A godunov-type method for the seven-equation model of compressible two-phase flow. *Computers & Fluids*, 54:67–91, 2012.

[78] W. Wagner and H.-J. Kretzschmar. *International Steam Tables-Properties of Water and Steam based on the Industrial Formulation IAPWS-IF97: Tables, Algorithms, Diagrams, and CD-ROM Electronic Steam Tables-All of the equations of IAPWS-IF97 including a complete set of supplementary backward equations for fast calculations of heat cycles, boilers, and steam turbines*. Springer Science & Business Media, 2007.

[79] J.L. Weisbach. *Lehrbuch der Ingenieur- und Maschinenmechanik*. Vieweg, 1845.

[80] M Ishii and T.C. Chawla. Local drag laws in dispersed two-phase flow. Technical report, Argonne National Lab., 1979.

[81] M. Wirz. *Optical and thermal modeling of parabolic trough concentrator systems*. PhD thesis, Diss., Eidgenössische Technische Hochschule ETH Zürich, Nr. 21800, 2014, 2014.

[82] M. Mertins. *Technische und wirtschaftliche Analyse von horizontalen Fresnel-Kollektoren*. PhD thesis, Karlsruher Institut für Technologie, 2009.

[83] F. Burkholder and C. Kutscher. Heat loss testing of Schott's 2008 PTR70 parabolic trough receiver. Technical report, NREL/TP-550–45633, 2009.

[84] H. Walter. *Modellbildung und numerische Simulation von Naturumlaufdampferzeugern.* Fortschritt-Berichte VDI, Reihe 6 Energietechnik, Nr. 457, VDI Verlag, 2001.

[85] S. Nukiyama. The maximum and minimum values of the heat Q transmitted from metal to boiling water under atmospheric pressure. *International Journal of Heat and Mass Transfer*, 9(12):1419–1433, 1966.

[86] K. Lee and D.J. Ryley. The evaporation of water droplets in superheated steam. *Journal of Heat Transfer*, 90:445, 1968.

[87] M.S. Plesset and S.A. Zwick. The growth of vapor bubbles in superheated liquids. *Journal of Applied Physics*, 25(4):493–500, 1954.

[88] H.C. Unal. Maximum bubble diameter, maximum bubble-growth time and bubble-growth rate during the subcooled nucleate flow boiling of water up to 17.7 MN/m^2. *International Journal of Heat and Mass Transfer*, 19:643–649, 1976.

[89] R.T. Lahey. A mechanistic subcooled boiling model. In *Proceedings of the 6th International Heat Transfer Conference*, volume 1, pages 293–297, 1978.

[90] G. Brown. Heat transmission by condensation of steam on a spray of water drops. *Proceedings of General Discussion on Heat Transfer*, pages 49–52, 1951.

[91] T.G. Theofanous. Modeling of basic condensation processes. In *WRSR Workshop on Condensation, Silver Spring, Maryland*, 1979.

[92] F.W. Dittus and L.M.K. Boelter. Heat transfer in automobile radiators of the tubular type. *International Communications in Heat and Mass Transfer*, 12(1):3–22, 1985.

[93] R.J. LeVeque. *Numerical Methods for Conservation Laws.* Lectures in Mathematics ETH Zürich, Department of Mathematics Research Institute of Mathematics. Springer, 1992.

[94] Frédéric Coquel, Jean-Marc Hérard, Khaled Saleh, and Nicolas Seguin. Two properties of two-velocity two-pressure models for two-phase flows. *Communications in Mathematical Sciences*, 12(3), 2014.

[95] R. Saurel, F. Petitpas, and R. Abgrall. Modelling phase transition in metastable liquids: application to cavitating and flashing flows. *Journal of Fluid Mechanics*, 607:313–350, 2008.

[96] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of computational physics*, 49(3):357–393, 1983.

[97] K. Saleh. *Analyse et Simulation Numérique par Relaxation d'Ecoulements Diphasiques Compressibles. Contribution au Traitement des Phases Evanescentes.* PhD thesis, Université Pierre et Marie Curie-Paris VI, 2012.

[98] J.-M. Hérard. A three-phase flow model. *Mathematical and computer modelling*, 45(5):732–755, 2007.

[99] E. Godlewski and P.-A. Raviart. *Numerical approximation of hyperbolic systems of conservation laws*, volume 118. Springer, 1996.

[100] H. Städtke. *Gasdynamic Aspects of Two-Phase Flow: Hyperbolicity, Wave Propagation Phenomena and Related Numerical Methods.* John Wiley & Sons, 2006.

[101] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, 100(1):32–74, 1928.

[102] J.-M. Hérard and O. Hurisse. A fractional step method to compute a class of compressible gas–liquid flows. *Computers & Fluids*, 55:57–69, 2012.

[103] G. Strang. Accurate partial difference methods. *Numerische Mathematik*, 6(1):37–46, 1964.

[104] V.A. Titarev and E.F. Toro. Finite-volume weno schemes for three-dimensional conservation laws. *Journal of Computational Physics*, 201(1):238–260, 2004.

[105] E.F. Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.

[106] C. Parés and M.L. Muñoz-Ruiz. On some difficulties of the numerical approximation of nonconservative hyperbolic systems. *Bol. Soc. Esp. Mat. Apl*, 47(47):23–52, 2009.

[107] G. Dal Maso, P. G. LeFloch, and F. Murat. Definition and weak stability of nonconservative products. *J. Math. Pures Appl., IX. Sér.*, 74(6):483–548, 1995.

[108] M.J. Castro, U.S. Fjordholm, S. Mishra, and C. Pars. Entropy conservative and entropy stable schemes for nonconservative hyperbolic systems. *SIAM Journal on Numerical Analysis*, 51(3):1371–1391, 2013.

[109] R. Abgrall and S. Karni. A comment on the computation of non-conservative products. *Journal of Computational Physics*, 229(8):2759–2763, 2010.

[110] A. Canestrelli, A. Siviglia, M. Dumbser, and E. F. Toro. Well-balanced high-order centred schemes for non-conservative hyperbolic systems. Applications to shallow water equations with fixed and mobile bed. *Advances in Water Resources*, 32(6):834–844, 2009.

[111] C. Parés. Numerical methods for nonconservative hyperbolic systems: a theoretical framework. *SIAM Journal on Numerical Analysis*, 44(1):300–321, 2006.

[112] M. Dumbser and D.S. Balsara. A new efficient formulation of the hllem riemann solver for general conservative and non-conservative hyperbolic systems. *Journal of Computational Physics*, 304:275–319, 2016.

[113] M.J. Castro, P.G. LeFloch, M.L. Muñoz-Ruiz, and C. Parés. Why many theories of shock waves are necessary: Convergence error in formally path-consistent schemes. *Journal of Computational Physics*, 227(17):8107–8129, 2008.

[114] E. Tadmor and W. Zhong. Entropy stable approximations of navier–stokes equations with no artificial numerical viscosity. *Journal of Hyperbolic Differential Equations*, 3(03):529–559, 2006.

[115] U.S. Fjordholm and S. Mishra. Accurate numerical discretizations of non-conservative hyperbolic systems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(01):187–206, 2012.

[116] J.-L. Guermond and B. Popov. Viscous regularization of the euler equations and entropy principles. *SIAM Journal on Applied Mathematics*, 74(2):284–305, 2014.

[117] D. Serre. *Systemes de lois de conservation. I. Hyperbolicité, entropies, ondes de choc. Fondations.* Diderot Editeur, Paris, 1996.

[118] I. Suliciu. On modelling phase transitions by means of rate-type constitutive equations. shock wave structure. *International Journal of Engineering Science*, 28(8):829–841, 1990.

[119] C. Chalons and J.-F. Coulombel. Relaxation approximation of the euler equations. *Journal of Mathematical Analysis and Applications*, 348(2):872–893, 2008.

[120] T.-P. Liu. Hyperbolic conservation laws with relaxation. *Communications in Mathematical Physics*, 108(1):153–175, 1987.

[121] G.-Q. Chen, C. D. Levermore, and T.-P. Liu. Hyperbolic conservation laws with stiff relaxation terms and entropy. *Communications on Pure and Applied Mathematics*, 47(6):787–830, 1994.

[122] S. Jin and Z. Xin. The relaxation schemes for systems of conservation laws in arbitrary space dimensions. *Communications on Pure and Applied Mathematics*, 48(3):235–276, 1995.

[123] F. Coquel, K. Saleh, and N. Seguin. A robust and entropy-satisfying numerical scheme for fluid flows in discontinuous nozzles. *Mathematical Models and Methods in Applied Sciences*, 24(10):2043–2083, 2014.

[124] F. Bouchut, C. Klingenberg, and K. Waagan. A multiwave approximate riemann solver for ideal mhd based on relaxation. i: theoretical framework. *Numerische Mathematik*, 108(1):7–42, 2007.

[125] D. Serre. *Systems of Conservation Laws 1: Hyperbolicity, entropies, shock waves.* Cambridge University Press, 1999.

[126] T. Flåtten and H. Lund. Relaxation two-phase flow models and the subcharacteristic condition. *Mathematical Models and Methods in Applied Sciences*, 21(12):2379–2407, 2011.

[127] G.B. Whitham. *Linear and Nonlinear Waves.* John Wiley & Sons, 1974.

[128] A. Ambroso, C. Chalons, F. Coquel, and T. Galié. Relaxation and numerical approximation of a two-fluid two-pressure diphasic model. *ESAIM: Mathematical Modelling and Numerical Analysis*, 43(06):1063–1097, 2009.

[129] F. Coquel, Q. Nguyen, M. Postel, and Q. Tran. Entropy-satisfying relaxation method with large time-steps for euler ibvps. *Mathematics of Computation*, 79(271):1493–1533, 2010.

[130] C. Chalons, M. Girardin, and S. Kokh. Large time step and asymptotic preserving numerical schemes for the gas dynamics equations with source terms. *SIAM Journal on Scientific Computing*, 35(6):A2874–A2902, 2013.

[131] C. Chalons, M. Girardin, and S. Kokh. An all-regime lagrange-projection like scheme for the gas dynamics equations on unstructured meshes. *Communications in Computational Physics*, 2014.

[132] C. Chalons, F. Coquel, S. Kokh, and N. Spillane. Large time-step numerical scheme for the seven-equation model of compressible two-phase flows. In *Finite Volumes for Complex Applications VI Problems & Perspectives*, pages 225–233. Springer, 2011.

[133] G.A. Sod. A numerical study of a converging cylindrical shock. *Journal of Fluid Mechanics*, 83(04):785–794, 1977.

[134] D.W. Schwendeman, C.W. Wahle, and A.K. Kapila. The riemann problem and a high-resolution godunov method for a model of compressible two-phase flow. *Journal of Computational Physics*, 212(2):490–526, 2006.

[135] V.H. Ransom. Numerical benchmark tests. *Multiphase Science and Technology*, 3(1-4), 1987.

[136] C. Berthon. Schéma nonlinéaire pour l'approximation numérique d'un système hyperbolique non conservatif. *Comptes Rendus Mathematique*, 335(12):1069–1072, 2002.

[137] C.F. Colebrook. Turbulent flow in pipes with particular reference to the transition region between the smooth and rough pipe laws. *J. Inst. Civ. Eng*, 11(4):133–156, 1939.

[138] D.J. Zigrang and N.D. Sylvester. A review of explicit friction factor equations. *Journal of energy resources technology*, 107(2):280–283, 1985.

[139] Angelo Tesauro, Pierre-Elouan Réthoré, and Gunner Chr Larsen. State of the art of wind farm optimization. In *EWEA 2012-European Wind Energy Conference & Exhibition*, 2012.