

*Ecole doctorale Mathématiques, sciences de l'information
et de l'ingénieur*

Institut de Recherche Mathématiques Avancée

THÈSE présentée par / **DISSERTATION** presented by :
Gauthier LAZARE

soutenue le /defended on : **16 octobre 2025**

pour obtenir le grade de / to obtain the grade of :
Docteur de l'Université de Strasbourg / Strasbourg University Doctor
Discipline/S spécialité / Discipline/Specialty : **Mathématiques**

**Développement d'une méthode numérique
performante pour la résolution d'un modèle
diphasique homogène partiellement
déséquilibré en milieu poreux hétérogène.**

THÈSE dirigée par / DISSERTATION supervisor :

HELLUY Philippe

Professeur, Université de Strasbourg

RAPPORTEURS :

KOKH Samuel

Directeur de recherche, Maison de la simulation - Paris

MATHIS Hélène

Professeure, Université de Montpellier

AUTRES MEMBRES DU JURY / OTHER MEMBERS OF THE JURY :

FACCANONI Gloria

Maitre de conférences, Université de Toulon

HERARD Jean-Marc

Ingénieur chercheur - Expert Senior EDF, EDF R&D Lab Chatou

SALEH Khaled

Professeur, Institut de Mathématiques de Marseille

INVITÉS / INVITED MEMBERS : **Erwan LE COUPANEC** – Ingénieur chercheur EDF

Thèse

INSTITUT DE
RECHERCHE
MATHÉMATIQUE
AVANCÉE

UMR 7501

Strasbourg

présentée pour obtenir le grade de docteur de
l'Université de Strasbourg
Spécialité MATHÉMATIQUES

Gauthier Lazare

**Développement d'une méthode numérique
performante pour la résolution d'un modèle
diphasique homogène partiellement déséquilibré
en milieu poreux hétérogène.**

Soutenue le 16 octobre 2025
devant la commission d'examen

Philippe Helluy, directeur de thèse
Samuel Kokh, rapporteur
Hélène Mathis, rapportrice
Gloria Faccanoni, examinatrice
Jean-Marc Hérard, examinateur
Khaled Saleh, examinateur

<https://irma.math.unistra.fr>



Université

de Strasbourg



INSTITUT DE RECHERCHE MATHÉMATIQUE AVANCÉE (IRMA) - UNIVERSITÉ DE
STRASBOURG

EDF R&D - CHATOU

DÉVELOPPEMENT D'UNE MÉTHODE NUMÉRIQUE
PERFORMANTE POUR LA RÉOLUTION D'UN MODÈLE
DIPHASIQUE HOMOGÈNE PARTIELLEMENT DÉSÉQUILIBRÉ EN
MILIEU POREUX HÉTÉROGÈNE.

DEVELOPMENT OF AN EFFICIENT NUMERICAL METHOD FOR SOLVING A PARTIALLY
NON-EQUILIBRIUM HOMOGENEOUS TWO-PHASE MODEL IN A HETEROGENEOUS POROUS
MEDIUM.

GAUTHIER LAZARE

DIRECTOR:

PROF. PHILIPPE HELLUY

IRMA - STRASBOURG



Abstract - English version

EDF has developed the thermo-hydraulic code THYC-coeur to simulate water flow in the cores of PWRs (Pressurized Water Reactors) and to evaluate the Departure from Nucleate Boiling Ratio (DNBR), an indicator of the margin to the boiling crisis that could threaten core integrity. THYC-coeur mainly solves steady-state problems using finite volume schemes. Although each computation is fast (about 10 seconds for an industrial configuration), the very large number of required simulations (several million) makes code acceleration necessary, which is the main goal of this thesis. The drift-flux models are first presented and analyzed. Several analytical solutions are then constructed. To more easily study certain numerical phenomena, a simplified prototype called ThermoTorch has been developed. After verifying the global numerical scheme in both steady-state and transient regimes, an acceleration method using initialization from neural networks was tested.

Abstract - French version

EDF a développé le code thermo-hydraulique THYC-cœur pour simuler l'écoulement d'eau dans le cœur des Réacteurs à Eau Pressurisée (REP) et évaluer le Rapport de Flux Thermique Critique (RFTC), indicateur de la marge à la crise d'ébullition pouvant menacer l'intégrité du cœur. THYC-cœur résout principalement des problèmes stationnaires via des schémas volumes finis. Bien que le calcul soit rapide (10 s pour une configuration industrielle), le grand nombre de simulations requises (plusieurs millions) rend nécessaire une accélération du code, objectif de cette thèse. Les modèles drift-flux sont d'abord présentés et analysés. Plusieurs solutions analytiques sont ensuite construites. Afin d'étudier plus facilement certains phénomènes numériques, un prototype simplifié, appelé ThermoTorch est développé. Après une vérification en stationnaire et en instationnaire du schéma numérique global, une méthode d'accélération utilisant une initialisation issue de réseaux de neurones a été testée.

Remerciements

Tout d'abord, je tiens à remercier Philippe Helluy, mon directeur de thèse, qui m'a accompagné durant ces trois années. Malgré la distance et les rares occasions de nous voir, j'ai énormément apprécié apprendre à tes côtés, me casser la tête avec toi sur nos problèmes et échanger sur des sujets variés, scientifiques ou non. Tu as toujours su te tenir informé de mes avancées depuis Strasbourg, et accepté les contraintes industrielles tout en maintenant un lien vital avec le monde académique. Merci pour tout, j'espère pouvoir continuer à travailler avec toi à l'avenir.

Je pense ensuite à mon encadrement industriel à EDF. Avec une équipe de cinq personnes, je craignais que certains décrochent au fil du temps. Pourtant, tout le monde m'a accompagné jusqu'au bout, me fournissant des avis critiques et pertinents sur mes travaux et des conseils précieux pour la rédaction et la préparation de la soutenance.

Merci Qingqing, pour tes conseils et ton accompagnement constant, malgré ton départ à Lyon.

Merci Frank, d'avoir suivi ma thèse depuis Saclay et pour tous tes retours sur mes travaux, malgré des orientations qui ne nous ont pas permis de collaborer directement.

Merci Erwan, d'avoir porté mon sujet de thèse au sein de ton projet. Malgré un planning chargé, tu as toujours suivi mes travaux en m'offrant un cadre de travail sans pression sur la performance des méthodes développées. J'ai hâte de continuer à travailler avec toi par la suite.

Merci Stéphane, pour ton soutien infailible. Tu es l'oreille attentive du groupe, celle qui maintient la cohésion humaine. Jamais dans le jugement, tu prodigues des conseils sages dans les moments difficiles. Merci aussi pour ton apport industriel à cette thèse, qui a considérablement enrichi le manuscrit, notamment lors de cette soirée au bar avant tes vacances.

Enfin, merci Jean-Marc, sans qui une partie importante de mes travaux n'existerait pas. Tu m'as beaucoup apporté, tant sur le plan scientifique, humain qu'organisationnel. Tu as été un soutien constant pendant ces trois années. Tu n'étais pas censé y consacrer autant de temps : merci infiniment pour ton engagement.

Je souhaite également remercier mon jury : mes rapporteurs Hélène et Samuel, et mes examinateurs Gloria et Khaled. Merci pour votre relecture attentive du manuscrit et pour les échanges passionnants que nous avons eus lors de la soutenance. . Ce fut un honneur d'avoir reçu de personnes aussi qualifiées de précieux avis sur mon travail.

Je veux aussi remercier mes collègues au sein d'EDF : Ueva qui, accompagné de Stéphane, m'a permis de rejoindre la belle aventure d'EDF dans les meilleures conditions; Patrick et Sylvie qui m'ont accueilli au sein du département MFEE; Damien qui m'a guidé à mes débuts à I8C; et Leticia, qui lui a succédé et s'est battue pour que je puisse poursuivre l'aventure en tant qu'ingénieur-chercheur à I8C.

Je voudrais remercier tous les membres du groupe I8C (et plus largement de la R&D) qui m'ont intégré au sein du groupe. J'ai hâte de continuer à travailler avec vous l'année prochaine : apprendre tout sur THYC-GV avec Pierre et Arthur, comprendre les transitoires avec Paul et Romain, et tester diverses méthodes d'IA avec Thibault et Pauline. Merci aussi aux anciens doctorants qui m'ont montré la voie : Luc, Martin, Jacques et Pierre.

Enfin, merci à toutes les équipes extra-professionnelles d'EDF qui m'ont permis de m'aérer l'esprit durant ces trois années: la team ludothèque, la team coinche, la super team Golfech - Morgane et Pauline - et bien sûr, l'équipe Surf/Tennis/Soulac : PV, Igor, Ueva, Nicolas et Olivier.

Enfin, je tiens aussi à remercier ma famille, ma femme et mes amis qui ont été à mes côtés durant toute ma thèse, qui m'ont soutenu quand je leur parlais de mes problèmes de bulles ou de valeurs propres. Grâce à eux, j'ai toujours eu le moral au beau fixe pendant ces trois ans.

A Marine

Contents

Introduction	9
1 Nuclear energy in France	9
1.1 Operation of a pressurized water reactor	9
1.2 Nuclear reactor core composition	10
2 Boiling two-phase flow	12
2.1 Boiling regimes	12
2.2 Departure From Nucleate Boiling Ratio	13
3 THYC-coeur: a component code simulating 3D flows in nuclear reactor cores	13
3.1 A brief review of reactor core codes	14
3.2 Porous approach	15
3.3 Characteristics of THYC-coeur code	16
3.4 Industrial motivations to accelerate THYC-coeur	17
4 Contents of the thesis	18
Synthesis of thesis work	19
1 Chapter 1	19
2 Chapter 2	20
3 Chapter 3	21
4 Chapter 4	22
5 Chapter 5	23
6 Chapter 6	24
7 Chapter 7	25
8 Chapter 8	26
Communications and publications	28
I Two-phase flow models	29
1 Two-phase flow models with a relative velocity between phases	30
1.1 Governing equations for each phase	30
1.1.1 Local instant formulation	31
1.1.2 Averaged equations	32
1.2 Mixture model	36
1.2.1 Mixture and disequilibrium quantities	37
1.2.2 Mixture equations	38
1.2.3 Assumptions to reduce the number of equations	39
1.2.4 Disequilibrium equation	40
1.3 4-equation models	40
1.3.1 Total energy 4-equation model	41
1.3.2 Enthalpy 4-equation model	42
1.3.3 Mixture Equation of State	42

1.4	Closure laws	44
1.4.1	Mass transfer source term	44
1.4.2	Relative velocity between phases	45
1.4.3	Additional closure laws	51
1.5	THYC-coeur model	51
Appendix		
1.A	Detailed computation of the model	53
1.A.1	Time averaging	53
1.A.2	Space averaging	57
1.B	Thermodynamic equilibrium - 3-equation model	61
1.C	Discussion on the Bestion correlation	62
1.C.1	Terminal velocity of a slug	62
1.C.2	Approximated Bestion correlation	63
2	Hyperbolicity of the simplified models	65
2.1	Introduction	65
2.2	Methodology	66
2.3	Hyperbolicity of the barotropic drift-flux model without energy equation	66
2.4	Hyperbolicity of the 4-equation drift-flux model	69
2.4.1	Four-equation model with null relative velocity	69
2.4.2	Four-equation model with constant relative velocity	70
Appendix		
2.A	Detailed computations for the hyperbolicity of the 4-equation model with a constant relative velocity	75
2.A.1	Computation to obtain the model in variables $(\tilde{\rho}, u, \tilde{p}, y)$	75
2.A.2	Latent heat	76
2.A.3	Detailed computation of the characteristic polynomial	77
2.B	Study of the sufficient condition for hyperbolicity when considering a constant relative velocity	79
3	Some analytical solutions of the simplified two phase flow models	81
3.1	Analytical solutions for the disequilibrium equation	81
3.1.1	Steady-state solution for constant parameters with constant relative velocity	82
3.1.2	Steady-state solution for constant parameters with the approximated Bestion correlation	82
3.1.3	Unsteady solution with approximated Bestion correlation	83
3.2	Unsteady solutions for the 3-equation model	85
3.2.1	A self-similar unsteady solution without heat exchange.	86
3.2.2	A self-similar unsteady solution in free medium with a specific heat profile.	89
3.2.3	Conclusion	91
Appendix		
3.A	Steady-state for 3-equation model	92
3.B	Unsteady solutions for 1D Riemann problems - one-phase flow	93
3.B.1	Symmetric double rarefaction wave	94
3.B.2	Symmetric double shock wave	95
3.B.3	Sod shock tube	97
3.C	Analytical solution for the 4-equation model - 1D Riemann Problem	100
3.C.1	Simplified mixture EoS	101
3.C.2	1D Riemann Problem - Analytical solution	102

II	THYC-coeur with relative velocity	105
4	Finite volume schemes for the disequilibrium equation	106
4.1	Continuous equations for the disequilibrium	106
4.1.1	Governing equations	106
4.1.2	Continuous maximum principle	107
4.2	Finite volume schemes	108
4.2.1	Discretization and notations	108
4.2.2	Finite volume discretization	110
4.2.3	Global scheme with QG Scheme	111
4.2.4	Global Scheme with QRd Scheme	113
4.2.5	Global scheme with QRq Scheme	114
4.2.6	Conclusion for the boundary conditions	115
4.3	Numerical verification of the schemes	116
4.3.1	Consistency and convergence rate of the schemes for steady state solutions . .	116
4.3.2	Comparison for steady-state simulations with constant parameters	118
4.3.3	Comparison for steady-states with the approximated Bestion correlation . . .	120
4.3.4	Comparison of the schemes for unsteady simulations	120
4.3.5	Conclusion	122
Appendix		
4.A	Continuous maximum principle	123
4.B	Extension of the schemes to boundaries conditions	125
4.B.1	Boundary condition for QG scheme	125
4.B.2	Boundary condition for QRd scheme	126
4.B.3	Boundary condition for QRq scheme	127
4.C	Why do the QRd Scheme and the QRq Scheme not converge for unsteady simulations?	128
4.D	Production control scheme	132
4.D.1	Finite volume scheme	133
4.D.2	Numerical verification of the scheme	134
4.D.3	Conclusion	135
5	Benchmark of the numerical schemes in THYC-coeur	137
5.1	Presentation of the SPIN protection system and of the Bias Curves Database	138
5.1.1	SPIN protection system	138
5.1.2	Bias Curves Database	140
5.2	Definition of the performance indicators	141
5.3	Numerical results of the Bias Curves Database with the Bestion correlation	143
5.3.1	Verification of the implementation	143
5.3.2	Impact on the performance at an industrial stopping criterion	144
5.4	Numerical results of the Bias Curves Database with the Chexal-Lellouche correlation	146
5.4.1	Verification of the implementation	146
5.4.2	Impact on the performance at an industrial stopping criterion	148
5.5	Schemes comparison	150
III	ThermoTorch 1D without relative velocity	152
6	ThermoTorch: a 1D finite volume code	153
6.1	System of equations	153
6.1.1	Conservative balance equations	153
6.1.2	Unknowns considered and implemented equations	154

6.2	Discretization framework for finite volume schemes	157
6.2.1	Time discretization	157
6.2.2	Spatial discretization with a staggered grid	158
6.2.3	Control of the time step	159
6.3	Finite volume numerical scheme	159
6.3.1	Time stepping	159
6.3.2	Spatial integration and global numerical scheme	161
6.3.3	Matrix system formulation and variable resolution order	164
6.3.4	Steady-state computations	165
6.4	Configurations considered with the code	165
6.4.1	Steady-state Heated Channel Configuration	165
6.4.2	Unsteady configuration for Riemann problems	167

Appendix

6.A	Closure laws	169
6.A.1	Fraction used for vaporization	169
6.A.2	Relaxation time	169
6.B	Adjustement of thermodynamic parameters from water-steam data under reactor conditions for the EoS of the heated channel configuration	170
6.C	Thermodynamic coefficients for the density linearization	171
6.C.1	General case	172
6.C.2	Configuration for Riemann problems	172
6.C.3	Heated Channel Configuration	173

7 Verification of finite volume schemes in ThermoTorch 174

7.1	Verification of ThermoTorch for steady-state solutions	174
7.1.1	Solutions of ThermoTorch for two steady-state cases	175
7.1.2	Consistency and convergence rate of ThermoTorch scheme for steady-state . .	179
7.2	Verification of ThermoTorch for unsteady one-dimensional Riemann problems with the 3-equation model	181
7.2.1	Symmetric double rarefaction wave	181
7.2.2	Symmetric double shock wave	185
7.2.3	Numerical evaluation of the consistency error for solutions with shocks	191
7.2.4	Impact of the CFL value on the shock profile	193
7.2.5	Sod shock tube	194
7.3	Verification of ThermoTorch for a smooth one-dimensional Riemann problem with the 4-equation model	195
7.3.1	Analytical solution and methodology	196
7.3.2	Two-rarefaction waves solution with a constant mass fraction	198
7.3.3	Numerical evaluation of the consistency error due to the pseudo-entropy approximation	204
7.3.4	Conclusion on the consistency error	209

Appendix

7.A	Derivation details of the solution for the unsteady Riemann problem with the 4-equations model	211
7.A.1	General case	211
7.A.2	Symmetric configuration	212
7.B	Analysis of two THYC-coeur computations in accidental conditions.	213
7.C	Estimation of the consistency error due to pressure	214
7.D	Two rarefaction waves solution with a mass fraction jump	217

8	Steady-state convergence acceleration using a neural network	224
8.1	Methodology	225
8.1.1	Training a neural network to predict steady state solution	225
8.1.2	Characteristics of the Neural Network	226
8.1.3	Data generation and normalization	227
8.1.4	Proof of concept	228
8.2	Sensitivity analysis and performance obtained with the MSE Loss	229
8.2.1	Size of the training, validation and test datasets	230
8.2.2	Impact of the numerical parameters of ThermoTorch on the performance . . .	230
8.2.3	Optimization of the hyperparameters	231
8.2.4	Acceleration results with optimized neural network using MSE Loss	231
8.3	Optimization with regularization on normal modes	232
8.3.1	Correlation between prediction accuracy and acceleration performance	232
8.3.2	Normal mode analysis of the model	234
8.3.3	Acceleration results with regularization on normal modes	235
8.4	Conclusion and perspectives	236
Appendix		
8.A	Implementation	238
8.B	Sensitivity analysis on the size of the datasets	238
8.B.1	Training dataset size	238
8.B.2	Test dataset size	239
8.C	Sensitivity analysis on ThermoTorch numerical parameters	240
8.D	Sensitivity analysis on hyperparameters of the NN	241
8.D.1	Stopping number of epochs for training	241
8.D.2	Batch size	242
8.D.3	Number of neurons per layer and number of layers	243
8.E	Sensitivity analysis on the parameters of the MSE+DFT Loss	244
8.F	Using the 3-equation NN to predict the 4-equation model	245
Conclusion		246
Résumé en français		250
A	Introduction	251
A.1	Fonctionnement d'un réacteur à eau pressurisée	251
A.2	Présentation du coeur de réacteur nucléaire	252
A.3	Régimes d'ébullition	254
A.4	Rapport de Flux Thermique Critique	255
A.5	Etude bibliographique des codes de réacteur nucléaire	256
A.6	Approche poreuse	257
A.7	Caractéristiques du code THYC-coeur	258
A.8	Motivations industrielles d'accélérer le code THYC-coeur	258
B	Synthèse des travaux de thèse	259
B.1	Chapitre 1	259
B.2	Chapitre 2	260
B.3	Chapitre 3	261
B.4	Chapitre 4	262
B.5	Chapitre 5	263
B.6	Chapitre 6	264
B.7	Chapitre 7	266
B.8	Chapitre 8	267

Introduction

Cette thèse a été réalisée dans le cadre du contrat EDF CIFRE 2022-1027, sous la direction de Philippe Helluy. Elle s'est déroulée au sein du département Mécaniques des Fluides, Energies et Environnement (MFEE) sur le site EDF Lab R&D à Chatou, en collaboration avec l'Institut de Recherche Mathématiques Avancées (IRMA, UMR 7501) à l'Université de Strasbourg.

1 Nuclear energy in France

1.1 Operation of a pressurized water reactor

In 2023, nuclear energy enabled the production of 320,4 TWh of the energy in France, accounting for 65.3% of total electricity production. In 2025, Electricité de France (EDF) operates 57 nuclear power plants. These reactors have different powers. 32 reactors have an electrical power of 900 MWe and 20 reactors of 1300 MWe. There are also 4 N4-type reactors with a power of 1450 MWe and one EPR, with a power of 1650 MWe. All of these nuclear reactors are **Pressurized Water Reactors** (PWRs). A simplified diagram of a PWR is shown in Figure 1. A PWR relies on three circuits: the primary circuit, the secondary circuit, and the tertiary circuit (also called the cooling loop).

The primary circuit consists of several major components: the reactor core, the pressurizer, the steam generator, and the primary pump. A reactor includes 3 or 4 primary loops (steam generator, pump) supplied by a single reactor core. To extract the energy released from fission, a heat transfer fluid (water) is pumped into the core by the primary pumps. It heats up as it comes into contact with the fuel. During normal operation of a nuclear power plant, the water remains mostly in liquid form thanks to the high pressure maintained by the pressurizer. The heated water then flows into the steam generator, which is a heat exchanger where heat from the primary circuit is transferred to the secondary circuit. In the secondary circuit, the water is vaporized into the steam generator and drives the various turbines to generate electricity. After delivering its work in the turbine, the steam is condensed in the condenser and then injected again in the steam generator. Cooling in the condenser is provided by the tertiary circuit, which draws water from a river or the sea. Depending on the plant's configuration (open or closed), a cooling tower may be added to the tertiary circuit. This thesis focuses on the flow within a single component of the primary circuit: the reactor core. This component is detailed in the following section.

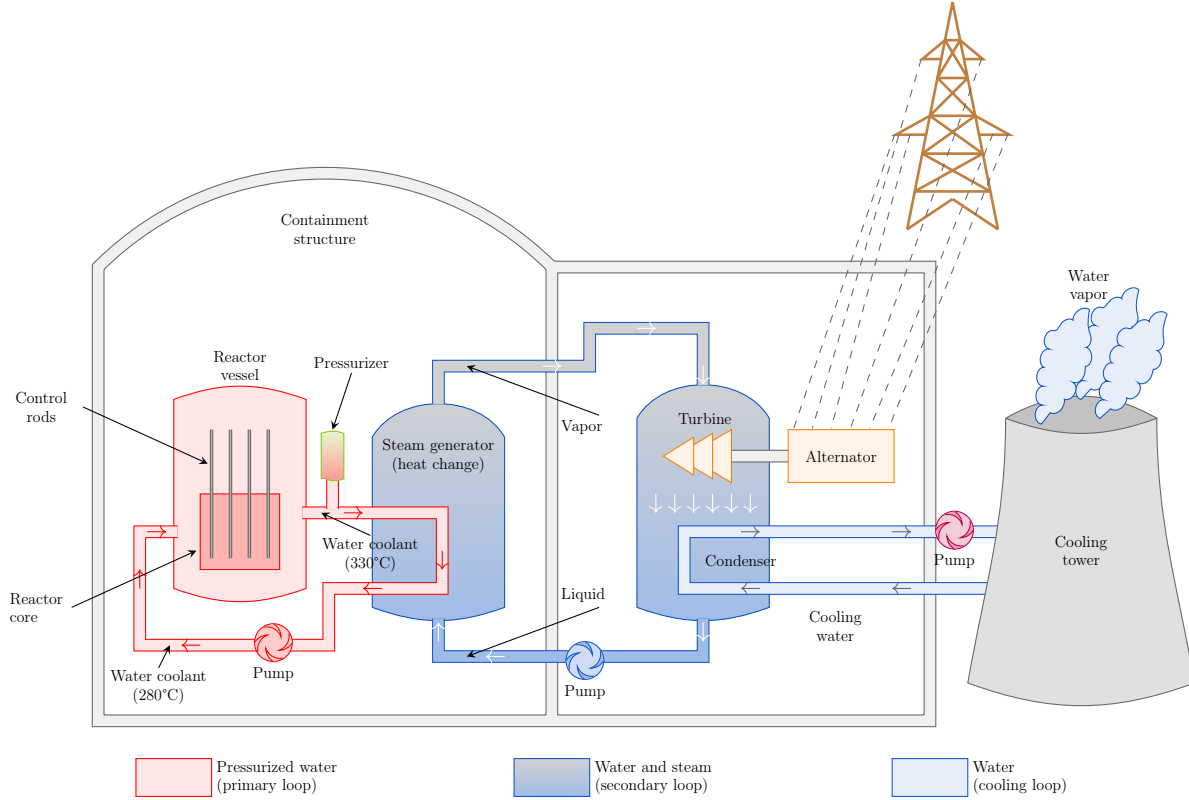


FIGURE 1

Diagram of a Pressurized Water Reactor (PWR). Tikz diagram from Gloria Faccanoni's thesis [41].

1.2 Nuclear reactor core composition

In the reactor core, water flows bottom-up. It enters at a temperature of about 290°C and exits the core at approximately 320°C . It is pressurized to around 155 bar, which enables the water to remain mostly in liquid form despite the very high temperature. Under normal operating conditions, a small portion of the flow may vaporize into steam. Under accident conditions, a two-phase water-steam mixture with a non-negligible amount of steam may be encountered. The mixture velocities are roughly $5\text{ m}\cdot\text{s}^{-1}$ for nominal conditions.

The reactor core is loaded with nuclear fuel composed of uranium oxide (UO_2) enriched to about 4% uranium-235. The fissile element can also be plutonium for MOX fuel. This fuel is manufactured in the form of 1.35 cm high **pellets**. Several pellets are inserted into **fuel rods**, which are encased in cladding around the fuel. This cladding, made of zirconium alloy, ensures the fuel rod's sealing to prevent the release of radioactive material into the primary circuit. An **assembly** is made up of 264 rods grouped together and is 4 or 5 m long. Figure 2 shows the pellets, rods, and a fuel assembly. An assembly also includes guide tubes for inserting control rods, which regulate the nuclear reaction. These control rods are attached to a cluster located above the assembly. Within the assembly, spacer and mixing grids are placed at regular intervals. The spacer grids keep the fuel rods in place within the assembly, while the mixing grids promote flow mixing to homogenize the water. This homogenization prevents hot spots by introducing turbulence. These grids help to avoid safety-critical conditions such as the boiling crisis described below.

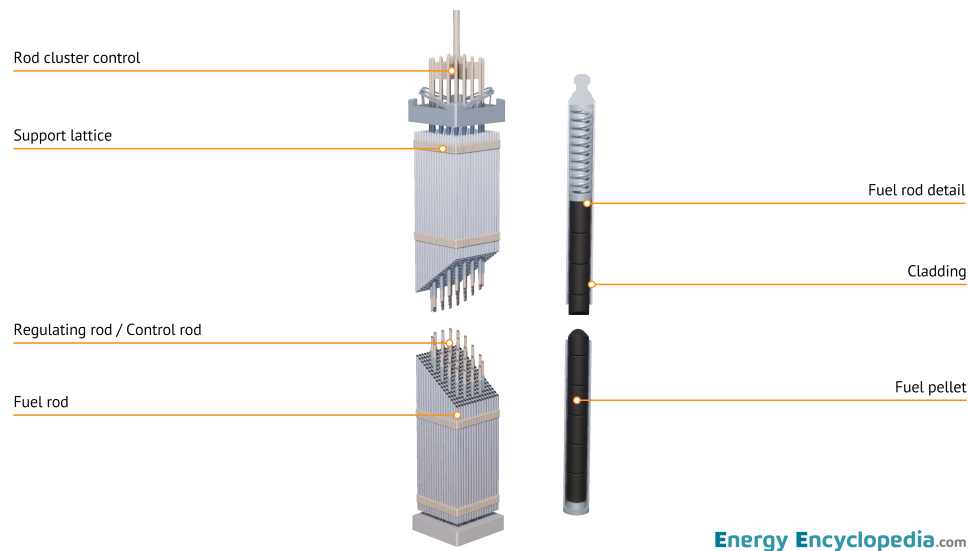


FIGURE 2

Scheme of a fuel assembly composed of fuel rods containing pellets of uranium. Credit: EnergyEncyclopedia.com.

The reactor core is composed of a steel vessel called the Reactor Pressure Vessel, which withstands the pressure inside the core. The vessel is filled with a variable number of assemblies depending on the desired core power: 157 assemblies for a 900 MWe reactor, 193 for 1300 MWe, 205 for 1450 MWe and 241 for EPR. Water flows through this highly obstructed environment, featuring very complex solid geometries (grids, rods, guide tubes, etc.). Despite the fact that the flow is mainly axial from bottom to top, transverse flows can occur between assemblies, as the space between them is left open. Inside the assemblies, water primarily flows through the spaces between the fuel rods and remove the heat produced by the nuclear reaction. The space between four fuel rods is called a **sub-channel**, as shown in Figure 3.

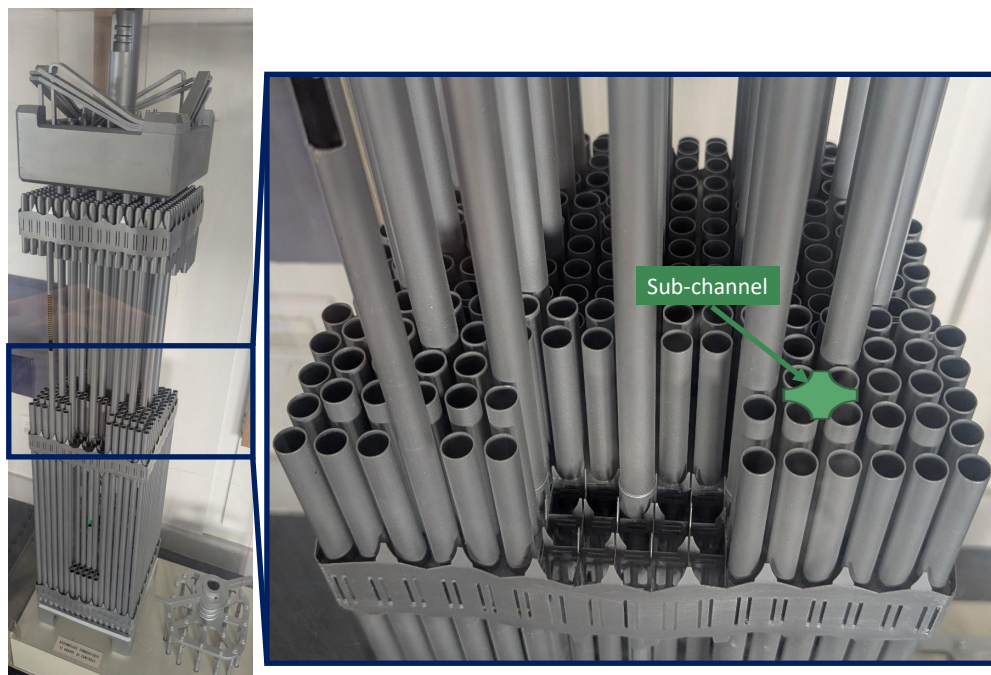


FIGURE 3

Image of a fuel assembly model with a zoom on the sub-channels. Source EDF R&D.

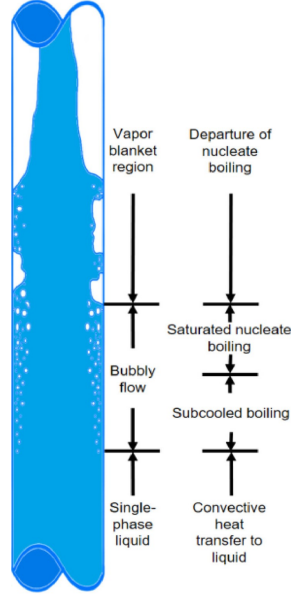
2 Boiling two-phase flow

2.1 Boiling regimes

At reactor core pressure of 155 bar, the saturation temperature of water is approximately 350°C . Water enters the reactor core as a turbulent single-phase liquid flow, with a Reynolds number of $Re \sim 500000$. As the fluid is heated during its upward passage through the reactor core, it may undergo various boiling regimes and heat transfer mechanisms. The two phases, liquid and gas, can exhibit a variety of complex structures. Steam can appear in the form of small bubbles of vapor (of a size of around $\sim 10^{-4}$ m) to large vapor pockets encompassing the entire space between fuel rods (around $\sim 10^{-2}$ m). The different regimes are described in Figure 4.

The heat transfer begins with a pure liquid convective heat transfer, as the liquid is heated toward saturation. As the heat flux increases, the wall temperature T_w increases. Once the wall temperature exceeds the saturation temperature T_{sat} , the **Onset of Nucleate Boiling (ONB)** begins. From the ONB, the **nucleate boiling** (see Figure 4) occurs. The wall surface is sufficiently hot to vaporize water at its surface. Vapor bubbles appear at the wall surface and can detach. The liquid is not at saturation on average but vapor already appears on the rods: this is the **subcooled boiling**. When the enthalpy of the fluid mixture increases, the nucleate boiling continues. Slugs and columns of vapor can appear. These bigger structures detach from the wall surface and are convected upward.

As the enthalpy increases further, the vapor structures agglomerate into vapor pockets. Nucleate boiling may abruptly transition to film boiling: when this point is reached the heat flux is called **critical heat flux**. This phenomenon is called the **Departure from Nucleate Boiling (DNB)**. The difference between the temperature of the cladding and the fluid temperature varies violently from several degrees to hundreds of degrees. This is the boiling crisis during which the wall temperature is so high ($> 1000^{\circ}\text{C}$) that thermo-mechanical damage can occur in the cladding, threatening the integrity of the fuel rods in a nuclear reactor core. The boiling crisis is especially dangerous because it involves an hysteresis phenomenon. Once the critical heat flux is exceeded, the heat flux must be reduced all the way to the Leidenfrost point [55] to return to nucleate boiling. It is therefore crucial to ensure that the critical heat flux is not reached at any location within the nuclear reactor core. Under normal reactor core operating conditions, the heat flux remains within the subcooled nucleate boiling region.



(a) DNB

FIGURE 4

Boiling regimes for an upward low quality flow. Illustration extracted from [141].

2.2 Departure From Nucleate Boiling Ratio

The distance to **DNB** is measured with the **Departure from Nucleate Boiling Ratio (DNBR)**. It is defined by the ratio between the local heat flux ϕ and the critical heat flux ϕ_{DNB} that would trigger the **DNB**:

$$DNBR = \frac{\phi_{DNB}}{\phi}. \quad (1)$$

DNBR is a local quantity defined at each point in the reactor core. It is essential to ensure that the value is always greater than one everywhere in the reactor core ($DNBR > 1$). The flux ϕ_{DNB} is obtained using a **critical heat flux correlation**. This correlation predicts the local value of ϕ_{DNB} as a function of the surface average values in a sub-channel of pressure P , mass flux of the mixture G , and equilibrium quality X (see Equation (5.1)), such that

$$\phi_{DNB} = \phi_{DNB}(P, G, X). \quad (2)$$

The parameters of the correlation are fixed based on experimental critical heat flux test results. To evaluate the DNBR in a reactor core during design or fuel reload planning, a thermal-hydraulic code can be used. This code must be able to provide local fields of pressure, mass flux, and dynamic quality throughout the entire core for a two-phase flow with subcooled boiling. Once the DNBR field is determined, the minimum value over the entire core is considered, as it is the most limiting point in terms of margin to DNB. This minimum value is called **DNBRmin** which should satisfy

$$DNBR_{min} > 1. \quad (3)$$

3 THYC-coeur: a component code simulating 3D flows in nuclear reactor cores

To ensure the safety of PWRs, EDF developed the thermal-hydraulic code THYC-coeur [7], which simulates the two-phase flow in the reactor core. This code enables the calculation of the 3D field of

Departure from Nucleate Boiling Ratio (DNBR) in the core. This section begins with a brief review of nuclear reactor core codes. Two possible approaches to handle geometry are described in Section 3.1. The porous media approach is the one used in THYC-coeur and is presented in Section 3.2. The characteristics of THYC-coeur code are detailed in Section 3.3. Finally, Section 3.4 presents the safety studies done with THYC-coeur and explains the motivation for accelerating the code.

3.1 A brief review of reactor core codes

For thermal-hydraulic simulation, several scales can be considered (see Figure 5).

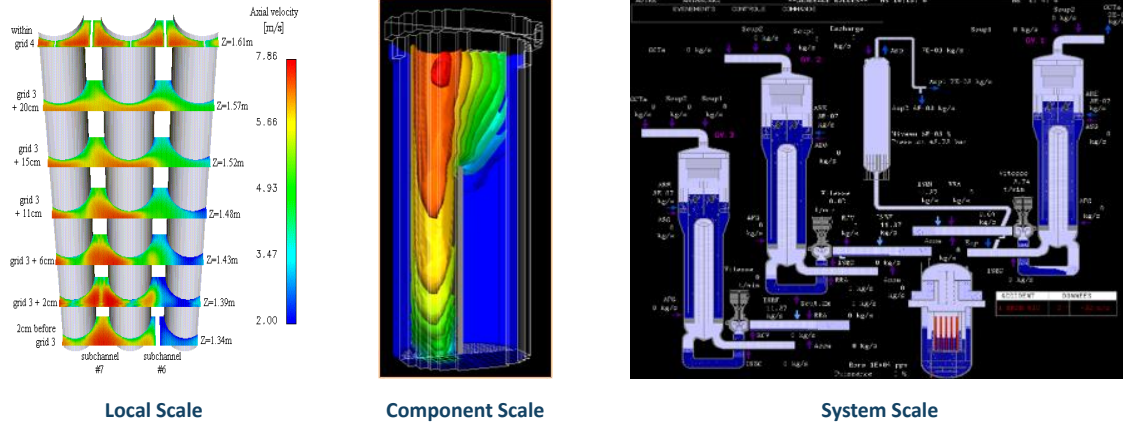


FIGURE 5

Illustration of the different simulation scales: local, component and system. Source: EDF R&D.

- The first and most macroscopic scale is the **system scale**, where several components of a circuit are simulated simultaneously. These system codes include CATHARE [14] in France, TRACE [8, 112], RELAP5 [46] in the USA, and MARS [26] in South Korea. To precisely evaluate the DNBR field over the entire reactor core, this scale is too macroscopic.
- The second scale that can be considered is the **local CFD scale**, using CFD codes such as code_saturne [4] developed by EDF or NEPTUNE_CFD [65, 103], jointly developed by EDF, Framatome, CEA and ASNR. The local scale is precise but is unable to simulate a full reactor core with current computational capabilities. It would require hundreds of billions of cells to represent the full core with a wall resolved mesh. These codes are currently used to simulate flow locally, for a bundle of just a few rods at most when resolving the wall or an assembly without resolving the wall.
- The third scale is the **component scale**. The entire reactor core is represented with a mesh of cells of a size in the order of centimeters, which makes it possible to calculate the flow in the entire core in an acceptable CPU time. With a component scale code, fields such as P , G , and X can be obtained over the entire reactor core, allowing DNBR field and DNBRmin to be evaluated. The trade-off of a fast component-scale code is that the model operates at a macroscopic scale, involving macroscopic terms that require closure laws. Experiments and high-fidelity CFD simulations are necessary to determine the parameters of the correlations used in these closure laws. Therefore, these closure laws are only validated over specific ranges of physical quantities and configurations.

Many component-scale codes exist worldwide to simulate two-phase flows in a reactor core. Two approaches are possible to account for the complex geometry of fuel assemblies in a reactor core.

- The **coupled sub-channel approach** simulates each sub-channel independently and couples them numerically. It is highly efficient in terms of computational time.

- The **porous media approach** integrates the solid structures into a cartesian mesh through a porosity field (see Section 3.2). With its true 3D model, the porous approach enables a finer representation of transverse flows.

For both approaches, two main physical models are used.

- The **drift-flux model** [81, 82] considers the water and steam phases as a homogeneous mixture. It can be formulated with 3 or 4 equations depending on the physical phenomena taken into account. The drift-flux model is robust and enables fast simulations. The velocity difference between the two phases is modeled by a relative velocity, obtained from a drift-flux correlation law [145]. A partial differential equation for relative velocity can be added to the model instead of the drift-flux correlation.
- The **two-fluid model** [82] represents each phase separately by its conservation laws. In this model, the difficulty lies in the modeling of the transfer terms between phases. This model easily simulates configurations with two phases out of thermodynamic equilibrium or cases where the vapor and liquid velocities differ greatly. The two-fluid model can also be extended to a three-phase framework with a 9-equation model. It is the case for COBRA-TF [135] and CTF [125] codes.

Table 1 lists some component codes according to their choice of geometric approach (sub-channel / porous media) and physical model (drift-flux, two-fluid model).

	Sub-channel approach	Porous approach
Drift-Flux model	FLICA III-F (Framatome - France) COBRA III-C [119] / IV [134] (PNNL - USA) THINC IV [21, 25] (Westinghouse - USA) VIPRE-01 [30, 132] (EPRI - USA) VIPRE-W [94] (Westinghouse - USA) SUBCHANFLOW [79, 126] (KIT - Germany) ASSERT-PV [118] (AECL - Canada) MATRA [142] (KAERI - South Korea) ATHAS [128] (XJTU - China) CORTH-v2.0 [34] (NPIC - China) LINDEN [10] (CGN - China)	THYC-coeur [7] (EDF/Framatome - France) FLICA IV [136] and FLICA V [102] (CEA - France) ESCOT [43, 97] (KAERI - South Korea)
Two-fluid model	COBRA-TF [135] (USA) CTF [125] (USA) VIPRE-02 [87] (EPRI - USA) ATHAS-02 [77] (XJTU - China)	COMMIX [35, 104] (Argonne National Laboratory - USA) CUPID [143, 144] (KAERI - South Korea)

TABLE 1
Comparison table of schemes characteristics for THYC-coeur applications.

Several codes are also derived from the American COBRA code: the Bulgarian code COBSOFM [114], which is derived from COBRA III-C and adapted for VVER reactors, or the code developed by Areva NP called COBRA FLX [5]. Framatome also developed a code from COBRA-TF called F-COBRA-TF [57].

3.2 Porous approach

The porous approach used in THYC-coeur considers control volumes containing both a solid volume V_s and a fluid volume V_f (see Figure 6). The exact geometry of the solid is disregarded. For each

control volume with a total volume $V_f + V_s$, the porosity is defined as

$$\varepsilon = \frac{V_f}{V_f + V_s}. \quad (4)$$

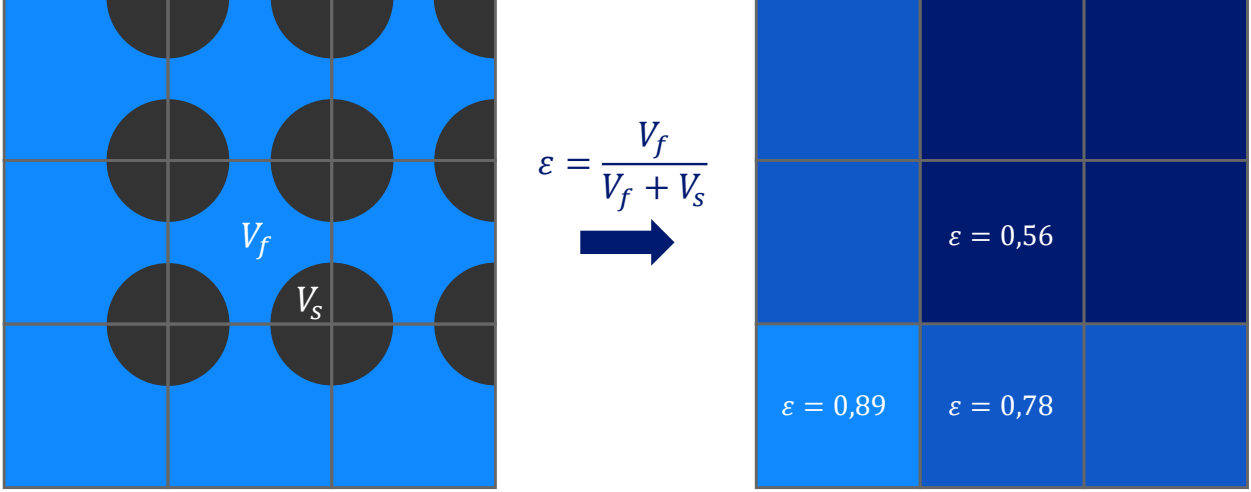


FIGURE 6
Diagram explaining the porous approach.

The model equations are integrated over these control volumes. The presence of solid matter is therefore only taken into account through porosity ε and source terms appearing in the equations (heat input, pressure losses). These closure terms are modeled by physical laws that are representative of the case under consideration. For example, mixing and spacer grids are represented by singular pressure losses in the model.

This method first allows to get rid of the complex geometry of solids in the reactor core. It enables modeling flows in a cluttered medium with mesh sizes on the order of a centimeter. It uses a 3D cartesian mesh to represent the reactor core, making the code efficient in terms of CPU time.

3.3 Characteristics of THYC-coeur code

First developments of the THYC-coeur code began in 1986. A validation report on THYC-coeur was written in 1998, based on numerous experimental test results. In 2006, the French nuclear safety authority approved the use of THYC-coeur for safety demonstrations. In 2018, co-development between Framatome and EDF began to integrate THYC-coeur into the new ODYSSEE calculation chain. A new validation report was submitted to the safety authority in 2025. In EDF, the THYC-coeur code is developed in the department *Mécanique des Fluides, Energies et Environnement* (MFEE) of EDF R&D. In order to meet the needs listed in Sections 1.2 and 2.1, several modeling choices were made for THYC-coeur code.

- Fast and accurate computation of 3D flows throughout an entire reactor core.
→ **Component scale code.**
- Consideration of actual 3D flows, particularly transverse flows between sub-channels and assemblies; Component-scale simulation of flows in a cluttered medium with complex geometry.
→ **3D porous media approach.**
- Simulation of two-phase flows with phase change at high temperature and high pressure; Consideration of the velocity difference between the vapor phase and the liquid phase.
→ **Drift-flux model** [81, 82].

- Consideration of the subcooled boiling phenomenon.
→ Addition of the **disequilibrium equation** on the mass fraction of vapor, leading to a **4-equation** drift-flux model.
- Conservation of physical quantities (mass, energy...).
→ **Finite volume formulation**.

The subcooled boiling phenomenon has a significant impact on the DNBR value. Indeed, subcooled boiling increases the dynamic quality X , which can greatly reduce the critical heat flux ϕ_{DNB} . Therefore, it is important to take this phenomenon into account to predict the DNBR value observed in real conditions.

To define the drift-flux model in THYC-coeur with a porous approach, several closure laws are required for the macroscopic quantities appearing in the equations (these terms are discussed in **Chapter 1**). Correlations (parameterized functions) are used as closure laws. These correlations are developed and calibrated based on results from experimental tests or high-fidelity CFD calculations (DNS, LES). In particular, the drift-flux model relies on the drift-flux correlation [145] for the relative velocity between phases, noted u_r . This correlation is based on a formulation that takes into account certain physical phenomena such as the buoyancy of the vapor in the liquid, but also the effects of non-uniform distribution of the vapor void fraction and flow velocity (lower velocity near the wall and higher at the center of sub-channels). In THYC-coeur, these parameters are obtained either with the Chexal-Lellouche correlation [22, 23, 24], or with the Bestion correlation [14]. In THYC-coeur, the power at the surface of fuel rods is an input parameter. It is either provided by the user or comes from a coupling with a neutronics code and a fuel rod thermal code.

3.4 Industrial motivations to accelerate THYC-coeur

THYC-coeur is mainly used to obtain the steady-state of two-phase flows for nominal or accidental conditions, through an unsteady simulation. As part of the safety studies carried out for a new reactor, for design studies during ten-year inspections or for fuel reloading safety analysis, a very large number of accidental scenarios must be analyzed. At the time the code was developed, a small number of calculations with very conservative assumptions were performed for each study, leading to very penalizing studies, which sometimes deviate from reality. Recently, a more realistic treatment of the core's initial conditions and physical parameters has led to a significant increase in the number of configurations studied. Moreover, in 2017, THYC-coeur was selected to be integrated into the new industrial computational chain ODYSSEE for future studies, starting with the design of the EPR2 plants. As a result, the number of computations performed with THYC-coeur has been increasing significantly over the years, now reaching millions of simulations each year.

Compared to a sub-channel approach, the 3D modeling with a porous approach enables for better flow modeling, particularly the transverse flows between sub-channels. However, this choice impacts the computation time. A reactor core computation with an industrial mesh takes about 10 seconds with THYC-coeur (on a mono-core workstation with an industrial mesh using the quarter symmetry of the core), whereas a sub-channel code calculation takes only a few seconds, with FLICA for example. The computations with THYC-coeur are already fast, especially compared to CFD calculations. But millions of THYC-coeur calculations are launched simultaneously for industrial studies, which represent a non-negligible CPU time. Reducing computational time is therefore critical. This thesis contributes to the ongoing effort to accelerate THYC-coeur code, aiming to reduce computational time without compromising the accuracy of the results.

4 Contents of the thesis

The objective of this work is to propose methods to accelerate the THYC-coeur code, a partially unbalanced homogeneous two-phase model in a heterogeneous porous medium. To this end, the thesis is divided into three main parts, which are organized as follows.

Part I: Two-phase flow model The first part, which includes the first three chapters, is dedicated to the presentation and analysis of drift-flux models for two-phase flows. The drift-flux models [81, 82] are presented in **Chapter 1**. In particular, the chapter details the steps to go from the local conservation equations of each phase to the drift-flux model. Drift-flux models are mixture models that include several macroscopic terms that need to be modeled. **Chapter 1** presents some closure laws used to obtain these terms. In particular, the closure law used for the relative velocity using the *Zuber and Findlay* model [145] is discussed. The resulting PDEs models are analyzed in **Chapter 2**, where the model is studied to determine whether it is hyperbolic, in order to ensure meaningful time-dependent solutions.

Part II: THYC-coeur with relative velocity The second part, which includes the fourth and fifth chapters, proposes new numerical schemes for the fourth equation on the vapor mass fraction for the industrial THYC-coeur code, with a relative velocity. **Chapter 4** presents three new schemes for the fourth equation. Several test cases are simulated to verify the schemes and their convergence rates. In **Chapter 5**, the schemes are implemented in the industrial THYC-coeur code, and a database of industrial cases is used to continue the verification of the schemes and compare the performance in terms of CPU time and accuracy of the schemes.

Part III: ThermoTorch 1D without relative velocity **Chapter 6** presents the ThermoTorch code. It is a simplified prototype with zero relative velocity that reproduces certain characteristics of THYC-coeur. This code is used in **Chapter 7** to verify the numerical schemes of the code for both steady and unsteady configurations. Finally, **Chapter 8** presents an AI-driven acceleration method and shows the results obtained with ThermoTorch.

A more detailed summary of the content of each chapter can be found in the **Synthesis of thesis work** section below.

Synthesis of the thesis work

1 Chapter 1

In this chapter, the physical models considered in the work of this thesis are presented. These models allow for the simulation of two-phase flow (water-steam) in a medium obstructed by the presence of solid material. They are derived from the models proposed by Ishii in [81, 82].

For each phase, noted $k \in \{g, l\}$, with g denoting the gas and l the liquid, the local conservation equations (mass, momentum, and energy) are considered. In order to define the quantities for each phase at any instant, the local equations are time-averaged. This time-averaging introduces the local void fraction α_k , which corresponds to the probability of finding phase k at a given time and location. The time-averaging also introduces interfacial exchange terms between phases (mass transfer, momentum transfer, and energy transfer).

The equations are then averaged in space over a volume V_0 containing a solid volume V_s and a fluid volume V_f . This allows to avoid dealing with the complex geometry of the solids. This introduces the fluid porosity ε given by

$$\varepsilon = \frac{V_f}{V_f + V_s}. \quad (5)$$

The spatial averaging also introduces source terms due to the presence of solids. A six-equation model averaged in time and space is obtained. Considering that the two-phase flow behaves similarly to a homogeneous mixture, the objective here is to simplify the model in order to obtain a robust physical model that is not costly in CPU time to simulate. After defining mixture quantities, the equations are summed for each phase to obtain three mixture equations. This summation eliminates the interfacial transfer terms. By taking the difference between the energy equation of each phase, a disequilibrium equation on the relative specific enthalpy is obtained. The same process is done with the momentum equations to obtain an equation on the relative velocity between phases, denoted $\mathbf{u}_r$ and defined by

$$\mathbf{u}_r = \mathbf{u}_g - \mathbf{u}_l, \quad (6)$$

with $\mathbf{u}_g$ the velocity of the gas phase and $\mathbf{u}_l$ the velocity of the liquid phase. The last disequilibrium equation is the equation on the mass fraction of vapor. To reduce the number of equations, several assumptions are made:

- The minority phase is always considered at saturation, which allows the removal of the disequilibrium equation for specific relative enthalpy,
- The relative velocity is obtained through a closure law, which eliminates the equation on the relative velocity. This closure law is derived from the drift-flux theory developed in [145].

If the equation for the mass fraction of vapor is considered, an unbalanced 4-equation model is obtained. This allows modeling of non-equilibrium thermodynamic flow, particularly the phenomenon of subcooled boiling, where boiling can begin even when the liquid is not yet at saturation in average (a spatial averaging operator is used). To do this, the interfacial mass transfer term involved in the

mass fraction of vapor equation must be modeled. If the mixture is considered to be at saturation, this equation disappears and the 3-equation model at equilibrium is obtained.

In this chapter, two 4-equation models are presented: a conservative model in total energy, and an enthalpy-based model. The enthalpy-based model corresponds to the one implemented in the industrial code THYC-coeur [7]. To obtain it, several additional assumptions were made: the heat contribution due to viscous friction between phases and to solid friction has been neglected, and the sum of the transfer of thermal energy between phases has been neglected in the enthalpy balance.

To close the systems of equations presented, several terms must be modeled, such as fluid friction, solid friction, relative velocity, mass transfer. For this, closure laws modeling the main physical phenomena are used. In this chapter, these closure laws are described, in particular for the mass transfer term and for the relative velocity. For the relative velocity, two different correlations are proposed: the Bestion correlation [14] and the Chexal-Lellouche correlation [22].

2 Chapter 2

The models from **Chapter 1** are studied in **Chapter 2** in simplified configurations to determine whether the models derived from the 4-equation drift-flux model are hyperbolic in order to ensure stable time-dependent solutions. The following models are analyzed in the chapter:

- **Drift-flux barotrop model with a constant relative velocity:** The drift-flux model is considered without an equation on energy and for a constant relative velocity. The eigenvalues of the system can not be obtained analytically. Using the intermediate value theorem, the model is **strictly hyperbolic** for $u_{r0} \neq 0$.
- **4-equation model without relative velocity:** This model is equivalent to an HRM model [16]. Considering c the speed of sound of the mixture, the eigenvalues are $\lambda \in \{u, u, u+c, u-c\}$. This model is **hyperbolic**. This implies having an equation of state for the mixture that ensures a real speed of sound.
- **Drift-flux 4-equation model with a constant relative velocity u_{r0} :** This model is more complex than the HRM model. Only one analytical eigenvalue is obtainable $u_l = u - yu_{r0}$. The other analytical eigenvalues are not obtainable. The characteristic polynomial is evaluated for the gas velocity $u_g = u + (1 - y)u_{r0}$. Using the intermediate value theorem, the system is hyperbolic if the derivative of the characteristic polynomial is negative in $u_l = u - yu_{r0}$. This is a sufficient condition for hyperbolicity. With p the pressure, ρ the density of the mixture and y the mass fraction, the system is **hyperbolic** if

$$C_1^h(p, \rho, y) < 0 \quad \text{or} \quad |u_{r0}| < u_r^{max}(p, \rho, y) = \sqrt{\frac{C_2^h}{C_1^h}} c(p, \rho, y), \quad (7)$$

with

$$\begin{cases} C_1^h(p, \rho, y) = ya^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = ((1 - \rho y v_g(p)), \end{cases} \quad (8)$$

where c is the mixture speed of sound and $a = \left(\frac{\partial e}{\partial p} \right)_{\rho, y}$. Once again, this result is valid when the mixture equation of state ensures a real speed of sound. In the appendix to this chapter, this sufficient condition is studied for a water-steam mixture at 155 bar.

3 Chapter 3

The objective of this chapter is to propose analytical solutions for simplified versions of the models presented in **Chapter 1**. The first part focuses on the fourth equation concerning the mass fraction of vapor only, in order to provide analytical solutions for testing finite volume schemes in **Chapter 4** by isolating the equation from the complete system. A free medium is considered here. Three new exact solutions are presented. First, two **steady-state** solutions, with most of the equation parameters taken constant $(\rho_0, \tau_0, u_0, \bar{y}_0)$, are presented. The first one uses a constant relative velocity u_r . The second one uses the approximated Bestion correlation presented in **Chapter 1**. This correlation proposes the relative velocity as a function of the mass fraction:

$$u_r(y) = \frac{1 + (\delta - 1)y}{(1 - y)u_b}, \quad (9)$$

with δ and u_b constant parameters. These analytical solutions are used in **Chapter 4** to verify steady-state numerical schemes for the fourth equation on the mass fraction. The third solution is an **unsteady** solution of the fourth equation, once again using the approximated Bestion correlation $u_r(y)$. This self-similar solution gives the mass fraction as a function of the variable ξ :

$$\xi(x, t) = \frac{x}{u_0 t}. \quad (10)$$

This solution is also used in **Chapter 4** to verify the scheme for unsteady solutions. The second section of the chapter is extracted from an article published during the PhD thesis [70]. The three-equation model is considered without relative velocity $u_r = 0$, corresponding to an HEM-type model [16] in a porous medium of porosity $\varepsilon(x)$ and with an energy input $\phi(x, t)$. Two new self-similar analytical solutions from [70] are presented for two different configurations of this model. These solutions will be used for verification of the industrial code THYC-coeur in future work. The solutions depend on the variable ξ :

$$\xi(x, t) = \frac{x}{t + t_0}, \quad t_0 > 0. \quad (11)$$

The first solution is obtained without heat input and with a non uniform porosity $\varepsilon(x)$ satisfying:

$$\varepsilon = \left(\frac{x}{x_0} \right)^\alpha, \quad (12)$$

with α and x_0 constants. The analytical solution is proposed for a general equation of state, and the particular case of the ideal gas is detailed at the end. For the second solution, a flow with an arbitrary equation of state in a free medium is considered. A heat input of the following form is applied:

$$\phi(x, t) = \frac{1}{t + t_0} \psi(\xi). \quad (13)$$

The derivation of this solution leads to an ODE system in ξ that must be solved numerically.

Chapter 3 also includes three appendices to detail already known solutions used in this work. The first appendix presents a steady-state solution of the three-equation model from [78], adapted to the case under consideration. This analytical solution is obtained using a Newton method and for a general equation of state. It is used in **Chapter 7** to verify the ThermoTorch code in the steady-state three-equation configuration.

In the second appendix, solutions to Riemann problems for the three-equation model with an ideal gas equation of state are presented. In particular, the three Riemann problems considered are a symmetric double rarefaction wave, a symmetric double shock wave, and the Sod shock tube. They

are used in **Chapter 7** to study the behavior of the ThermoTorch numerical schemes for both regular solutions and shock solutions when using the 3-equation model.

Finally, the last appendix presents a mixture equation of state derived from a mixture of ideal gases with the same polytropic index γ . The equation of state is written as:

$$e(p, \rho, y) = ye_g + (1 - y)e_l = \frac{p}{\rho(\gamma - 1)} + h_0(y), \quad (14)$$

with

$$h_0(y) = yh_{0g} + (1 - y)h_{0l}, \quad (15)$$

where γ , h_{0g} and h_{0l} are constant parameters to set. This particular equation of state is used to find a solution to a Riemann problem for the 4-equation model. This solution is used in **Chapter 7** to study the unsteady numerical scheme of ThermoTorch with the four-equation model, in particular the scheme used for the energy equation.

4 Chapter 4

This chapter is an extended version of an article published during the PhD thesis [93]. It focuses on the disequilibrium equation for the mass fraction of vapor y :

$$\frac{\partial(\rho y)}{\partial t} + \nabla \cdot (y\mathbf{q}) + \nabla \cdot (y(1 - y)\mathbf{q}_r) = \rho \frac{\bar{y} - y}{\tau} + \Gamma_p, \quad (16)$$

where ρ is the mixture density, $\mathbf{q}$ is the mixture mass flux, $\mathbf{q}_r = \rho\mathbf{u}_r$ the relative mass flux, $\bar{y}$ the mass fraction at equilibrium, τ the relaxation time to equilibrium and $\Gamma_p > 0$ the vapor production. This equation is similar to the one studied in [53]. It models non-equilibrium flow, with a mass fraction y that tends to deviate from the equilibrium mass fraction $\bar{y}$ due to a direct production term Γ_p , and is brought back to equilibrium after a characteristic time τ . The originality of this transport equation comes from the drift-flux term, which is a convection term of the nonlinear quantity $y(1 - y)$ by the relative mass flux $\mathbf{q}_r$.

First, the continuous maximum principle is studied on Equation (16) to ensure that the mass fraction remains between zero and one. Then, three finite volume schemes - called QRd, QRq, and QG - are proposed for this equation. These schemes are presented in an article published during the PhD thesis [93]. These are linear implicit schemes, presented in a multidimensional unstructured framework. The study is limited to linear numerical schemes, so that these can be used in the THYC-coeur code. They respect the maximum principle for the vapor mass fraction, which must remain between 0 and 1. To achieve this, no time step condition is required for the QRd and QRq schemes. The QG scheme has a constraint on the time step, which is **not limiting in the concerned applications**. In this context, the production term, which does not necessarily respect the continuous maximum principle, is not considered in these schemes.

The schemes are implemented in a 1D prototype that simulates only the fourth equation. Using analytical solutions of the fourth equation from **Chapter 3**, the schemes are first tested on one-dimensional **steady-state** solutions. This steady-state solution is obtained for constant parameters $(\rho, q, q_r, \tau, \bar{y})$. Two test cases representative of reactor flow conditions are studied. This verification demonstrates the consistency of the three schemes, which have a convergence rate equal to one in space. The QG and QRq schemes appear to be more accurate than the QRd scheme for a fixed mesh, especially when a significant negative relative velocity is considered.

The proposed QRd and QRq schemes are not conservative in unsteady conditions. They are therefore not suitable for simulating unsteady solutions, as shown in the appendix of this chapter. The QG scheme can be verified on an unsteady solution. By considering a constant CFL value, a convergence curve is produced for the unsteady solution of the fourth equation proposed in **Chapter 3**.

A convergence rate close to one is obtained for the QG scheme, as expected. In **Chapter 5**, these schemes are then implemented in the industrial code THYC-coeur in order to evaluate their performance impact compared to the reference scheme of THYC-coeur. In this study on the THYC-coeur code, the disequilibrium equation is no longer treated independently as in **Chapter 4** but as part of the complete THYC-coeur system (see **Chapter 1** for the model).

In the appendix of this chapter, a numerical scheme is proposed for the production term Γ_p in order to recover the maximum principle at the discrete level, that is, to ensure that the mass fraction always remains between 0 and 1. Indeed, without control, an arbitrary model for the term $\Gamma_p > 0$ can cause the mass fraction to evolve to values greater than one. At the discrete level, the proposed GAMc scheme allows the equation to be modified locally to ensure that the solution remains below one. This corrective scheme can only be used for steady-state solution search, since the unsteady behavior is altered due to the control.

5 Chapter 5

The three schemes proposed in **Chapter 4** have been implemented in the industrial code THYC-coeur [7]. To verify their implementation and consistency, a database of 36 148 steady-state industrial cases is used. These cases correspond to a configuration of an N4 reactor, a French nuclear reactor type, under different physical conditions (inlet temperature and mass flux, outlet pressure, power shape). This database is called the Bias Curve Database. It was originally created to configure a protection system called SPIN (*Système de Protection Intégré Numérique* in French), used to ensure the safety of an operating facility. First, the Bias Curve Database is presented, along with the SPIN protection system. Then, the three schemes introduced in **Chapter 4** [93] are compared to the reference scheme of THYC-coeur on the Bias Curve Database. Two different models are considered, depending on the correlation used for the relative velocity. The first model is based on the Bestion correlation [14], while the second is based on the Chexal-Lellouche correlation [22]. For each scheme, the performance in terms of accuracy (compared to the reference scheme) and in terms of CPU time are evaluated. The comparisons are carried out using industrial numerical parameters (mesh size, time step, etc.) and aim to evaluate the performance of the schemes in predicting the solution of the continuous equation model when the industrial numerical parameters are fixed.

The results obtained validate the implementation and consistency of the three new schemes for both models. These new schemes provide increased robustness for the code. For the model using the Chexal-Lellouche correlation, the schemes also provide a 50% gain in CPU time compared to the reference scheme, when considering the same numerical parameters (time step, stopping criteria, etc.). The findings of this comparison are summarized in Table 2, which lists the advantages and limitations of using each scheme in the THYC-coeur context. It shows that the QG scheme is the most appropriate scheme for use in THYC-coeur. In particular, it is very robust, provides the best accuracy for a given mesh with the Chexal-Lellouche correlation, and can be used for unsteady simulations. The QG scheme has a weak constraint on the time step, unlike the QRd and QRq schemes, but this constraint is not actually limiting in practice. In particular, it is not reached for the applications tested here.

Scheme	Constraints time step	Other constraints	Steady-state Bestion	Steady-state Chexal-Lellouche	Unsteady simulations
REF	Complex	-	Valid	Less recommended	-
QRd	None	Outlet BC Not limiting	Valid	Less recommended	Not suitable
QRq	None	Co-current flows	Valid	Recommended	Not suitable
QG	Low Not limiting	None	Valid	Recommended	Recommended

TABLE 2

Comparison table of schemes characteristics for THYC-coeur applications.

6 Chapter 6

The ThermoTorch 1D code (see the article published during the PhD thesis [68]) is a one-dimensional finite volume code simulating a two-phase flow. It is a prototype that shares the same general characteristics with the industrial THYC-coeur code [7], but with simplified physics. Thus, the 4-equation model and the 3-equation model from THYC-coeur presented in **Chapter 1** are considered for ThermoTorch without second-order terms (diffusion, friction...) and with a simplified mixture equation of state. Moreover, a null relative velocity is considered here. This allows for a simple model that can be used in the next two chapters to verify the numerical schemes of ThermoTorch 1D for steady and unsteady solutions (see **Chapter 7**) or to evaluate the performance of a method to accelerate the search for a steady state using an initialization from a neural network (see **Chapter 8**).

Chapter 6 begins with the presentation of the models implemented in ThermoTorch: the 3-equation model and the 4-equation model. The model is manipulated to express the system with the following unknowns: pressure, mass flux $q = \rho u$, specific enthalpy $h = e + p/\rho$, and mass fraction y (for the 4-equation model only). Next, the spatial and temporal discretization is detailed. In particular, a staggered grid mesh is used with a uniform mesh size Δx . In ThermoTorch, the solution fields are initialized. Then, the evolution in time of the solution is computed with a time discretization based on a time step Δt . The time step Δt can be either constant or computed based on a CFL number derived from the speed of one of the system's waves. This method allows for the simulation of unsteady flows as well as reaching the steady state by advancing in time until the solution no longer evolves.

The finite volume numerical schemes used to discretize the continuous equation system are developed in the third part of the chapter. Semi-implicit schemes are proposed, allowing each equation to be written in terms of the time increment of the unknowns. A pseudo-entropy function $\mathfrak{s}$ is introduced to decouple the energy equation from the rest of the system. It corresponds to a mixture quantity derived from the entropies of each phase:

$$\mathfrak{s} = y s_g + (1 - y) s_l, \quad (17)$$

with s_g the gas entropy and s_l the liquid entropy. For the 4-equation model, this function is not the entropy of the system. It satisfies

$$T_l d\mathfrak{s} = dh - (v + y(T_l - T_s(p)) \frac{ds_g}{dp}) dp + \mathfrak{s}_y dy, \quad (18)$$

with

$$\mathfrak{s}_y = (h_l - \bar{h}_g) - T_l(s_l - \bar{s}_g) = \mu_l(p, T_l) - \bar{\mu}_g(p, T_s) + (T_s - T_l) \bar{s}_g. \quad (19)$$

The blue terms are neglected to fully decouple the energy equation. The function $\mathfrak{s}$ is only used in the unsteady part of the scheme. Thus, this approximation only affects **unsteady** solutions for the 4-equation model. The impact of this approximation is evaluated in **Chapter 7**. The matrix system of the 4-equation model, obtained by considering the time increments of unknowns $(p, \mathfrak{s}, q, y)$, can be written in the following form,

$$\begin{pmatrix} A_{\mathfrak{s},\mathfrak{s}} & 0 & 0 & 0 \\ 0 & A_{y,y} & 0 & 0 \\ A_{p,\mathfrak{s}} & A_{p,y} & A_{p,p} & A_{p,q} \\ 0 & 0 & A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta\mathcal{S} \\ \delta Y \\ \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_{\mathfrak{s}} \\ B_y \\ B_p \\ B_q \end{pmatrix}, \quad (20)$$

where $\delta\mathcal{S}$, δY , δP and δQ (with $\delta\phi = \phi^{n+1} - \phi^n$, $\phi \in \{S, Y, P, Q\}$) are the vector solutions corresponding to the temporal increments of function $\mathfrak{s}$, mass fraction y , pressure p and mass flux q on their respective meshes. For example, $\delta\mathcal{S} = (\mathfrak{s}^{n+1} - \mathfrak{s}^n)_{i \in [0, n_x]}$. This finite volume scheme allows solving a time step in several independent stages, decoupling the mass fraction and the function $\mathfrak{s}$ from the other variables. The vector $\delta\mathcal{S}$ is obtained using

$$\delta\mathcal{S} = A_{\mathfrak{s},\mathfrak{s}}^{-1} B_{\mathfrak{s}}, \quad (21)$$

and the vector δY using

$$\delta Y = A_{y,y}^{-1} B_y. \quad (22)$$

Then, the coupled momentum-pressure system can be written

$$\begin{pmatrix} A_{p,p} & A_{p,q} \\ A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_p - A_{p,\mathfrak{s}}\delta\mathcal{S} - A_{p,y}\delta Y \\ B_q \end{pmatrix}. \quad (23)$$

The matrix of this system is mostly sparse, so the Python representation `scipy.sparse.csc_matrix` is used. The vectors δP and δQ are obtained by inverting directly the sparse matrix, using the Python method `scipy.sparse.linalg.spsolve`. The 3-equation model can be solved in a similar way, without considering the equation for the mass fraction y . The final part of the chapter aims to present the code configurations used in the following chapters. The equations of state available in the code are also presented.

7 Chapter 7

In this chapter, the numerical schemes of ThermoTorch presented in **Chapter 6** are tested. The objective is to verify the implementation of the numerical schemes in ThermoTorch and to evaluate the convergence rate of these schemes in space and time. The configurations from **Chapter 3** are implemented, and the approximate numerical solutions obtained with ThermoTorch are compared with the analytical solutions described in **Chapter 3**. Throughout this chapter, no relative velocity is considered, as no analytical solution is known up to our knowledge for the model with relative velocity. Several types of solutions are studied in this chapter to verify different characteristics of the schemes.

- **Steady-state solutions of the 3-equation model:** These solutions make it possible to study the spatial convergence rate of the ThermoTorch scheme for regular 1D solutions of the 3-equation model. The configuration studied is a heated channel with conditions close to the one of a flow in a reactor core. A convergence rate of one in space is obtained.
- **Unsteady solutions of 1D Riemann problems for the 3-equation model:** These solutions are used to study the 3-equation model of ThermoTorch for unsteady configurations. The different 1D Riemann problems studied are: a symmetric double rarefaction wave case

(regular solution), a symmetric double shock wave case (solutions with shocks) and the Sod shock tube case [131]. Simulations are obtained with a constant CFL number (when considering fast waves):

$$\text{CFL}_{u+c} = 0.5. \quad (24)$$

This study shows that the ThermoTorch schemes for the 3-equation model have a convergence rate superior to $1/2$ (and which appears to approach one) in space and time for the unsteady regular solution considered (symmetric two-rarefaction wave) [52]. For solutions with shocks, a consistency error is introduced by the non-conservative schemes used here as expected [51, 76]. As a consequence, the pressure and density plateaus are not perfectly predicted for the symmetric two-shock wave case. It is shown that the magnitude of this consistency error is negligible compared to the error introduced by using an industrial mesh for conditions close to those of a reactor core flow.

- **Unsteady solutions of 1D regular Riemann problems for the 4-equation model:** These Riemann problems are used to evaluate the behavior of the ThermoTorch scheme with the 4-equation model in unsteady conditions. In particular, this study assesses the impact of using pseudo-entropy and the approximation made in its linearization (see Equation (18)) compared to using the system's actual entropy s . This is made possible by the equation of state proposed in **Chapter 3**, for which the mixture entropy is known.

The pseudo-entropy approximation introduces a consistency error in the energy equation for unsteady solutions. This error is highlighted with a convergence study. When pseudo-entropy is used, the error between the analytical and numerical solution reaches a plateau as the mesh is refined. On the contrary, with entropy, the ThermoTorch scheme shows a convergence rate of $1/2$ (expected due to the contact wave in the solution). The magnitude of the consistency error due to the approximation on pseudo-entropy is evaluated and compared to the error introduced by using an industrial mesh. The errors are estimated for various flow conditions encountered in nuclear reactor cores. It is observed that in most cases, the consistency error is negligible.

In ThermoTorch, the use of the pseudo-entropy $\mathfrak{s}$ allows the energy equation to be completely decoupled from the other equations. As a result, this yields a numerically robust model for steady-state computation with excellent run-time performance. In return, a consistency error is introduced in the transient regime, which becomes more significant as the configuration deviates from thermodynamic equilibrium. This confirms that it is negligible in most cases for nuclear reactor core unsteady applications.

8 Chapter 8

This chapter is an extended version of a conference article [92]. It proposes a Machine Learning-based method to accelerate convergence of a finite-volume code when looking for steady-state solutions. A Deep Neural Network is developed to predict the steady-state solutions. These predicted solutions are used to initialize the computation, aiming to reduce the number of external iterations (the desired effect is represented in Figure 7) compared to another initialization such as a constant field initialization. The development of this method has first appeared in the article [68] published during the PhD thesis.

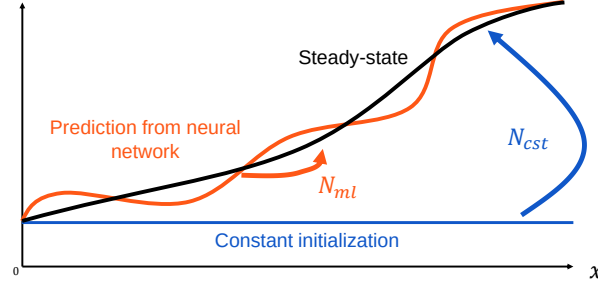


FIGURE 7

Illustration of the neural network methodology to accelerate the convergence toward the steady-state.

The performance of this method, i.e. the gain in number of iterations before convergence towards steady-state, is evaluated with the 1D prototype ThermoTorch, presented in **Chapter 6**. The heated channel configuration (from **Chapter 6**) is used, with four boundary conditions (inlet mass flux q_{in} and temperature T_{in} , outlet pressure p_{out} and uniform heat input ϕ_0). The detailed methodology for training and using the neural network is represented in Figure 8. Both 3-equation and 4-equation models of ThermoTorch are tested.

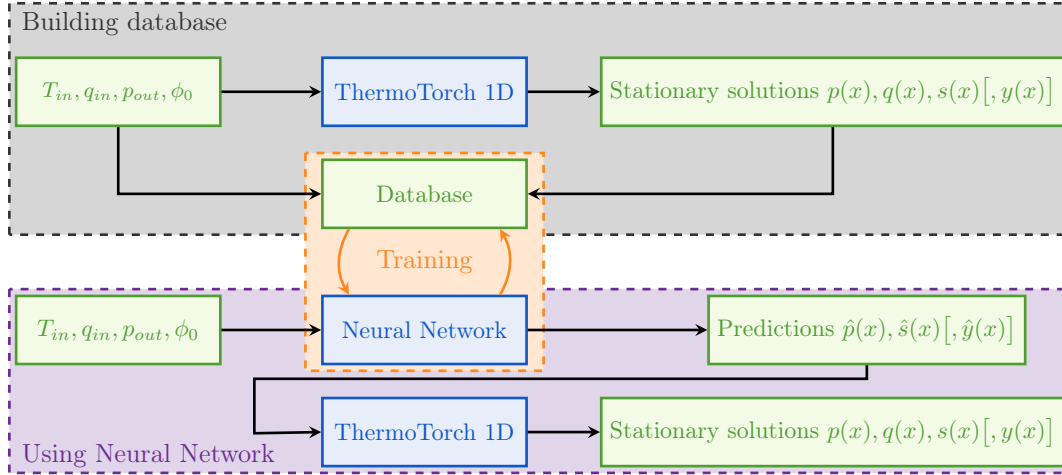


FIGURE 8

Construction of the database used for training and acceleration of ThermoTorch 1D with a neural network.

A first neural network has been developed with the mean square error between the prediction and the steady-state solution (for each field) as a loss function. After optimization of the hyperparameters of the network, good results were observed for a neural network with 2 hidden layers of 200 neurons per layer. A training data set of 10000 cases and a test data set of 1000 cases are used. The method accelerates all cases, regardless of the model. The average gain obtained is $70 \pm 19\%$ when focusing on the 3-equation model and $48 \pm 21\%$ with the 4-equation model.

The loss function is not entirely correlated with the observed gain. A study on the frequencies of the error between the prediction and the solution was carried out. It has been observed that the gain is not the same depending on the frequency of the error applied (with the same amplitude). In particular, the perturbations of low frequencies for the entropy have a huge impact on the performance to reach the steady-state.

In the second part of the chapter, a new loss function is proposed. It takes into account the first frequencies of the Discrete Fourier Transform (DFT) of the error in entropy. With this loss function, the gain is increased by more than 10%. It becomes $83 \pm 11\%$ for the 3-equation model and $61 \pm 18\%$ for the 4-equation model.

Communications and publications

Conference

- Journées scientifiques 2023 du GdR MaNu - 23-27 octobre 2023 – Le Croisic
Oral presentation of 30 minutes
Title: Equation d'advection-convection du titre massique d'une espèce : schémas volumes finis vérifiant le principe du maximum discret sans contrainte sur le pas de temps.
URL: <https://indico.math.cnrs.fr/event/9765/timetable/?print=1&view=standard>
- Sixth Workshop on Compressible Multiphases Flows - 17-19 juin 2024 - Strasbourg
Oral presentation of 30 minutes
Title: Hyperbolicité d'un modèle diphasique déséquilibré en vitesse.
Presentation of a poster
Title: Schémas Volumes Finis préservant le principe du min-max pour l'équation de déséquilibre d'un modèle diphasique homogène.
URL: <https://indico.math.cnrs.fr/event/11216/contributions/11818/>
- Seventh Workshop on Compressible Multiphases Flows - 26-28 mai 2025 - Strasbourg
Presentation of a poster
Title: Acceleration of the Convergence of a Core Thermal Hydraulic Code Using Initialization from a Neural Network.
URL: <https://indico.math.cnrs.fr/event/13690/overview>
- NURETH-21 (accepted) - 31 août – 5 septembre 2025
Conference paper and oral presentation of 30 minutes
Title: Acceleration of the Convergence of a Core Thermal Hydraulic Code Using Initialization from a Neural Network.
URL: <https://www.nureth-21.org/>

Publications

- Gauthier Lazare, Qingqing Feng, Philippe Helluy, Jean-Marc Hérard, Frank Hulsemann, Stéphane Pujet, *Maximum principle for the mass fraction in a system with two mass balance equations*, Comptes Rendus Mécanique de l'Académie des Sciences, Volume 352 (2024), pp. 81-98. doi: 10.5802/crmeca.244. <https://comptes-rendus.academie-sciences.fr/mecanique/articles/10.5802/crmeca.244/>
- Philippe Helluy, Gauthier Lazare, *Analysis and optimization of a liquid-vapor thermohydraulic model*, ESAIM Proceedings and Surveys (accepted), 2025. hal-04535413 <https://hal.science/hal-04535413>
- Jean-Marc Hérard, Gauthier Lazare. Two unsteady solutions of a homogeneous equilibrium model in a porous medium, ESAIM Proceedings and Surveys (accepted), 2025. hal-04733944 <https://hal.science/hal-04733944v1>

Part I

Two-phase flow models

Chapter 1

Two-phase flow models with a relative velocity between phases

The volume of fluid in a core is in the order of 10 m^3 , whereas the size of a steam bubble is in the order of 10^{-10} m^3 . It is therefore inconceivable to explicitly simulate the vapor and liquid phases of the flow. Simulation codes designed to simulate two-phase core flows are therefore based on time- and space-averaged equations, solved on mesh sizes of 10^{-2} to 10^{-6} m^3 . As it will be shown in this chapter, the 3 time- and space-averaged equations for each phase (mass, momentum and energy) can be rewritten as 3 equations for mixture and 3 equations for phase disequilibrium. The simplest model is to consider the liquid-vapor mixture as a homogeneous fluid with local kinematic and thermodynamic equilibrium between the phases. Taking into account the difference of velocity between the phases, linked to their density difference, and the thermodynamic disequilibrium, mainly generated by the heat flux from the fuel rod, improves the prediction of the thermal-hydraulic quantities used as inputs to critical flow predictors. The way in which disequilibrium are taken into account differs from one code to another. Three methods often considered are:

- 3-equation mixture models including closure laws modeling the resultant of phase disequilibrium (industrial reactor core codes such as THINC [21], VIPRE [132]).
- 4-equation models, which add a transport equation based on the thermodynamic disequilibrium between phases (industrial reactor core codes such as THYC [7], FLICA [136]).
- 6-equation models, known as two-fluid models, or even 9-equation models (industrial reactor core codes such as CTF [120]).

The aim of this chapter is to present the establishment of a 4-equation model from local instantaneous equations. The steps leading to the THYC-coeur model will be detailed, and the assumptions made will be explained. An alternative 4-equation model will also be proposed, which has the advantage of being conservative. The models will be used in several chapters of this thesis.

1.1 Governing equations for each phase

The objective is to model water-steam flows (or more generally, two-phase flows) in a nuclear reactor core, a large-scale domain that is particularly cluttered with complex solid structures with a large range of sizes. To achieve this, a two-phase Eulerian model is considered. This model is first averaged in time to obtain phase equations where phase quantities are defined (non zero) at any instant. A porous approach is then considered through spatial integration in order to bypass the complex detailed geometry of the solid medium (see Introduction - Section 3.2). This porous approach allows for a fast robust model at the component scale but additional terms due to averaging

must be modeled by physical closure laws, which accounts for the geometry and physical conditions considered.

The model must simulate a high-pressure, high-flow regime with phase change and account for the relative velocity between phases. The phenomenon of subcooled boiling (see Introduction - Section 2.1) must also be included in the model, as this phenomenon has a significant impact on the Departure from Nucleate Boiling Ratio (DNBR) (see Introduction - Sections 2.1 and 2.2). These considerations lead to the use of either a 3-equation model with analytical closure laws for subcooled boiling, either a 4-equation drift-flux mixture model. In [2], a drift-flux model with relative velocity between phases is considered. This model does not take into account the energy equation. The 4-equation model proposed in this work is an extension of this drift-flux model.

First, the local instant equations are written for each phase before averaging them in time and space. The method and the hypothesis for obtaining a mixture model are taken from [82] and are recalled here. Then, the two different sets of equations used in this work are discussed: the **total energy 4-equation model**, for mathematical analysis and then the **enthalpy 4-equation model**, model used in the industrial code THYC-coeur [7]. Finally, some of the closure laws required are discussed, especially the relative velocity between the phases.

1.1.1 Local instant formulation

A two-phase flow in which two phases are mixed is considered: a liquid phase (denoted by the subscript l) and a gas phase (denoted by the subscript g). Each phase k is described by the following local quantities: its density ρ_k , its velocity $\mathbf{u}_k$, its pressure p_k , and its specific internal energy e_k . Furthermore, each phase is governed by an Equation of State (noted EoS) such that

$$e_k = e_k(p_k, \rho_k). \quad (1.1)$$

The specific total energy E_k can be defined by

$$E_k = e_k + \frac{|\mathbf{u}_k|^2}{2}, \quad (1.2)$$

and the specific enthalpy h_k by

$$h_k = e_k + \frac{p_k}{\rho_k}. \quad (1.3)$$

For each phase, a specific entropy function $s_k(p_k, \rho_k)$ can be obtained with

$$c_k^2(p_k, \rho_k) \left(\frac{\partial s_k}{\partial p_k} \right)_{\rho_k} + \left(\frac{\partial s_k}{\partial \rho_k} \right)_{p_k} = 0, \quad (1.4)$$

where the speed of sound $c_k(p_k, \rho_k)$ is defined by

$$(\rho_k c_k)^2 = \left(\frac{\partial e_k}{\partial p_k} \right)_{\rho_k}^{-1} \left(p_k - \rho_k^2 \left(\frac{\partial e_k}{\partial \rho_k} \right)_{p_k} \right). \quad (1.5)$$

The temperature T_k of phase k is defined by the relation

$$T_k = \left(\frac{\partial e_k}{\partial s_k} \right)_{\rho_k}. \quad (1.6)$$

It gives the following differential

$$T_k ds_k = de_k + p_k dv_k, \quad (1.7)$$

with the specific volume v_k of phase k defined by

$$v_k = \frac{1}{\rho_k}. \quad (1.8)$$

Each phase locally satisfies three conservation laws: the mass balance (1.9), the momentum balance (1.10) and the energy balance (1.11). The local instant mass balance for phase k writes

$$\frac{\partial \rho_k}{\partial t} + \nabla \cdot (\rho_k \mathbf{u}_k) = 0. \quad (1.9)$$

The momentum balance writes

$$\frac{\partial}{\partial t} (\rho_k \mathbf{u}_k) + \nabla \cdot (\rho_k \mathbf{u}_k \otimes \mathbf{u}_k) + \nabla \cdot (p_k \mathbf{I}_d - \mathbf{T}_k) = \rho_k \mathbf{g}, \quad (1.10)$$

with $\mathbf{T}_k$ the viscous stress and $\mathbf{g}$ the gravity constant. The energy balance is

$$\frac{\partial}{\partial t} (\rho_k E_k) + \nabla \cdot (\rho_k E_k \mathbf{u}_k) + \nabla \cdot ((p_k \mathbf{I}_d - \mathbf{T}_k) \cdot \mathbf{u}_k) = -\nabla \cdot \boldsymbol{\varphi}_k + \rho_k \mathbf{g} \cdot \mathbf{u}_k, \quad (1.11)$$

with $\boldsymbol{\varphi}_k$ the heat flux and $\mathbf{I}_d$ the identity matrix.

1.1.2 Averaged equations

Time averaging

Equations (1.9), (1.10) and (1.11) are instantaneous local formulations. The quantities of each phase are therefore not continuous, as phase k is not present at all times. They are averaged in time to extend their definition to every instant. At an instant t_0 and a position $\mathbf{x}_0$, a time interval Δt is considered centered on instant t_0 such that it is large enough to smooth out the local variations of properties but small enough compared to the macroscopic time constant representing the unsteadiness of the bulk flow. The details of the computations are provided in **Appendix 1.A.1**. Several definitions are recalled in this section. The time interval during which phase k is present is noted Δt_k . The local void fraction (or local time fraction) can be defined by

$$\alpha_k(\mathbf{x}_0, t_0) = \frac{\Delta t_k}{\Delta t} \in [0, 1]. \quad (1.12)$$

It corresponds to the probability of finding phase k at a given time and location. Considering interfaces between phases of null thickness, the immiscibility condition writes

$$\alpha_l + \alpha_g = 1. \quad (1.13)$$

The quantity $\bar{f}_k$ is the mean value of a function f_k over the interval Δt such that

$$\bar{f}_k(\mathbf{x}_0, t_0) = \frac{1}{\Delta t} \int_{t \in [t_0 - \Delta t/2, t_0 + \Delta t/2]} f_k(\mathbf{x}_0, t) dt. \quad (1.14)$$

This operation is applied to the local instant equations. A more physical average is the mean of the function f_k on the time interval Δt_k where f_k is not null. It is noted $\bar{\bar{f}}_k$ such that

$$\bar{\bar{f}}_k(\mathbf{x}_0, t_0) = \frac{1}{\Delta t_k} \int_{t \in [\Delta t_k]} f_k(\mathbf{x}_0, t) dt = \frac{\bar{f}_k(\mathbf{x}_0, t_0)}{\alpha_k}, \quad (1.15)$$

where $[\Delta t_k]$ is the union of all intervals where phase k is present during $[t_0 - \frac{\Delta t}{2}, t_0 + \frac{\Delta t}{2}]$. The density ρ_k is defined (strictly positive) only at times when phase k is present and zero otherwise.

The mean value weighted by the density is used for extensive variable ψ_k . It is noted $\widehat{\psi}_k$ and defined by

$$\widehat{\psi}_k = \frac{\overline{\overline{\rho_k \psi_k}}}{\overline{\overline{\rho_k}}}. \quad (1.16)$$

Time averaging introduces source terms due to interface between phases (see **Appendix 1.A.1**). Using these definitions and Equation (1.9), the time averaged mass balance for phase k writes

$$\frac{\partial}{\partial t} (\alpha_k \overline{\overline{\rho_k}}) + \nabla \cdot (\alpha_k \overline{\overline{\rho_k}} \widehat{\mathbf{u}}_k) = \Gamma_k, \quad (1.17)$$

where Γ_k is the mass transfer from phase k to the other, which satisfies

$$\sum_k \Gamma_k = 0. \quad (1.18)$$

From Equation (1.10), the time averaged of momentum balance for phase k writes

$$\frac{\partial}{\partial t} (\alpha_k \overline{\overline{\rho_k}} \widehat{\mathbf{u}}_k) + \nabla \cdot (\alpha_k \overline{\overline{\rho_k}} \widehat{\mathbf{u}}_k \otimes \widehat{\mathbf{u}}_k) + \nabla (\alpha_k \overline{\overline{p_k}}) = \nabla \cdot (\alpha_k (\overline{\overline{\mathbf{T}}_k} + \mathbf{T}_k^t)) + \alpha_k \overline{\overline{\rho_k}} \widehat{\mathbf{g}} + \mathbf{m}_k, \quad (1.19)$$

where $\mathbf{T}_k^t$ is the viscous stress due to turbulent effects defined in **Appendix 1.A.1** (see Equation (1.139)) and $\mathbf{m}_k$ the transfer of momentum from phase k to the other, which satisfies, with σ the surface tension (depending on several quantities) between phases,

$$\sum_k \mathbf{m}_k = \mathbf{m}_m(\sigma). \quad (1.20)$$

The term $\mathbf{m}_m(\sigma)$ is detailed in [83]. It is null when the surface tension is neglected ($\sigma \approx 0$). The turbulent fluctuations of kinetic energy are neglected in the energy equation (see **Appendix 1.A.1**). Using Equation (1.11), the time averaged total energy balance is

$$\begin{aligned} \frac{\partial}{\partial t} \left(\alpha_k \overline{\overline{\rho_k}} (\widehat{e}_k + \frac{|\widehat{\mathbf{u}}_k|^2}{2}) \right) + \nabla \cdot \left(\alpha_k \overline{\overline{\rho_k}} (\widehat{e}_k + \frac{|\widehat{\mathbf{u}}_k|^2}{2}) \widehat{\mathbf{u}}_k \right) + \nabla \cdot (\alpha_k \overline{\overline{p_k}} \mathbf{u}_k) = \\ - \nabla \cdot (\alpha_k (\overline{\overline{\varphi}}_k + \varphi_k^t)) \\ + \nabla \cdot (\alpha_k (\overline{\overline{\mathbf{T}}_k} + \mathbf{T}_k^t) \cdot \widehat{\mathbf{u}}_k) \\ + \alpha_k \overline{\overline{\rho_k}} \widehat{\mathbf{g}}_k \cdot \widehat{\mathbf{u}}_k + \Xi_k, \end{aligned} \quad (1.21)$$

with φ_k^t the heat flux due to turbulent effects defined in Equation (1.141) and Ξ_k the transfer of total energy from phase k to the other, which satisfies

$$\sum_k \Xi_k = \Xi_m(\sigma). \quad (1.22)$$

The term $\Xi_m(\sigma)$ is defined in [83] and is null when $\sigma = 0$. The functions $\mathbf{m}_m$ and Ξ_m depend exclusively on the surface tension σ between phases. In this work, the effect of the surface tension is neglected so that $\mathbf{m}_m = \mathbf{0}$ and $\Xi_m = 0$.

Space averaging

The equations are also spatially averaged. This allows the integration of solid obstacles into the equations. This porous approach models flows in a complex geometry (fuel rods, mixing grids, supporting grids, etc.) without considering the exact geometry of the solid. The method for spatial integration in a cluttered environment is derived from methods developed initially for natural porous media [116], especially for Stokes flow using Darcy's Law [140]. The variables are averaged over a volume V_0 centered on a position $\mathbf{x}_0$. This volume contains a volume V_f of fluid and a volume V_s of solid (see Figure 1.2), such that

$$V_0 = V_f + V_s. \quad (1.23)$$

The fluid porosity can be defined as

$$\varepsilon = \frac{V_f}{V_0}. \quad (1.24)$$

The space-averaging process is similar to the time-averaging one, but porosity $\varepsilon(x)$ is independent of time ($\partial_t \varepsilon = 0$) and provided as input data, unlike the local void fraction $\alpha_k(x, t)$ which is an unknown depending on time and space. Spatial integration introduces source terms in the equations due to the presence of solid in the flow. No terms appear in the mass conservation equation since the solid does not produce matter. In the momentum equation, solid friction introduces a pressure drop on phase k noted $\mathbf{m}_{ks}$. In the energy equation, energy transfer from the solid to phase k is considered through the term Ξ_{ks} . Similarly to time-averaging, several space averages are defined. For a quantity f_k , the mean value on volume V_0 is noted $\langle f_k \rangle$ such that

$$\langle f_k \rangle (\mathbf{x}_0, t) = \frac{1}{V_0} \int_{V_0} f_k(\mathbf{x}, t) d\mathbf{x}. \quad (1.25)$$

The mean value of quantity f_k on the volume of fluid V_f is noted $\langle\langle f_k \rangle\rangle$ and defined by

$$\langle\langle f_k \rangle\rangle (\mathbf{x}_0, t) = \frac{1}{V_f} \int_{V_f} f_k(\mathbf{x}, t) d\mathbf{x} = \frac{\langle f_k \rangle (\mathbf{x}_0, t)}{\varepsilon}. \quad (1.26)$$

As a consequence, the local void fraction averaged in space writes:

$$\langle\langle \alpha_k \rangle\rangle (\mathbf{x}_0, t_0) = \frac{V_k}{V_f}, \quad (1.27)$$

with V_k the volume occupied by the phase k at instant t_0 (in volume V_0). $\langle\langle \alpha_k \rangle\rangle$ is simply called void fraction and still satisfies the immiscibility condition. A mean value weighted by the local void fraction is defined for an intensive variable f_k (density, pressure, friction...) such that

$$\tilde{f}_k = \frac{\langle\langle \alpha_k \bar{f}_k \rangle\rangle}{\langle\langle \alpha_k \rangle\rangle}. \quad (1.28)$$

Finally, a mean value weighted by $\alpha_k \bar{\rho}_k$ is defined for the extensive variable ψ_k (velocity, energy...) such that

$$\widetilde{\widetilde{\psi}}_k = \frac{\langle\langle \alpha_k \bar{\rho}_k \hat{\psi}_k \rangle\rangle}{\langle\langle \alpha_k \bar{\rho}_k \rangle\rangle} = \frac{\langle\langle \alpha_k \bar{\rho}_k \hat{\psi}_k \rangle\rangle}{\langle\langle \alpha_k \rangle\rangle \tilde{\rho}_k}. \quad (1.29)$$

The computation to obtain the space-averaged equations from the time-averaged equations is detailed in **Appendix 1.A.2**. Spatial turbulent terms appear when averaging the terms of the form

$\alpha_k \bar{\rho}_k \hat{\psi}_k \hat{\mathbf{u}}_k$ in the momentum and energy balances. They are noted with the superscript st to distinguish it from the turbulent flux due to time-averaging (noted with superscript t). Averaging Equation (1.17), the space-averaged mass balance for phase k writes

$$\frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{\mathbf{u}}_k) = \varepsilon \langle \Gamma_k \rangle. \quad (1.30)$$

Using Equation (1.19), the space-averaged momentum balance for phase k writes

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{\mathbf{u}}_k) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{\mathbf{u}}_k \otimes \widetilde{\mathbf{u}}_k) + \varepsilon \langle \alpha_k \rangle \nabla \tilde{p}_k = \\ \nabla \cdot (\varepsilon \langle \alpha_k \rangle (\widetilde{\mathbf{T}}_k + \widetilde{\mathbf{T}}_k^t + \mathbf{T}_k^{st})) \\ + \varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \mathbf{g} \\ + \varepsilon \langle \mathbf{m}_k \rangle + \mathbf{m}_{ks}, \end{aligned} \quad (1.31)$$

where $\mathbf{m}_{ks}$ is the friction on phase k due to the solid and $\mathbf{T}_k^{st}$ the turbulent friction due to space averaging. As for the time average, the spatial fluctuations of kinetic energy have been neglected. Using Equation (1.21), the space-averaged total energy balance for phase k writes

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{E}_k) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{E}_k \widetilde{\mathbf{u}}_k) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \tilde{p}_k \widetilde{\mathbf{u}}_k) = \\ - \nabla \cdot (\varepsilon \langle \alpha_k \rangle (\widetilde{\varphi}_k + \widetilde{\varphi}_k^t + \varphi_k^{st})) \\ + \nabla \cdot (\varepsilon \langle \alpha_k \rangle (\widetilde{\mathbf{T}}_k + \widetilde{\mathbf{T}}_k^t + \mathbf{T}_k^{st}) \cdot \widetilde{\mathbf{u}}_k) \\ + \varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \mathbf{g} \cdot \widetilde{\mathbf{u}}_k + \varepsilon \langle \Xi_k \rangle + \Xi_{ks}, \end{aligned} \quad (1.32)$$

where $\widetilde{E}_k = \widetilde{e}_k + \frac{|\widetilde{\mathbf{u}}_k|^2}{2}$ is the total energy for phase k , Ξ_{ks} is the energy received by phase k from the solid and φ_k^{st} the turbulent heat flux due to space averaging. Manipulating these equations (see **Appendix 1.A.2** for detailed computation), the enthalpy equation can be written

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{h}_k) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \tilde{\rho}_k \widetilde{h}_k \widetilde{\mathbf{u}}_k) = \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \tilde{p}_k) + \varepsilon \langle \alpha_k \rangle \widetilde{\mathbf{u}}_k \cdot \nabla \tilde{p}_k \\ - \nabla \cdot (\varepsilon \langle \alpha_k \rangle (\widetilde{\varphi}_k + \widetilde{\varphi}_k^t + \varphi_k^{st})) \\ + \varepsilon \langle \alpha_k \rangle (\widetilde{\mathbf{T}}_k + \widetilde{\mathbf{T}}_k^t + \mathbf{T}_k^{st}) : \nabla \widetilde{\mathbf{u}}_k \\ + \varepsilon \left(\langle \Xi_k \rangle - \langle \mathbf{m}_k \rangle \cdot \widetilde{\mathbf{u}}_k + \langle \Gamma_k \rangle \frac{\widetilde{u}_k^2}{2} \right) \\ + \Xi_{ks} - \mathbf{m}_{ks} \cdot \widetilde{\mathbf{u}}_k. \end{aligned} \quad (1.33)$$

The heat given by the solid to the phase k is noted $\phi_{ks} = \Xi_{ks} - \mathbf{m}_{ks} \cdot \widetilde{\mathbf{u}}_k$. The transfer of thermal energy between phases is written Λ_k and defined by

$$\Lambda_k = \langle \Xi_k \rangle + \langle \mathbf{m}_k \rangle \cdot \widetilde{\mathbf{u}}_k - \langle \Gamma_k \rangle \frac{|\widetilde{\mathbf{u}}_k|^2}{2}. \quad (1.34)$$

Remark: It is important to point out here that, despite neglecting surface tension, the sum of Λ_k terms is not, a priori, equal to zero because the phases have not the same velocity $\widetilde{\mathbf{u}}_g \neq \widetilde{\mathbf{u}}_l$, and thus

$$\sum_k \Lambda_k = \langle \Gamma_g \rangle \left(\frac{|\widetilde{\mathbf{u}}_g|^2}{2} - \frac{|\widetilde{\mathbf{u}}_l|^2}{2} \right) - \langle \mathbf{m}_g \rangle \cdot (\widetilde{\mathbf{u}}_g - \widetilde{\mathbf{u}}_l) \neq 0. \quad (1.35)$$

For the application concerned, the heat provided by the solid is large compared to the dissipation due to friction (including also the space and turbulent friction due to averaging) and to the interfacial transfer of thermal energy between phase. Hence, neglecting these terms in Equation (1.33), the equation for thermal energy becomes

$$\begin{aligned} \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k h_k} \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k h_k} \widetilde{\mathbf{u}_k} \right) &= \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{p_k} \right) + \varepsilon \langle \alpha_k \rangle \widetilde{\mathbf{u}_k} \cdot \nabla \widetilde{p_k} \\ &\quad - \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\boldsymbol{\varphi}_k} + \widetilde{\boldsymbol{\varphi}_k^t} + \boldsymbol{\varphi}_k^{st} \right) \right) \\ &\quad + \varepsilon \Lambda_k + \phi_{ks}, \end{aligned} \quad (1.36)$$

with

$$\sum_k \Lambda_k = 0. \quad (1.37)$$

1.2 Mixture model

Considering the 3 space- and time-averaged Equations (1.30), (1.31) and (1.32) (or Equation (1.36)) for each phase k , a 6-equation model is obtained. This model approaches a Baer-Nunziato type model [9] with porous medium, for which a 7th equation on the void rate should be considered.

Considering that the two-phase flow behaves similarly to a homogeneous mixture, the objective here is to simplify the model (to a 4-equation model) in order to obtain a robust physical model that is not costly in CPU time. First, mixture quantities are defined based on the weighted-sum of the variables of each phase. The equations are summed for each phase to obtain three mixture equations. Two disequilibrium equations are also derived for momentum and total energy by subtracting the gas phase equations from the liquid ones. Then, assumptions detailed later allow to free ourselves of these disequilibrium equations. The last disequilibrium equation which is the mass balance for the gaseous phase is retained. The methodology is a 3D extension of [72, 81] where the space was averaged on a section to obtain the one-dimensional drift-flux model.

Two different sets of equations can be obtained depending on the energy equation considered (total energy or enthalpy). After defining the mixture and disequilibrium quantities, the two different 4-equation models are discussed according to the energy equation considered. In order to simplify the notations, the averaging operators (for time and space) are abandoned. So, from now on, the time- and space-averaged quantities are the density ρ_k , the velocity $\mathbf{u}_k$, the pressure p_k , the enthalpy h_k , the total energy E_k , the friction terms $\mathbf{T}_k + \mathbf{T}_k^t + \mathbf{T}_k^{st}$, the heat fluxes $\boldsymbol{\varphi} + \boldsymbol{\varphi}^t + \boldsymbol{\varphi}^{st}$, the interfacial mass transfer Γ_k , the interfacial momentum transfer $\mathbf{m}_k$ and the interfacial energy transfer Ξ_k . The void fraction for phase k averaged in space is noted α_k . Using these notations, the mass balance writes

$$\frac{\partial}{\partial t} (\varepsilon \alpha_k \rho_k) + \nabla \cdot (\varepsilon \alpha_k \rho_k \mathbf{u}_k) = \varepsilon \Gamma_k, \quad (1.38)$$

the momentum balance writes

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \alpha_k \rho_k \mathbf{u}_k) + \nabla \cdot (\varepsilon \alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) + \varepsilon \alpha_k \nabla p_k &= \\ \nabla \cdot \left(\varepsilon \alpha_k (\mathbf{T}_k + \mathbf{T}_k^t + \mathbf{T}_k^{st}) \right) & \\ + \varepsilon \alpha_k \rho_k \mathbf{g} & \\ + \varepsilon \mathbf{m}_k + \mathbf{m}_{ks}, & \end{aligned} \quad (1.39)$$

and the total energy balance writes

$$\begin{aligned}
 \frac{\partial}{\partial t} (\varepsilon \alpha_k \rho_k E_k) + \nabla \cdot (\varepsilon \alpha_k \rho_k E_k \mathbf{u}_k) + \nabla \cdot (\varepsilon \alpha_k p_k \mathbf{u}_k) = \\
 - \nabla \cdot (\varepsilon \alpha_k (\varphi_k + \varphi_k^t + \varphi_k^{st})) \\
 + \nabla \cdot (\varepsilon \alpha_k (\mathbf{T}_k + \mathbf{T}_k^t + \mathbf{T}_k^{st}) \cdot \mathbf{u}_k) \\
 + \varepsilon \alpha_k \rho_k \mathbf{g} \cdot \mathbf{u}_k + \varepsilon \Xi_k + \Xi_{ks}.
 \end{aligned} \tag{1.40}$$

If the enthalpy equation is considered instead of the total energy (these two different equations are not equivalent due to different approximations), it writes

$$\begin{aligned}
 \frac{\partial}{\partial t} (\varepsilon \alpha_k \rho_k h_k) + \nabla \cdot (\varepsilon \alpha_k \rho_k h_k \mathbf{u}_k) = \frac{\partial}{\partial t} (\varepsilon \alpha_k p_k) + \varepsilon \alpha_k \mathbf{u}_k \cdot \nabla p_k \\
 - \nabla \cdot (\varepsilon \alpha_k (\varphi_k + \varphi_k^t + \varphi_k^{st})) \\
 + \varepsilon \Lambda_k + \phi_{ks},
 \end{aligned} \tag{1.41}$$

with $\sum_k \Lambda_k = 0$.

1.2.1 Mixture and disequilibrium quantities

The mass fraction of phase k is defined and noted y_k . It is obtained from the void fraction α_k using

$$y_k = \frac{\alpha_k \rho_k}{\rho_m} \in [0, 1], \tag{1.42}$$

with ρ_m the mixture density defined by $\rho_m = \alpha_g \rho_g + \alpha_l \rho_l$. It gives

$$y_g + y_l = 1. \tag{1.43}$$

The following notations are used

$$\begin{cases} \alpha_g = \alpha & , & \alpha_l = 1 - \alpha, \\ y_g = y & , & y_l = 1 - y. \end{cases} \tag{1.44}$$

To obtain mixture quantities, variables based on unit mass are weighted by the mass fraction y_k , whereas the ones based on unit volume or surface are weighted by void fraction α_k . It can be summarized by

$$\begin{cases} f_m = \sum_k \alpha_k f_k \text{ for } f = \rho, p, \mathbf{T}, \mathbf{T}^t, \mathbf{T}^{st}, \varphi, \varphi^t, \varphi^{st}, \\ \psi_m = \sum_k y_k \psi_k \text{ for } \psi = \mathbf{u}, e, h, E. \end{cases} \tag{1.45}$$

Mixture momentum and energy source terms due to the solid are obtained by summing the contribution of each phase such that

$$\begin{cases} \mathbf{m}_s = \sum_k \mathbf{m}_{ks}, \\ \Xi_s = \sum_k \Xi_{ks}, \\ \phi_s = \sum_k \phi_{ks}. \end{cases} \tag{1.46}$$

Disequilibrium quantities are introduced, such as the relative velocity between phases $\mathbf{u}_r$ defined by

$$\mathbf{u}_r = \mathbf{u}_g - \mathbf{u}_l, \quad (1.47)$$

and the relative specific enthalpy L , defined by

$$L = h_g - h_l, \quad (1.48)$$

where the enthalpy of each phase are defined by (1.3). The relative specific enthalpy L does not necessarily correspond to the classical latent heat because the two phases are not always saturated in the model.

1.2.2 Mixture equations

To obtain the mixture equations, surface tension is neglected, so that only one pressure is considered for both phases

$$p_m = p_l = p_g, \quad (1.49)$$

As explained above, this hypothesis also enables to neglect the mixture source terms $\mathbf{m}_m(\sigma)$ and $\Xi_m(\sigma)$. Considering an intensive variable f based on unit volume, the following identity is used

$$\sum_k \alpha_k f_k \mathbf{u}_k = f_m \mathbf{u}_m + \rho_m y(1 - y) \left(\frac{f_g}{\rho_g} - \frac{f_l}{\rho_l} \right) \mathbf{u}_r. \quad (1.50)$$

For a variable ψ based on unit mass, a similar identity is

$$\sum_k \alpha_k \rho_k \psi_k \mathbf{u}_k = \rho_m \psi_m \mathbf{u}_m + \rho_m y(1 - y) (\psi_g - \psi_l) \mathbf{u}_r, \quad (1.51)$$

These equations use the mixture quantities for variables based on unit volume (density, friction tensors, heat fluxes) or unit mass (velocity, internal energy, enthalpy, total energy) defined in Equations (1.45).

Summing Equation (1.38) for each phase k , mixture mass balance is obtained

$$\frac{\partial}{\partial t} (\varepsilon \rho_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m) = 0. \quad (1.52)$$

Summing Equation (1.39) for each phase k , mixture momentum balance writes

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \rho_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m \otimes \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1 - y) \mathbf{u}_r \otimes \mathbf{u}_r) + \varepsilon \nabla p_m = \\ \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st})) \\ + \varepsilon \rho_m \mathbf{g} + \mathbf{m}_s. \end{aligned} \quad (1.53)$$

Summing Equation (1.40) for each phase k , a first equation for energy is

$$\begin{aligned}
 & \frac{\partial}{\partial t} (\varepsilon \rho_m E_m) + \nabla \cdot (\varepsilon \rho_m E_m \mathbf{u}_m) + \nabla \cdot \left(\varepsilon \rho_m y (1 - y) (E_g - E_l) \mathbf{u}_r \right) \\
 & + \nabla \cdot \left(\varepsilon p_m \left[\mathbf{u}_m + \rho_m y (1 - y) \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \right) = \\
 & - \nabla \cdot \left(\varepsilon (\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st}) \right) \\
 & + \nabla \cdot \left(\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st}) \cdot \mathbf{u}_m \right) \\
 & + \nabla \cdot \left(\varepsilon \rho_m y (1 - y) \left(\frac{\mathbf{T}_g + \mathbf{T}_g^t + \mathbf{T}_g^{st}}{\rho_g} - \frac{\mathbf{T}_l + \mathbf{T}_l^t + \mathbf{T}_l^{st}}{\rho_l} \right) \cdot \mathbf{u}_r \right) \\
 & + \varepsilon \rho_m \mathbf{u}_m \cdot \mathbf{g} \\
 & + \Xi_s,
 \end{aligned} \tag{1.54}$$

where the mixture total energy E_m is defined by

$$E_m = \sum_k y_k E_k = e_m + \sum_k y_k \frac{|\mathbf{u}_k|^2}{2}. \tag{1.55}$$

When the relative velocity is not null, the mixture total energy can not be defined as usual with the mixture internal energy and the mixture velocity because

$$E_m = e_m + \frac{|\mathbf{u}_m|^2}{2} + y(1 - y) \frac{|\mathbf{u}_r|^2}{2} \neq e_m + \frac{|\mathbf{u}_m|^2}{2}. \tag{1.56}$$

If Equation (1.41) on enthalpy is considered for each phase and if several assumptions are made (neglecting friction contribution to dissipation and the transfer of mechanical energy between phases), the mixture enthalpy equation writes (see **Appendix 1.A.2** for details)

$$\begin{aligned}
 & \frac{\partial}{\partial t} (\varepsilon \rho_m h_m) + \nabla \cdot (\varepsilon \rho_m h_m \mathbf{u}_m) + \nabla \cdot \left(\varepsilon \rho_m y (1 - y) L \mathbf{u}_r \right) = \\
 & \varepsilon \frac{\partial p_m}{\partial t} + \varepsilon \left[\mathbf{u}_m + y(1 - y) \rho_m \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \cdot \nabla p_m \\
 & - \nabla \cdot \left(\varepsilon (\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st}) \right) + \phi_s,
 \end{aligned} \tag{1.57}$$

where the mixture enthalpy is defined by

$$h_m = \sum_k y_k h_k = \sum_k y_k \left(e_k + \frac{p_m}{\rho_k} \right) = e_m + \frac{p_m}{\rho_m}. \tag{1.58}$$

This equation is the one used in the THYC-coeur code, and is not strictly equivalent to the total energy equation, as some terms have been neglected in between.

1.2.3 Assumptions to reduce the number of equations

To eliminate the disequilibrium equation for the relative specific enthalpy L , it is assumed that the dispersed phase (most often the gas phase) is saturated. In this case, the scalar L is no longer an unknown and can be expressed from other unknowns (pressure, mixture enthalpy and mass fraction). This makes it possible to eliminate the energy disequilibrium equation, reducing the model from six equations to five.

The relative velocity is approximated by a closure law that accounts for the main physical phenomena causing a velocity difference between the phases. The correlations used for the relative velocity are derived from the drift-flux model [145] and detailed in Section 1.4.2. Thanks to this closure law, the complex equation for the relative velocity involving source terms to be modeled is not considered. This assumptions allow to reduce the number of equation from five to four.

1.2.4 Disequilibrium equation

The only disequilibrium equation considered is the equation on the vapor mass fraction y . This equation could be eliminated by considering the thermal equilibrium, which means that both phases are at saturation, i.e. when both phases are at the same temperature: the saturation temperature noted T_{sat} and defined by the equilibrium of chemical potentials at pressure p_m (see Section 1.3.3 for details on saturation). The mass fraction at equilibrium writes

$$\bar{y}(h_m, p_m) = \frac{h_m - \bar{h}_l(p_m)}{\bar{h}_g(p_m) - \bar{h}_l(p_m)}, \quad (1.59)$$

where $\bar{h}_g(p_m)$ (resp. $\bar{h}_l(p_m)$) is the specific enthalpy of the gas phase (resp. liquid phase) at saturation (see Equation (1.66) for quantities at saturation). This thermal equilibrium leads to a 3-equation model (see **Appendix 1.B**). However, within the framework of THYC-coeur code, the phenomenon of subcooled boiling plays a significant role in predicting the Departure from Nucleate Boiling Ratio (DNBR). To account for this, the equation for the vapor mass fraction is retained. It is expressed as

$$\frac{\partial}{\partial t}(\varepsilon \rho_m y) + \nabla \cdot (\varepsilon \rho_m y \mathbf{u}_m + \varepsilon \rho_m y (1 - y) \mathbf{u}_r) = \varepsilon \Gamma_g, \quad (1.60)$$

where the source term Γ_g corresponds to the mass transfer from the gaseous phase to the liquid one. It should take into account several physical phenomena (subcooled boiling, evaporation at saturation, recondensation...). It will be discussed in Section 1.4.1. The disequilibrium equation is also called the vapor mass fraction balance or the fourth equation.

1.3 4-equation models

The two energy Equations (1.54) and (1.57) proposed here are not equivalent because additional assumptions have been made to obtain the equation on enthalpy. As a consequence, two different systems can be defined: the total energy 4-equation model (using Equation (1.54)) and the enthalpy 4-equation model (using Equation (1.57)). They are described below with the assumptions detailed for each model. In other chapters, the subscript m used for mixture quantities is sometimes dropped to improve readability.

1.3.1 Total energy 4-equation model

The total energy 4-equation model is

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t}(\varepsilon \rho_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m) = 0, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m \otimes \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) \mathbf{u}_r \otimes \mathbf{u}_r) + \varepsilon \nabla p_m = \\ \quad \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st})) \\ \quad + \varepsilon \rho_m \mathbf{g} + \mathbf{m}_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m E_m) + \nabla \cdot (\varepsilon \rho_m E_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y)(E_g - E_l) \mathbf{u}_r) \\ \quad + \nabla \cdot \left(\varepsilon \left(p_m \left[\mathbf{u}_m + \rho_m y(1-y) \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \right) \right) = \\ \quad - \nabla \cdot (\varepsilon (\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st})) \\ \quad + \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st}) \cdot \mathbf{u}_m) \\ \quad + \nabla \cdot \left(\varepsilon \rho_m y(1-y) \left(\frac{\mathbf{T}_g + \mathbf{T}_g^t + \mathbf{T}_g^{st}}{\rho_g} - \frac{\mathbf{T}_l + \mathbf{T}_l^t + \mathbf{T}_l^{st}}{\rho_l} \right) \cdot \mathbf{u}_r \right) \\ \quad + \varepsilon \rho_m \mathbf{u}_m \cdot \mathbf{g} + \Xi_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m y) + \nabla \cdot (\varepsilon \rho_m y \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) \mathbf{u}_r) = \varepsilon \Gamma_g. \end{array} \right. \quad (1.61)$$

The main assumptions made to obtain this model are

- The surface tension has been neglected, so that the same pressure p_m is considered for both phases. Moreover, the mixture source terms $\mathbf{m}_m$ (respectively Ξ_m) in the momentum (respectively energy) balance are null,
- Space and time turbulent fluctuations of kinetic energy have been neglected in the energy equation.

If this model is considered with $\varepsilon = 1$, without source terms due to solid, gravity and mass transfer, the total energy formulation is a system of conservative equations. In this case, it can be likened to an extension of the HRM-type model [16] with the addition of a relative velocity $\mathbf{u}_r$ between phases. This model will be used to study the hyperbolicity (see **Chapter 2**) of the model and to provide analytical solutions for one dimensional Riemann problems (see **Chapter 3**).

1.3.2 Enthalpy 4-equation model

The enthalpy 4-equation model writes

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t}(\varepsilon \rho_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m) = 0, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m \otimes \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) \mathbf{u}_r \otimes \mathbf{u}_r) + \varepsilon \nabla p_m = \\ \quad \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st})) \\ \quad + \varepsilon \rho_m \mathbf{g} + \mathbf{m}_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m h_m) + \nabla \cdot (\varepsilon \rho_m h_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) L \mathbf{u}_r) = \\ \quad \varepsilon \frac{\partial p_m}{\partial t} + \varepsilon \left[\mathbf{u}_m + y(1-y) \rho_m \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \cdot \nabla p_m \\ \quad - \nabla \cdot (\varepsilon (\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st})) + \phi_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m y) + \nabla \cdot (\varepsilon \rho_m y \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) \mathbf{u}_r) = \varepsilon \Gamma_g. \end{array} \right. \quad (1.62)$$

The same assumptions as for the total energy model have been made. The additional assumptions compared to the total energy model are:

- The heat contributions due to viscous friction between phases and to solid friction have been neglected in the enthalpy balance. This approximation is justified for nominal conditions of a reactor core. It is not relevant during the start-up of a reactor core, where friction is used to heat the primary circuit before initiating the nuclear reaction, but this configuration is not studied here.
- The sum of the transfer of thermal energy between phases $\sum_k \Lambda_k$ has been neglected in the enthalpy balance.

The enthalpy 4-equation model is the one used in the code THYC-coeur, in **Chapter 5**, and a simplified version is also implemented in the ThermoTorch code presented in **Chapter 6**.

Generally, the two models presented are not equivalent. When the relative velocity is zero and when friction terms and diffusive flux are neglected, the models with total energy and enthalpy become equivalent. The first model proposed in total energy is a conservative model (when considering a free medium). This model allows for the study of hyperbolicity and is suitable for handling shock waves. The enthalpy-based model, on the other hand, is not written in conservative form. Certain terms have been neglected in order to express the enthalpy equation without source terms representing interphase transfer. These assumptions are valid for the applications of THYC-coeur code. This leads to an energy equation with several advantages for numerical simulation. In particular, the decoupling of the energy equation from the rest of the system allows for solving energy independently (see **Chapter 6**).

1.3.3 Mixture Equation of State

The terms of frictions (laminar and turbulent) and of heat flux (laminar and turbulent) due to solid, the relative velocity and the mass transfer are modeled with closure laws. The sets of equations are composed of four equations but five mixture unknowns: density ρ_m , velocity $\mathbf{u}_m$, enthalpy h_m (or total energy E_m), mass fraction y and pressure p_m . An equation of state for the mixture is needed to link the energy to density and pressure. For each phase (liquid and gas), the equation of state from Equation (1.1) is still considered valid for the averaged quantities such that

$$e_k = e_k(p_m, \rho_k), k \in \{l, g\}. \quad (1.63)$$

Saturation is reached when the chemical potentials of the two phases are at equilibrium. It defines the saturation temperature T_{sat} such that

$$\mu_g(p_m, T_{sat}(p_m)) = \mu_l(p_m, T_{sat}(p_m)), \quad (1.64)$$

where the chemical potential is defined by

$$\mu_k(p_m, T_k) = h_k - T_k s_k. \quad (1.65)$$

The thermodynamic quantities of a saturated phase k considered depend only on pressure p_m . In this case, and for the rest of this work, a quantity f_k at saturation is noted

$$f_k(p_m, T_k = T_{sat}(p_m)) = \bar{f}_k(p_m). \quad (1.66)$$

From now on, time averaging is no longer used, so the operator $\bar{\cdot}$ always corresponds to saturation, without the risk of conflicting notations. To simplify the discussions, the dispersed phase considered at saturation here is the gaseous phase such that

$$\begin{aligned} T_g &= T_{sat}(p_m), \\ \bar{\rho}_g(p_m) &= \rho_g(p_m, T_{sat}(p_m)), \\ \bar{e}_g(p_m) &= e_g(p_m, \bar{\rho}_g), \\ \bar{h}_g(p_m) &= \bar{e}_g(p_m) + \frac{p_m}{\bar{\rho}_g}. \end{aligned} \quad (1.67)$$

The equation of state for the mixture (noted Mixture EoS) writes

$$e_m(p_m, \rho_m, y) = y\bar{e}_g(p_m) + (1-y)e_l \left(p_m, \rho_l = \frac{1-y}{\frac{1}{\rho_m} - \frac{y}{\bar{\rho}_g(p_m)}} \right). \quad (1.68)$$

It can also be written with enthalpy such that

$$h_m(p_m, \rho_m, y) = y\bar{h}_g(p_m) + (1-y)h_l \left(p_m, \rho_l = \frac{1-y}{\frac{1}{\rho_m} - \frac{y}{\bar{\rho}_g(p_m)}} \right). \quad (1.69)$$

The mixture speed of sound is defined by

$$(\rho_m c_m)^2 = \left(\frac{\partial e_m}{\partial p_m} \right)_{\rho_m, y}^{-1} \left(p_m - \rho_m^2 \left(\frac{\partial e_m}{\partial \rho_m} \right)_{p_m, y} \right). \quad (1.70)$$

The speed of sound in the mixture is smaller than the two speeds of sound for one-phase flow (either liquid or gas) which is well known [107]. It is assumed here that the EoS for each phase are chosen such that the mixture speed of sound is real. This mixture speed of sound corresponds to the wave velocity in the case $u_r = 0$ (see Section 2.4.1). Using $h_m = y\bar{h}_g + (1-y)h_l$, the relative specific enthalpy $L = \bar{h}_g - h_l$ is given by

$$L(p_m, h_m, y) = \frac{\bar{h}_g(p_m) - h_m}{1-y} > 0. \quad (1.71)$$

1.4 Closure laws

Closure laws are needed for the following terms: the stress tensors (laminar and turbulent) $\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st}$, the heat fluxes (laminar and turbulent) $\varphi_m + \varphi_m^t + \varphi_m^{st}$, the relative velocity $\mathbf{u}_r$, the friction due to solid $\mathbf{m}_s$, the heat due to solide Φ_s and the mass transfer source term Γ_g . For each term, many closure laws have been proposed based on experimental data. Only closure laws for the mass transfer source term and relative velocity are presented hereafter. The form of the friction and flow terms is given in Section 1.4.3. The other closure laws are described in [7].

1.4.1 Mass transfer source term

Taking into account the fourth equation allows for the consideration of flows that are out of thermal equilibrium, which means that liquid and gas can be present at different temperatures in the same location. It can model, for example, the phenomenon of subcooled boiling, where the liquid is not yet at saturation on average, but vapor production has already begun at the wall. The gas temperature is noted T_g , the liquid temperature T_l and the saturation temperature $T_{sat}(p_m)$ at pressure p_m . If the gas phase is the dispersed phase, the gas phase is at saturation (but not the liquid a priori) so that

$$T_g = T_{sat}(p_m) \neq T_l. \quad (1.72)$$

In the case where the gas phase becomes dominant, the liquid phase is at saturation, and the gas is out of thermodynamic equilibrium, such that

$$T_l = T_{sat}(p_m) \neq T_g. \quad (1.73)$$

The fourth equation models the deviation of the vapor mass fraction y from the thermodynamic equilibrium fraction $\bar{y}$ (when both phases are at saturation), defined by Equation (1.59).

It is necessary to model the mass transfer term Γ_g , in particular to account for two major physical phenomena. The first phenomenon is the vapor production Γ_p due to the heat flux from the solid Φ_s supplied to the fluid. The second is the mass transfer due to interfacial exchanges between the phases, denoted Γ_{ie} . This last term accounts for different configurations depending on which phase is out of equilibrium: the recondensation of saturated gas bubbles in the subcooled liquid phase, the vaporization triggered by pressure variations, the vaporization of saturated droplets in a superheated vapor flow, etc. These two contributions are summed so that

$$\Gamma_g = \Gamma_{ie} + \Gamma_p. \quad (1.74)$$

Mass transfer due to the heat flux Γ_p

The fraction of the heat flux Φ_s directly used for vaporization is noted $\chi \in [0, 1]$. The latent heat is defined by $\bar{L} = \bar{h}_g - \bar{h}_l$, which is the energy needed to vaporize 1kg of saturated liquid. The vapor production due to the heat flux is

$$\Gamma_p = \frac{\chi \Phi_s}{\bar{L}} > 0. \quad (1.75)$$

The parameter χ must then be modeled to account for the physical phenomena occurring in a reactor core. The closure laws used for χ are not detailed here. A law is proposed in the modeling of the ThermoTorch 1D code in **Chapter 6**. If no precautions are taken in the closure law for χ , the production term Γ_p won't respect the maximum principle for the mass fraction (see **Chapter 4**). This source term can then produce so much steam that the mass fraction exceeds 1.

Mass transfer due to interfacial exchanges Γ_{ie}

If the mixture were at thermodynamic equilibrium, no heat exchange would occur between the phases, and the mass transfer due to interfacial exchanges Γ_{ie} would be null, so that $\Gamma_{ie}(\bar{y}) = 0$. By using a first-order Taylor expansion close to thermodynamic equilibrium and knowing that Γ_{ie} is expressed in $kg.m^{-3}.s^{-1}$, Γ_{ie} can be written in the form

$$\Gamma_{ie}(y) = \rho_m \frac{\bar{y} - y}{\tau}, \quad (1.76)$$

where τ is a relaxation time and $\bar{y}$ the mass fraction at equilibrium from Equation (1.59). Like the parameter χ , the relaxation time τ is modeled by a closure law. One possibility is a constant relaxation time or one that depends on the deviation from saturation. A correlation is detailed in **Chapter 6**. If $\tau > 0$ and $\bar{y} \in [0, 1]$, this source term for the return to equilibrium ensures the maximum principle (see Section 4.1.2), i.e. the mass fraction remains between 0 and 1.

1.4.2 Relative velocity between phases

The simplest model would consider a zero relative velocity. In this case, both phases have the same velocity, and several terms disappear from the equations. However, the relative velocity has a major impact in the model results. Neglecting the relative velocity would mean ignoring many physical phenomena in the reactor core, such as the buoyancy of the gas phase rising faster than the liquid or the growth of gas bubbles before detachment from the wall. Moreover, spatially averaged equations are considered here. The average velocity is not representative of the velocity profile. For example, in a sub-channel, bubbles close to the wall experience a lower liquid velocity than the average velocity, whereas once in the center of the flow, the entrainment velocity is much higher. These spatial distribution phenomena of void fraction and velocity are not accounted for in the absence of relative velocity.

Zuber and Findlay [145] developed a drift-flux model where the gas velocity results from the superposition of the entrainment velocity of the two-phase flow and a drift velocity that reflects the balance between buoyancy and the interfacial friction of gas bubbles in the liquid flow. The model depends on two parameters, the drift velocity u_{gj} and the distribution parameter C_0 . In the literature, plenty of drift-flux correlations have been proposed to represent different geometric configurations and operating conditions.

First, a brief history of the Zuber & Findlay model [145] is presented. Next, the drift-flux model is derived before explaining the numerical constraints of using this model in the 4-equation system. The two correlations used in this work will then be detailed: the Chexal-Lellouche correlation and the Bestion correlation. A brief literature review will present the validation of those two correlations.

History of the Zuber & Findlay model

A one-dimensional two-phase flow is considered through a section A . Using the local void fraction α , the superficial flux of the flow is

$$j = j_g + j_l, \quad (1.77)$$

with $j_g = \alpha u_g$ being the gas superficial flux and $j_l = (1 - \alpha)u_l$ the liquid superficial flux. Behringer [13] proposed the first model in 1936 by writing the gas phase velocity as

$$u_g = j + u_{gj}. \quad (1.78)$$

This formulation considers the gas drift velocity u_{gj} relative to the liquid (often due to gravity and surface tension between the gas and liquid) resulting from the local effect of the relative velocity.

However, it does not account for the non-uniform distribution of the mass flux and void fraction in the section. Bankoff's work [11] considers this non-uniform distribution of the mass flux by writing the slip ratio S using a Bankoff parameter $K \in [0.5, 1.0]$ (for circular pipes) such that

$$S = \frac{u_g}{u_l} = \frac{1 - \langle \alpha \rangle}{K - \langle \alpha \rangle}, \quad (1.79)$$

with $\langle \alpha \rangle$ the mean of the void fraction on a surface normal to the flow. Armand [6] adds pressure in this type of model by exploring a wide range of pressures for high mass fluxes. Numerous subsequent studies have attempted to account for both the effects of local relative velocity and non-uniform distribution. In particular, Griffith [63] expresses the terminal velocity of a bubble to determine the drift velocity for a slug flow in two-phase flow. The parallel works of *Nicklin and al.* [109] and *Neal* [108] propose a distribution coefficient C_0 to account for the non-uniform effects of the flow by writing

$$u_g = C_0 j + u_{gj}. \quad (1.80)$$

While Nicklin et al. [109] refer to the actual velocity of the vapor plug, Neal [108] already speaks of the averaged velocity as total cross-sectional average gas velocity.

Zuber & Findlay model

To truly consider these distribution effects, Zuber & Findlay [145] average the physical quantities over the section A of the flow such that

$$\langle f \rangle = \frac{1}{A} \int_A f dA. \quad (1.81)$$

The average $\langle f \rangle_\alpha$ weighted by the surface distribution of the void fraction is also used such that

$$\langle f \rangle_\alpha = \frac{\langle \alpha f \rangle}{\langle \alpha \rangle}. \quad (1.82)$$

The drift-flux model introduces two parameters: the drift velocity $\langle u_{gj} \rangle_\alpha$ and the distribution coefficient C_0 . Equation (1.78) can be averaged weighted by the void fraction

$$\langle u_g \rangle_\alpha = \frac{\langle \alpha j \rangle}{\langle \alpha \rangle} + \langle u_{gj} \rangle_\alpha. \quad (1.83)$$

From now on, the problem is not symmetric anymore and the gas phase is considered dispersed as the average are weighted by the void fraction of the gas phase. Generally, the non uniform distribution of the void fraction and velocity cause $\langle u_g \rangle_\alpha \neq \langle u_g \rangle$. Indeed, considering non-uniform spatial distributions:

$$\langle u_g \rangle_\alpha = \frac{\langle \alpha u_g \rangle}{\langle \alpha \rangle} = \frac{\langle j_g \rangle}{\langle \alpha \rangle} \neq \left\langle \frac{j_g}{\alpha} \right\rangle = \langle u_g \rangle. \quad (1.84)$$

Noting the distribution coefficient $C_0 = \frac{\langle \alpha j \rangle}{\langle \alpha \rangle \langle j \rangle}$, Equation (1.83) rewrites

$$\langle u_g \rangle_\alpha = C_0 \langle j \rangle + \langle u_{gj} \rangle_\alpha. \quad (1.85)$$

Noting $\langle \beta \rangle = \frac{\langle j_g \rangle}{\langle j \rangle}$, Zuber & Findlay [145] propose an experimentally usable formulation to determine the average void fraction based on the volumetric mass fluxes of each fluid (through $\langle \beta \rangle$) and the chosen drift-flux model (C_0 and $\langle u_{gj} \rangle_\alpha$) by using

$$\langle \alpha \rangle = \frac{\langle \beta \rangle}{C_0 + \frac{\langle u_{gj} \rangle_\alpha}{\langle j \rangle}} = \frac{\langle j_g \rangle}{\langle u_{gj} \rangle_\alpha + C_0 \langle j \rangle}. \quad (1.86)$$

The distribution coefficient C_0 accounts for the non-uniform distribution of the flow and void fraction, while the drift velocity $\langle u_{gj} \rangle_\alpha$ accounts for the local relative velocity effects (due to gravity for example). To truly account for the physics of the flow, the coefficient C_0 must verify certain conditions depending on the void fraction at the wall α_{wall} and the void fraction at the center of the flow α_{center} :

1. When the flow satisfies $\alpha_{wall} < \alpha_{center}$, $C_0 \in [1, 1.5]$,
2. When the flow satisfies $\alpha_{wall} > \alpha_{center}$, $0 < C_0 < 1$,
3. For uniform profiles ($\alpha_{wall} = \alpha_{center}$), $C_0 = 1$.

In particular, for boiling flows, at low void fractions, the bubbles remain closed to the wall and are convected at a slower speed than the average speed ($C_0 < 1$). As the void fraction increases, the bubbles eventually detach completely from the wall and are carried in the central flow, which is faster than the average flow ($C_0 > 1$). When the flow becomes single-phase gas, the distribution coefficient must be 1. Therefore, C_0 should approach 1 as α approaches 1.

Many studies have followed to propose correlations for the parameters of the drift-flux model to better match experimental results. While some propose flow maps as advised by Zuber & Findlay, others suggest correlations that apply over large operating ranges for specific geometries, neglecting the flow pattern. The two correlations proposed here belong to this second category, which are suitable for industrial applications where the application framework is specific and where the very high pressure (> 70 bar) does not allow for the consideration of classical flow maps. They have been developed for flows in confined spaces such as a tube or a rod bundle of a combustible assembly. The hydraulic diameter d_h is defined as the characteristic size of the subchannel of the rod bundle (or the diameter of the tube) considered.

The drift-flux model has been developed in a 1D framework. It can be extended to consider 3D flows such as in [61] where the distribution parameter is adapted in 3D. It is not necessary when the configuration is three dimensional but the flow is essentially axial, i.e. the flow along the fuel rods from the bottom to the top of the core. Indeed, the transverse velocities u_{xy} , i.e. the flows between fuel rods and between assemblies are negligible compared to the axial one u_z . For nominal cases, $u_z \sim 3 - 5 \text{ m.s}^{-1} \gg u_{xy} \sim 0.2 \text{ m.s}^{-1}$. For transverse flows, the relative velocity due to the drift model is not considered.

Numerical considerations for drift-flux models

The drift-flux velocity was initially created to estimate the average of void fraction over a section of a 1D flow from the average superficial fluxes of each phase. This model was later considered in 6-equation two-fluid numerical codes, such as the TRACE code [112] or in system codes such as the RELAP5 code [46] or CATHARE [14]. In this context, this model is used to evaluate the friction coefficient between the liquid phase and the gas phase.

For these applications, the drift-flux velocity model is involved in an intermediate calculation and does not need to verify numerical conditions for implementation. In THYC-coeur code, the correlation is used to directly evaluate the relative velocity between phases. This methodology can be found in the FLICA code [136] or in recent work in the ESCOT code [43, 97]. To obtain the formula for the relative velocity, some computations are necessary. Using $j = \alpha u_g + (1 - \alpha)u_l$, Equation (1.83) gives

$$\langle u_g \rangle_\alpha = C_0 \left[\langle \alpha \rangle \langle u_g \rangle_\alpha + (1 - \langle \alpha \rangle) \langle u_l \rangle_\alpha \right] + \langle u_{gj} \rangle_\alpha. \quad (1.87)$$

From now on, the average operators are omitted. Using $\delta = \frac{\rho_l}{\rho_g}$, the following identities are used

$$\begin{cases} \alpha = \frac{\delta y}{1 + y(\delta - 1)}, \\ u_g = u_m + (1 - y)u_r, \\ u_l = u_m - yu_r. \end{cases} \quad (1.88)$$

Using Equations (1.87) and (1.88), the relative velocity writes

$$u_r = \frac{1 + y(\delta - 1)}{1 - y} \cdot \frac{u_{gj} + (C_0 - 1)u_m}{1 + y(\delta - 1)(1 - C_0)}. \quad (1.89)$$

The relative velocity appears in several terms of Equations (1.62). Considering Equation (1.89), the relative velocity proportional to $\frac{1}{1-y}$. The most limiting term that may not remain bounded in Equations (1.62), is $y(1-y)u_r^2$, which appears in the Momentum Balance. To ensure that the computation of the relative velocity is numerically feasible, regardless of the void fraction, the parameters must verify three constraints.

1. The drift velocity u_{gj} must remain bounded for $0 \leq \alpha \leq 1$ and must verify

$$u_{gj} = \mathcal{O}_{\alpha \rightarrow 1}((1 - \alpha)^{1/2}). \quad (1.90)$$

2. The distribution coefficient C_0 must remain bounded and verify

$$(1 - C_0) = \mathcal{O}_{\alpha \rightarrow 1}((1 - \alpha)^{1/2}). \quad (1.91)$$

3. The distribution coefficient must verify

$$C_0 < 1 + \frac{1}{y(\delta - 1)} = \frac{\delta}{\alpha(\delta - 1)}. \quad (1.92)$$

Chexal-Lellouche correlation

The Chexal-Lellouche correlation [22, 23, 24] has been developed to cover a full range of pressures, flows, void fractions, and different fluid types typical of flows in nuclear reactor core, for PWR and BWR fuel assemblies. Only the parameters for an ascending vertical flow of water/steam (co-current flows) are detailed here. Other formulations for different fluids and geometries can be found in [23, 24]. The parameters u_{gj} and C_0 depend upon the density of each phase, the void fraction α and the hydraulic diameter d_h . The Reynolds Number for each phase also appears, defined by

$$Re_k = \frac{\alpha_k \rho_k u_k d_h}{\mu_k}, \quad k = \{g, l\}, \quad (1.93)$$

with μ_k the dynamic viscosity of phase k . The flow quality of the vapor is noted x_d and is obtained through

$$x_d = \frac{\alpha \rho_g u_g}{\rho_m u_m}, \quad (1.94)$$

so that the Reynolds Numbers rewrite

$$\begin{cases} Re_g = \frac{x_d \rho_g u_g d_h}{\mu_g}, \\ Re_l = \frac{(1 - x_d) \rho_l u_l d_h}{\mu_l}. \end{cases} \quad (1.95)$$

A fluid Reynolds Number is defined by

$$Re_f = \max(Re_g, Re_l). \quad (1.96)$$

Using these definitions and the ratio of densities noted $\delta = \frac{\rho_g}{\rho_l}$, the drift velocity (oriented in the same direction as gravity) writes

$$u_{gj} = 1.41 C_1(\alpha, Re_f) \times C_2(\rho_l, \rho_g) \times C_3(Re_l) \times C_4(d_h) \left(\frac{\sigma g(\delta - 1)}{\rho_l} \right)^{0.25}, \quad (1.97)$$

with σ the surface tension between water and steam, g the gravitational constant and

$$\begin{aligned} C_1 &= (1 - \tilde{\alpha})^{B_1} \text{ where } B_1 = \min \left(0.8; \left[1 + \exp \left(-\frac{Re_f}{60000} \right) \right]^{-1} \right), \\ C_2 &= \begin{cases} \text{If } \delta \leq 18, & C_2 = 0.4757 [\ln(\delta)]^{0.7} \\ \text{If } \delta > 18, & \begin{cases} \text{If } C_5 \geq 1 & C_2 = 1 \\ \text{If } C_5 < 1 & C_2 = \left(1 - \exp \left[-\frac{C_5}{1-C_5} \right] \right)^{-1} \end{cases} \end{cases} \quad \text{with } C_5 = \sqrt{\frac{150}{\delta}}, \\ C_3 &= \max \left(0.5; 2 \exp \left[-\frac{Re_f}{300000} \right] \right), \\ C_4 &= \begin{cases} \text{If } C_6 \geq 1, & C_4 = 1, \\ \text{If } C_6 < 1, & C_4 = \left(1 - \exp \left[-\frac{C_6}{1-C_6} \right] \right)^{-1} \end{cases} \quad \text{with } C_6 = \left(\frac{0.09144}{d_h} \right)^{0.6}. \end{aligned} \quad (1.98)$$

The distribution coefficient C_0 writes

$$C_0 = \frac{L_v}{K_0 + (1 - K_0)\alpha^{r_v}}, \quad (1.99)$$

with

$$\begin{cases} K_0 = B_1 + (1 - B_1)\delta^{0.25}, \\ L_v = \frac{1 - e^{-\alpha C_p}}{1 - e^{-C_p}}, \\ C_p = \frac{4p_{cr}^2}{p(p_{cr} - p)}, \\ r_v = \frac{1 + 1.57\delta}{1 - B_1}. \end{cases} \quad (1.100)$$

Here p_{cr} denotes the critical pressure for water. This correlation is valid for a wide range of parameters:

$$\begin{cases} 0.01 < \alpha < 0.95, \\ 0.02 \text{ kg.m}^{-2}.\text{s}^{-1} < G < 2160 \text{ kg.m}^{-2}.\text{s}^{-1}, \\ 1 \text{ bar} < p < 150 \text{ bar}, \\ 1.1 \text{ kW.m}^{-2} < \phi_s < 2210 \text{ kW.m}^{-2}, \\ 0^\circ\text{C} < \Delta T_{sat} < 30^\circ\text{C}, \\ 0.009 \text{ m} < d_h < 0.048 \text{ m}, \end{cases} \quad (1.101)$$

where ΔT_{sat} is the deviation from saturation. In reactor core conditions at low void fraction, the relative velocity obtained with this correlation is negative and close to the mixture velocity (in absolute value). This models the growth of gas bubbles during their formation before they detach and are carried away in the flow. The Chexal-Lellouche correlation satisfies the numerical constraints (1.90) and (1.91) as

$$\begin{aligned} u_{gj} &\underset{\alpha \rightarrow 1}{\sim} (1 - \alpha)^{B_1}, B_1 \in [0.5, 0.8] > 0.5, \\ C_0 &= 1 + K(1 - \alpha) + o_{\alpha \rightarrow 1}(1 - \alpha), K \in \mathbb{R}. \end{aligned} \quad (1.102)$$

The Chexal-Lellouche correlation does not satisfy condition (1.92), at low pressure and high void fraction. In practice, these conditions are never reached in industrial applications. This correlation is for example implemented in the 6-equation code RELAP5 [46] and in the 4-equation code ESCOT [97]. This correlation is used in THYC-coeur code [7] until V6.0.

Bestion correlation

The Bestion correlation [14] has been developed for dispersed flows in PWR applications. It is based on the terminal velocity of a vapor bubble with a characteristic size d_h in a vertical flow (see **Appendix 1.C**). Despite its simplicity, it shows good results for dispersed and stratified flows [28, 66, 105]. The drift-flux parameters for a vertical flow are

$$\begin{cases} u_{gj} = 0.188 \sqrt{g d_h (\delta - 1)}, \\ C_0 = 1, \end{cases} \quad (1.103)$$

with the drift velocity pointing upward (opposite to gravity). The coefficient 0.188 has been set to match several experimental results [14]. Despite automatically satisfying constraints (1.91) and (1.92), this correlation does not verify the numerical constraint (1.90). Numerical precautions must be considered at very high mass fraction ($y \rightarrow 1$). In PWRs conditions, the mass fraction never reaches this value for steady-state solutions. This correlation is used in the system code CATHARE [14] or the code TRACE [8, 112]. In the original article [14], the distribution coefficient C_0 is not specified. While *Miwa and al.* [105] use the formula $C_0 = 1$, *Gui and al.* [66] present a different formula:

$$C_0 = \begin{cases} 1.2 - 0.2\sqrt{\delta - 1} & \text{for } \alpha < 0.4, \\ 1.0 & \text{for } \alpha \geq 0.4. \end{cases} \quad (1.104)$$

The formula $C_0 = 1$ is the most accurate for PWR applications [66]. It was retained for THYC-coeur V7.0.

Validation of the correlations

It is difficult to compare relative velocity models for the 4-equation model because this model is directly coupled to the closure laws used for the relaxation time τ and the fraction of the heat flux used for vaporization χ . Many studies compare relative velocity correlations [28, 66, 72, 73, 85, 113] with experimental results approximating the operating conditions of PWR and BWR as closely as possible. In most articles, the tests considered are at low pressure. In the articles [66, 85, 105], the pressures are $p \in [0.1, 2.0]$ MPa. In *Coddington and al.* [28], the experimental data in PWR/BWR geometry does not exceed 10 MPa. *Ozaki and al.* [113] use experimental data with rod bundles geometry but still at low pressure.

The Chexal-Lellouche and Bestion correlations are among the most accurate in all these comparisons, particularly for PWR-type geometries. *Gui et al.* [66] notes that Bestion tends to underestimate the void fraction for $\alpha > 0.75$, whereas Chexal-Lellouche seems to better predict this region.

However, it emphasizes that Chexal-Lellouche significantly overestimates the void fraction for low void fractions. In *Julia and al.* [85], the Bestion correlation is the most accurate, with a relative prediction error of the void fraction of $\pm 23.8\%$ compared to Chexal-Lellouche, for which the error is $\pm 38.6\%$. The article by *Ozaki and al.* [113] analyzes the FRIGG experimental data ($p \in [1, 8.7]$ MPa, $G \in [470 - 2160]$ $kg.m^{-2}.s^{-1}$). The Bestion correlation underestimates the void fraction for all experimental points, especially for high void fractions. For Chexal-Lellouche, the void fraction is slightly underestimated at high void fractions. Unlike the *Inoue* [80] and *Maier and Coddington* [101] correlations, also implemented in the industrial code THYC-coeur, which consider only pressure and mass mass flux, the Bestion and Chexal-Lellouche correlations can also work at other operating points (e.g., at low pressure) even if the performance is less optimal.

Despite all these experimental comparison, there is a lack of results at the nominal pressure of a reactor core ($p > 100$ bar). High pressures make testing and visualization very complex. However, the analysis of the PSBT case results [121], which are the closest experimental data to PWR pressure conditions, shows that the Bestion correlation presents better results than Chexal-Lellouche. The two presented correlations are considered in the following.

For THYC-coeur code [7], an additional term is considered for relative velocity, due to turbulent effect and to mass fraction gradient. It is added to the drift-flux relative velocity noted $\mathbf{u}_r^{\text{drift-flux}}$. The total relative velocity writes

$$\mathbf{u}_r^{THYC} = \mathbf{u}_r^{\text{drift-flux}} - \frac{D}{y(1-y)} \nabla y, \quad (1.105)$$

where D is a turbulent diffusion coefficient modeled by a closure law. Unlike drift-flux relative velocity, which has a component only along the axial flow, the turbulent term has components in each direction.

1.4.3 Additional closure laws

Viscous terms $\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st}$ are modeled using a mixture viscosity μ and a turbulent viscosity μ_t such that

$$\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st} = (\mu + \mu_t) \left(\nabla \mathbf{u}_m + {}^t \nabla \mathbf{u}_m - \frac{2}{3} \nabla \cdot \mathbf{u}_m \mathbf{I}_d \right). \quad (1.106)$$

Thermal diffusive fluxes are expressed by Fourier's law as follows

$$\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st} = (\lambda + \lambda_t) \nabla T_m, \quad (1.107)$$

where λ is the mixture conductivity and λ_t is the mixture turbulent conductivity. The closure laws for viscosity (laminar and turbulent) and for conductivity (laminar and turbulent) are detailed in [7]. With the exception of **Chapter 5**, where the complete THYC-coeur model is considered, these 2nd-order terms are neglected in the work presented here.

1.5 THYC-coeur model

In THYC-coeur, the enthalpy 4-equation model is used. By omitting the indices m for the mixing quantities, the model is written as follows

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t}(\varepsilon \rho) + \nabla \cdot (\varepsilon \rho \mathbf{u}) = 0, \\ \frac{\partial}{\partial t}(\varepsilon \rho \mathbf{u}) + \nabla \cdot (\varepsilon \rho \mathbf{u} \otimes \mathbf{u}) + \nabla \cdot (\varepsilon \rho y(1-y) \mathbf{u}_r \otimes \mathbf{u}_r) + \varepsilon \nabla p = \\ \quad \nabla \cdot (\varepsilon (\mathbf{T} + \mathbf{T}^t + \mathbf{T}^{st})) \\ \quad + \varepsilon \rho \mathbf{g} + \mathbf{m}_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho h) + \nabla \cdot (\varepsilon \rho h \mathbf{u}) + \nabla \cdot (\varepsilon \rho y(1-y) L \mathbf{u}_r) = \\ \quad \varepsilon \frac{\partial p}{\partial t} + \varepsilon \left[\mathbf{u} + y(1-y) \rho \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \cdot \nabla p \\ \quad - \nabla \cdot (\varepsilon (\boldsymbol{\varphi} + \boldsymbol{\varphi}^t + \boldsymbol{\varphi}^{st})) + \Phi_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho y) + \nabla \cdot (\varepsilon \rho y \mathbf{u}) + \nabla \cdot (\varepsilon \rho y(1-y) \mathbf{u}_r) = \varepsilon \Gamma_g. \end{array} \right. \quad (1.108)$$

The porosity ε is a given function. The system is composed of 18 unknowns

$$\rho, \mathbf{u}, \mathbf{u}_r, h, y, p, \Gamma_g, \mathbf{T}, \mathbf{T}^t, \mathbf{T}^{st}, \mathbf{m}_s, L, \rho_g, \rho_l, \boldsymbol{\varphi}, \boldsymbol{\varphi}^t, \boldsymbol{\varphi}^{st}, \Phi_s. \quad (1.109)$$

The following relations and equations close the system:

- The **pressure** p , the mixture **velocity** $\mathbf{u}$, the mixture **enthalpy** h and the **mass fraction** y are the **main unknowns** of the system, determined by the four equations of the model.
- The density ρ is obtained with the Mixture Equation of State (1.69). The EoS gives $h(\rho, p, y)$. It can be reversed to obtain the density ρ .
- The phase densities ρ_g and ρ_l are obtained using the assumption that the minority phase is at saturation. If the gaseous phase is at saturation for example

$$\left\{ \begin{array}{l} \rho_g(p) = \bar{\rho}_g(p), \\ \rho_l(\rho, y, p) = \frac{1-y}{\frac{1}{\rho} - \frac{y}{\bar{\rho}_g(p)}}. \end{array} \right. \quad (1.110)$$

- The specific relative enthalpy L is obtained with Equation (1.71).
- The relative velocity $\mathbf{u}_r$ is computed with the drift-flux model from Equation (1.89). The drift-flux parameters u_{gj} and C_0 are obtained with the Bestion correlation (see Equation (1.103)) or with the Chexal-Lellouche correlation (see Equations (1.97), (1.98) and (1.99)). A turbulent contribution due to mass fraction gradient can be added with Equation (1.105).
- The mass transfer Γ_g is decomposed in two terms (see Equation (1.74)): the mass transfer due to heat flux from Equation (1.75) and the mass transfer due to interfacial exchanges from Equation (1.76).
- The viscous friction terms $\mathbf{T} + \mathbf{T}^t + \mathbf{T}^{st}$ are modeled with Equation (1.106) using a mixture viscosity and a turbulent viscosity.
- The diffusive heat fluxes $\boldsymbol{\varphi} + \boldsymbol{\varphi}^t + \boldsymbol{\varphi}^{st}$ are expressed with Fourier's law from Equation (1.107).
- The source terms due to solid $\mathbf{m}_s$ and Φ_s are detailed in [7].

1.A Detailed computation of the model

In this appendix, the computation to obtain the time- and space-averaged equations is detailed. The derivation of this model is based on [33, 82].

1.A.1 Time averaging

Definition of time intervals and local void fraction

For a point $\mathbf{x}_0$ in the domain, an instant t_0 is considered. The quantities are averaged over the time interval denoted Δt (see Figure 1.1) centered on t_0 , which is sufficiently large to smooth out the local variations of properties but small enough compared to the macroscopic time constant unsteadiness of the bulk flow. The following notation is adopted

$$[\Delta t] = [t_0 - \frac{\Delta t}{2}, t_0 + \frac{\Delta t}{2}]. \quad (1.111)$$

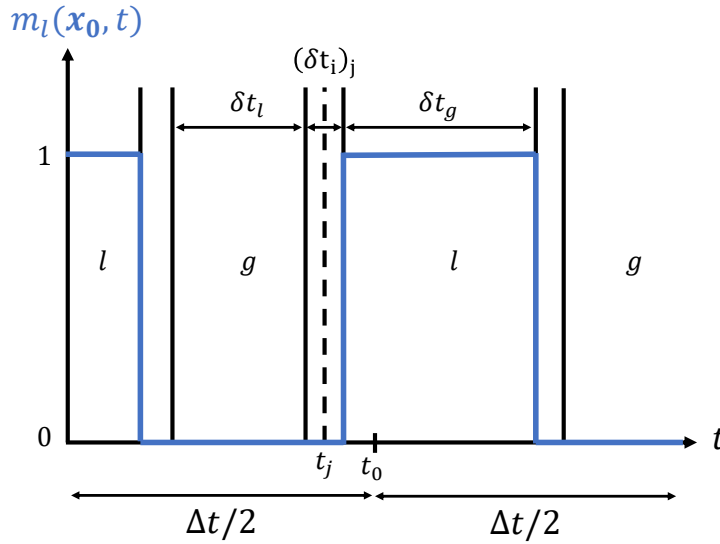


FIGURE 1.1

State density (defined in Equation (1.112)) for the liquid during interval Δt - Scheme representing the different time intervals.

During the interval Δt , the fluid is liquid for several time intervals $(\delta t_l)_j$ and gaseous for several intervals $(\delta t_g)_j$ (see Figure 1.1). The following notations are used: $\Delta t_l = \sum_j (\delta t_l)_j$ and $\Delta t_g = \sum_j (\delta t_g)_j$. To simplify the integration intervals, the notation $t \in [\Delta t_k]$ is used for an instant t when the phase k is present. The state density function for phase k can be defined by

$$m_k(x, t) = \begin{cases} 1 & \text{if } t \in [\Delta t_k], \\ 0 & \text{otherwise.} \end{cases} \quad (1.112)$$

The state density function enables to extend the definition of phase quantities f_k for any instant. The local void fraction α of phase k is

$$\alpha_k(\mathbf{x}_0, t_0) = \frac{1}{\Delta t} \int_{t \in [\Delta t]} m_k(x, t) dt = \frac{\Delta t_k}{\Delta t}. \quad (1.113)$$

It corresponds to the probability of finding phase k at the given time and location. It is also called the local time fraction. During phase change between liquid and gas, an interface j at instant t_j of thickness ϵ_j and velocity $\mathbf{v}_j$ exists during an interval $(\delta t_i)_j$ such that

$$(\delta t_i)_j = \frac{\epsilon_j}{v_{nj}}, \quad (1.114)$$

with $v_{nj} > 0$ the normal velocity of the interface. During the interval Δt , several interfaces j may appear (see Figure 1.1). Δt_i denotes the time interval where an interface is present, such that

$$\Delta t_i = \sum_j (\delta t_i)_j < \Delta t. \quad (1.115)$$

The time interval Δt is therefore decomposed as follows

$$\Delta t = \Delta t_i + \Delta t_l + \Delta t_g. \quad (1.116)$$

Non stationary interfaces of zero thickness are considered such that

$$\lim_{\epsilon_j \rightarrow 0} (\delta t_i)_j = 0, \forall j \text{ with } |\mathbf{v}_i^j| \neq 0. \quad (1.117)$$

With this assumption, $\Delta t_i = 0$ such that

$$\Delta t = \Delta t_g + \Delta t_l. \quad (1.118)$$

Using Equation (1.118), the immiscibility condition can be written for the local void fractions

$$\alpha_g + \alpha_l = 1. \quad (1.119)$$

Definition of several time averaging operators

A first time average $\bar{f}_k$ is defined as the mean value of a function f_k over the interval Δt such that

$$\bar{f}_k(\mathbf{x}_0, t_0) = \frac{1}{\Delta t} \int_{t \in [\Delta t]} f_k(\mathbf{x}_0, t) dt. \quad (1.120)$$

Using this definition, the local void fraction rewrites as a time average of the state density function

$$\alpha_k = \overline{m_k}. \quad (1.121)$$

A more natural mean value for quantity f_k is the average value using the state density function as a weighting function. This time average is noted $\overline{\overline{f}}_k$ such that

$$\overline{\overline{f}}_k(\mathbf{x}_0, t_0) = \frac{\overline{m_k f_k}}{\overline{m_k}} = \frac{\overline{f_k}}{\alpha_k} = \frac{1}{\Delta t_k} \int_{t \in [\Delta t_k]} f_k(\mathbf{x}_0, t) dt. \quad (1.122)$$

It corresponds to the mean value on interval $[\Delta t_k]$ where phase k exists. The time average $\overline{\overline{f}}_k$ is adapted to intensive quantities such as the pressure or the density. For extensive quantities (velocity, specific energy...), mass weighted mean value must be defined. For an extensive quantity ψ_k , the time average weighted by the density is noted $\widehat{\psi}_k$ and defined by

$$\widehat{\psi}_k = \frac{\overline{\rho_k \psi_k}}{\overline{\rho_k}} = \frac{\overline{\overline{\rho_k \psi_k}}}{\overline{\overline{\rho_k}}}. \quad (1.123)$$

The local instant equations are averaged using operator $\overline{\cdot}$. This operator is not commutative with temporal and spatial derivatives. The commutation of the operator introduces a source term that involves the jump of the derived quantity at interface. Each interface j is defined by its outward normal $\mathbf{n}_k^j$ for the side of phase k . The normal velocity of the interface is still denoted $v_{nj} = |\mathbf{v}_j \cdot \mathbf{n}_k^j| > 0$. The following relations are given by [82]:

$$\begin{aligned}\frac{\partial \overline{f_k}}{\partial t}(\mathbf{x}_0, t_0) &= \frac{\partial \overline{f_k}}{\partial t_0}(\mathbf{x}_0, t_0) - \frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} \left(f_k(\mathbf{x}_0, t_j) \mathbf{n}_k^j \cdot \mathbf{v}_j \right), \\ \overline{\nabla f_k}(\mathbf{x}_0, t_0) &= \nabla \overline{f_k}(\mathbf{x}_0, t_0) + \frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} f_k(\mathbf{x}_0, t_j) \mathbf{n}_k^j.\end{aligned}\tag{1.124}$$

Methodology

The methodology is detailed for a model equation on variable ψ_k defined by

$$\frac{\partial \rho_k \psi_k}{\partial t} + \nabla \cdot (\rho_k \psi_k \mathbf{u}_k) + \nabla \cdot \mathbf{f}_k - \rho_k \chi_k = 0,\tag{1.125}$$

with $\mathbf{f}_k$ a flux and χ_k a scalar volumetric source. This methodology is the same when ψ_k is a vector. The flux $\mathbf{f}_k$ becomes a tensor and χ_k a vector. Time fluctuating quantities are noted with superscript $'$ and defined by

$$\begin{cases} \rho_k = \overline{\overline{\rho_k}} + \rho_k', \\ \mathbf{u}_k = \widehat{\mathbf{u}}_k + \mathbf{u}_k', \\ \psi_k = \widehat{\psi}_k + \psi_k'. \end{cases}\tag{1.126}$$

Using the properties $\overline{\overline{\overline{f}}} = \overline{f}$ and $\widehat{\widehat{u}} = \widehat{u}$, the fluctuating quantities satisfy

$$\begin{cases} \overline{\overline{\rho_k'}} = 0, \\ \overline{\overline{\rho_k \mathbf{u}_k'}} = 0, \\ \overline{\overline{\rho_k \psi_k'}} = 0. \end{cases}\tag{1.127}$$

Using these properties, the following identity is obtained

$$\overline{\overline{\rho_k \psi_k \mathbf{u}_k}} = \overline{\overline{\rho_k}} \widehat{\widehat{\psi_k}} \widehat{\mathbf{u}} + \mathbf{f}_k^t,\tag{1.128}$$

with the turbulent flux $\mathbf{f}_k^t$ defined by

$$\mathbf{f}_k^t = \overline{\overline{\rho_k \psi_k' \mathbf{u}_k'}}.\tag{1.129}$$

The time-averaging of Equation (1.125) is detailed in several steps below.

- Time-averaging the local instant formulation:

$$\overline{\frac{\partial \rho_k \psi_k}{\partial t} + \nabla \cdot (\rho_k \psi_k \mathbf{u}_k) + \nabla \cdot \mathbf{f}_k - \rho_k \chi_k} = 0.\tag{1.130}$$

- Using the distributive property of time-average:

$$\overline{\frac{\partial \rho_k \psi_k}{\partial t}} + \overline{\nabla \cdot (\rho_k \psi_k \mathbf{u}_k)} + \overline{\nabla \cdot \mathbf{f}_k} - \overline{\rho_k \chi_k} = 0.\tag{1.131}$$

- Using Equations (1.124) for differential operators, which introduces a source term:

$$\frac{\partial \overline{\rho_k \psi_k}}{\partial t} + \nabla \cdot (\overline{\rho_k \psi_k \mathbf{u}_k}) + \nabla \cdot \bar{\mathbf{f}}_k - \overline{\rho_k \chi_k} = \Psi_k^i, \quad (1.132)$$

with

$$\Psi_k^i = -\frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} \mathbf{n}_k^j \cdot (\rho_k \psi_k (\mathbf{u}_k - \mathbf{v}_j) + \mathbf{f}_k). \quad (1.133)$$

- Using the weighted quantities by the density function (noted $\bar{\bar{f}}$):

$$\frac{\partial}{\partial t} (\alpha_k \overline{\overline{\rho_k \psi_k}}) + \nabla \cdot (\alpha_k \overline{\overline{\rho_k \psi_k \mathbf{u}_k}}) + \nabla \cdot (\alpha_k \bar{\bar{\mathbf{f}}}_k) - \alpha_k \overline{\overline{\rho_k \chi_k}} = \Psi_k^i. \quad (1.134)$$

- Using the weighted quantities by the density ρ_k (noted $\hat{f}$) and Equation (1.128):

$$\frac{\partial}{\partial t} (\alpha_k \bar{\bar{\rho}}_k \hat{\psi}_k) + \nabla \cdot (\alpha_k \bar{\bar{\rho}}_k \hat{\psi}_k \hat{\mathbf{u}}_k) + \nabla \cdot (\alpha_k (\bar{\bar{\mathbf{f}}}_k + \mathbf{f}_k^t)) - \alpha_k \bar{\bar{\rho}}_k \hat{\chi}_k = \Psi_k^i. \quad (1.135)$$

Time-averaged equations

The methodology described above is used on Equations (1.9), (1.10) and (1.11). Considering $\psi_k = 1$, $\mathbf{f}_k = \mathbf{0}$ and $\chi_k = 0$, the time-averaged mass balance writes

$$\frac{\partial \alpha_k \bar{\bar{\rho}}_k}{\partial t} + \nabla \cdot (\alpha_k \bar{\bar{\rho}}_k \hat{\mathbf{u}}_k) = \Gamma_k, \quad (1.136)$$

with

$$\Gamma_k = -\frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} \mathbf{n}_k^j \cdot (\rho_k (\mathbf{u}_k - \mathbf{v}_j)). \quad (1.137)$$

Considering $\psi_k = \mathbf{u}_k$, $\mathbf{f}_k = p_k \mathbf{I}_d - \mathbf{T}_k$ and $\chi_k = \mathbf{g}$, the time-averaged momentum balance is

$$\frac{\partial}{\partial t} (\alpha_k \bar{\bar{\rho}}_k \hat{\mathbf{u}}_k) + \nabla \cdot (\alpha_k \bar{\bar{\rho}}_k \hat{\mathbf{u}}_k \otimes \hat{\mathbf{u}}_k) + \nabla (\alpha_k \bar{\bar{p}}_k) = \nabla \cdot (\alpha_k (\bar{\bar{\mathbf{T}}}_k + \mathbf{T}_k^t)) + \alpha_k \bar{\bar{\rho}}_k \mathbf{g} + \mathbf{m}_k, \quad (1.138)$$

with

$$\begin{cases} \mathbf{T}_k^t = -\mathbf{f}_k^t = -\overline{\overline{\rho_k \mathbf{u}'_k \otimes \mathbf{u}'_k}}, \\ \mathbf{m}_k = -\frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} \mathbf{n}_k^j \cdot [\rho_k (\mathbf{u}_k - \mathbf{v}_j) \otimes \mathbf{u}_k + p_k \mathbf{I}_d - \mathbf{T}_k]. \end{cases} \quad (1.139)$$

Considering $\psi_k = E_k = e_k + \frac{u_k^2}{2}$, $\mathbf{f}_k = \varphi_k - (p_k \mathbf{I}_d - \mathbf{T}_k) \cdot \mathbf{u}_k$ and $\chi_k = \mathbf{g} \cdot \mathbf{u}_k$, the time-averaged energy balance is

$$\begin{aligned} \frac{\partial}{\partial t} \left(\alpha_k \bar{\bar{\rho}}_k \left(\hat{e}_k + \frac{\hat{u}_k^2}{2} + \frac{\widehat{(u'_k)^2}}{2} \right) \right) + \nabla \cdot \left(\alpha_k \bar{\bar{\rho}}_k \left(\hat{e}_k + \frac{\hat{u}_k^2}{2} + \frac{\widehat{(u'_k)^2}}{2} \right) \hat{\mathbf{u}}_k \right) = \\ - \nabla \cdot \left(\alpha_k (\bar{\bar{\varphi}}_k + \mathbf{f}_k^t - \overline{\overline{(-p_k \mathbf{I}_d + \mathbf{T}_k) \cdot \mathbf{u}'_k}}) \right) \\ + \nabla \cdot (\alpha_k (-\bar{\bar{p}}_k \mathbf{I}_d + \bar{\bar{\mathbf{T}}}_k) \cdot \hat{\mathbf{u}}_k) \\ + \alpha_k \bar{\bar{\rho}}_k \mathbf{g} \cdot \hat{\mathbf{u}}_k + \Xi_k, \end{aligned} \quad (1.140)$$

with $\mathbf{f}_k^t$ the turbulent term due to time averaging and Ξ_k the total energy transfer between phase k and the other such that

$$\begin{cases} \overline{\mathbf{f}_k^t} = \overline{\rho_k \left(e_k + \frac{u_k^2}{2} \right)' \mathbf{u}_k'}, \\ \Xi_k = -\frac{1}{\Delta t} \sum_j \frac{1}{v_{nj}} \mathbf{n}_k^j \cdot \left[\rho_k (\mathbf{u}_k - \mathbf{v}_j) \left(e_k + \frac{u_k^2}{2} \right) + (p_k \mathbf{I}_d - \mathbf{T}_k) \cdot \mathbf{u}_k + \varphi_k \right]. \end{cases} \quad (1.141)$$

Using the fluctuating terms for pressure and friction such that $p_k = \bar{p}_k + p'_k$ and $\mathbf{T}_k = \bar{\mathbf{T}}_k + \mathbf{T}'_k$, the following identity can be used using $\overline{\rho_k \mathbf{u}_k'} = \mathbf{0}$ and enthalpy $h_k = e_k + \frac{p_k}{\rho_k}$

$$\begin{aligned} \mathbf{f}_k^t - \overline{(-p_k \mathbf{I}_d + \mathbf{T}_k) \cdot \mathbf{u}_k'} &= \overline{\rho_k \left(e'_k + \frac{\hat{u}_k^2}{2} + \widehat{\mathbf{u}}_k \cdot \mathbf{u}_k' + \frac{(u'_k)^2}{2} \right) \mathbf{u}_k'} + \overline{p'_k \mathbf{u}_k'} - \overline{\mathbf{T}'_k \cdot \mathbf{u}_k'} \\ &= \overline{\rho_k h'_k \mathbf{u}_k'} + \overline{\mathbf{T}_k^t \cdot \mathbf{u}_k} + \overline{\rho_k \frac{(u'_k)^2}{2} \mathbf{u}_k'} - \overline{\mathbf{T}'_k \cdot \mathbf{u}_k'}. \end{aligned} \quad (1.142)$$

The turbulent enthalpy flux is noted φ_k^t and defined by

$$\varphi_k^t = \overline{\rho_k h'_k \mathbf{u}_k'}. \quad (1.143)$$

In the considered applications where high pressures and temperatures are considered, internal energy is dominant compared to the kinetic energy. Therefore, time fluctuations of kinetic energy can be neglected (in the time derivative, the convection and Equation (1.142)). The energy equation writes finally:

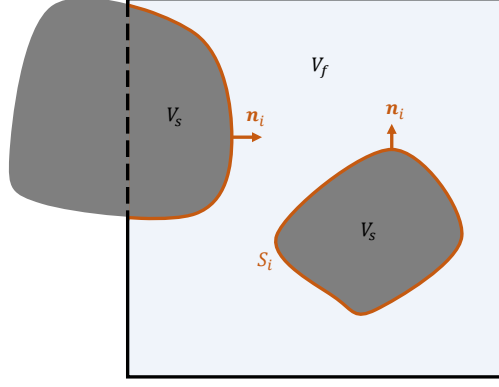
$$\begin{aligned} \frac{\partial}{\partial t} \left(\alpha_k \bar{\rho}_k \left(\hat{e}_k + \frac{\hat{u}_k^2}{2} \right) \right) + \nabla \cdot \left(\alpha_k \bar{\rho}_k \left(\hat{e}_k + \frac{\hat{u}_k^2}{2} \right) \hat{\mathbf{u}}_k \right) + \nabla \cdot (\alpha_k \bar{p}_k \mathbf{u}_k) = \\ - \nabla \cdot (\alpha_k (\bar{\varphi}_k + \varphi_k^t)) \\ + \nabla \cdot (\alpha_k (\bar{\mathbf{T}}_k + \mathbf{T}_k^t) \cdot \hat{\mathbf{u}}_k) \\ + \alpha_k \bar{\rho}_k \mathbf{g} \cdot \hat{\mathbf{u}}_k + \Xi_k. \end{aligned} \quad (1.144)$$

At the interface, there is no accumulation of mass. Using the results of [83], the sum of the source terms can be expressed as function of the surface tension between phases such that

$$\begin{cases} \sum_k \Gamma_k = 0, \\ \sum_k \mathbf{m}_k = \mathbf{m}_m(\sigma), \\ \sum_k \Xi_k = \Xi_m(\sigma). \end{cases} \quad (1.145)$$

1.A.2 Space averaging

Considering a volume V_0 centered at point $\mathbf{x}_0$, the fluid is occupying a volume V_f and several solids are occupying the volume V_s (see Figure 1.2). The total surface of solids contained in volume V_0 is noted S_i with an outward normal noted $\mathbf{n}_i$. The objective is to space-average using the averages defined from Equations (1.25), (1.26), (1.28) and (1.29). Considering the model of Equations (1.135), a methodology similar to time averaging is used.


FIGURE 1.2

Control volume V_0 considered with fluid occupying a volume V_f and several solids occupying a volume V_s of surface S_i and normal $\mathbf{n}_i$.

Similarly to time-averaging, spatial fluctuating moment (noted with " superscript) are considered such that

$$\begin{cases} \hat{\mathbf{u}}_k = \widetilde{\mathbf{u}}_k + \mathbf{u}_k'', \\ \hat{\psi}_k = \widetilde{\psi}_k + \psi_k'' \\ \alpha_k \bar{\bar{f}}_k = \langle\langle \alpha_k \rangle\rangle \tilde{f}_k + \alpha_k f_k'' \end{cases} \quad \begin{array}{l} \text{for extensive variables,} \\ \\ \text{for intensive variables.} \end{array} \quad (1.146)$$

A similar identity than for the time averaging is used

$$\langle\langle \alpha_k \bar{\bar{\rho}}_k \hat{\psi}_k \hat{\mathbf{u}}_k \rangle\rangle = \langle\langle \alpha_k \rangle\rangle \tilde{\rho}_k \widetilde{\psi}_k \widetilde{\mathbf{u}}_k + \langle\langle \alpha_k \rangle\rangle \mathbf{f}_k^{st}, \quad (1.147)$$

with $\mathbf{f}_k^{st}$ the spatial turbulent flux defined by

$$\mathbf{f}_k^{st} = \frac{\langle\langle \alpha_k \bar{\bar{\rho}}_k \psi_k'' \mathbf{u}_k'' \rangle\rangle}{\langle\langle \alpha_k \rangle\rangle} = \overline{\bar{\rho}_k \psi_k'' \mathbf{u}_k''}. \quad (1.148)$$

The spatial turbulent flux is noted with the superscript st to avoid confusion with the turbulence due to time averaging. Differential operators do not commute with space averaging. Using results from [110], Space-averaged of the time derivative writes

$$\left\langle \frac{\partial f_k}{\partial t} \right\rangle = \frac{\partial \langle f_k \rangle}{\partial t} - \frac{1}{V_0} \int_{S_i} f_k \mathbf{v}_i \cdot \mathbf{n}_i dS, \quad (1.149)$$

where $\mathbf{v}_i$ is the velocity at the interface between solid and fluid. For gradient operators, two different formulas can be used

$$\langle \nabla (\alpha_k f_k) \rangle = \nabla (\varepsilon \langle\langle \alpha_k \rangle\rangle \tilde{f}_k) + \frac{1}{V_0} \int_{S_i} \alpha_k f_k \mathbf{n}_i dS, \quad (1.150)$$

and

$$\langle \nabla (\alpha_k f_k) \rangle = \varepsilon \langle\langle \alpha_k \rangle\rangle \nabla \tilde{f}_k + \frac{1}{V_0} \int_{S_i} \alpha_k f_k'' \mathbf{n}_i dS. \quad (1.151)$$

These formulas are extensible to vectors by transforming gradient in a divergence. There is no general agreement on which form of Equation (1.150) or (1.151) should be applied for each gradient and divergence that appears in the governing equations. It depends on the correlation used for the closure laws. Special attention must be taken when using correlations from the literature, minding

to how the authors define their correlation function. Using Equation (1.149) and (1.150), the space average of the convective term can be written:

$$\begin{aligned} \left\langle \frac{\partial \alpha \bar{\rho}_k \widehat{\psi}_k}{\partial t} \right\rangle + \left\langle \nabla \cdot (\alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k) \right\rangle &= \frac{\partial \langle \alpha \bar{\rho}_k \widehat{\psi}_k \rangle}{\partial t} + \nabla \cdot \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k \rangle \\ &+ \int_{S_i} \alpha_k \bar{\rho}_k (\widehat{\mathbf{u}}_k - \mathbf{v}_i) dS. \end{aligned} \quad (1.152)$$

The solids are still in the medium, and the fluid does not penetrate the solid. Hence, for all instant t and for $\mathbf{x} \in S_i$, $\mathbf{v}_i(t) = \mathbf{0}$ and $\widehat{\mathbf{u}}_k(\mathbf{x}, t) = \mathbf{0}$ so that Equation (1.152) becomes

$$\left\langle \frac{\partial \alpha \bar{\rho}_k \widehat{\psi}_k}{\partial t} \right\rangle + \left\langle \nabla \cdot (\alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k) \right\rangle = \frac{\partial \langle \alpha \bar{\rho}_k \widehat{\psi}_k \rangle}{\partial t} + \nabla \cdot \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k \rangle. \quad (1.153)$$

The following methodology is used on Equation (1.135):

- Space-averaging the equation:

$$\left\langle \frac{\partial}{\partial t} (\alpha_k \bar{\rho}_k \widehat{\psi}_k) \right\rangle + \left\langle \nabla \cdot (\alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k) \right\rangle + \left\langle \nabla \cdot (\alpha_k (\bar{\mathbf{f}}_k + \mathbf{f}_k^t)) \right\rangle - \langle \alpha_k \bar{\rho}_k \widehat{\chi}_k \rangle = \langle \Psi_k^i \rangle. \quad (1.154)$$

- Using (1.153) and (1.150), which involves source terms due to solid:

$$\frac{\partial}{\partial t} \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \rangle + \nabla \cdot \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k \rangle + \nabla \cdot \langle \alpha_k (\bar{\mathbf{f}}_k + \mathbf{f}_k^t) \rangle - \langle \alpha_k \bar{\rho}_k \widehat{\chi}_k \rangle = \langle \Psi_k^i \rangle + \Psi_k^s, \quad (1.155)$$

with

$$\Psi_k^s = \int_{S_i} (\alpha_k \bar{\mathbf{f}}_k + \mathbf{f}_k^t) \cdot \mathbf{n}_i dS. \quad (1.156)$$

- Transforming space average on volume V_0 to fluid space average on volume V_f by using Equation (1.26):

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \rangle) + \nabla \cdot (\varepsilon \langle \alpha_k \bar{\rho}_k \widehat{\psi}_k \widehat{\mathbf{u}}_k \rangle) \\ + \nabla \cdot (\varepsilon \langle \alpha_k (\bar{\mathbf{f}}_k + \mathbf{f}_k^t) \rangle) - \varepsilon \langle \alpha_k \bar{\rho}_k \widehat{\chi}_k \rangle = \varepsilon \langle \Psi_k^i \rangle + \Psi_k^s. \end{aligned} \quad (1.157)$$

- Using identity from Equation (1.147) and weighted average:

$$\begin{aligned} \frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \psi_k}) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \psi_k \mathbf{u}_k}) \\ + \nabla \cdot (\varepsilon \langle \alpha_k \rangle (\widetilde{\mathbf{f}}_k + \widetilde{\mathbf{f}}_k^t + \mathbf{f}_k^{st})) - \varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \chi_k} = \varepsilon \langle \Psi_k^i \rangle + \Psi_k^s. \end{aligned} \quad (1.158)$$

This methodology is applied to Equations (1.136), (1.138) and (1.144). For spatial gradient and divergence, Equation (1.150) is mostly used. An exception is made for the treatment of pressure in the momentum balance where Equation (1.151) is used. This must be taken into account for modeling the source term (friction) due to the solid in this equation. This formulation allows for a pressure term that enables the recovery of hydrostatic pressure for a case with zero velocity. The space-averaged mass balance is

$$\frac{\partial}{\partial t} (\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k}) + \nabla \cdot (\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{u}_k}) = \varepsilon \langle \Gamma_k \rangle. \quad (1.159)$$

The space-averaged momentum balance is

$$\begin{aligned}
 \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{u}_k} \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{u}_k} \otimes \widetilde{\mathbf{u}_k} \right) + \varepsilon \langle \alpha_k \rangle \nabla \widetilde{p_k} = \\
 \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\mathbf{T}_k} + \widetilde{\mathbf{T}_k^t} + \mathbf{T}_k^{st} \right) \right) \\
 + \varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{g}} \\
 + \varepsilon \langle \mathbf{m}_k \rangle + \mathbf{m}_{ks},
 \end{aligned} \tag{1.160}$$

with $\mathbf{m}_{ks}$ the friction on phase k due to the solid and $\mathbf{T}_k^{st}$ the friction due to space turbulence such that

$$\mathbf{T}_k^{st} = - \frac{\langle \alpha_k \bar{\rho}_k \mathbf{u}_k'' \otimes \mathbf{u}_k'' \rangle}{\langle \alpha_k \rangle} = - \overline{\bar{\rho}_k \mathbf{u}_k'' \otimes \mathbf{u}_k''}. \tag{1.161}$$

Similarly to time-averaging, the spatial fluctuations of kinetic energy $\frac{\widetilde{(u'')^2}}{2}$ are neglected. The same manipulation (see Equation (1.142)) leads to the space-averaged energy balance

$$\begin{aligned}
 \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k} \left(\widetilde{e_k} + \frac{\widetilde{u_k^2}}{2} \right) \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k} \left(\widetilde{e_k} + \frac{\widetilde{u_k^2}}{2} \right) \widetilde{\mathbf{u}_k} \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{p_k \mathbf{u}_k} \right) = \\
 - \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\varphi_k} + \widetilde{\varphi_k^t} + \varphi_k^{st} \right) \right) \\
 + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\mathbf{T}_k} + \widetilde{\mathbf{T}_k^t} + \mathbf{T}_k^{st} \right) \cdot \widetilde{\mathbf{u}_k} \right) \\
 + \varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{g}} \cdot \widetilde{\mathbf{u}_k} + \varepsilon \langle \Xi_k \rangle + \Xi_{ks},
 \end{aligned} \tag{1.162}$$

with Ξ_{ks} the total energy given from the solid to phase k and φ_k^{st} the heat flux due to space turbulence such that

$$\varphi_k^{st} = \frac{\langle \alpha_k \bar{\rho}_k h_k'' \mathbf{u}_k'' \rangle}{\langle \alpha_k \rangle}. \tag{1.163}$$

Enthalpy equation

Equation (1.162) can be manipulated to obtain an equation on enthalpy. Equations (1.160) (noted E_{mass}^k) and (1.159) (noted $\mathbf{E}_{mom}^k$) are used to obtain the equation on kinetic energy with

$$(\mathbf{E}_{mom}^k - E_{mass}^k \widetilde{\mathbf{u}_k}) \cdot \widetilde{\mathbf{u}_k} + E_{mass}^k \frac{\widetilde{u_k^2}}{2}, \tag{1.164}$$

so that the equation for kinetic energy writes

$$\begin{aligned}
 \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k} \frac{\widetilde{u_k^2}}{2} \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho_k} \frac{\widetilde{u_k^2}}{2} \widetilde{\mathbf{u}_k} \right) + \varepsilon \langle \alpha_k \rangle \widetilde{\mathbf{u}_k} \cdot \nabla \widetilde{p_k} = \\
 \widetilde{\mathbf{u}_k} \cdot \left[\nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\mathbf{T}_k} + \widetilde{\mathbf{T}_k^t} + \mathbf{T}_k^{st} \right) \right) \right] \\
 + \varepsilon \langle \alpha_k \rangle \widetilde{\rho_k \mathbf{g}} \cdot \widetilde{\mathbf{u}_k} \\
 + \varepsilon \langle \mathbf{m}_k \rangle \cdot \widetilde{\mathbf{u}_k} - \langle \Gamma_k \rangle \frac{\widetilde{u_k^2}}{2} + \mathbf{m}_{ks} \cdot \widetilde{\mathbf{u}_k}.
 \end{aligned} \tag{1.165}$$

Subtracting Equation (1.165) from Equation (1.162), the internal energy equation writes

$$\begin{aligned}
 \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho}_k \widetilde{e}_k \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho}_k \widetilde{e}_k \widetilde{\mathbf{u}}_k \right) + \widetilde{p}_k \cdot \nabla \left(\varepsilon \langle \alpha_k \rangle \widetilde{\mathbf{u}}_k \right) = \\
 - \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\varphi}_k + \widetilde{\varphi}_k^t + \varphi_k^{st} \right) \right) \\
 + \varepsilon \langle \alpha_k \rangle \left(\widetilde{\mathbf{T}}_k + \widetilde{\mathbf{T}}_k^t + \mathbf{T}_k^{st} \right) : \nabla \widetilde{\mathbf{u}}_k \\
 + \varepsilon \left(\langle \Xi_k \rangle - \langle \mathbf{m}_k \rangle \cdot \widetilde{\mathbf{u}}_k + \langle \Gamma_k \rangle \frac{\widetilde{u}_k^2}{2} \right) \\
 + \Xi_{ks} - \mathbf{m}_{ks} \cdot \widetilde{\mathbf{u}}_k.
 \end{aligned} \tag{1.166}$$

Defining the averaged enthalpy of phase k as

$$\widetilde{h}_k = \widetilde{e}_k + \frac{\widetilde{p}_k}{\widetilde{\rho}_k}, \tag{1.167}$$

the enthalpy equation writes

$$\begin{aligned}
 \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho}_k \widetilde{h}_k \right) + \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \widetilde{\rho}_k \widetilde{h}_k \widetilde{\mathbf{u}}_k \right) = \frac{\partial}{\partial t} \left(\varepsilon \langle \alpha_k \rangle \widetilde{p}_k \right) + \varepsilon \langle \alpha_k \rangle \widetilde{\mathbf{u}}_k \cdot \nabla \widetilde{p}_k \\
 - \nabla \cdot \left(\varepsilon \langle \alpha_k \rangle \left(\widetilde{\varphi}_k + \widetilde{\varphi}_k^t + \varphi_k^{st} \right) \right) \\
 + \varepsilon \langle \alpha_k \rangle \left(\widetilde{\mathbf{T}}_k + \widetilde{\mathbf{T}}_k^t + \mathbf{T}_k^{st} \right) : \nabla \widetilde{\mathbf{u}}_k \\
 + \varepsilon \left(\langle \Xi_k \rangle - \langle \mathbf{m}_k \rangle \cdot \widetilde{\mathbf{u}}_k + \langle \Gamma_k \rangle \frac{\widetilde{u}_k^2}{2} \right) \\
 + \Xi_{ks} - \mathbf{m}_{ks} \cdot \widetilde{\mathbf{u}}_k.
 \end{aligned} \tag{1.168}$$

1.B Thermodynamic equilibrium - 3-equation model

Model (1.61) is a 4-equation model with a phase in thermodynamic disequilibrium. The model can be simplified by considering that the two phases are at saturation, when the regime becomes two-phase. In that case, the mass fraction is considered at thermodynamic equilibrium and is therefore no longer an unknown in the system. It can be expressed as a function of the other variables (h_m, p_m) such that

$$y(p_m, h_m) = \frac{h_m - \bar{h}_l(p_m)}{\bar{h}_g(p_m) - \bar{h}_l(p_m)}. \tag{1.169}$$

As a consequence, the disequilibrium Equation (1.60) is not solved anymore and the model, called

3-equation model from now on, is written (with the notations of Equations (1.61)) :

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t}(\varepsilon \rho_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m) = 0, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m \mathbf{u}_m \otimes \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y) \mathbf{u}_r \otimes \mathbf{u}_r) = \\ \quad - \varepsilon \nabla p_m + \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st})) \\ \quad + \varepsilon \rho_m \mathbf{g} + \mathbf{m}_s, \\ \frac{\partial}{\partial t}(\varepsilon \rho_m E_m) + \nabla \cdot (\varepsilon \rho_m E_m \mathbf{u}_m) + \nabla \cdot (\varepsilon \rho_m y(1-y)(E_g - E_l) \mathbf{u}_r) \\ \quad + \nabla \cdot \left(\varepsilon \left(p_m \left[\mathbf{u}_m + \rho_m y(1-y) \left(\frac{1}{\rho_g} - \frac{1}{\rho_l} \right) \mathbf{u}_r \right] \right) \right) = \\ \quad - \nabla \cdot (\varepsilon (\boldsymbol{\varphi}_m + \boldsymbol{\varphi}_m^t + \boldsymbol{\varphi}_m^{st})) \\ \quad + \nabla \cdot (\varepsilon (\mathbf{T}_m + \mathbf{T}_m^t + \mathbf{T}_m^{st}) \cdot \mathbf{u}_m) \\ \quad + \nabla \cdot \left(\varepsilon \rho_m y(1-y) \left(\frac{\mathbf{T}_g + \mathbf{T}_g^t + \mathbf{T}_g^{st}}{\rho_g} - \frac{\mathbf{T}_l + \mathbf{T}_l^t + \mathbf{T}_l^{st}}{\rho_l} \right) \cdot \mathbf{u}_r \right) \\ \quad + \varepsilon \rho_m \mathbf{u}_m \cdot \mathbf{g} + \Xi_s. \end{array} \right. \quad (1.170)$$

This 3-equation model is an extension of the HEM model [16] in a porous medium with a relative velocity between phases.

1.C Discussion on the Bestion correlation

1.C.1 Terminal velocity of a slug

The Bestion correlation can be derived from physical arguments about the terminal velocity of a gas bubble in a two-phase steady state flow without phase change and for low void fraction. The case considered is unidimensional and the following formulas are used:

$$\left\{ \begin{array}{l} j = \alpha u_g + (1 - \alpha) u_l, \\ u_{gj} = u_g - j = (1 - \alpha) u_r, \\ \delta = \frac{\rho_l}{\rho_g}. \end{array} \right. \quad (1.171)$$

Considering low void fractions, the frictional force from the two-phase flow on the gas is expressed as

$$F = -K(u_g - j), \quad (1.172)$$

with K a positive drag coefficient. For the steady-state, the momentum balance for each phase write

$$\begin{aligned} \alpha \partial_x p &= -\alpha \rho_g g - K(u_g - j), \\ (1 - \alpha) \partial_x p &= -(1 - \alpha) \rho_l g + K(u_g - j). \end{aligned} \quad (1.173)$$

By substituting the pressure gradient, the frictional force writes

$$F = -K u_{gj} = \alpha(1 - \alpha) \rho_g (\delta - 1) g. \quad (1.174)$$

The considered symmetric drag coefficient is

$$K \sim \frac{\alpha \rho_g (1 - \alpha) \rho_l |u_{gj}|}{\rho d}, \quad (1.175)$$

with d a characteristic length of the flow. For low void fraction, $\rho_l/\rho \sim 1$, such that

$$K \sim \alpha(1 - \alpha)\rho_g \frac{|u_{gj}|}{d}. \quad (1.176)$$

The drift velocity becomes

$$|u_{gj}| \sim \sqrt{(\delta - 1)gd}. \quad (1.177)$$

Considering the hydraulic diameter d_h as the characteristic length and using $u_{gj} = (1 - \alpha)u_r$, Bestion correlation is retrieved

$$u_r \sim \frac{1}{1 - \alpha} \sqrt{(\delta - 1)gd_h} \text{ where } \frac{1}{1 - \alpha} = \frac{1 + y(\delta - 1)}{1 - y}. \quad (1.178)$$

1.C.2 Approximated Bestion correlation

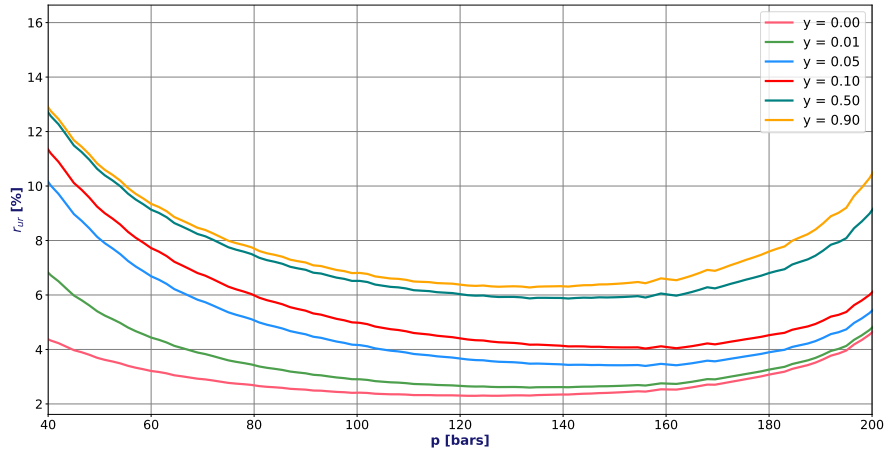
For a water-steam mixture, the pressures considered in reactor core applications are in the range $p \in [40, 200]$ bar. The pressure difference between the inlet and the outlet is approximately $\Delta p \sim 3$ bar for the 1300MWe/N4 PWR and $\Delta p \sim 2$ bar for the 900MWe PWR. The most constraining case is considered with $\Delta p = 3$ bar. To obtain some orders of magnitude, the value of the ratio δ is estimated by considering saturation $\delta(p) = \frac{\bar{\rho}_l(p)}{\bar{\rho}_g(p)}$. The density ratio varies in the range $\delta \in [3, 40]$ (data extracted from the thermodynamics of the industrial code THYC-coeur for a water-steam mixture at saturation). At a given pressure p and a given mass fraction y , the relative variation r_{ur} of the relative velocity for the Bestion correlation with a pressure variation $\Delta p = 3$ bar is computed such that

$$r_{ur}(p, y) = \frac{u_r(y, \delta(p + \frac{\Delta p}{2})) - u_r(y, \delta(p - \frac{\Delta p}{2}))}{u_r(y, \delta(p))} \quad [\%]. \quad (1.179)$$

Figure 1.3 shows the ratio r_{ur} as a function of the pressure for different mass fraction. It increases as the mass fraction increases but the error remains below 13% for any mass fraction or pressure considered. In more than 90% of considered case, the pressure satisfies $p \in [120, 160]$ bar and the mass fraction respects $y < 0.1$. Within these ranges, the relative variation never exceeds 4.5%. As a first approximation, it is possible to consider the relative velocity independent of the pressure. An approximate formulation of the Bestion correlation can then be proposed with δ_c being a constant chosen by the user according to the pressure of the case studied.

$$u_r^{app}(y) = \frac{1 + y(\delta_c - 1)}{1 - y} 0.188 \sqrt{gd_h(\delta_c - 1)}. \quad (1.180)$$

This approximated Bestion correlation, which depends only on the mass fraction y , allows the computation of an analytical solution for the disequilibrium equation to compare the accuracy of finite volume schemes (see **Chapter 3** - Section 3.1).

**FIGURE 1.3**

Relative variation of the relative velocity as a function of the pressure (for several mass fraction) for $\Delta p = 3$ bar - Bestion correlation.

Chapter 2

Hyperbolicity of the simplified models

2.1 Introduction

Hyperbolicity is a property that ensures the stability of solutions over time. If hyperbolicity is not guaranteed, numerical solutions may explode in finite time (see for example [50], or [67] - page 158, Figure 8.16/b for two-phase systems). In a hyperbolic system, waves propagate at finite real speeds. In this chapter, the objective is to study the hyperbolicity of the conservative model composed of Equations (1.61) proposed in **Chapter 1**. This model corresponds to a drift-flux two-phase flow model with a relative velocity u_r between the phases. The system of equations presented in Section 1.3.1 is considered in a 1D framework. Second-order differential terms (diffusion) and all zeroth-order differential source terms (solid friction, gravity, mass transfer, etc.) are not taken into account. Moreover, a free medium ($\varepsilon = 1$) is considered. The subscripts m for the mixture quantities are removed to improve readability. The studied model is therefore written as

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + \rho y(1-y)u_r^2 + p) = 0, \\ \partial_t(\rho E) + \partial_x(\rho E u + \rho y(1-y)(E_g - E_l)u_r) + \partial_x\left(p\left[u + \rho y(1-y)\left(\frac{1}{\rho_g} - \frac{1}{\rho_l}\right)u_r\right]\right) = 0, \\ \partial_t(\rho y) + \partial_x(\rho y u + \rho y(1-y)u_r) = 0, \end{cases} \quad (2.1)$$

with the total energy of the mixture defined by

$$E = yE_g + (1-y)E_l = e + \frac{u^2}{2} + y(1-y)\frac{u_r^2}{2}. \quad (2.2)$$

The relative quantities are noted with a subscript r and defined as:

$$\phi_r = \phi_g - \phi_l, \text{ with } \phi \in \{y, u, v, e\}. \quad (2.3)$$

An exception is made for pressure p_r defined in Equation (2.50).

An equation of state for the internal energy of the mixture is considered to close the system:

$$e = e(p, \rho, y). \quad (2.4)$$

The methodology for studying hyperbolicity is described in Section 2.2. Several systems of increasing complexity, derived from system of Equations (2.1), are considered. First, Equations (2.1) are studied without the energy equation in Section 2.3, assuming a barotropic relation between density and pressure. Then, the 4-equation model is considered in Section 2.4. First, the system is studied without relative velocity in Section 2.4.1. This system corresponds to an HRM system [16]. In Section 2.4.2, a constant relative velocity is introduced.

2.2 Methodology

A conservative system of equations can be written

$$\partial_t \mathbf{W} + \partial_x (\mathbf{F}(\mathbf{W})) = \mathbf{0}, \quad (2.5)$$

with $\mathbf{W}$ the vector of conservative quantities (for example $\mathbf{W} = (\rho, \rho u, \rho E, \rho y)$) and $\mathbf{F}(\mathbf{W})$ the conservative fluxes associated with the quantities $\mathbf{W}$. To study the hyperbolicity of the model, the system is expressed in a non-conservative form using the vector $\mathbf{Y}$:

$$\partial_t \mathbf{Y} + \mathbf{B}(\mathbf{Y}) \partial_x \mathbf{Y} = \mathbf{0}, \quad (2.6)$$

where $\mathbf{B}(\mathbf{Y})$ is the convective matrix. The system of equations (2.5) is **hyperbolic** if and only if the matrix $\mathbf{B}$ is diagonalizable with real eigenvalues. It is **strictly hyperbolic** if the eigenvalues are distinct. The characteristic polynomial of the matrix $\mathbf{B}$ is computed and defined as

$$P(\lambda) = \det(\mathbf{B}(\mathbf{Y}) - \lambda \mathbf{I}_d). \quad (2.7)$$

The polynomial $P(\lambda)$ is then examined to determine the eigenvalues λ satisfying

$$P(\lambda) = 0. \quad (2.8)$$

If an analytical computation of these roots is too complex, the intermediate value theorem can be applied to the polynomial P , which is a continuous function of λ , to ensure that the polynomial has real roots. To simplify the equations, the following notations are used in this chapter:

$$\begin{cases} \xi = y(1-y) & \in [0, \frac{1}{4}], \\ y_r = 2y - 1 & \in [-1, 1]. \end{cases} \quad (2.9)$$

2.3 Hyperbolicity of the barotropic drift-flux model without energy equation

Model of Equations (2.1) is considered without the total energy equation. The conservative unknowns are $(\rho, \rho u, \rho y)$. A barotropic relation is considered for density of each phase $\rho_k(p)$. We assume that the barotropic equation of state for each phase satisfies

$$\begin{cases} \frac{d\rho_g}{dp} = \frac{1}{c_g^2} > 0, \\ \frac{d\rho_l}{dp} = \frac{1}{c_l^2} > 0, \end{cases} \quad (2.10)$$

with $c_k, k \in \{l, g\}$ the speed of sound in phase k , assumed to be real. The density of the mixture is defined by

$$\frac{1}{\rho(p, y)} = \frac{y}{\rho_g(p)} + \frac{1-y}{\rho_l(p)}. \quad (2.11)$$

Using speeds of sound within each phase, the differential for mixture density is

$$d\rho = \frac{1}{c^2} dp + m dy, \quad (2.12)$$

with

$$\begin{cases} \frac{1}{c^2} = \frac{y}{c_g^2} \left(\frac{\rho}{\rho_g} \right)^2 + \frac{1-y}{c_l^2} \left(\frac{\rho}{\rho_l} \right)^2, \text{ where } c \text{ is the real mixture speed of sound,} \\ m = -\frac{v_g - v_l}{v^2}, \text{ where } v_k = \frac{1}{\rho_k} \text{ and } v = yv_g + (1-y)v_l. \end{cases} \quad (2.13)$$

The system of equations can be written

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + \rho \xi u_r^2 + p(\rho, y)) = 0, \\ \partial_t(\rho y) + \partial_x(\rho y u + \rho \xi u_r) = 0, \end{cases} \quad (2.14)$$

with $u_r(y)$ a relative velocity depending upon the mass fraction. It is rewritten in a non-conservative form, highlighting the terms due to the relative velocity in blue

$$\begin{cases} \partial_t \rho + u \partial_x \rho + \rho \partial_x u = 0, \\ \partial_t u + u \partial_x u + v \partial_x(\rho \xi u_r^2) + v \partial_x(p(\rho, y)) = 0, \\ \partial_t y + u \partial_x y + v \partial_x(\rho \xi u_r) = 0. \end{cases} \quad (2.15)$$

Using Equations (2.14), the system can be written in a matrix form, with $\mathbf{Y} = (\rho, u, y)$:

$$\partial_t \mathbf{Y} + \mathbf{B}(\mathbf{Y}) \partial_x \mathbf{Y} = 0, \quad (2.16)$$

with

$$\mathbf{B} = \begin{pmatrix} u & \rho & 0 \\ v(c^2 + \xi u_r^2) & u & \partial_y(\xi u_r^2) - v m c^2 \\ v \xi u_r & 0 & u + \partial_y(\xi u_r) \end{pmatrix}. \quad (2.17)$$

The characteristic polynomial is defined by

$$P(\lambda) = \det(\mathbf{B} - \lambda \mathbf{I}_d). \quad (2.18)$$

Using the notation

$$x = u - \lambda, \quad (2.19)$$

the polynomial writes

$$P(x) = x^3 + \partial_y(\xi u_r) x^2 - (c^2 + \xi u_r^2) x - (\partial_y(\xi u_r) + v m \xi u_r) c^2 + \xi^2 u_r^2 \partial_y u_r. \quad (2.20)$$

When considering a constant relative velocity

$$u_r = u_{r0}, \quad (2.21)$$

the following relations can be used

$$\begin{cases} \partial_y u_r = 0, \\ \partial_y(\xi u_r) = -y_r u_{r0}, \end{cases} \quad (2.22)$$

with y_r defined in Equation (2.3). The characteristic polynomial becomes

$$P(x, u_r) = x^3 - y_r u_{r0} x^2 - (c^2 + \xi u_{r0}^2) x - (v m \xi - y_r) u_{r0} c^2. \quad (2.23)$$

The polynomial P satisfies:

$$P(-x, -u_{r0}) = P(x, u_{r0}). \quad (2.24)$$

For $u_r = u_{r0} > 0$, if the polynomial has three real roots (x_1, x_2, x_3) , then for $u_r = -u_{r0} < 0$ the quantities $(-x_1, -x_2, -x_3)$ are real roots of the characteristic polynomial. This allows us to study only the case $u_{r0} > 0$. We first focus on cases where

$$\xi \neq 0 \Leftrightarrow y \neq 0 \text{ or } y \neq 1. \quad (2.25)$$

The case $\xi = 0$ is treated later. The velocities of each phase $u_g = u + (1 - y)u_{r0}$ and $u_l = u - yu_g$ are considered. We note

$$\begin{cases} x_g = u - u_g = -(1 - y)u_{r0} < 0, \\ x_l = u - u_l = yu_{r0} > 0, \end{cases} \quad (2.26)$$

with $x_g \neq x_l$ because $u_{r0} \neq 0$. The polynomial is evaluated for x_g and x_l :

$$\begin{cases} P(x = x_g) = u_{r0}c^2y(1 - mv(1 - y)) \\ \quad = \frac{v_g}{v}yu_{r0}c^2 > 0, \\ P(x = x_l) = -u_{r0}c^2(1 - y)(1 + mvy) \\ \quad = -\frac{v_l}{v}(1 - y)u_{r0}c^2 < 0. \end{cases} \quad (2.27)$$

It is interesting to note that, using the definition of the void fraction,

$$\alpha = \frac{yv_g}{v} \in [0, 1], \quad (2.28)$$

we can write

$$\begin{cases} P(x = x_g) = \alpha u_{r0}c^2 > 0, \\ P(x = x_l) = -(1 - \alpha)u_{r0}c^2 < 0. \end{cases} \quad (2.29)$$

Using

$$\begin{cases} P(x \rightarrow -\infty) = -\infty, \\ P(x = x_g \leq 0) > 0, \\ P(x = x_l \geq 0) < 0, \\ P(x \rightarrow +\infty) = +\infty, \end{cases} \quad (2.30)$$

and the intermediate value theorem, the polynomial P admit three distinct real roots (for $u_{r0} \neq 0$). The barotropic drift-flux system with a constant relative velocity is **strictly hyperbolic** when $u_{r0} \neq 0$ and $\xi \neq 0$.

Case $\xi = 0$: The characteristic polynomial from Equation (2.23) can be written:

$$P(x) = (x - y_ru_{r0})(x^2 - c^2) - \xi(u_{r0}^2x + vmc^2u_{r0}). \quad (2.31)$$

When $\xi = 0$, the characteristic polynomial has three roots $(u - c, u - y_ru_{r0}, u + c)$. The system is also **hyperbolic**.

2.4 Hyperbolicity of the 4-equation drift-flux model

In this Section, the 4-equation model is studied. Considering the equation of state for internal energy from Equation (2.4), the differential for internal energy writes

$$de = adp + bd\rho + mdy, \quad (2.32)$$

with

$$\begin{cases} a = \left(\frac{\partial e}{\partial p} \right)_{\rho, y}, \\ b = \left(\frac{\partial e}{\partial \rho} \right)_{p, y}, \\ m = \left(\frac{\partial e}{\partial y} \right)_{p, \rho}. \end{cases} \quad (2.33)$$

The mixture speed of sound is defined by

$$c^2 = a^{-1}(pv^2 - b), \quad \text{with } v = \frac{1}{\rho}. \quad (2.34)$$

It is assumed that the equation of state ensures a real speed of sound c for the mixture. Each phase k is characterized by an equation of state $e_k(p, \rho_k)$, such that

$$\begin{cases} de_k = a_k dp + b_k d\rho_k, \\ c_k^2 = a_k^{-1}(pv_k^2 - b_k), \end{cases} \quad (2.35)$$

with c_k the speed of sound in phase k . By considering the gas phase at saturation during this work, the thermodynamic coefficients write:

$$\begin{cases} a = ya_g + (1-y)a_l + y \frac{dv_g}{dp} (a_g \rho_g^2 c_g^2 - a_l \rho_l^2 c_l^2), \\ b = \left(\frac{\rho_l}{\rho} \right)^2 b_l, \\ m = L - \rho^2 a c^2 v_r, \\ c^2 = \left(\frac{\rho_l}{\rho} \right)^2 \cdot \frac{a_l}{a} c_l^2. \end{cases} \quad (2.36)$$

In the following, the model is first studied when the relative velocity is zero in Section 2.4.1. Then a constant relative velocity $u_r = u_{r0}$ is considered in Section 2.4.2.

2.4.1 Four-equation model with null relative velocity

System (2.1) is considered now without relative velocity. The obtained system is a HRM Model [16]:

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, \\ \partial_t(\rho E) + \partial_x(\rho E u) + \partial_x(pu) = 0, \\ \partial_t(\rho y) + \partial_x(\rho y u) = 0. \end{cases} \quad (2.37)$$

The relative velocity is null so that the mixture total energy writes

$$E = e + \frac{u^2}{2}. \quad (2.38)$$

The equation for kinetic energy is obtained by multiplying the momentum balance equation in its non-conservative form by u . It is then subtracted from the total energy equation to derive the equation on internal energy:

$$\partial_t(\rho e) + \partial_x(\rho e u) + p \partial_x u = 0. \quad (2.39)$$

Using the linearization of internal energy in Equation (2.32) and the speed of sound in Equation (2.34), the pressure equation is obtained

$$\partial_t p + u \partial_x p + \rho c^2 \partial_x u = 0. \quad (2.40)$$

Using variables $\mathbf{Y} = (\rho, u, p, y)$, the non-conservative system writes

$$\partial_t \mathbf{Y} + B(\mathbf{Y}) \partial_x \mathbf{Y} = 0, \quad (2.41)$$

with

$$B = \begin{pmatrix} u & \rho & 0 & 0 \\ 0 & u & v & 0 \\ 0 & \rho c^2 & u & 0 \\ 0 & 0 & 0 & u \end{pmatrix}. \quad (2.42)$$

The eigenvalues λ can be easily computed

$$\lambda \in \{u, u, u + c, u - c\}. \quad (2.43)$$

The eigenvalue u is a double eigenvalue and spans a two-dimensional vector space with the eigenvectors $r_1 = (1, 0, 0, 0)$ and $r_2 = (0, 0, 0, 1)$. The system is therefore **hyperbolic**.

2.4.2 Four-equation model with constant relative velocity

System (2.1) is now considered with a constant non-zero relative velocity $u_r = u_{r0} \neq 0$. The system writes, with the terms due to the relative velocity highlighted in blue,

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, & (a) \\ \partial_t(\rho u) + \partial_x(\rho u^2 + \rho \xi u_{r0}^2 + p) = 0, & (b) \\ \partial_t(\rho E) + \partial_x(\rho E u + \rho \xi (E_g - E_l) u_{r0}) + \partial_x\left(p \left[u + \rho \xi (v_g - v_l) u_{r0}\right]\right) = 0, & (c) \\ \partial_t(\rho y) + \partial_x(\rho y u + \rho \xi u_{r0}) = 0, & (d) \end{cases} \quad (2.44)$$

with

$$\begin{cases} E = y e_g + (1 - y) e_l + y \frac{u_g^2}{2} + (1 - y) \frac{u_l^2}{2}, \\ E_g - E_l = e_g - e_l + \frac{u_g^2}{2} - \frac{u_l^2}{2}. \end{cases} \quad (2.45)$$

Using

$$\begin{cases} u_g = u + (1 - y)u_{r0}, \\ u_l = u - yu_{r0}, \end{cases} \quad (2.46)$$

we have

$$\begin{cases} E = e + \frac{u^2}{2} + \xi \frac{u_{r0}^2}{2}, \\ E_g - E_l = e_r + uu_{r0} + \frac{1 - 2y}{2} u_{r0}^2, \end{cases} \quad (2.47)$$

with e_r defined in Equation (2.3). The latent heat $L = h_g - h_l$ appears when using:

$$E_g - E_l + p(v_g - v_l) = L + uu_{r0} + \frac{1 - 2y}{2} u_{r0}^2. \quad (2.48)$$

The system of equations is transformed to more easily obtain the characteristic polynomial. We recall the notations used:

$$\begin{cases} y_r = 2y - 1, \\ \xi = y(1 - y). \end{cases} \quad (2.49)$$

The following new notations are introduced:

$$\begin{cases} p_r = \rho \xi u_{r0}^2, \\ \tilde{\rho} = \rho \xi, \\ \tilde{p} = p + p_r. \end{cases} \quad (2.50)$$

Using the notations from Equations (2.50), Equations (2.44) write in a non conservative way

$$\begin{cases} \partial_t \rho + \rho \partial_x u + u \partial_x \rho = 0, & (a) \\ \rho \partial_t u + \rho u \partial_x u + \partial_x \tilde{p} = 0, & (b) \\ \rho \partial_t E + \rho u \partial_x E + \partial_x (\tilde{\rho} L u_{r0}) + \partial_x (\tilde{p} u) - \partial_x \left(\tilde{\rho} \frac{y_r}{2} u_{r0}^3 \right) = 0, & (c) \\ \rho \partial_t y + \rho u \partial_x y + \partial_x (\rho \xi u_{r0}) = 0. & (d) \end{cases} \quad (2.51)$$

Non conservative system in $(\tilde{\rho}, u, \tilde{p}, y)$

The system is written in the new set of variables: $(\tilde{\rho}, u, \tilde{p}, y)$, with $\tilde{\rho}$ the modified density and $\tilde{p}$ the modified pressure, as defined in Equations (2.50). The computations are detailed in **Appendix 2.A.1**. During the manipulations, the equation on internal energy is obtained:

$$\rho \left(\partial_t e + u \partial_x e \right) + \partial_x (\tilde{\rho} u_{r0} L) + \tilde{p} \partial_x u - \tilde{\rho} u_{r0} \partial_x \left(\frac{y_r u_{r0}^2}{2} \right) = 0. \quad (2.52)$$

The final system in non-conservative form using the variables $(\tilde{\rho}, u, \tilde{p}, y)$ writes:

$$\begin{cases} \partial_t \tilde{\rho} + (u - y_r u_{r0}) \partial_x \tilde{\rho} + \tilde{\rho} \partial_x u = 0, \\ \partial_t u + u \partial_x u + v \partial_x \tilde{p} = 0, \\ \partial_t \tilde{p} + u \partial_x \tilde{p} + (\rho c^2 + A_u u_{r0}) \partial_x u + A_{\tilde{p}} u_{r0} \partial_x \tilde{\rho} + A_y u_{r0} \partial_x y + a^{-1} \xi u_{r0} \partial_x \tilde{L}(\tilde{\rho}, \tilde{p}, y) = 0, \\ \partial_t y + u \partial_x y + v u_{r0} \partial_x \tilde{\rho} = 0, \end{cases} \quad (2.53)$$

with

$$\begin{cases} A_{\tilde{\rho}} = a^{-1} v (L - m) - y_r u_{r0}^2, \\ A_u = \rho \xi u_{r0} (1 + v a^{-1}), \\ A_y = -\xi u_{r0}^2 a^{-1}. \end{cases} \quad (2.54)$$

The latent heat $\tilde{L}$ is written here as a function of the variables $(\tilde{\rho}, \tilde{p}, y)$. Its derivatives with respect to these variables are detailed in **Appendix 2.A.2**. In the following, the notation used for partial derivatives of any function f is:

$$f_x(x, y, z) = \left(\frac{\partial f}{\partial x} \right)_{y,z} (x, y, z). \quad (2.55)$$

Equations (2.53) can be written in a matrix formulation:

$$\frac{\partial \mathbf{Y}}{\partial t} + \mathbf{B}(\mathbf{Y}) \frac{\partial \mathbf{Y}}{\partial x} = 0, \quad \mathbf{Y} = \begin{pmatrix} \tilde{\rho} \\ u \\ \tilde{p} \\ y \end{pmatrix}, \quad (2.56)$$

with

$$\mathbf{B}(\mathbf{Y}) = \begin{pmatrix} u - y_r u_{r0} & \rho \xi & 0 & 0 \\ 0 & u & v & 0 \\ A_{\tilde{\rho}} u_{r0} + a^{-1} \xi u_{r0} \tilde{L}_{\tilde{\rho}} & \rho c^2 + A_u u_{r0} & u + a^{-1} \xi u_{r0} \tilde{L}_{\tilde{p}} & A_y u_{r0} + a^{-1} \xi u_{r0} \tilde{L}_y \\ v u_{r0} & 0 & 0 & u \end{pmatrix}. \quad (2.57)$$

The characteristic polynomial of matrix $\mathbf{B}(\mathbf{Y})$ is computed in the following to study the eigenvalues of the system.

Characteristic polynomial

The following notations are used in the following:

$$\begin{cases} g = a^{-1} v, \\ \beta = 1 + g, \\ G = g(L - m) = \rho c^2 v_r, \text{ with } v_r \text{ from Equation (2.3),} \\ F = a^{-1} L_p = \frac{a^{-1} \frac{dh_g}{dp} - \beta}{1 - y}. \end{cases} \quad (2.58)$$

We are looking for the eigenvalues λ of the matrix $\mathbf{B}(\mathbf{Y})$. By setting $x = u - \lambda$, the characteristic polynomial of the matrix can be written as:

$$P(x = u - \lambda) = \det(B(\mathbf{W}) - \lambda \mathbf{I}_d) = x^4 + Ax^3 + Bx^2 + Cx + D, \quad (2.59)$$

with

$$\begin{cases} A = (F\xi - y_r) u_{r0}, \\ B = -(c^2 + (\beta + F y_r) \xi u_{r0}^2), \\ C = (G\xi + (3y - 1)c^2) u_{r0} + (g y_r - F\xi) \xi u_{r0}^3, \\ D = g\xi^2 u_{r0}^4 - (G\xi + c^2 y_r) y u_{r0}^2. \end{cases} \quad (2.60)$$

The calculation of the polynomial $P(x)$ is detailed in **Appendix 2.A.3**. As expected, we retrieve the eigenvalues $\lambda \in \{u + c, u - c, u, u\}$ when $u_{r0} = 0$. Similarly to Section 2.3, the polynomial $P(x)$ satisfies the following property when $u_{r0} \neq 0$:

$$P(-x, -u_{r0}) = P(x, u_{r0}). \quad (2.61)$$

This allows us to study only the case where $u_{r0} > 0$. We evaluate the characteristic polynomial for the velocities of each phase u_g and u_l . We again use the notations:

$$\begin{cases} x_g = u - u_g = -(1 - y)u_{r0} & \leq 0, \\ x_l = u - u_l = yu_{r0} & \geq 0. \end{cases} \quad (2.62)$$

The calculations are detailed in **Appendix 2.A.3**. We obtain:

$$\begin{cases} P(x = x_g = -(1 - y)u_{r0} \leq 0) = -\rho_g y c^2 u_{r0}^2 \leq 0, \\ P(x = x_l = yu_{r0} \geq 0) = 0, \end{cases} \quad (2.63)$$

with $x_g \neq x_l$ as $u_{r0} \neq 0$. Using the following properties when considering $y \neq 0$,

$$\begin{aligned} (1) \quad & P(x \rightarrow -\infty) = +\infty, \\ (2) \quad & P(x = x_g \leq 0) = -\rho v_g(p) y c^2 u_{r0}^2 < 0, \\ (3) \quad & P(x = x_l \geq 0) = 0, \\ (4) \quad & P(x \rightarrow +\infty) = +\infty, \end{aligned} \quad (2.64)$$

the intermediate value theorem allows us to assert that when the derivative of the polynomial P satisfies

$$P'(x = x_l) < 0, \quad (2.65)$$

it is sufficient for the characteristic polynomial to have four distinct real roots. The derivative of polynomial P in $x = x_l$ writes

$$P'(x_l) = C_1^h u_{r0}^3 - C_2^h c^2 u_{r0}, \quad (2.66)$$

with

$$\begin{cases} C_1^h(p, \rho, y) = y a^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = (1 - y) \rho v_l = (1 - \rho y v_g(p)) > 0. \end{cases} \quad (2.67)$$

If the thermodynamic function C_1^h satisfies

$$C_1^h(p, \rho, y) < 0, \quad (2.68)$$

hyperbolicity is ensured for any constant relative velocity u_{r0} . If this condition is not satisfied, the constant relative velocity u_{r0} should satisfy:

$$|u_{r0}| \leq u_r^{max}(p, \rho, y) = \sqrt{\frac{C_2^h(p, \rho, y)}{C_1^h(p, \rho, y)}} c(p, \rho, y). \quad (2.69)$$

Sufficient condition for hyperbolicity: The drift-flux system of Equations (2.44) with a constant relative velocity $u_{r0} \neq 0$ is **strictly hyperbolic** if

$$C_1^h(p, \rho, y) < 0 \quad \text{or} \quad |u_{r0}| < u_r^{max}(p, \rho, y) = \sqrt{\frac{C_2^h(p, \rho, y)}{C_1^h(p, \rho, y)}} c(p, \rho, y), \quad (2.70)$$

with

$$\begin{cases} C_1^h(p, \rho, y) = y a^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = 1 - \rho y v_g(p) = 1 - \alpha, \end{cases} \quad (2.71)$$

where α is the void fraction.

When the first condition is not met, the maximum relative speed u_r^{max} tends towards zero as the mass fraction approaches one, i.e. when the liquid phase disappear. Here, this result is obtained with the vapor phase at saturation. In practice, it is the liquid phase that is considered at saturation when mass fraction approaches one (and another condition for hyperbolicity would be obtained). The sufficient condition for hyperbolicity is discussed more broadly in **Appendix 2.B** when considering a mixture of water and vapor at 155 bar.

Case $y = 0$: When considering the case where $y = 0$, the characteristic polynomial from Equation (2.59) writes:

$$P(x) = x(x + u_{r0})(x^2 - c^2). \quad (2.72)$$

In that case, the system is **hyperbolic**, with four distinct real roots $\lambda \in \{u - c, u + u_{r0}, u + c\}$.

2.A Detailed computations for the hyperbolicity of the 4-equation model with a constant relative velocity

In this appendix, the detailed calculations for the hyperbolicity of the drift-flux 4-equation model with a constant relative velocity $u_{r0} \neq 0$ are presented. First, the manipulations of Equations (2.44) to obtain a model in the variables $(\tilde{\rho}, u, \tilde{p}, y)$ are shown. Then, a discussion is carried out on latent heat and its derivative. Finally, the detailed computation of the characteristic polynomial of the Jacobian matrix of the fluxes is provided. As a reminder, the following notations are used:

$$\begin{cases} y_r = 2y - 1, \\ \xi = y(1 - y), \\ p_r = \rho \xi u_{r0}^2, \\ \tilde{\rho} = \rho \xi, \\ \tilde{p} = p + p_r. \end{cases} \quad (2.73)$$

2.A.1 Computation to obtain the model in variables $(\tilde{\rho}, u, \tilde{p}, y)$

The steps to go from the system of Equations (2.51) in (ρ, u, E, y) to the system in $(\tilde{\rho}, u, \tilde{p}, y)$ are described.

- **Internal energy equation:** The internal energy equation is obtained using the same mechanism as in Section 2.4.1 with some difference due to the relative velocity. First, the kinetic energy equation is calculated using the momentum balance from Equation (2.51)-(b) and by multiplying it by u :

$$\partial_t \left(\rho \frac{u^2}{2} \right) + \partial_x \left(\rho u \frac{u^2}{2} \right) + u \partial_x (\tilde{p}) = 0. \quad (2.74)$$

Equation (2.74) is subtracted from the total energy Equation (2.51)-(c). An equation on the quantity $e + \xi u_{r0}^2/2$ is obtained:

$$\rho \left(\partial_t (e + \xi u_{r0}^2/2) + u \partial_x (e + \xi u_{r0}^2/2) \right) + \partial_x (\rho \xi u_{r0} L) + \tilde{p} \partial_x u - \partial_x \left(\rho \xi u_{r0} \frac{y_r u_{r0}^2}{2} \right) = 0. \quad (2.75)$$

Assuming that relative velocity is constant, an equation on $\xi u_{r0}^2/2$ is obtained from Equation (2.51)-(d) on mass fraction y multiplied by $(2y - 1)u_{r0}^2$:

$$\rho \left(\partial_t (\xi u_{r0}^2/2) + u \partial_x (\xi u_{r0}^2/2) \right) - \frac{y_r u_{r0}^2}{2} \partial_x (\rho \xi u_{r0}) = 0. \quad (2.76)$$

Subtracting Equation (2.76) from Equation (2.75), the internal energy equation is

$$\rho \left(\partial_t e + u \partial_x e \right) + \partial_x (\tilde{\rho} u_{r0} L) + \tilde{p} \partial_x u - \tilde{\rho} u_{r0} \partial_x \left(\frac{y_r u_{r0}^2}{2} \right) = 0. \quad (2.77)$$

- **Equation on modified pressure $\tilde{p}$:** The internal energy Equation (2.77) is transformed into an equation on the pressure p thanks to the linearization from Equation (2.32). Subtracting Equation (2.51)-(a) multiplied by the thermodynamic coefficient b and Equation (2.51)-(d) by thermodynamic coefficient m from the internal energy equation, the pressure equation writes

$$\begin{aligned} \partial_t p + u \partial_x p + \rho a^{-1} (\tilde{p} v^2 - b) \partial_x u \\ - a^{-1} \xi u_{r0}^3 \partial_x y \\ + a^{-1} v (L - m) u_{r0} \partial_x \tilde{p} \\ + a^{-1} \xi u_{r0} \partial_x L = 0. \end{aligned} \quad (2.78)$$

Using Equation (2.76) in its conservative form gives an equation on the relative pressure p_r :

$$\partial_t p_r + (u - y_r u_{r0}) \partial_x p_r + p_r \partial_x u. \quad (2.79)$$

The equation on the modified pressure can be written by summing Equations (2.78) and (2.79):

$$\begin{aligned} \partial_t \tilde{p} + u \partial_x \tilde{p} + (\rho c^2 + A_u u_{r0}) \partial_x u \\ + A_{\tilde{p}} u_{r0} \partial_x \tilde{p} \\ + A_y u_{r0} \partial_x y \\ + a^{-1} \xi u_{r0} \partial_x L = 0, \end{aligned} \quad (2.80)$$

with

$$\begin{cases} A_{\tilde{p}} = a^{-1} v (L - m) - y_r u_{r0}^2, \\ A_u = \rho \xi u_{r0} (1 + v a^{-1}), \\ A_y = -\xi u_{r0}^2 a^{-1}. \end{cases} \quad (2.81)$$

- **Equation on modified density $\tilde{\rho}$:** The equation on modified density $\tilde{\rho} = \rho \xi$ is obtained by summing Equation (2.51)-(a) on density ρ multiplied by ξ and Equation (2.51)-(d) on mass fraction y multiplied by $(1 - 2y)$:

$$\partial_t \tilde{\rho} + (u - y_r u_{r0}) \partial_x \tilde{\rho} + \tilde{\rho} \partial_x u = 0. \quad (2.82)$$

Considering Equation (2.82) on modified density $\tilde{\rho}$, Equation (2.80) on modified pressure $\tilde{p}$, Equation (2.51)-(b) on velocity and Equation (2.51)-(d) on mass fraction, the system in variables $(\tilde{\rho}, u, \tilde{p}, y)$ is obtained

$$\begin{cases} \partial_t \tilde{\rho} + (u - y_r u_{r0}) \partial_x \tilde{\rho} + \tilde{\rho} \partial_x u = 0, \\ \partial_t u + u \partial_x u + v \partial_x \tilde{p} = 0, \\ \partial_t \tilde{p} + u \partial_x \tilde{p} + (\rho c^2 + A_u u_{r0}) \partial_x u + A_{\tilde{p}} u_{r0} \partial_x \tilde{p} + A_y u_{r0} \partial_x y + a^{-1} \xi u_{r0} \partial_x \tilde{L}(\tilde{\rho}, \tilde{p}, y) = 0, \\ \partial_t y + u \partial_x y + v u_{r0} \partial_x \tilde{\rho} = 0, \end{cases} \quad (2.83)$$

with

$$\begin{cases} A_{\tilde{\rho}} = a^{-1} v (L - m) - y_r u_{r0}^2, \\ A_u = \rho \xi u_{r0} (1 + v a^{-1}), \\ A_y = -\xi u_{r0}^2 a^{-1}, \\ \tilde{L}(\tilde{\rho}, \tilde{p}, y) = L(\rho = \tilde{\rho}/\xi, p = \tilde{p} - \tilde{\rho} u_{r0}^2, y = y). \end{cases} \quad (2.84)$$

Here the latent heat $\tilde{L}$ is expressed in variables $(\tilde{\rho}, \tilde{p}, y)$. It is discussed below.

2.A.2 Latent heat

Latent heat $L = h_g(p) - h_l(p, h, y)$ appears in the system when $u_{r0} \neq 0$. The following notation is used for partial derivative of any function f depending upon (x, y, z) :

$$f_x(x, y, z) = \left(\frac{\partial f}{\partial x} \right)_{y,z}(x, y, z). \quad (2.85)$$

The differential on enthalpy is obtained from the linearization on internal energy (see Equation (2.32)):

$$dh = (a + v)dp - ac^2d\rho + mdy. \quad (2.86)$$

Starting from

$$h = h_g(p) + (1 - y)L(p, \rho, y), \quad (2.87)$$

the derivatives for the latent heat $L(p, \rho, y)$ are:

$$\begin{cases} \left(\frac{\partial L}{\partial p} \right)_{\rho, y} = L_p = \frac{\frac{dh_g}{dp} - (a + v)}{1 - y}, \\ \left(\frac{\partial L}{\partial \rho} \right)_{p, y} = L_\rho = \frac{ac^2}{1 - y}, \\ \left(\frac{\partial L}{\partial y} \right)_{\rho, p} = L_y = \frac{L - m}{1 - y} = \frac{\rho^2 ac^2 v_r}{1 - y}. \end{cases} \quad (2.88)$$

The latent heat is considered as a function of $(\tilde{\rho}, \tilde{p}, y)$:

$$\tilde{L}(\tilde{\rho}, \tilde{p}, y) = L(\rho = \tilde{\rho}/\xi, p = \tilde{p} - \tilde{\rho}u_{r0}^2, y = y). \quad (2.89)$$

The derivative of $\tilde{L}$ are

$$\begin{cases} \tilde{L}_{\tilde{\rho}} = \frac{1}{\xi}L_\rho - u_{r0}^2L_p, \\ \tilde{L}_{\tilde{p}} = L_p, \\ \tilde{L}_y = L_y + \frac{\rho y_r}{\xi}L_\rho, \end{cases} \quad (2.90)$$

with L_p , L_ρ et L_y given by Equations (2.88).

2.A.3 Detailed computation of the characteristic polynomial

The characteristic polynomial, defined by

$$P(x = u - \lambda) = \det(B(\mathbf{W}) - \lambda \mathbf{I}_d), \quad (2.91)$$

writes for Equations (2.53):

$$P(x) = \begin{vmatrix} x - y_r u_{r0} & \rho\xi & 0 & 0 \\ 0 & x & v & 0 \\ A_{\tilde{\rho}} u_{r0} + a^{-1}\xi u_{r0} \tilde{L}_{\tilde{\rho}} & \rho c^2 + A_u u_{r0} & x + a^{-1}\xi u_{r0} \tilde{L}_{\tilde{p}} & A_y u_{r0} + a^{-1}\xi u_{r0} \tilde{L}_y \\ v u_{r0} & 0 & 0 & x \end{vmatrix}, \quad (2.92)$$

with, using notations of Equations (2.58),

$$\begin{cases} A_{\tilde{\rho}} = G - y_r u_{r0}^2, \\ A_u = \rho\xi\beta u_{r0}, \\ A_y = -\rho\xi g u_{r0}^2. \end{cases} \quad (2.93)$$

Using the Leibniz formula for determinants, the determinant is developed along the last line:

$$\begin{aligned} P(x) = & -v u_{r0} \begin{vmatrix} \rho\xi & 0 & 0 \\ x & v & 0 \\ \rho c^2 + A_u u_{r0} & x + \rho g \xi u_{r0} \tilde{L}_{\tilde{p}} & A_y u_{r0} + \rho g \xi u_{r0} \tilde{L}_y \end{vmatrix} \\ & + x \begin{vmatrix} x - y_r u_{r0} & \rho\xi & 0 \\ 0 & x & v \\ A_{\tilde{\rho}} u_{r0} + \rho g \xi u_{r0} \tilde{L}_{\tilde{\rho}} & \rho c^2 + A_u u_{r0} & x + \rho g \xi u_{r0} \tilde{L}_{\tilde{p}} \end{vmatrix}. \end{aligned} \quad (2.94)$$

Developing it once again along the first line for each determinant:

$$P(x) = -\xi u_{r0}^2 \left(v A_y + g \xi \tilde{L}_y \right) + x(x - y_r u_{r0}) \left| \begin{array}{cc} x & v \\ \rho c^2 + A_u u_{r0} & x + \rho g \xi u_{r0} \tilde{L}_{\tilde{\rho}} \end{array} \right| + \rho \xi x \left(v A_{\tilde{\rho}} u_{r0} + g \xi u_{r0} \tilde{L}_{\tilde{\rho}} \right). \quad (2.95)$$

Finally the characteristic polynomial writes

$$P(x) = x^4 + (a^{-1} \xi \tilde{L}_{\tilde{\rho}} - y_r) u_{r0} x^3 - (c^2 + v A_u u_{r0} + \rho g \xi y_r \tilde{L}_{\tilde{\rho}} u_{r0}^2) x^2 + \left(y_r u_{r0} (c^2 + v A_u u_{r0}) + \xi u_{r0} (A_{\tilde{\rho}} + \rho g \xi \tilde{L}_{\tilde{\rho}}) \right) x - \xi (v A_y + g \xi \tilde{L}_y) u_{r0}^2. \quad (2.96)$$

Using Equations (2.90), Equations (2.93) and notations from Equations (2.58), it simplifies to

$$P(x = u - \lambda) = \det(B(\mathbf{W}) - \lambda \mathbf{I}_d) = x^4 + Ax^3 + Bx^2 + Cx + D, \quad (2.97)$$

with

$$\begin{cases} A = (F\xi - y_r) u_{r0}, \\ B = -(c^2 + (\beta + F y_r) \xi u_{r0}^2), \\ C = (G\xi + (3y - 1)c^2) u_{r0} + (g y_r - F\xi) \xi u_{r0}^3, \\ D = g \xi^2 u_{r0}^4 - (G\xi + c^2 y_r) y u_{r0}^2. \end{cases} \quad (2.98)$$

Evaluation of the characteristic polynomial in particular values

Polynomial P from Equation (2.97) is evaluated for $x = z u_{r0}$, $z \in \mathbb{R}$. It writes

$$P(x = z u_{r0}) = P_1(z) u_{r0}^2 + P_2(z) u_{r0}^4, \quad (2.99)$$

with

$$\begin{cases} P_1(z) = -c^2 z^2 + ((3y - 1)c^2 + G\xi)z - y(G\xi + y_r c^2), \\ P_2(z) = z^4 + (F\xi - y_r)z^3 - (\beta + F y_r)\xi z^2 + (g y_r - F\xi)\xi z + g \xi^2. \end{cases} \quad (2.100)$$

The roots of polynomials P_1 and P_2 are studied.

- The roots of polynomial P_1 are:

$$z_1^1 = y \quad ; \quad z_2^1 = \frac{G\xi}{c^2} + y_r = \rho(y^2 v_g - (1 - y)^2 v_l). \quad (2.101)$$

- The roots of polynomial P_2 are:

$$\begin{aligned} z_1^2 &= y \quad ; \quad z_2^2 = -(1 - y); \\ z_{3,4}^2 &= -\frac{F\xi}{2} \pm \frac{\sqrt{\xi(F^2\xi + 4g)}}{2}. \end{aligned} \quad (2.102)$$

$z = y$ is a root for $P_1(z)$ and $P_2(z)$, so $x = y u_{r0}$ is a root of the characteristic polynomial $P(x)$. Moreover, the polynomial P_1 is evaluated for $z_2^2 = -(1 - y)$:

$$P_1(z = -(1 - y)) = -(y c^2 + G\xi) = -y v_g \rho c^2 \leq 0. \quad (2.103)$$

It is interesting to note that the value $z = y$ and $z = -(1 - y)$ corresponds to compute respectively the value of the polynomial for the liquid velocity $u_l = u - yu_{r0}$ and for the vapor velocity $u_g = u + (1 - y)u_{r0}$. To properly use the intermediate value theorem, the derivative of the characteristic polynomial should be evaluated in $x_l = yu_{r0}$. It writes:

$$P'(x_l) = C_1^h u_{r0}^3 - C_2^h c^2 u_{r0}, \quad (2.104)$$

with

$$\begin{cases} C_1^h(p, \rho, y) = ya^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = (1 - y)\rho v_l = (1 - \rho y v_g(p)) > 0. \end{cases} \quad (2.105)$$

2.B Study of the sufficient condition for hyperbolicity when considering a constant relative velocity

The sufficient condition for hyperbolicity of the drift-flux 4-equation model with a constant relative velocity writes:

$$C_1^h(p, \rho, y) < 0 \quad \text{or} \quad |u_{r0}| < u_r^{max}(p, \rho, y) = \sqrt{\frac{C_2^h(p, \rho, y)}{C_1^h(p, \rho, y)}} c(p, \rho, y), \quad (2.106)$$

with

$$\begin{cases} C_1^h(p, \rho, y) = ya^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = 1 - \rho y v_g(p). \end{cases} \quad (2.107)$$

A two-phase flow of liquid water and steam is considered here and the sufficient condition is evaluated for this mixture. The physical properties of vapor and liquid water are obtained using the Python module `iapws`¹ [139]. It is developed by the International Association for the Properties of Water and Steam (IAPWS). The version IAPWS97 is used here, corresponding to the industrial formulation. A constant pressure characteristic of a reactor core is considered:

$$p = 155 \text{ bar}. \quad (2.108)$$

An out-of-equilibrium two-phase flow is considered with the deviation from equilibrium defined by

$$\Delta y_0^{eq} = y - \bar{y}, \quad (2.109)$$

where y is the mass fraction of vapor and $\bar{y}$ the mass fraction at equilibrium. Several values for the deviation are considered:

$$\Delta y_0^{eq} \in \{0, 0.01, 0.05\}. \quad (2.110)$$

A zero deviation corresponds to a flow at thermodynamic equilibrium. A deviation of 0.01 is commonly encountered in real reactor core flows in the subcooled boiling region, whereas a deviation of 0.05 is a maximum value that is very rarely reached in reality.

The ratio of the maximum relative velocity to the speed of sound, called **maximum ratio** and noted u_r^{max}/c , is shown in Figure 2.1 as a function of the mass fraction y for the different deviations from equilibrium considered. In the same figure, the ratio between the relative velocity obtained with Bestion's correlation [14] (see Equation (1.103), with an hydraulic diameter of $d_h = 0.011185$ m) and the speed of sound of the mixture, called **Bestion ratio** and noted u_r^b/c , is also shown. The

¹<https://iapws.readthedocs.io/en/latest/modules.html>

maximum ratio u_r^{max}/c varies little with the deviation from equilibrium except at very high mass fractions. In most of the domain, the Bestion ratio is well below the maximum ratio, which ensures hyperbolicity. When the mass fraction approaches one, the Bestion ratio exceeds the maximum ratio, and it is no longer possible to conclude on the system's hyperbolicity. These mass fraction values are never encountered in practice for flows in reactor cores. Therefore, the sufficient condition for hyperbolicity is not limiting in the vast majority of cases.

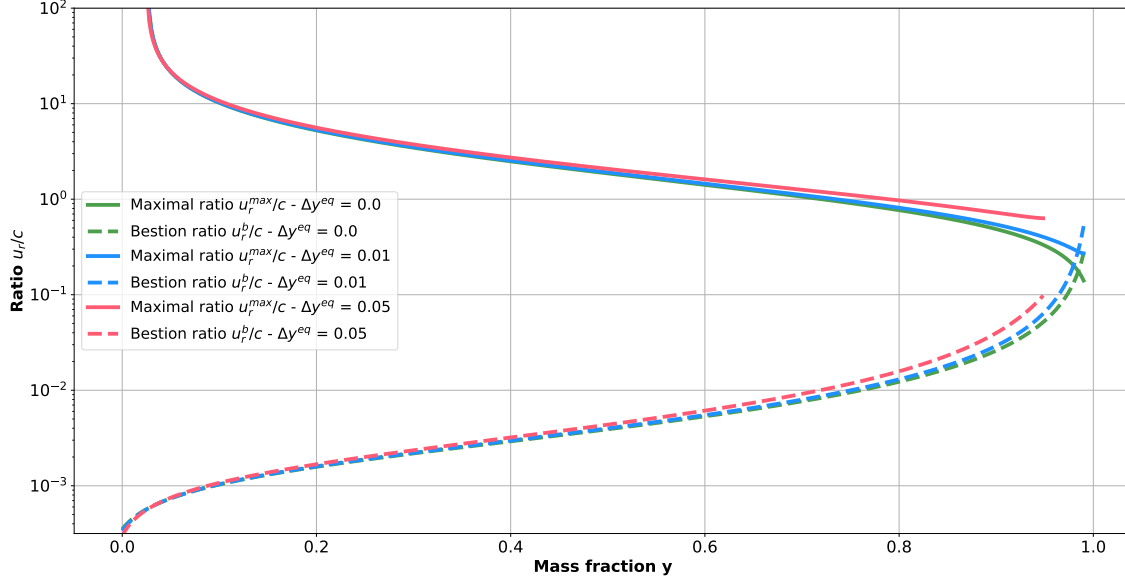


FIGURE 2.1

Maximum ratio u_r^{max}/c and Bestion ratio u_r^b/c as functions of the mass fraction y for different deviations from equilibrium and at pressure $p = 155$ bar.

Chapter 3

Some analytical solutions of the simplified two phase flow models

In this chapter, several analytical solutions are derived from the simplified equations of the model proposed in Section 1.3.1, where no friction or heat flux are considered. These analytical solutions are used for verification of finite volume schemes. In Section 3.1, the disequilibrium equation alone is considered with most parameters constant. Two steady-state solutions and one unsteady solution are detailed for use in **Chapter 4** to verify the various schemes proposed, in particular for the non-linear relative velocity convection term. In Section 3.2, two self-similar analytical solutions are detailed for the 3-equation model (see **Appendix 1.B**). The first solution uses the model for a porous flow, while the second considers a free medium but with a heating term. These solutions can be used to check finite-volume scheme. In appendices, several other analytical solutions are proposed. They are taken from literature or derived from known results. They are used in **Chapter 7** to verify the global finite-volume schemes proposed in **Chapter 6**. In **Appendix 3.A**, a solution from [78] is derived for the 3-equation model. **Appendix 3.B** presents the solutions of various 1D Riemann problems for a single-phase Euler 3-equation model. In **Appendix 3.A**, the 4-equation two-phase model is considered. An analytical solution is proposed when a simplified mixture equation of state is used.

3.1 Analytical solutions for the disequilibrium equation

In this section, the 1D disequilibrium equation is considered on the domain $\Omega = [0, 1]$ and in a free medium ($\varepsilon = 1$). The production term Γ_p due to heat flow is not considered. Mixture density and velocity are considered constant, hence

$$\begin{cases} \rho(x, t) = \rho_0, \\ u(x, t) = u_0. \end{cases} \quad (3.1)$$

As a consequence, the mass conservation is satisfied. The disequilibrium equation writes in this case

$$\frac{\partial y}{\partial t} + \frac{\partial}{\partial x} (yu_0 + y(1-y)u_r(x, t)) = \frac{\bar{y}(x, t) - y}{\tau(x, t)}, \quad (3.2)$$

where $u_r(x, t)$, $\tau(x, t)$ and $\bar{y}(x, t)$ are given functions, called input functions here. A Dirichlet boundary solution is applied on the left of the domain such that

$$y(x = 0, t) = y_0. \quad (3.3)$$

Density ρ_0 no longer appears in the disequilibrium equation. However, a value must be specified numerically. This value has no impact on the solution. It is therefore taken at $\rho_0 = 1 \text{ kg.m}^{-3}$. Two steady-state solutions are presented first. The first considers constant uniform input functions. The second solution uses the same input functions except for relative velocity, where the approximated Bestion correlation is used. A third unsteady solution is derived with the approximated Bestion correlation. The functions τ and $\bar{y}$ are chosen so as to obtain a self-similar solution.

3.1.1 Steady-state solution for constant parameters with constant relative velocity

The input functions are taken uniform such that, $\forall x$,

$$\begin{cases} u_r(x) = u_{r0}, \\ \bar{y}(x) = \bar{y}_0, \\ \tau(x) = \tau_0. \end{cases} \quad (3.4)$$

Equation (3.2) writes in this case, for a steady-state solution,

$$(u_0 + (1 - 2y)u_{r0}) \frac{\partial y}{\partial x} = \frac{\bar{y}_0 - y}{\tau_0}, \quad (3.5)$$

where the Boundary Condition (3.3) is chosen to respect $y_0 < \bar{y}_0$. Defining the following notations

$$\begin{cases} \lambda_r = \frac{u_{r0}}{u_0}, \\ l_0 = u_0 \tau_0, \end{cases} \quad (3.6)$$

Equation (3.5) writes

$$(1 + (1 - 2y)\lambda_r) \frac{\partial y}{\partial x} = \frac{\bar{y}_0 - y}{l_0}, \quad (3.7)$$

It can be rewritten

$$(\lambda_{lin}(\bar{y}_0 - y) + \lambda_{log}) \frac{\partial y}{\partial x} = \frac{\bar{y}_0 - y}{l_0}, \quad (3.8)$$

with

$$\begin{cases} \lambda_{lin} = 2\lambda_r, \\ \lambda_{log} = 1 + \lambda_r(1 - 2\bar{y}_0). \end{cases} \quad (3.9)$$

Integrating between 0 and x , the solution y is given by the implicit formulation

$$\lambda_{lin}(y(x) - y_0) - \lambda_{log} \ln \left(\frac{\bar{y}_0 - y(x)}{\bar{y}_0 - y_0} \right) = \frac{x}{l_0}. \quad (3.10)$$

Numerically, the solution is obtained with a dichotomy. It is used in Section 4.3.1 and Section 4.3.2 to verify the spatial convergence rate of the schemes proposed for the disequilibrium equation.

3.1.2 Steady-state solution for constant parameters with the approximated Bestion correlation

A very similar configuration to the one presented before is considered here. Some parameters are still taken constant such that

$$\begin{cases} \bar{y}(x, t) = \bar{y}_0, \\ \tau(x, t) = \tau_0. \end{cases} \quad (3.11)$$

For relative velocity, the approximated Bestion correlation from Equation (1.180) is considered such that

$$u_r(y) = \frac{1 + (\delta - 1)y}{1 - y} u_b \text{ with } u_b = 0.188\sqrt{gd_h(\delta - 1)}. \quad (3.12)$$

Using notations

$$\begin{cases} \lambda_b = \frac{u_b}{u_0}, \\ l_0 = u_0\tau_0, \\ \alpha_c = 2(\delta - 1)\lambda_b, \end{cases} \quad (3.13)$$

Equation (3.2) writes

$$(1 + \lambda_b + \alpha_c y) \frac{\partial y}{\partial x} = \frac{\bar{y}_0 - y}{l_0}. \quad (3.14)$$

Defining

$$\begin{cases} \lambda_{lin}^b = \alpha_c, \\ \lambda_{log}^b = 1 + \lambda_b + \alpha_c \bar{y}_0, \end{cases} \quad (3.15)$$

Equation (3.14) rewrites

$$(-\lambda_{lin}^b(\bar{y}_0 - y) + \lambda_{log}^b) \frac{\partial y}{\partial x} = \frac{\bar{y}_0 - y}{l_0}. \quad (3.16)$$

Integrating once again between 0 and x , the solution y satisfies the implicit formulation

$$\lambda_{lin}^b(y(x) - y_0) + \lambda_{log}^b \ln \left(\frac{\bar{y}_0 - y(x)}{\bar{y}_0 - y_0} \right) = -\frac{x}{l_0}. \quad (3.17)$$

The solution is once again obtained through a dichotomy. This solution is used in Section 4.3.3 to evaluate an approximate performance of the proposed scheme for steady-state with Bestion correlation.

3.1.3 Unsteady solution with approximated Bestion correlation

The third analytical solution is unsteady. The equation is considered on an interval $[t_0, t_1]$. Using once again, the approximated Bestion correlation (3.12), unsteady Equation (3.2) writes

$$\begin{cases} \frac{\partial y}{\partial t} + u_0 \frac{\partial y}{\partial x} + \frac{\partial}{\partial x} \left(y(1 + (\delta - 1)y) u_b \right) = \frac{\bar{y}(x, t) - y}{\tau(x, t)}, \\ y(x = 0, t = t_0) = y_0, \end{cases} \quad (3.18)$$

where $\bar{y}(x, t)$ and $\tau(x, t)$ are chosen input functions. The following notations are once again used

$$\begin{cases} \lambda_b = \frac{u_b}{u_0}, \\ \alpha_c = 2(\delta - 1)\lambda_b. \end{cases} \quad (3.19)$$

Equation (3.18) rewrites

$$\frac{\partial y}{\partial t} + u_0(1 + \lambda_b + \alpha_c y) \frac{\partial y}{\partial x} = \frac{\bar{y}(x, t) - y}{\tau(x, t)}. \quad (3.20)$$

Input functions $\bar{y}(x, t)$ and $\tau(x, t)$ are chosen so that the solutions of Equation (3.20) are self-similar solutions depending upon the variable

$$\xi(x, t) = \frac{x}{u_0 t}. \quad (3.21)$$

The time and space derivative for a function $y(\xi)$ are

$$\begin{cases} \frac{\partial y}{\partial t} = -\frac{\xi}{t}y'(\xi), \\ \frac{\partial y}{\partial x} = \frac{1}{u_0 t}y'(\xi). \end{cases} \quad (3.22)$$

When looking for $y(\xi)$, Equation (3.20) writes

$$\left[(1 + \lambda_b + \alpha_c y(\xi)) - \xi \right] y'(\xi) = \frac{t}{\tau(x, t)} (\bar{y}(x, t) - y(\xi)). \quad (3.23)$$

The following forms are taken for input functions with

$$\begin{cases} \tau(x, t) = \tau(\xi, t) = \frac{t}{\chi_0(1 + \lambda_b + \alpha_c y(\xi) - \xi)}, \text{ with } \chi_0 \text{ a constant parameter,} \\ \bar{y}(x, t) = \bar{y}(\xi = \frac{x}{u_0 t}) = \bar{y}_0 + \bar{y}_1 \cos(\chi_0 \xi) + \bar{y}_2 \sin(\chi_0 \xi), \end{cases} \quad (3.24)$$

where $\bar{y}_0$, $\bar{y}_1$ and $\bar{y}_2$ are constant parameters. To ensure the maximum principle for this Equation (see Section 4.1.2), the proposed functions must respect the constraints $\forall (x, t) \in [0, 1] \times [0, T]$,

$$\begin{cases} \tau > 0, \\ \bar{y} \in [0, 1]. \end{cases} \quad (3.25)$$

As $y(\xi) \in [0, 1]$, the first constraint of Equation (3.25) implies that

$$t_0 \geq \frac{x_{max}}{u_0(1 + \lambda_b)}, \quad (3.26)$$

with $x_{max} = 1$ m here.

Considering $\bar{y}_0 \in [0, 1]$, the second constraint is automatically satisfied when

$$\begin{cases} \bar{y}_0 \geq |\bar{y}_1| + |\bar{y}_2|, \\ \bar{y}_0 \leq 1 - |\bar{y}_1| - |\bar{y}_2|. \end{cases} \quad (3.27)$$

Using input functions of Equations (3.24), Equation (3.23) becomes

$$y'(\xi) + \chi_0 y(\xi) = \chi_0 \bar{y}. \quad (3.28)$$

Using the Boundary Condition (3.3), the solution of this equation is

$$y(\xi) = \bar{y}_0 + y_1 \cos(\chi_0 \xi) + y_2 \sin(\chi_0 \xi) + (y_0 - \bar{y}_0 - y_1) e^{-\chi_0 \xi}, \quad (3.29)$$

with

$$\begin{cases} y_1 = \frac{\bar{y}_1 - \bar{y}_2}{2}, \\ y_2 = \frac{\bar{y}_1 + \bar{y}_2}{2}. \end{cases} \quad (3.30)$$

The following parameters are chosen

$$\begin{cases} \bar{y}_0 = y_0, \\ \bar{y}_1 = \frac{1}{2}y_0, \\ \bar{y}_2 = \frac{1}{2}y_0, \end{cases} \quad (3.31)$$

so that $y_1 = 0$, $y_2 = y_0$ and $\bar{y}_0 = \bar{y}_1 + \bar{y}_2$. To respect Conditions (3.27), the parameter y_0 should satisfy

$$y_0 \in [0, \frac{2}{3}]. \quad (3.32)$$

In that case, the solution is

$$y(\xi) = (1 + \frac{1}{2} \sin(\chi_0 \xi)) y_0 \in [0, 1]. \quad (3.33)$$

This unsteady analytical solution is used to verify the performance of finite volume schemes for the disequilibrium equation in Section 4.3.4.

3.2 Unsteady solutions for the 3-equation model

This section is extracted from Article [70], a work published during the PhD Thesis in collaboration with Jean-Marc Hérard. Some exact solutions are available for Euler equations, both in the steady and the unsteady framework, at least in a one-dimensional setting. Regular solutions in rarefaction waves are well known from long (see [58, 130] for instance), and are also mandatory in order to build the one-dimensional unsteady solution of the Riemann problem when no source term arises. For the Homogeneous Equilibrium Model (HEM) [16, 17] or the Homogeneous Relaxation Model (HRM) [12, 16, 17], steady and unsteady numerical solutions can also be found when considering fluid flows in ducts with variable cross section (see for instance [45, 60, 69, 98]). However, few exact solutions are available when investigating flows with heat transfer, even in the steady case (see [29, 44, 78, 127]), and to our knowledge none in the unsteady case, when retaining the HEM model (or Euler equations) in a porous medium (or equivalently in a duct with variable cross section).

Hence we give focus herein on two reference **unsteady** solutions of the HEM model, in a 1D framework, when the fluid flows in a **porous medium**, or when **some heat source term** arises. Using the notations ε for porosity, ρ for density, u for velocity, p for pressure and $e(p, \rho)$ for internal energy the corresponding system writes

$$\begin{cases} \partial_t(\varepsilon \rho) + \partial_x(\varepsilon \rho u) = 0, \\ \partial_t(\varepsilon \rho u) + \partial_x(\varepsilon \rho u^2) + \varepsilon \partial_x p = 0, \\ \partial_t(\varepsilon E) + \partial_x(\varepsilon u(E + p)) = \varepsilon \phi(x, t), \\ \partial_t \varepsilon = 0, \end{cases} \quad (3.34)$$

with the total energy $E = \rho(\frac{1}{2}u^2 + e(p, \rho))$, the internal energy $e(p, \rho)$ given by a thermodynamic law, the porosity $\varepsilon(x)$ a given spatial function and the heating source term $\phi(x, t)$ also given by the user.

3.2.1 A self-similar unsteady solution without heat exchange.

Considering System (3.34) without heat exchange, it writes

$$\begin{cases} \partial_t(\varepsilon\rho) + \partial_x(\varepsilon\rho u) = 0, \\ \partial_t(\varepsilon\rho u) + \partial_x(\varepsilon\rho u^2) + \varepsilon\partial_x p = 0, \\ \partial_t(\varepsilon E) + \partial_x(\varepsilon u(E + p)) = 0, \\ \partial_t \varepsilon = 0. \end{cases} \quad (3.35)$$

We note $v = 1/\rho$ the specific volume and we define the entropy as the function satisfying

$$c^2(p, \rho) \left(\frac{\partial s}{\partial p} \right)_\rho + \left(\frac{\partial s}{\partial \rho} \right)_p = 0, \quad (3.36)$$

with c the speed of sound defined by

$$(\rho c)^2 = \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} \left(p - \rho^2 \left(\frac{\partial e}{\partial \rho} \right)_p \right). \quad (3.37)$$

By making the classical variable change from $(\rho, \rho u, E, \varepsilon)$ to (s, u, p, ε) , the system becomes

$$\begin{cases} \partial_t s + u \partial_x s = 0, \\ \partial_t u + u \partial_x u + v \partial_x p = 0, \\ \partial_t p + u \partial_x p + \rho c^2 \partial_x u + \rho c^2 \frac{u}{\varepsilon} \partial_x \varepsilon = 0, \\ \partial_t \varepsilon = 0. \end{cases} \quad (3.38)$$

By denoting the variable vector $\mathbf{Y} = (s, u, p, \varepsilon)$, the system is first rewritten in matrix form

$$\partial_t \mathbf{Y} + \mathbf{A} \partial_x \mathbf{Y} = 0, \quad (3.39)$$

$$\text{with } \mathbf{A} = \begin{bmatrix} u & 0 & 0 & 0 \\ 0 & u & v & 0 \\ 0 & \rho c^2 & u & \rho c^2 \frac{u}{\varepsilon} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.40)$$

and then we search for solutions depending only on $\xi = \frac{x}{t}$. In this case, the solutions $u(\xi)$, $s(\xi)$ and $p(\xi)$ must satisfy, $\forall t > 0$,

$$\begin{cases} \frac{1}{t}(u - \xi)s'(\xi) = 0, \\ \frac{1}{t}(u - \xi)u'(\xi) + \frac{1}{t}vp'(\xi) = 0, \\ \frac{1}{t}(u - \xi)p'(\xi) + \frac{1}{t}\rho c^2 u'(\xi) = -\rho c^2 u \frac{\varepsilon'(x)}{\varepsilon(x)}. \end{cases} \quad (3.41)$$

In the following, we exclude the degenerate constant case $\varepsilon(x) = \varepsilon_0$. The third equation (multiplied by x) imposes a particular form to the porosity $\varepsilon(x)$, such that, with $x_L \in \mathbb{R}$,

$$\varepsilon(x) = \left(\frac{x}{x_L} \right)^\alpha, \text{ with a constant } \alpha \in \mathbb{R}^*. \quad (3.42)$$

Remark 1: Specific values for α .

We can highlight some specific values for α . The specific value $\alpha = 1$ corresponds to a radial solution corresponding to $\varepsilon(x) \propto 2\pi x$, while $\alpha = 2$ corresponds to a spherical solution where $\varepsilon(x) \propto 4\pi x^2$. $\square$

The first equation imposes, for $u(\xi) \neq \xi$, a constant entropy

$$s(\xi) = s_0. \quad (3.43)$$

The solutions $(u(\xi), p(\xi))$ must be such that

$$\begin{cases} p'(\xi) = \rho(\xi - u(\xi))u'(\xi), \\ \rho c^2 u'(\xi) + (u - \xi)p'(\xi) = -\rho c^2 u(\xi) \frac{\alpha}{\xi}. \end{cases} \quad (3.44)$$

Excluding the void case ($\rho = 0$), the equations combine into

$$[c^2 - (u - \xi)^2] u'(\xi) + \alpha \frac{u c^2(\xi)}{\xi} = 0. \quad (3.45)$$

Remark 2:

- For the particular case $\alpha = 0$, we recover the trivial solutions $u(\xi) = u_0$ and $p(\xi) = p_0$.

Moreover, for $u'(\xi) \neq 0$, the solutions satisfy

$$c^2 - (u - \xi)^2 = 0 \Leftrightarrow \xi = u \pm c, \quad (3.46)$$

which correspond to the two characteristics associated with the two Genuinely Non Linear fields (GNL) for Euler equations $\lambda = u - c$ and $\lambda = u + c$.

- In the **general case** $\alpha \neq 0$, the constant solution $(u, p) = (u_0, p_0)$ is admissible **if and only if** $u_0 = 0$.

$\square$

In the case $\alpha \neq -1$, we look for velocity solutions of the form

$$u(\xi) = a\xi, \quad (3.47)$$

with an unknown constant a . Using Equation (3.45), we have

$$a((1 + \alpha)c^2 - (a - 1)^2 \xi^2) = 0. \quad (3.48)$$

For any EOS, we note

$$f^{EOS}(c, s_0) = \int \frac{dp}{(\rho c)(s_0, p)}. \quad (3.49)$$

For $a \neq 0$, we deduce from Equation (3.48)

$$c(\xi) = \pm \frac{(1 - a)\xi}{\sqrt{1 + \alpha}}. \quad (3.50)$$

- If $c = \frac{(a-1)\xi}{\sqrt{1+\alpha}}$, using the first Equation from (3.44) and Equation (3.49), an equation linking unknowns a and c is obtained

$$a\xi\sqrt{1 + \alpha} + f^{EOS}(c(\xi), s_0) = 0. \quad (3.51)$$

- If $c = \frac{(1-a)\xi}{\sqrt{1+\alpha}}$, the relation is

$$a\xi\sqrt{1+\alpha} - f^{EOS}(c(\xi), s_0) = 0. \quad (3.52)$$

Summary: A one-dimensional self-similar regular solution $(s(\xi), u(\xi), c(\xi))$ of Equations (3.35), with $\xi(x, t) = \frac{x}{t+t_0}$, $t_0 > 0$ and $\varepsilon(x) = \left(\frac{x}{x_0}\right)^\alpha \in [0, 1]$, $x_0 \in \mathbb{R}^{+*}$, is given by

$$\begin{cases} s(\xi) = s_0, \\ u(\xi) = a\xi, \\ c(\xi) = \frac{|(1-a)\xi|}{\sqrt{1+\alpha}}, \end{cases} \quad (3.53)$$

with $a \in \mathbb{R}$, such that Equation (3.51) is satisfied when $(a-1)\xi > 0$, or fulfilling Equation (3.52) when $(1-a)\xi > 0$.

Equation (3.51) (or Equation (3.52)), together with Equation (3.50), which must be valid for all ξ , lead to a **compatibility** condition on a . We consider two specific EOS in the following remarks.

Remark 3: Perfect gas EOS: We consider a perfect gas thermodynamics respecting $p = (\gamma - 1)\rho e$, $\gamma > 1$. We can then write

$$\begin{cases} s = \ln(p\rho^{-\gamma}), \\ c^2(p, s) = \gamma e^{s/\gamma} p^{\frac{\gamma-1}{\gamma}}. \end{cases} \quad (3.54)$$

We have, using $s = s_0$,

$$\int \frac{dp}{(\rho c)(s_0, p)} = \frac{2c(s_0, p)}{\gamma - 1}. \quad (3.55)$$

- If $c(\xi) = \frac{(a-1)\xi}{\sqrt{1+\alpha}}$, from Equation (3.51), we get

$$\xi \left(a\sqrt{1+\alpha} + \frac{2}{\gamma-1} \frac{(a-1)}{\sqrt{1+\alpha}} \right) = 0. \quad (3.56)$$

Thus

$$a = \frac{2}{\gamma + 1 + \alpha(\gamma - 1)}. \quad (3.57)$$

When the porosity is uniform ($\varepsilon = \varepsilon_0$ i.e. $\alpha = 0$), we retrieve the standard value $a = \frac{2}{\gamma+1}$. Moreover Equation (3.51) corresponds to the Riemann invariant associated with the 1-rarefaction wave in the Riemann problem [29, 130].

- If $c(\xi) = \frac{(1-a)\xi}{\sqrt{1+\alpha}}$, we get the same result $a = \frac{2}{\gamma + 1 + \alpha(\gamma - 1)}$, with a similar significance for Equation (3.52) that is a Riemann invariant of the 3-wave, when $\varepsilon = \varepsilon_0$.

□

Remark 4: Stiffened gas EOS: Considering a stiffened gas EOS

$$p + \gamma\pi = (\gamma - 1)\rho e, \quad (3.58)$$

with $\pi > 0$ and $\gamma > 1$, we get once again

$$f^{EOS}(c, s_0) = \frac{2c}{\gamma - 1}. \quad (3.59)$$

It leads to the same solution for the constant a as in Equation (3.57). $\square$

Numerical application for a radial case: We consider the solution for a choice $\alpha = 1$ on a domain $[0, x_L]$. A perfect gas of constant $\gamma = 1.2$ is chosen. In this case, for $T = 1$ s, $x_L = 2$ m and $u_L = ax_L/T$ with $a = 1/\gamma$, the solutions are represented in Figure 3.1. The initial (respectively boundary) conditions are deduced from the analytical solution given in Equation (3.53), setting $t = 0$ (respectively $x = x_L$, x_L being a boundary).

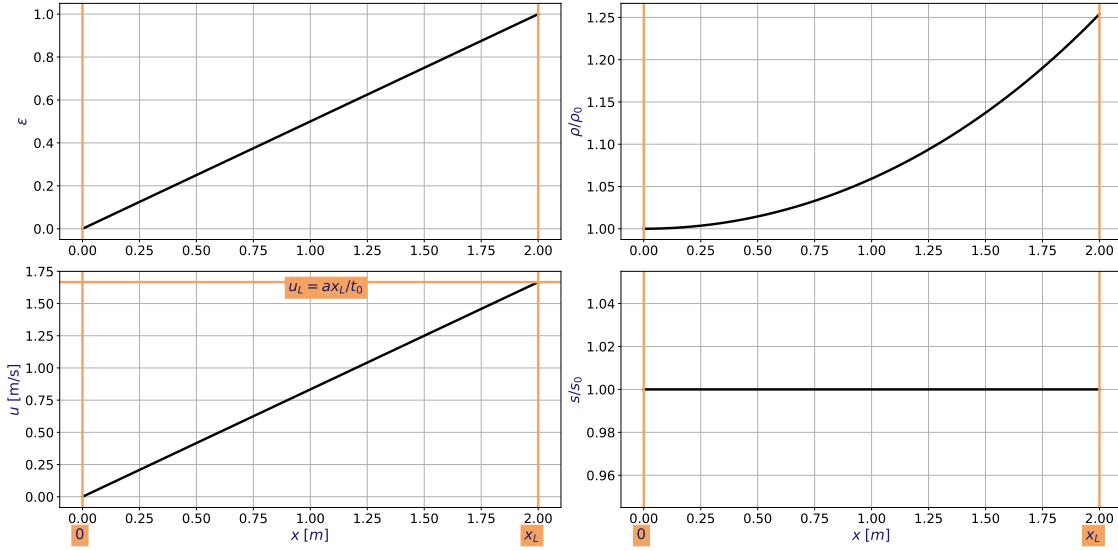


FIGURE 3.1
Analytical solution for $\alpha = 1$, $\gamma = 1.2$ and $x_L = 2.0$ m at $T = 1.0$ s.

3.2.2 A self-similar unsteady solution in free medium with a specific heat profile.

For this solution, we consider a free medium ($\varepsilon = 1$) and a specific heat source term noted $\phi(x, t)$. System (3.34) becomes

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, \\ \partial_t E + \partial_x(u(E + p)) = \phi(x, t). \end{cases} \quad (3.60)$$

Using a change of variables from $(\rho, \rho u, E)$ to (u, p, s) , the system writes

$$\begin{cases} \partial_t u + u \partial_x u + v \partial_x p = 0, \\ \partial_t p + u \partial_x p + \rho c^2 \partial_x u = \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \phi(x, t), \\ \partial_t s + u \partial_x s = \left(\frac{\partial s}{\partial p} \right)_\rho \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \phi(x, t). \end{cases} \quad (3.61)$$

We look for solutions depending on

$$\xi = \frac{x^a}{(t + t_0)^b}, \quad (3.62)$$

i.e. by normalizing $\xi = \frac{x^\beta}{t+t_0}$, $\beta \in \mathbb{R}^*$, for variables $s(\xi)$, $u(\xi)$ and $p(\xi)$ where $t_0 > 0$ is a constant. The momentum equation shows that non-trivial solutions require $\beta = 1$. It constrains the heating profile to satisfy

$$\phi(x, t) = (t + t_0)^{-1} \psi(\xi), \quad (3.63)$$

with $\psi(\xi)$ to be given by the user.

The entropy equation gives

$$(u(\xi) - \xi)s'(\xi) = \left(\frac{\partial s}{\partial p} \right)_\rho \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \psi(\xi). \quad (3.64)$$

For this particular form of the heating source term, the variables must satisfy

$$\begin{bmatrix} u - \xi & 0 & 0 \\ 0 & u - \xi & v \\ 0 & \rho c^2 & u - \xi \end{bmatrix} \cdot \begin{bmatrix} s' \\ u' \\ p' \end{bmatrix} = \begin{pmatrix} \left(\frac{\partial s}{\partial p} \right)_\rho \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \psi(\xi) \\ 0 \\ \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \psi(\xi) \end{pmatrix}. \quad (3.65)$$

Remark 5: Case $\psi = 0$.

For the particular case where the heating term is not considered, we find the classic framework. The solution $(s, u, p)(\xi) = (s_0, u_0, p_0)$ is then trivial. In the general case, a non-trivial solution (u, p) then imposes

$$(u - \xi)^2 - c^2 = (u - \xi + c)(u - \xi - c) = 0, \quad (3.66)$$

with $s'(\xi) = 0 \Leftrightarrow s(\xi) = s_0$.

□

We consider **the general case** $\psi(\xi) \neq 0$. We can note that the profile $(u(\xi), p(\xi)) = (u_0, p_0)$ is no longer a solution of System (3.65). The first two equations can be rewritten as:

$$p'(\xi) = -\rho(\xi)(u(\xi) - \xi)u'(\xi), \quad (3.67)$$

and

$$\begin{aligned} \rho c^2(\xi)u'(\xi) - \rho(\xi)(u(\xi) - \xi)^2u'(\xi) &= \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v \psi(\xi), \\ \Leftrightarrow (c^2(\xi) - (u(\xi) - \xi)^2)u'(\xi) &= \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v^2(\xi)\psi(\xi). \end{aligned} \quad (3.68)$$

Equation (3.68) imposes $c^2 - (u - \xi)^2 \neq 0$. The solutions must satisfy

$$\begin{cases} s'(\xi) = \left(\frac{\partial s}{\partial p} \right)_\rho \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v(\xi) \frac{\psi(\xi)}{(u(\xi) - \xi)}, \\ u'(\xi) = \frac{1}{c^2 - (u - \xi)^2} \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v^2(\xi)\psi(\xi), \\ p'(\xi) = \frac{u - \xi}{(u - \xi)^2 - c^2} \left(\frac{\partial e}{\partial p} \right)_\rho^{-1} v(\xi)\psi(\xi). \end{cases} \quad (3.69)$$

Summary: A one-dimensional self-similar regular solution $(s(\xi), u(\xi), p(\xi))$ of Equations (3.60), with a heat source given by Equation (3.63), is obtained through Equations (3.69).

For an arbitrary $\psi(\xi)$, the integration of Equations (3.69) is not trivial. It can be done numerically using an implicit Euler method, proceeding backwards, and using a very dense mesh of the ξ axis, see [60] for the numerical method. In particular, the specific case with perfect gas EOS of [3] can be mentioned, where the considered heating is in the form of $\phi = \frac{A}{27} \frac{\gamma-2}{\gamma-1} t^{-1} \xi^4$, $A \in \mathbb{R}$ and fits within the framework proposed here.

Remark 6: Perfect gas EOS. Considering the perfect gas thermodynamics law defined in (3.54), $\left(\frac{\partial s}{\partial p}\right)_\rho = \frac{1}{p}$ and $\left(\frac{\partial e}{\partial p}\right)_\rho = \frac{v}{\gamma-1}$, so we have

$$\begin{bmatrix} u - \xi & 0 & 0 \\ 0 & u - \xi & v \\ 0 & \gamma p & u - \xi \end{bmatrix} \cdot \begin{bmatrix} s' \\ u' \\ p' \end{bmatrix} = \begin{pmatrix} (\gamma-1)\psi(\xi)p^{-1} \\ 0 \\ (\gamma-1)\psi(\xi) \end{pmatrix}. \quad (3.70)$$

When $\psi = 0$, some manipulations of Equations (3.65) lead to

$$(u - \xi) \left(2(u - \xi)' + (\gamma - 1)u' \right) = 0. \quad (3.71)$$

We recover the results from Section 3.2.1 for a constant porosity $\varepsilon = 1$. In this case, the velocity writes

$$u(\xi) - u_0 = \frac{2}{\gamma + 1} (\xi - \xi_0), \quad (3.72)$$

leading to the profile of the sound velocity with $c(\xi) = \pm(u(\xi) - \xi)$. $\square$

3.2.3 Conclusion

We have presented two analytical solutions of the HEM model in a 1D framework when the fluid flows through a porous medium or when a heating profile is applied to the fluid. For the first solution with porosity, exact solutions are given for a perfect gas and a stiffened gas EOS but this can be extended to more complex EOS. The second solution with heating is valid for any EOS. Obviously, it would be interesting to numerically test the two proposed solutions and compare them with solutions obtained from numerical schemes.

We refer the readers who are not familiar with classical exact solutions to references [29, 58, 130], among others for pure one-dimensional exact solutions of Euler equations, when $\alpha = 0$ (or equivalently when the porosity is uniform ($\epsilon(x) = \epsilon_0$)) and without any heat source term, focusing on a perfect gas EOS.

3.A Steady-state for 3-equation model

The 3-equation model of Section 1.B is considered in a free medium ($\varepsilon = 1$). A steady-state analytical solution is proposed for this model in a 1D framework. This solution is a simplified version of the one from [78] and is used in **Chapter 7** to verify ThermoTorch's global 3-equation finite volume scheme. The configuration for this analytical solution is a 1D tube on the domain $[0, L]$ with an incoming flow of single-phase liquid water undersaturated at $x = 0$ such that

$$\begin{cases} u(x=0) = u_{in}, \\ p(x=0) = p_{in}, \\ T(x=0) = T_{in} \leq T_s(p_{in}). \end{cases} \quad (3.73)$$

In this section, the notations for the specific volume and the specific enthalpy are

$$\begin{cases} v = \frac{1}{\rho}, \\ h = e + pv. \end{cases} \quad (3.74)$$

From Equations (3.73), it can be deduced

$$\begin{cases} h(x=0) = h_l(p_{in}, T_{in}), \\ \rho(x=0) = \rho_l(p_{in}, T_{in}). \end{cases} \quad (3.75)$$

Second order differential terms, gravity and solid friction are neglected. Relative velocity is assumed to be zero. The resulting model is an Euler model with a total energy equation that includes a known heating term $\phi(x)$ such that

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, \\ \partial_t(\rho E) + \partial_x(\rho u E + pu) = \phi(x). \end{cases} \quad (3.76)$$

The equation of state for the mixture is initially written in internal energy (see Equation (1.68)). It can be written in enthalpy. The 3-equation model is considered, so the equation of state depends only on pressure and density. It is noted $h^{EoS}(p, \rho)$. Transforming the equation on total-energy in an equation on enthalpy, the steady-state system writes

$$\begin{cases} \partial_x(\rho u) = 0, \\ \partial_x(\rho u^2) = -\partial_x p, \\ \partial_x(\rho h u) = u \partial_x p + \phi(x). \end{cases} \quad (3.77)$$

In this configuration, the mass flux is uniform such that

$$q(x) = \rho(x)u(x) = q_0 > 0. \quad (3.78)$$

The constant mass flux ensures

$$\partial_x(\rho u^2) = \partial_x(q_0^2 v) = q_0^2 \partial_x v. \quad (3.79)$$

Using constant mass flux q_0 , the momentum and energy equations become

$$\begin{cases} \partial_x p = -q_0^2 \partial_x v, \\ \partial_x h = \frac{\phi(x)}{q_0} - q_0^2 \partial_x \left(\frac{v^2}{2} \right). \end{cases} \quad (3.80)$$

Let's suppose that State (a) at point x_a is known. It is noted (p_a, v_a, h_a) . Set of Equations (3.80) is integrated between State (a) and an unknown State (b) such that

$$\begin{cases} p_b(v_b) = p_a - q_0^2(v_b - v_a), \\ h_b(v_b) = h_a + \frac{1}{q_0} \int_{x=a}^b \phi(x), dx - \frac{q_0^2}{2}(v_b^2 - v_a^2). \end{cases} \quad (3.81)$$

The function $\mathcal{G}(v_b)$ can be defined by

$$\mathcal{G}(v_b) = h^{EoS}(p_b(v_b), v_b) - h_b(v_b). \quad (3.82)$$

A Newton method is used to find v_b such that $\mathcal{G}(v_b) = 0$, giving the solution for state (b) , knowing state (a) .

3.B Unsteady solutions for 1D Riemann problems - one-phase flow

1D Riemann problems are considered here in the framework of Euler equations (with an energy equation) combined with a perfect gas Equation of State (EoS). A spatial one-dimensional domain Ω centered on 0 is used. At initial time $t = 0$, two constant states are defined on either side of the discontinuity in $x = 0$: the left state is denoted L and the right state is denoted R. Figure 3.2 shows the initial configuration of the problem.

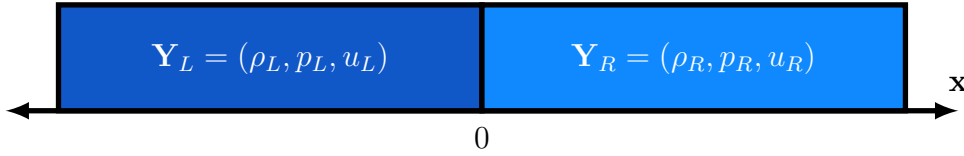


FIGURE 3.2
Initial configuration for 1D Riemann problems with the 3-equation model.

Three different test cases are studied here according to the initial values for each quantity: the symmetric double rarefaction wave in Section 3.B.1, the symmetric double shock wave in Section 3.B.2 and the Sod shock tube in Section 3.B.3. The first two problems are symmetric problems corresponding to classical test cases to simulate two different wall boundary conditions when using the mirror technique. The third problem is the classical Sod Shock Tube problem from [131]. First, the equation of state and the system of equations used for these Riemann problems are detailed. Then, a general methodology based on the results of [58] and [130] is presented. This methodology is then applied to determine the analytical solutions of the three test cases presented here.

The perfect gas EoS writes:

$$\begin{cases} p = \rho e(\gamma - 1), \\ s(p, \rho) = c_v \ln(\rho p^{-\gamma}), \\ c(p, \rho) = \sqrt{\gamma \frac{p}{\rho}}, \end{cases} \quad (3.83)$$

with $\gamma = 7/5$ and $c_v = 1000 \text{ J.kg}^{-1}.\text{K}^{-1}$. We consider the Euler equations

$$\partial_t \mathbf{w} + \partial_x \mathbf{F}(\mathbf{w}) = \mathbf{0}, \text{ with } \begin{cases} \mathbf{w} = (\rho, u, E), \\ \mathbf{F} = (\rho u, \rho u^2 + p, \rho u E + pu). \end{cases} \quad (3.84)$$

The fluxes matrix $\mathbf{F}$ is diagonalizable with three real eigenvalues noted λ such that

$$\lambda \in \{u - c(p, \rho), u, u + c(p, \rho)\}. \quad (3.85)$$

Across a shock, the Rankine-Hugoniot jump conditions between the left state l and the right state r are

$$-\sigma(\mathbf{w}_r - \mathbf{w}_l) + (\mathbf{F}_r - \mathbf{F}_l) = \mathbf{0}, \quad (3.86)$$

with σ the propagation speed. Using this condition on the mass conservation equation enables to obtain the propagation speed

$$\sigma = \frac{\rho_r u_r - \rho_l u_l}{\rho_r - \rho_l}. \quad (3.87)$$

A 1D tube is considered in the domain $x \in [-1, 1]$ m. It is initialized with two different states: a left state (noted L) for $x < 0$ and a right state (noted R) for $x > 0$. For each configuration, the condition

$$u_R - u_L < \frac{2}{\gamma - 1}(c(p_L, \rho_L) + c(p_R, \rho_R)), \quad (3.88)$$

with $c(p, \rho)$ the speed of sound, is respected so that no vacuum appears. The existence and uniqueness is assured for the considered problem (see for instance [130]). We look for self-similar solutions depending upon the variable

$$\xi = \frac{x}{t}. \quad (3.89)$$

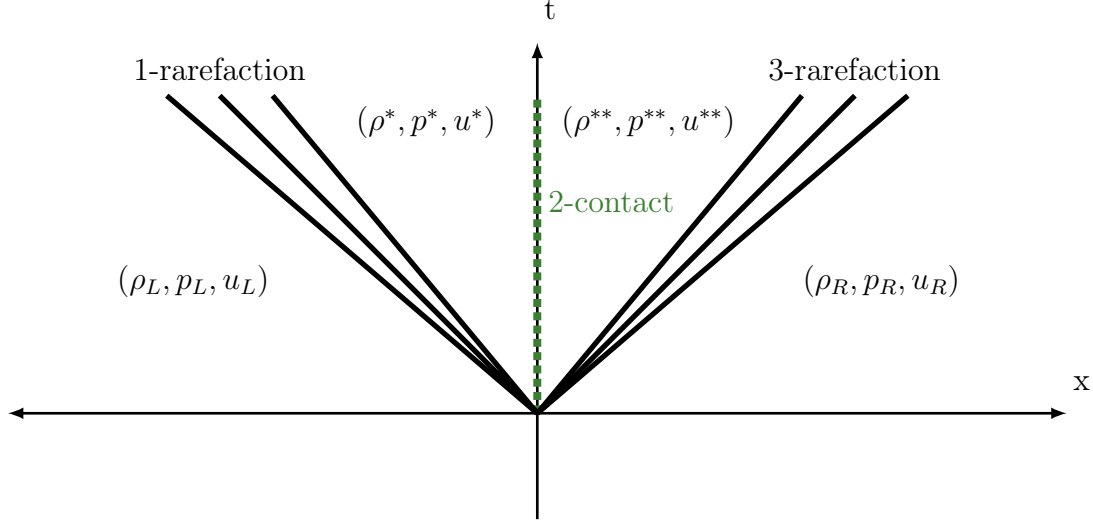
3.B.1 Symmetric double rarefaction wave

Table 3.1 describes the initial conditions with ρ_0, u_0 and p_0 positive constants. For this configuration, two symmetric rarefaction waves propagate in opposite directions. Due to symmetrical initial conditions, the contact discontinuity is a "ghost wave". This configuration corresponds to the behavior of the scheme when applying the mirror technique close to wall boundary conditions where the fluid is sucked.

Left state	Right state
$\rho_L = \rho_0$	$\rho_R = \rho_0$
$u_L = -u_0$	$u_R = u_0 > 0$
$p_L = p_0$	$p_R = p_0$

TABLE 3.1
Generic initial conditions for symmetric double rarefaction wave.

For a time $t > 0$, the space is divided into four distinct regions with constant states, separated by three waves (see Figure 3.3). A 1-rarefaction wave propagates to the left in the domain $x < 0$ and $t > 0$, the 2-contact wave is a "ghost wave" fixed at $x = 0$ for $t > 0$ and the 3-rarefaction wave propagates to the right in the domain $x > 0$ and $t > 0$.


FIGURE 3.3

Scheme of the solution for double symmetric rarefaction wave test case.

The invariants at the 2-contact wave are

$$\begin{cases} u^{**} = u^*, \\ p^{**} = p^*. \end{cases} \quad (3.90)$$

The problem is symmetric, so that

$$\begin{cases} u^* = 0, \\ \rho^{**} = \rho^*. \end{cases} \quad (3.91)$$

The symmetry of the problem enables to study only the domain $x < 0$, so that only the 1-rarefaction wave is considered. The Riemann invariants for the 1-rarefaction wave are

$$\begin{cases} u_L + \frac{2}{\gamma-1}c(p_L, \rho_L) = u^* + \frac{2}{\gamma-1}c(p^*, \rho^*), \\ s^* \equiv s(p^*, \rho^*) = s(p_L, \rho_L). \end{cases} \quad (3.92)$$

The perfect gas EoS (3.83) gives

$$p\rho^{-\gamma}(\xi) = p_L\rho_L^{-\gamma}. \quad (3.93)$$

The 1-rarefaction induces a relaxation for $\xi \in [\xi_L, \xi^*]$ with $\xi^* = u^* - c(p^*, \rho^*)$ and $\xi_L = u_L - c(p_L, \rho_L)$. Using Equations (3.92), the solution for the 1-rarefaction wave is $\forall \xi \in [\xi_L, \xi^*]$,

$$\begin{cases} u(\xi) = u_L + \frac{2}{\gamma+1}(\xi - \xi_L), \\ c(\xi) = u(\xi) - \xi, \\ \rho(\xi) = \left(\frac{c(\xi)^2}{\gamma p_L} \right)^{\frac{1}{\gamma+1}} \rho_L, \\ p(\xi) = \left(\frac{\rho(\xi)}{\rho_L} \right)^{\gamma} p_L. \end{cases} \quad (3.94)$$

3.B.2 Symmetric double shock wave

Table 3.2 describes the initial conditions with ρ_0, u_0 and p_0 positive constants. In this case, two symmetric shock waves propagate in opposite directions. Due to symmetrical initial conditions, the

contact discontinuity is still a "ghost wave". This configuration corresponds to the behavior of the scheme when applying the mirror technique close to wall boundary conditions where the fluid is impinging the wall.

Left state	Right state
$\rho_L = \rho_0$	$\rho_R = \rho_0$
$u_L = u_0 > 0$	$u_R = -u_0$
$p_L = p_0$	$p_R = p_0$

TABLE 3.2
Generic initial conditions for symmetric double shock wave.

For a time $t > 0$, two shock waves and one contact discontinuity divide the domain into four distinct regions with constant states (see Figure 3.4). The 2-contact discontinuity is a "ghost wave" that remains stationary at $x = 0$ for $t > 0$. For $t > 0$, a 1-shock wave propagates to the left in the domain $x < 0$, while a 3-shock wave propagates to the right in the domain $x > 0$.

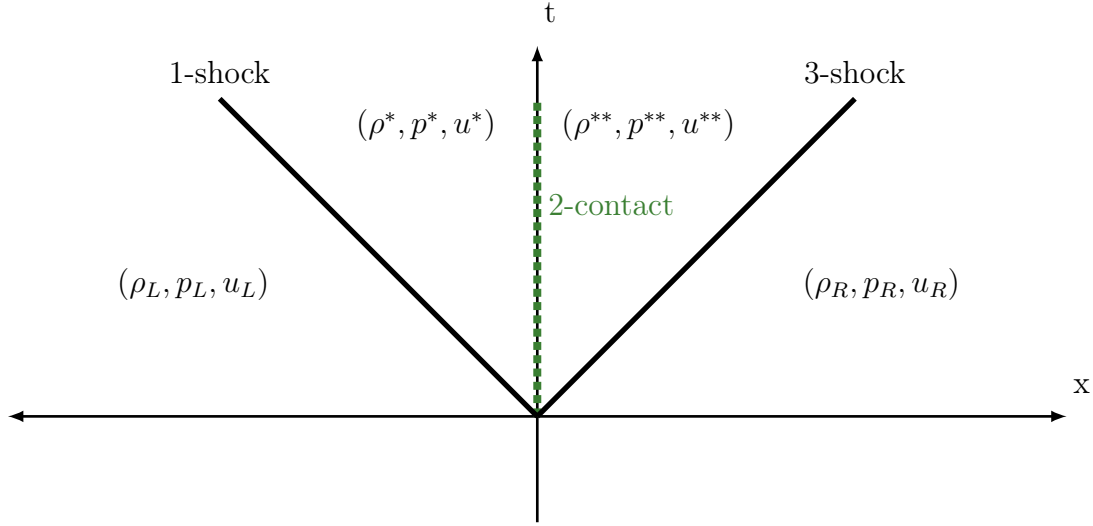


FIGURE 3.4
Scheme of the solution for double symmetric shock wave test case.

Once again, the problem is symmetric and (p, u) are preserved through the contact wave. Hence

$$\begin{cases} u^{**} = u^* = 0, \\ p^{**} = p^*, \\ \rho^{**} = \rho^*. \end{cases} \quad (3.95)$$

For shock waves, the jump conditions (3.86) are adjusted to eliminate the dependence on the propagation speed such that

$$\begin{aligned} \rho_l \rho_r (u_l - u_r)^2 &= (p_r - p_l)(\rho_r - \rho_l), \\ e(p_r, \rho_r) - e(p_l, \rho_l) + \frac{p_l + p_r}{2} \left(\frac{1}{\rho_r} - \frac{1}{\rho_l} \right) &= 0. \end{aligned} \quad (3.96)$$

Due to symmetry, only the domain $x > 0$ needs to be studied, with the 3-shock wave to be considered. The state (ρ^*, p^*) needs to be determined. Noting

$$z = \frac{\rho^*}{\rho_R} > 1. \quad (3.97)$$

and using perfect gas EoS (3.83), the second equation of (3.96) becomes

$$\frac{p^*}{p_R} = \frac{\beta z - 1}{\beta - z} > 1, \quad (3.98)$$

with

$$\beta = \frac{\gamma + 1}{\gamma - 1} > 1. \quad (3.99)$$

By substituting this formula into the first equation of (3.96), we obtain

$$P_{SDS}(z) = (\beta + 1)(1 - z)^2 - \zeta z(\beta - z) = 0, \text{ with } \zeta = \frac{\rho_R u_R^2}{p_R} \text{ and } z \in [1, \beta]. \quad (3.100)$$

Using $P_{SDS}(1) < 0$ and $P_{SDS}(\beta) > 0$, the solution $z \in [1, \beta[$ exists and can be computed with the relation

$$z = \frac{2 + (2 + \zeta)\beta + \sqrt{\Delta}}{2(1 + \beta + \zeta)} \text{ avec } \Delta = (2 + (2 + \zeta)\beta)^2 - 4(1 + \beta + \zeta)(1 + \beta). \quad (3.101)$$

The state (p^*, ρ^*) can be computed from the value of z . Using Equation (3.87), the speed of the 3-shock wave is

$$\sigma^{(3)} = -\frac{\rho_R u_R}{\rho^* - \rho_R} > 0. \quad (3.102)$$

3.B.3 Sod shock tube

The Sod Shock Tube is a classical test case from [131]. Unlike the two previous tests, the initial conditions, summarized in Table 3.3, are not symmetrical.

Left state	Right state
$\rho_L = 1$	$\rho_R = 0.125$
$u_L = 0$	$u_R = 0$
$p_L = 10^5$	$p_R = 10^4$

TABLE 3.3
Initial conditions for Sod shock tube test case.

Four distinct regions with constant states divide the domain for a time $t > 0$. These regions are separated by different type of wave (for $t > 0$): the 1-rarefaction wave propagates in the domain $x < 0$, the 2-contact wave propagates in the domain $x > 0$ and the 3-shock wave also propagates in the domain $x > 0$ but at a higher speed than the 2-contact wave. A scheme of this configuration is shown in Figure 3.5.

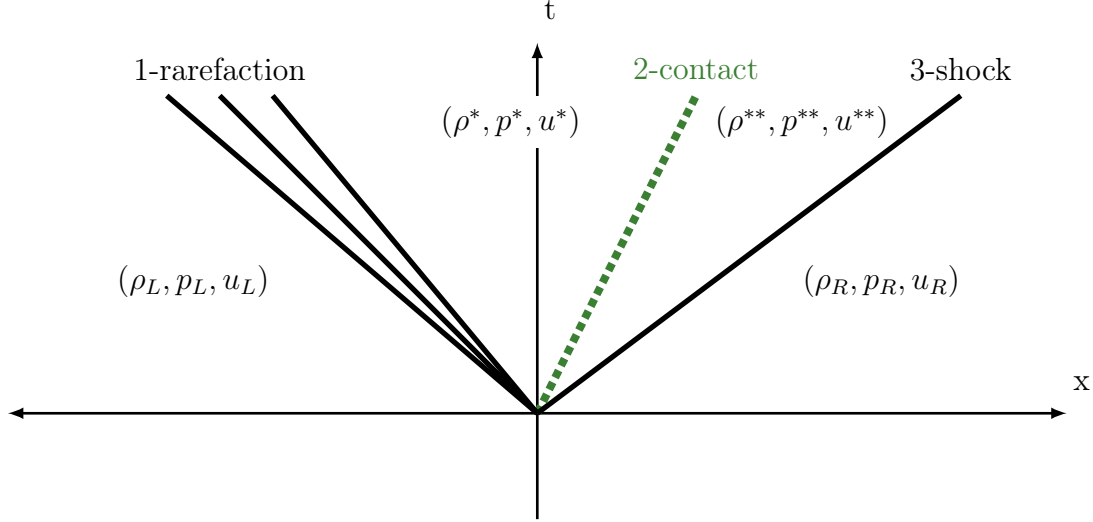


FIGURE 3.5
Scheme of the solution for the Sod shock tube test case.

At the 2-contact wave, we have $u^{**} = u^*$ and $p^{**} = p^*$. The system has 4 unknowns $(u^*, p^*, \rho^*, \rho^{**})$. Using the Riemann invariant (3.92) for the 1-rarefaction wave, we can write, with $c_L \equiv c(p_L, \rho_L)$,

$$\begin{cases} u_L + \frac{2}{\gamma-1}c_L = u^* + \frac{2}{\gamma-1}c(p^*, \rho^*), \\ s(p_L, \rho_L) = s(p^*, \rho^*) \Leftrightarrow p_L \rho_L^{-\gamma} = p^* (\rho^*)^{-\gamma}. \end{cases} \quad (3.103)$$

Using the entropy conservation, we compute the speed of sound $c(p^*, \rho^*)$ as a function of the left state

$$c(p^*, \rho^*) = \sqrt{\gamma \frac{p^*}{\rho^*}} = \left(\frac{p^*}{p_L} \right)^{\frac{\gamma-1}{2\gamma}} c_L. \quad (3.104)$$

Injecting (3.104) in the first Riemann invariant of Equations (3.103) gives a first formula for the velocity u^* as a function of p^* noted $u_{(1)}^*$

$$u_{(1)}^*(p^*) = u_L + \frac{2}{\gamma-1} \left(1 - \left(\frac{p^*}{p_L} \right)^{\frac{\gamma-1}{2\gamma}} \right) c_L. \quad (3.105)$$

Now considering the 3-shock, Equations (3.96) for jump conditions write

$$\begin{cases} u_R - u^* = -\sqrt{\frac{(p_R - p^*)(\rho_R - \rho^{**})}{\rho^{**}\rho_R}}, \\ \frac{p^*}{p_R} = \frac{\beta z - 1}{\beta - z} > 1 \text{ with } z = \frac{\rho^{**}}{\rho_R} > 1, \end{cases} \quad (3.106)$$

with the sign of $u_R - u^*$ given by the entropy inequality. Combining the two equations, the velocity writes

$$u^* = u_R + \sqrt{\frac{p_R}{\rho_R} \frac{(\beta+1)(1-z)}{\beta-z} \frac{1-z}{z}}. \quad (3.107)$$

The variable z can be rewritten

$$z(p^*) = \frac{\beta \frac{p^*}{p_R} + 1}{\frac{p^*}{p_R} + \beta}. \quad (3.108)$$

Using Equations (3.99) and (3.108), we obtain a second equation for u^* as a function of p^* , noted $u_{(2)}^*$

$$u_{(2)}^*(p^*) = u_R + c_R \sqrt{\frac{2}{(\gamma - 1)(\beta - z(p^*))z(p^*)}} (z(p^*) - 1), \quad (3.109)$$

with $c_R = \sqrt{\gamma p_R \rho_R^{-1}}$. The solution for p^* is obtained numerically by solving

$$u_{(1)}^*(p^*) = u_{(2)}^*(p^*) \text{ with } p^* \in [p_R, p_L]. \quad (3.110)$$

A dichotomy is used to get the pressure p^* from Equation (3.110). The 1-rarefaction wave induces a relaxation for $\xi \in [\xi_L, \xi^*]$ with $\xi_L = u_L - c_L$ and $\xi^* = u^* - c(p^*, \rho^*)$. Using Equation (3.87), the speed of the 2-contact wave is

$$\sigma^{(2)} = u^*. \quad (3.111)$$

Using again Equation (3.87), the speed of the 3-shock wave is

$$\sigma^{(3)} = \frac{\rho^{**} u^*}{\rho^{**} - \rho_R} > \sigma^{(2)}. \quad (3.112)$$

3.C Analytical solution for the 4-equation model - 1D Riemann Problem

In this section, the objective is to determine an analytical solution of the 4-equation model with a simplified mixture Equation of State (EoS). This solution is used to verify the numerical schemes for the 4-equation model in **Chapter 7**. In order to find an analytical solution, the conservative Model (1.61) is used in a one-dimensional simplified framework. Zeroth-order and 2nd-order terms are neglected. Relative velocity is taken to be zero. In this configuration, the 4-equation conservative model in total energy from Equations (1.61) is equivalent to the 4-equation enthalpy model from Equations (1.108). The system is written in conservative variables $\mathbf{W} = (\rho, \rho u, \rho E, \rho y)$

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2) + \partial_x p = 0, \\ \partial_t(\rho E) + \partial_x(\rho E u) + \partial_x(pu) = 0, \\ \partial_t(\rho y) + \partial_x(\rho y u) = 0, \end{cases} \quad (3.113)$$

with E the total energy defined by

$$E = e + \frac{1}{2}u^2. \quad (3.114)$$

A mixture EoS depending upon the mass fraction y is used here to verify the resolution of the 4-equation model. It is constructed with a mixture of perfect gas in Section 3.C.1. The Equation of State writes, with $\gamma > 1$,

$$e(p, \rho, y) = \frac{p}{\rho(\gamma - 1)} + h_0(y), \quad (3.115)$$

with $h_0(y)$ a function depending only upon the mass fraction.

The system is rewritten in a non-conservative form. The internal energy equation is obtained using the momentum equation multiplied by the velocity and the total energy equation. The classical internal energy equation of an Euler-type system is obtained:

$$\partial_t e + u \partial_x e + p v \partial_x u = 0, \quad (3.116)$$

using the notation $v = \frac{1}{\rho}$ for the specific volume. Defining the sound velocity by the formulation

$$\rho c^2 = \left(\frac{\partial e}{\partial p} \right)_{\rho, y}^{-1} \left(\frac{p}{\rho} - \rho \left(\frac{\partial e}{\partial \rho} \right)_{p, y} \right), \quad (3.117)$$

the pressure equation writes

$$\partial_t p + u \partial_x p + \rho c^2 \partial_x u = 0. \quad (3.118)$$

An entropy s for the system should satisfy the relationship

$$c^2 \left(\frac{\partial s}{\partial p} \right)_{\rho, y} + \left(\frac{\partial s}{\partial \rho} \right)_{p, y} = 0. \quad (3.119)$$

If $s(p, \rho, y)$ is an entropy of the system, the function $\hat{s}(p, \rho, y) = f(y) \cdot s(p, \rho, y) + s_0(y)$, with $f(y)$ and $s_0(y)$ any regular functions, is also an entropy of the system. If a function s satisfies the Equation (3.119), the entropy equation simply writes

$$\partial_t s + u \partial_x s = 0. \quad (3.120)$$

The non-conservative system writes

$$\begin{cases} \partial_t y + u \partial_x y = 0, \\ \partial_t s + u \partial_x s = 0, \\ \partial_t u + u \partial_x u + v \partial_x p = 0, \\ \partial_t p + u \partial_x p + \rho c^2 \partial_x u = 0. \end{cases} \quad (3.121)$$

A 1D Riemann problem is studied in Section 3.C.2 for the Equations (3.121) with EoS (3.115).

3.C.1 Simplified mixture EoS

A mixture as defined in Section 1.3.3 is considered, where the two phases are in pressure equilibrium, at pressure p . For each phase (liquid and gaseous), a perfect gas EoS is considered here, with the same constant polytropic index γ for both phases

$$e_k(p, \rho_k) = \frac{p}{\rho_k(\gamma - 1)} + h_{0k}, \quad (3.122)$$

where h_{0k} is a constant. The entropy associated to phase k is

$$s_k(p, \rho_k) = c_{vk} \ln \left(\frac{p \rho_k^{-\gamma}}{\gamma - 1} \right) + s_{0k}, \quad (3.123)$$

with c_{vk} the specific heat capacity at constant volume of phase k and s_{0k} a constant. This gives the phase temperature T_k with

$$c_{vk} T_k(p, \rho_k) = \frac{p}{\rho_k(\gamma - 1)}. \quad (3.124)$$

The parameters of the two EoS can be set to approximate the thermodynamics of water-vapor under nuclear reactor conditions. These parameters are listed below:

$$\gamma, h_{0g}, h_{0l}, c_{vg}, c_{vl}, s_{0g}, s_{0l}. \quad (3.125)$$

The mixture EoS writes, by using $v = \frac{1}{\rho} = y v_g + (1 - y) v_l$,

$$e(p, \rho, y) = y e_g + (1 - y) e_l = \frac{p}{\rho(\gamma - 1)} + h_0(y), \quad (3.126)$$

with

$$h_0(y) = y h_{0g} + (1 - y) h_{0l}. \quad (3.127)$$

Using Equation (3.117), the speed of sound of the mixture can be computed. It is independent of the mass fraction such that

$$c^2(p, \rho) = \frac{\gamma p}{\rho}. \quad (3.128)$$

Remark on entropy Using Condition (3.119), an entropy for the mixture is

$$s(p, \rho, y) = c_v(y) \ln \left(\frac{p \rho^{-\gamma}}{\gamma - 1} \right) + s_0(y), \quad (3.129)$$

with

$$\begin{cases} s_0(y) = y s_{0g} + (1 - y) s_{0l}, \\ c_v(y) = y c_{vg} + (1 - y) c_{vl}. \end{cases} \quad (3.130)$$

This choice of $c_v(y)$ and $s_0(y)$ enables to recover the entropy of each phase for limits $y \rightarrow 0$ and $y \rightarrow 1$. The temperature of the mixture can then be written

$$T(p, \rho, y) = \frac{p}{\rho c_v(y)(\gamma - 1)} = \frac{y c_{vg} T_g + (1 - y) c_{vl} T_l}{c_v(y)}. \quad (3.131)$$

The proposed entropy satisfies the following thermodynamic relationship

$$T ds = de + p dv + s_y dy, \quad (3.132)$$

with

$$s_y = -\frac{dh_0}{dy}(y) + T \left(\frac{ds_0}{dy}(y) + \frac{dc_v}{dy}(y) \frac{s - s_0(y)}{c_v(y)} \right). \quad (3.133)$$

These results will be used in **Chapter 6** to study the resolution algorithm and the entropy used numerically.

3.C.2 1D Riemann Problem - Analytical solution

An infinite 1D domain is considered. At initial time $t = 0$, two constant states are defined on either side of the discontinuity in $x = 0$: the left state is denoted L and the right state is denoted R. Figure 3.6 shows the initial configuration of the problem. The solution proposed here is based on the methodology developed in [130] or in [58].

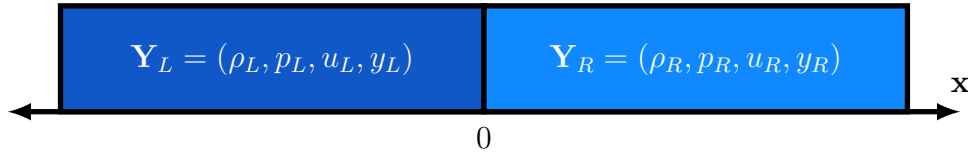


FIGURE 3.6
Initial configuration for a 1D Riemann problem with the 4-equation model.

System (3.121) writes, using $\mathbf{Y} = (y, s, u, p)$,

$$\partial_t \mathbf{Y} + \mathbf{B}(\mathbf{Y}) \partial_x \mathbf{Y} = 0, \quad (3.134)$$

with

$$\mathbf{B} = \begin{pmatrix} u & 0 & 0 & 0 \\ 0 & u & 0 & 0 \\ 0 & 0 & u & v \\ 0 & 0 & \rho c^2 & u \end{pmatrix}. \quad (3.135)$$

The eigenvalues of this matrix are

$$\begin{cases} \lambda_1 = u - c, \\ \lambda_2 = u, \\ \lambda_3 = u, \\ \lambda_4 = u + c. \end{cases} \quad (3.136)$$

The corresponding eigenvectors are

$$\begin{cases} \mathbf{r}_1 = (0, 0, 1, -\rho c), \\ \mathbf{r}_2 = (1, 0, 0, 0), \\ \mathbf{r}_3 = (0, 1, 0, 0), \\ \mathbf{r}_4 = (0, 0, 1, \rho c). \end{cases} \quad (3.137)$$

Four waves propagate in the domain, associated to each eigenvalue. For eigenvalue λ_i , the associated wave is denoted i -wave. The 2- and 3-waves are LD waves because $\nabla_{\mathbf{Y}} \lambda_i \cdot \mathbf{r}_i(\mathbf{Y}) = 0$ for $i = 2, 3$. They propagate at the same velocity u . The 1- and 4-waves are GNL waves. The structure of the solution is represented in Figure 3.7, where the constant states $\mathbf{Y}_L$, $\mathbf{Y}^*$, $\mathbf{Y}^{**}$ and $\mathbf{Y}_R$ are separated by waves (LD and GNL).

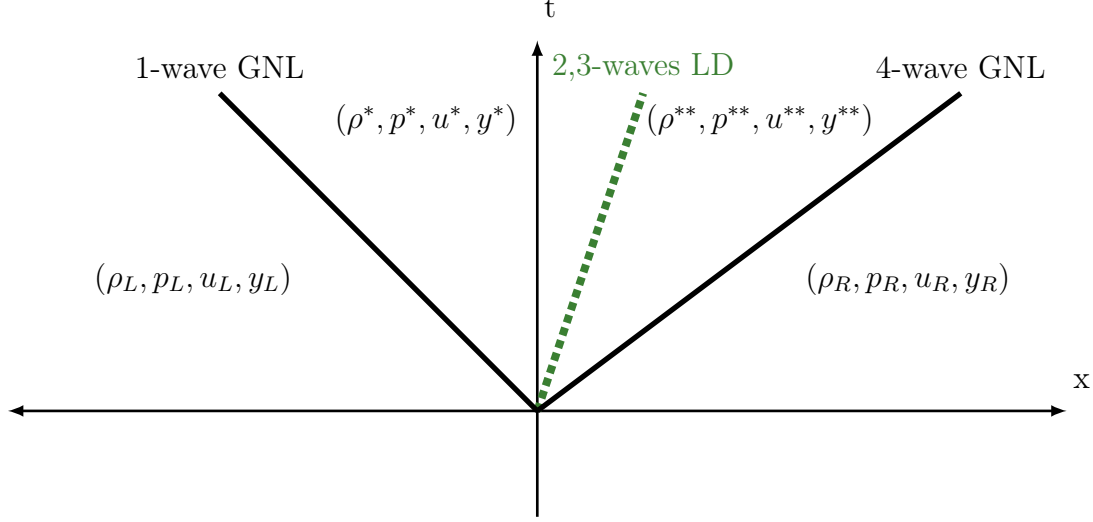


FIGURE 3.7
Structure of the solution for a 1D Riemann Problem - 4-equation model.

The Riemann invariants ϕ_i associated with the i -wave are determined by computing

$$\nabla_{\mathbf{Y}}(\phi_i(\mathbf{Y})) \cdot \mathbf{r}_i(\mathbf{Y}) = 0. \quad (3.138)$$

The Riemann invariants are

$$\begin{cases} \phi_1 = \left\{ y, s, u + \int_0^p \frac{dp}{\rho c(s, p, y)} \right\}, \\ \phi_{2,3} = \{u, p\}, \\ \phi_4 = \left\{ y, s, u - \int_0^p \frac{dp}{\rho c(s, p, y)} \right\}. \end{cases} \quad (3.139)$$

On either side of a shock wave propagating at speed σ , the left state is denoted l and the right state r . The following notations are adopted for the jump of a function f and its mean value across the shock

$$\begin{cases} [f] = f_r - f_l, \\ \bar{f} = \frac{f_r + f_l}{2}. \end{cases} \quad (3.140)$$

The Rankine-Hugoniot conditions for Sytem (3.113) write

$$\begin{cases} -\sigma[\rho] + [\rho u] = 0, \\ -\sigma[\rho u] + [\rho u^2 + p] = 0, \\ -\sigma \left[\rho e + \rho \frac{u^2}{2} \right] + \left[\rho u e + \frac{1}{2} \rho u^3 \right] + [up] = 0, \\ -\sigma[\rho y] + [\rho u y] = 0. \end{cases} \quad (3.141)$$

Using the following notation

$$w = u - \sigma, \quad (3.142)$$

the jump conditions become

$$\begin{cases} [\rho w] = 0, \\ \rho_l \rho_r [u]^2 = [p][\rho], \\ \rho w ([e] + \bar{p}[v]) = 0, \\ \rho w [y] = 0. \end{cases} \quad (3.143)$$

The relationships obtained above are valid for any EoS. For cases where $w \neq 0$, thus in GNL waves, the mass fraction jump is zero

$$[y] = 0. \quad (3.144)$$

When the mixture EoS (3.115) is used, the internal energy jump writes

$$[e] = \left[\frac{pv}{\gamma - 1} + h_0(y) \right] = \left[\frac{pv}{\gamma - 1} \right]. \quad (3.145)$$

By manipulating Relationships (3.143) with EoS (3.115), the pressure ratio as a function of the density ratio is obtained for shock waves

$$\frac{p_l}{p_r} = \frac{\beta z - 1}{\beta - z}, \quad (3.146)$$

with

$$\begin{cases} \beta = \frac{\gamma + 1}{\gamma - 1}, \\ z = \frac{\rho_l}{\rho_r}, \end{cases} \quad (3.147)$$

with $z < 1$ in the 1-shock wave and $z > 1$ in the 4-shock wave. Finally, using the entropy inequality, the relations are

$$\begin{cases} z = \frac{\rho_l}{\rho_r}, \\ \frac{p_l}{p_r} = \frac{\beta z - 1}{\beta - z}, \\ [u] = -\sqrt{\frac{[p][\rho]}{\rho_l \rho_r}}, \\ [y] = 0, \end{cases} \quad (3.148)$$

with $z \in]\frac{1}{\beta}, 1]$ (resp. $z \in [1, \beta[$) in a 1-shock wave (resp. in a 4-shock wave).

Considering the initial conditions $\mathbf{Y}_L$ and $\mathbf{Y}_R$ and using all these results, it is possible to prove (using a similar methodology as in [130]) that a unique self-similar solution $\mathbf{Y}(\xi = \frac{x}{t}), x \in \mathbb{R}, t > 0$ exists to the 1D Riemann problem, with positive density, without vacuum, if the following condition is satisfied:

$$u_R - u_L < \frac{2}{\gamma - 1}(c_L + c_R). \quad (3.149)$$

This unique solution is composed of four constant states $\mathbf{Y}_L, \mathbf{Y}^*, \mathbf{Y}^{**}, \mathbf{Y}_R$, separated by contact waves, rarefaction waves and shock waves.

Similar results as for a perfect gas are obtained.

Part II

THYC-coeur with relative velocity

Chapter 4

Finite volume schemes for the disequilibrium equation

The disequilibrium equation defined in Section 1.2.4 is considered. The maximum principle for the mass fraction is widely investigated, either for diffusive problems (see among others [40]) or in hybrid convection-diffusion problems (see among others [47, 53, 54, 91, 100]). Most of the time, a null relative velocity is considered in the convective flux model. In the sequel, the system involving two mass balance equations with a non null relative velocity (see also [53, 54]) is considered: Equations (1.52) and (1.60). For now, the production term of the disequilibrium equation is not considered. It is studied separately in **Appendix 4.D**.

First, the disequilibrium equation is recalled and notations are settled in Section 4.1.1. Once the continuous maximum principle has been examined in Section 4.1.2, three distinct linear finite volume schemes complying with the discrete maximum principle with no (or with a weak) restriction on the time step are proposed in Section 4.2. Eventually, numerical simulations are used to assess the accuracy of the schemes in Section 4.3 using analytical solutions from Section 3.1. The proposed schemes can be used in a broader framework such as two-phase flow models like those from **Chapter 1**.

This chapter is an extended version of a scientific article published during the thesis in *Comptes Rendus Mécanique de l'Académie des Sciences* (see [93]). Sections 4.3.2, 4.3.3 and 4.3.4 have been added to the published version to enhance the numerical verification database of the schemes.

4.1 Continuous equations for the disequilibrium

4.1.1 Governing equations

The system considered is composed of Equations (1.52) and (1.60) (see also [53] for a similar system):

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{q} = 0, \\ \frac{\partial(\rho y)}{\partial t} + \nabla \cdot (y \mathbf{q}) + \nabla \cdot (y(1-y) \mathbf{q}_r) = \rho \frac{\bar{y} - y}{\tau} + \Gamma_p, \end{cases} \quad (4.1)$$

where $\mathbf{q}$ is the mixture mass flux and $\mathbf{q}_r$ the relative mass flux

$$\begin{cases} \mathbf{q} = \rho \mathbf{u}, \\ \mathbf{q}_r = \rho \mathbf{u}_r. \end{cases} \quad (4.2)$$

In this section, the production term due to the heat flux Γ_p (see Section 1.4.1) is not addressed; it has been removed from the disequilibrium equation. This study focuses on the evolution of the mass fraction y in this flow. It evolves according to a mass flux $\mathbf{q}_g = \mathbf{q} + (1 - y)\mathbf{q}_r$. The system of equations is studied on a spatial domain Ω , over a time period $[0, T]$. The boundary of Ω is noted Γ . Noting the outward unit normal $\mathbf{n}_\Gamma$, the boundary can be split according to the sign of the flow as

$$\Gamma = \Gamma_+ \cup \Gamma_- \cup \Gamma_w, \quad (4.3)$$

where

$$\begin{cases} \Gamma_+ = \{\mathbf{x} \in \Gamma, \mathbf{q}(\mathbf{x}, t) \cdot \mathbf{n}_\Gamma < 0\}, \\ \Gamma_- = \{\mathbf{x} \in \Gamma, \mathbf{q}(\mathbf{x}, t) \cdot \mathbf{n}_\Gamma > 0\}, \\ \Gamma_w = \{\mathbf{x} \in \Gamma, \mathbf{q}(\mathbf{x}, t) \cdot \mathbf{n}_\Gamma = 0\}. \end{cases} \quad (4.4)$$

In this Chapter, the following notation, already used in Chapter 2, is once again used:

$$\xi = y(1 - y). \quad (4.5)$$

To study the continuous maximum principle, the mass flux noted $\mathbf{q}_\xi$ is defined by

$$\mathbf{q}_\xi = \mathbf{q} + (1 - 2y)\mathbf{q}_r. \quad (4.6)$$

The boundary Γ can also be split according to the sign of the mass flux $\mathbf{q}_\xi$:

$$\Gamma = \Gamma_+^\xi \cup \Gamma_-^\xi \cup \Gamma_w^\xi, \quad (4.7)$$

where

$$\begin{cases} \Gamma_+^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma < 0\}, \\ \Gamma_-^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma > 0\}, \\ \Gamma_w^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma = 0\}. \end{cases} \quad (4.8)$$

The four parameters $\mathbf{q}$, $\mathbf{q}_r$, $\bar{y}$ and τ are given functions here:

- $\mathbf{q}(\mathbf{x}, t)$ the mixture mass flux, given by a momentum conservation equation or provided as input data,
- $\mathbf{q}_r(\mathbf{x}, t)$ the relative mass flux, provided as input data. It is often given with a drift-flux closure law related to the other parameters,
- $\bar{y}(\mathbf{x}, t) \in [0, 1]$ the equilibrium mass fraction, reached after a characteristic relaxation time $\tau(\mathbf{x}, t) > 0$. $\bar{y}$ and τ are obtained through closure laws.

This system of equations must be completed by initial conditions for the mass fraction and the density

$$\begin{cases} y(\mathbf{x}, t = 0) = y_0(\mathbf{x}), \\ \rho(\mathbf{x}, t = 0) = \rho_0(\mathbf{x}), \end{cases} \quad (4.9)$$

and by suitable boundary conditions for the mass fraction (and also for the density on Γ_+).

4.1.2 Continuous maximum principle

Property 1 (Continuous maximum principle): For the density $\rho(\mathbf{x}, t) > 0$, assume initial conditions such that $y(\mathbf{x}, t = 0) \in [0, 1]$ and boundary conditions such that $y(\mathbf{x} \in \Gamma_+^\xi, t) \in [0, 1]$, with $\Gamma_+^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma < 0\}$. Consider closure laws for source terms such that $\bar{y}(\mathbf{x}, t) \in [0, 1]$ and $\tau(\mathbf{x}, t) > 0$. If the quantity $((1 - 2y)\nabla \cdot \mathbf{q}_r + 2\mathbf{q}_r \cdot \nabla y)/\rho$ is bounded on $\Omega \times [0, T]$, then the mass fraction $y(\mathbf{x}, t)$ solution of (4.1) lies in $[0, 1]$ on $\Omega \times [0, T]$. $\square$

Using the same methodology as [53] and [99], the proof of this property is given in **Appendix 4.A**. We recall that the positivity of the density is ensured as long as the divergence of the mixture velocity $\mathbf{u}$ is bounded and that the incoming flow has a positive density $\rho(\mathbf{x} \in \Gamma_+, t) > 0$. In the following, the focus shifts on the mass fraction governing equation with unknown y . A discrete finite volume scheme for the density ρ will be assumed (see Equation (4.14)).

Remark: In order to ensure the continuous maximum principle for the mass fraction y and the positivity of the density ρ , two different input boundaries Γ_+ and Γ_+^ξ are considered in the most general case. For applications concerned, co-current flows entering the domain are considered. Hence the phase velocities have the same sign, such that

$$\forall \mathbf{x} \in \Gamma_+, \mathbf{u}_g \cdot \mathbf{u}_l \geq 0 \text{ with } \begin{cases} \mathbf{u}_g &= \mathbf{u} + (1 - y)\mathbf{u}_r, \\ \mathbf{u}_l &= \mathbf{u} - y\mathbf{u}_r. \end{cases} \quad (4.10)$$

Hence, noting that

$$\begin{aligned} \mathbf{q} &= \rho(y\mathbf{u}_g + (1 - y)\mathbf{u}_l), \\ \mathbf{q}_\xi &= \rho((1 - y)\mathbf{u}_g + y\mathbf{u}_l), \end{aligned} \quad (4.11)$$

the dot product of these mass fluxes is $\mathbf{q}_\xi \cdot \mathbf{q} = \rho^2(y(1 - y)(\mathbf{u}_g - \mathbf{u}_l)^2 + \mathbf{u}_g \cdot \mathbf{u}_l) \geq 0$ but also $\Gamma_w^\xi = \Gamma_w$, $\Gamma_+^\xi = \Gamma_+$ and $\Gamma_-^\xi = \Gamma_-$.

4.2 Finite volume schemes

4.2.1 Discretization and notations

The time interval $[0, T]$ is discretised in N_T intervals $[t^n, t^{n+1}]$, $n \in \llbracket 0, N_T - 1 \rrbracket$ with $\Delta t^n = t^{n+1} - t^n$, $n \in \llbracket 0, N_T - 1 \rrbracket$ such that

$$t^0 = 0 \quad ; \quad \forall n \in \llbracket 0, N_T - 1 \rrbracket, t^{n+1} = t^n + \Delta t^n \text{ and } T = t^{N_T}. \quad (4.12)$$

The domain Ω is also discretised in N_Ω cells (cell i is noted Ω_i) such that

$$\Omega = \cup_{i \in \llbracket 1, N \rrbracket} \Omega_i. \quad (4.13)$$

A staggered mesh is considered. These two different meshes are described in Figure 4.1 for a 2D cartesian mesh. The mesh from Equation (4.13) is the pressure mesh (yellow cell in Figure 4.1) where the scalar is defined at the center of the cell. The velocities and mass fluxes are determined on the velocity meshes (green and blue cells in Figure 4.1), defined by the edges of the pressure mesh.

FIGURE 4.1

Scheme 4.2 represents a pressure cell i with three neighboring pressure cells j, k, l , a boundary wall face w and an inlet/outlet boundary face noted ∞ . Configuration presented here is an unstructured mesh because the proposed schemes are available for any unstructured mesh. However, numerical results presented in this chapter are obtained on cartesian staggered meshes.

FIGURE 4.2

Considering the pressure mesh, Equations (4.1) are integrated on each cell i between time t^n (superscript n for variables) and time t^{n+1} (superscript $n + 1$ for variables). A superscript $*$ is used

for data taken at an intermediate state t^* , which is either t^n or t^{n+1} . The explicit or implicit choice will be decided according to the wanted properties of the scheme. The following notations are used:

- θ_i^n the approximate value of the continuous quantity θ on cell i at time t^n ,
- ω_i the volume of cell i ,
- $j \in v(i)$ the neighbors of cell i ,
- S_{ij} the surface of intersection of cell i with its neighbor cell j ,
- $\mathbf{n}_{ij}$ the normal unit vector of the surface S_{ij} outward cell i ,
- $q_{ij}^* = \mathbf{q}_{ij}^* \cdot \mathbf{n}_{ij}$ the normal mixture mass flux between cell i and cell j ,
- $(q_r)_{ij}^* = (\mathbf{q}_r)_{ij}^* \cdot \mathbf{n}_{ij}$ the normal relative mass flux between cell i and cell j ,
- $(q_g)_{ij}^* = q_{ij}^* + (1 - y_{ij}^n)(q_r)_{ij}^*$ the normal gas mass flux between cell i and cell j . The explicit choice for y_{ij}^n will result in a linear numerical scheme. The spatial discretisation for the mass fraction y_{ij}^n at interface ij is detailed in the sequel.

Here, $\mathbf{q}_{ij}^*$ (respectively $(\mathbf{q}_r)_{ij}^*$) is an estimation of the mass flux $\mathbf{q}$ (respectively $\mathbf{q}_r$) at interface ij at time $t^* \in [t^n, t^{n+1}]$.

4.2.2 Finite volume discretization

Finite volume methods are well adapted to treat conservation issues and to keep physical quantities in valid bounds. Focusing on the system of mass balances (4.1), *Gastaldo and al.* [53] introduces a **non linear implicit** scheme (with respect to y), which guarantees the discrete maximum principle with no restriction on the time step, even if some non-zero relative velocity is accounted for. Here, a **linear implicit** scheme is proposed first with some restriction on the time step in order to comply with the discrete maximum principle. Then two other **linear implicit** schemes satisfying the maximum principle without any condition on the time step are presented.

The finite volume scheme for the total mass conservation is assumed to be

$$\omega_i(\rho_i^{n+1} - \rho_i^n) + \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* = 0. \quad (4.14)$$

Turning to the mass fraction equation, the following choices have been made:

- The unsteady term is decomposed in two parts:

$$\begin{aligned} \Delta t^n \int_{\omega_i} \frac{\partial \rho y}{\partial t} d\Omega &\approx \left((\rho y)_i^{n+1} - (\rho y)_i^n \right) \omega_i \\ &= \rho_i^n (y_i^{n+1} - y_i^n) \omega_i + y_i^{n+1} \underbrace{(\rho_i^{n+1} - \rho_i^n) \omega_i}_{= -\Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^*}, \text{ using (4.14).} \end{aligned} \quad (4.15)$$

- The mass fraction is taken implicit in the source term

$$\int_{\omega_i} \rho \frac{\bar{y} - y}{\tau} d\Omega \approx \rho_i^\# \frac{\bar{y}_i^\# - y_i^{n+1}}{\tau_i^\#} \omega_i, \quad (4.16)$$

where $t^\# \in [t^n, t^{n+1}]$ such that:

$$\begin{cases} \rho_i^\# > 0, \\ \bar{y}_i^\# \in [0, 1], \\ \tau_i^\# > 0. \end{cases} \quad (4.17)$$

In the sequel, the choice made is $\rho_i^\# = \rho_i^n$, $\bar{y}_i^\# = \bar{y}_i^n$ and $\tau_i^\# = \tau_i^n$.

- Turning to convection contributions, two different methods are used. The first method simply considers the total convection: $\nabla \cdot (y\mathbf{q}) + \nabla \cdot (y(1-y)\mathbf{q}_r) = \nabla \cdot (y\mathbf{q}_g)$. The scheme using this method is nicknamed QG and is detailed in the sequel. It introduces a condition on the time step to preserve the discrete maximum principle. The second method treats the mixture convection and the relative convection separately. Two different schemes are detailed, labeled QRd scheme and QRq scheme. The latter two schemes approximate the non-linear convection so that the problem is well-posed and the discrete mass fraction remains in physical bounds $[0, 1]$ whatever the time step is.

In the sequel, the following notation is used for the sign of z :

$$\text{sg}(z) = \begin{cases} 1 & \text{if } z \geq 0, \\ 0 & \text{if } z < 0. \end{cases} \quad (4.18)$$

In order to simplify notations, $\text{sg}_{ij} := \text{sg}(q_{ij}^*)$, $\text{sg}_{ij}^g := \text{sg}((q_g)_{ij}^*)$ and $\text{sg}_{ij}^r := \text{sg}((q_r)_{ij}^*)$ are used. In the following, the proposed schemes are detailed for cells within the domain in Sections 4.2.3, 4.2.4 and 4.2.5. The treatment of boundary conditions is carried out in **Appendix 4.B**, and the conclusions are summarized in Section 4.2.6.

4.2.3 Global scheme with QG Scheme

Definition of QG Scheme

For the gas mass flux convection, a standard implicit upwind scheme is used [39]. With QG scheme, the global scheme is written with the sign convention, for each cell Ω_i :

$$\begin{aligned} \rho_i^n \omega_i (y_i^{n+1} - y_i^n) - \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* y_i^{n+1} \\ + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_g)_{ij}^* \left\{ \text{sg}_{ij}^g y_i^{n+1} + (1 - \text{sg}_{ij}^g) y_j^{n+1} \right\} = \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}. \end{aligned} \quad (4.19)$$

The mass fraction y_{ij}^n at interface ij involved in the gas mass flux

$$(q_g)_{ij}^* = q_{ij}^* + (1 - y_{ij}^n)(q_r)_{ij}^*, \quad (4.20)$$

is approximated here using an upwind scheme based on sign of the mixture mass flux q_{ij}^* :

$$y_{ij}^n \approx \begin{cases} y_i^n & \text{if } q_{ij}^* \geq 0, \\ y_j^n & \text{if } q_{ij}^* < 0. \end{cases} \quad (4.21)$$

Other consistent choices could be made as long as $y_{ij}^n \in [0, 1]$.

Discrete maximum principle

Now, we examine if the discrete mass fraction remains within $[0, 1]$ on the N cells with this scheme.

Property 2 (Maximum principle for the mass fraction with QG scheme): Assume that the physical parameters are such that: $\tau_k^n > 0$ and $\bar{y}_k^n \in [0, 1]$, $k \in \llbracket 1, N \rrbracket$. If the initial conditions are such that $\forall k \in \llbracket 1, N \rrbracket$, $y_k^n \in [0, 1]$, then the global scheme with QG ensures that y_i^{n+1} remains in $[0, 1]$, $\forall i \in \llbracket 1, N \rrbracket$, when the time step Δt^n satisfies, $\forall i \in \llbracket 1, N \rrbracket$:

$$(1 - y_i^n) + (1 - \bar{y}_i^n) \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n \omega_i} \sum_{j \in v(i)} S_{ij} (1 - y_{ij}^n) (q_r)_{ij}^* \geq 0. \quad (4.22)$$

□

Obviously, the condition on the time step is automatically satisfied when $\mathbf{q}_r = \mathbf{0}$.

PROOF Using equation (4.19) obtained for each cell $i \in \llbracket 1, N \rrbracket$, the system can be expressed in the following matrix form

$$\mathbf{A} \mathbf{Y}^{n+1} = \mathbf{B} \text{ with } \forall (i, j) \in \llbracket 1, N \rrbracket^2, \begin{cases} \mathbf{A} &= (a_{ij}), \\ \mathbf{B} &= (b_i), \\ \mathbf{Y}^n &= (y_i^n). \end{cases} \quad (4.23)$$

The discrete system for $\widehat{\mathbf{Y}} = \mathbf{1} - \mathbf{Y}$ can be expressed too as

$$\widehat{\mathbf{A}} \widehat{\mathbf{Y}}^{n+1} = \widehat{\mathbf{B}} = \mathbf{A} \times \mathbf{1} - \mathbf{B} \text{ with } \forall (i, j) \in \llbracket 1, N \rrbracket^2, \begin{cases} \widehat{\mathbf{B}} &= (\widehat{b}_i) = \left(\sum_j a_{ij} - b_i \right), \\ \widehat{\mathbf{A}} &= (\widehat{a}_{ij}) = (a_{ij}), \\ \widehat{\mathbf{Y}}^n &= (\widehat{y}_i^n) = (1 - y_i^n). \end{cases} \quad (4.24)$$

We also introduce the quantity $\Lambda_i(\mathbf{A})$ on each cell by $\Lambda_i(\mathbf{A}) = |a_{ii}| - \sum_{j \in v(i)} |a_{ij}|$. Its definition implies that $\Lambda_i(\mathbf{A}) = \Lambda_i(\widehat{\mathbf{A}})$. The coefficients of the matrix system and the quantity Λ_i are

$$\begin{aligned} a_{ii} &= \widehat{a}_{ii} = 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* \text{sg}_{ij}^g - q_{ij}^*), \\ a_{ij} &= \widehat{a}_{ij} = \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} (q_g)_{ij}^* (1 - \text{sg}_{ij}^g), \\ b_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n, \\ \widehat{b}_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* - q_{ij}^*). \end{aligned} \quad (4.25)$$

Assuming that $a_{ii} > 0$,

$$\Lambda_i(\mathbf{A}) = 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* - q_{ij}^*).$$

Remark: The following two formulations are equivalent:

$$\frac{1}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* - q_{ij}^*) = \frac{1}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (1 - y_{ij}^n) (q_r)_{ij}^*. \quad (4.26)$$

The conditions $\forall (i, j) \in \llbracket 1, N \rrbracket^2, a_{ii} > 0, a_{ij} \leq 0$ and $\Lambda_i > 0$ ensure that the matrix $\mathbf{A}$ is invertible and its inverse has positive coefficients (see [27, 137]). In order to satisfy the maximum principle, the vectors $\mathbf{B}$ and $\widehat{\mathbf{B}}$ also need to fulfill $b_i \geq 0$ and $\widehat{b}_i \geq 0$ with $i \in \llbracket 1, N \rrbracket$. To summarize, the conditions for the scheme are

$$\forall i \in \llbracket 1, N \rrbracket, \begin{cases} a_{ii} &> 0, & a_{ij} &\leq 0, j \in v(i), \\ \Lambda_i(\mathbf{A}) &> 0, \\ b_i &\geq 0, & \widehat{b}_i &\geq 0. \end{cases} \quad (4.27)$$

For $\tau_k^n > 0, \bar{y}_k^n \in [0, 1], k \in \llbracket 1, N \rrbracket$ and when $y_i^n \in [0, 1], i \in \llbracket 1, N \rrbracket$, the maximum principle for the mass fraction with QG scheme is satisfied when the properties (4.27) are fulfilled. Actually, the condition on $(\widehat{b}_i)$ is equivalent to (4.22). If condition (4.22) is satisfied, then the coefficients (a_{ii}) and (Λ_i) are positive as $a_{ii} \geq \Lambda_i \geq \widehat{b}_i$. The other two conditions $a_{ij} \leq 0$ and $b_i \geq 0$ are always satisfied.

□

4.2.4 Global Scheme with QRd Scheme

Definition of QRd Scheme

For the mixture mass flux contribution to the convection flux, a standard upwind scheme is once again used as in equation (4.19). The global scheme with QRd scheme is written

$$\begin{aligned} \rho_i^n \omega_i (y_i^{n+1} - y_i^n) + \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* \left\{ (1 - \text{sg}_{ij}) (y_j^{n+1} - y_i^{n+1}) \right\} \\ + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_r)_{ij}^* \left\{ \text{sg}_{ij}^r y_i^{n+1} (1 - y_j^n) + (1 - \text{sg}_{ij}^r) y_j^n (1 - y_i^{n+1}) \right\} \\ = \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}. \end{aligned} \quad (4.28)$$

Discrete maximum principle

Property 3 (Maximum principle for the mass fraction with QRd scheme): Assume that the physical parameters are such that: $\tau_k^n > 0$ and $\bar{y}_k^n \in [0, 1]$, $\forall k \in \llbracket 1, N \rrbracket$. If the initial conditions are such that $\forall k \in \llbracket 1, N \rrbracket$, $y_k^n \in [0, 1]$, then the global scheme with QRd ensures that y_i^{n+1} remains in $[0, 1]$, $\forall i \in \llbracket 1, N \rrbracket$, whatever the time step Δt^n is.

□

The scheme is rigorously conservative in space only for the discrete steady states, i.e. when $y_i^{n+1} = y_i^n$, $i \in \llbracket 1, N \rrbracket$. This is not an issue in practice, when the method is applied to the computations of steady states.

PROOF Using the same methodology as for QG scheme, a similar discrete system (noted d) is written from (4.28) for QRd scheme:

$$\mathbf{A}_d \mathbf{Y}^{n+1} = \mathbf{B}_d \text{ with } \forall (i, j) \in \llbracket 1, N \rrbracket^2, \quad \begin{cases} \mathbf{A}_d &= ((a_d)_{ij}), \\ \mathbf{B}_d &= ((b_d)_i), \\ \mathbf{Y}^n &= (y_i^n). \end{cases} \quad (4.29)$$

Using the blue color to denote the terms linked to QRd scheme, the coefficients of the matrix system and the quantity Λ_i are

$$\begin{aligned} (a_d)_{ii} &= (\hat{a}_d)_{ii} = 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} q_{ij}^* (\text{sg}_{ij} - 1) + \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - y_j^n) \right), \\ (a_d)_{ij} &= (\hat{a}_d)_{ij} = \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} q_{ij}^* (1 - \text{sg}_{ij}), \\ (b_d)_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - 1) y_j^n \right), \\ (\hat{b}_d)_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* \text{sg}_{ij}^r (1 - y_j^n) \right), \\ \Lambda_i(\mathbf{A}_d) &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - y_j^n) \right). \end{aligned} \quad (4.30)$$

For given values q_{kl}^n and $(q_r)_{kl}^n$, $(k, l) \in \llbracket 1, N \rrbracket^2$, and for $\tau_k^n > 0$, $\bar{y}_k^n \in [0, 1]$, $k \in \llbracket 1, N \rrbracket$, the following conditions are fulfilled when $y_i^n \in [0, 1]$, $i \in \llbracket 1, N \rrbracket$:

$$\forall i \in \llbracket 1, N \rrbracket, \begin{cases} (a_d)_{ii} > 0, & (a_d)_{ij} \leq 0, j \in v(i), \\ \Lambda_i(\mathbf{A}_d) > 0, \\ (b_d)_i \geq 0, & (\hat{b}_d)_i \geq 0, \end{cases} \quad (4.31)$$

whatever the time step Δt^n is. Once again using the results [27, 137], the matrix $\mathbf{A}_d$ is invertible and its inverse has positive coefficients. Hence the maximum principle for the mass fraction y is satisfied whatever the time step Δt^n is when using QRd scheme.

4.2.5 Global scheme with QRq Scheme

Definition of QRq Scheme

QRq scheme not only takes into account the sign of the relative mass flux (as QRd Scheme) but also the sign of the mixture mass flux inside the relative mass flux contribution to the convection flux. The global QRq scheme is

$$\begin{aligned} & \rho_i^n \omega_i (y_i^{n+1} - y_i^n) + \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* \left\{ (1 - \text{sg}_{ij}) (y_j^{n+1} - y_i^{n+1}) \right\} \\ & + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_r)_{ij}^* \left[\begin{aligned} & \text{sg}_{ij} \left\{ y_i^n (1 - y_i^{n+1}) (1 - \text{sg}_{ij}^r) + y_i^{n+1} (1 - y_i^n) \text{sg}_{ij}^r \right\} \\ & + (1 - \text{sg}_{ij}) \left\{ y_j^n (1 - y_j^{n+1}) (1 - \text{sg}_{ij}^r) + y_j^{n+1} (1 - y_j^n) \text{sg}_{ij}^r \right\} \end{aligned} \right] \\ & = \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}. \end{aligned} \quad (4.32)$$

Discrete maximum principle

Property 4 (Maximum principle for the mass fraction with QRq scheme): Assume that the physical parameters are such that: $\tau_k^n > 0$ and $\bar{y}_k^n \in [0, 1]$, $k \in \llbracket 1, N \rrbracket$. If the initial conditions are such that $k \in \llbracket 1, N \rrbracket$, $y_k^n \in [0, 1]$, then QRq scheme ensures that y_i^{n+1} remains in $[0, 1]$, $i \in \llbracket 1, N \rrbracket$, whatever the time step Δt^n is, provided that the mass fluxes fulfill the following conditions, when $q_{ij}^* < 0$:

- If $(q_r)_{ij}^* \geq 0$

$$q_{ij}^* + (q_r)_{ij}^* (1 - y_j^n) \leq 0. \quad (4.33)$$

- Otherwise if $(q_r)_{ij}^* < 0$

$$q_{ij}^* - (q_r)_{ij}^* y_j^n \leq 0. \quad (4.34)$$

No condition arises when $q_{ij}^* \geq 0$.

□

For a co-current flow (considered here), these conditions are automatically satisfied. Once again, the scheme is rigorously conservative in space only for the steady states ($y_i^{n+1} = y_i^n$, $i \in \llbracket 1, N \rrbracket$).

PROOF Using the same methodology as for QRd scheme, the discrete system from (4.32) is written

$$\mathbf{A}_q \mathbf{Y}^{n+1} = \mathbf{B}_q \text{ with } \forall (i, j) \in \llbracket 1, N \rrbracket^2, \begin{cases} \mathbf{A}_q &= ((a_q)_{ij}), \\ \mathbf{B}_q &= ((b_q)_i), \\ \mathbf{Y}^n &= (y_i^n). \end{cases} \quad (4.35)$$

Using the blue color to denote the terms linked to QRq scheme, the coefficients of the matrix system

and the quantity $\Lambda_i(\mathbf{A}_q)$ are

$$\begin{aligned}
 (a_q)_{ii} &= 1 + \frac{\Delta t^n}{\tau_i^n} \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} q_{ij}^* (\text{sg}_{ij} - 1) + \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* \text{sg}_{ij} (\text{sg}_{ij}^r - y_i^n) \right), \\
 (a_q)_{ij} &= \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} q_{ij}^* (1 - \text{sg}_{ij}) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (1 - \text{sg}_{ij}) (\text{sg}_{ij}^r - y_j^n), \\
 (b_q)_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - 1) (\text{sg}_{ij} y_i^n + (1 - \text{sg}_{ij}) y_j^n) \right), \\
 (\hat{b}_q)_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* \text{sg}_{ij}^r (1 - [\text{sg}_{ij} y_i^n + (1 - \text{sg}_{ij}) y_j^n]) \right).
 \end{aligned} \tag{4.36}$$

Assuming that $\forall j \in v(i), (a_q)_{ij} < 0$,

$$\Lambda_i(\mathbf{A}_q) = 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - [\text{sg}_{ij} y_i^n + (1 - \text{sg}_{ij}) y_j^n]) \right).$$

Examining the sign of these coefficients raises two conditions on the mass fluxes .

- When $q_{ij}^* < 0$ and $(q_r)_{ij}^* \geq 0$

$$q_{ij}^* + (q_r)_{ij}^* (1 - y_j^n) \leq 0. \tag{4.37}$$

- When $q_{ij}^* < 0$ and $(q_r)_{ij}^* < 0$

$$q_{ij}^* - (q_r)_{ij}^* y_j^n \leq 0. \tag{4.38}$$

When q_{kl}^n and $(q_r)_{kl}^n, (k, l) \in \llbracket 1, N \rrbracket^2$ satisfy the previous conditions (4.37)-(4.38), and if $\tau_k^n > 0$, $\bar{y}_k^n \in [0, 1], k \in \llbracket 1, N \rrbracket$, the following properties are satisfied when $y_i^n \in [0, 1], i \in \llbracket 1, N \rrbracket$:

$$\forall i \in \llbracket 1, N \rrbracket, \begin{cases} (a_q)_{ii} > 0 & , & (a_q)_{ij} \leq 0, j \in v(i), \\ \Lambda_i(\mathbf{A}_q) > 0, \\ (b_q)_i \geq 0 & , & (\hat{b}_q)_i \geq 0, \end{cases} \tag{4.39}$$

whatever the time step Δt^n is. Once more, using common results of [27, 137], the matrix $\mathbf{A}_q$ is invertible and its inverse has positive coefficients. Hence the maximum principle for the mass fraction with QRq scheme is satisfied whatever the time step Δt^n is, as long as the conditions (4.37)-(4.38) on the mass fluxes are respected.

4.2.6 Conclusion for the boundary conditions

For each scheme, a study of the finite volume scheme for boundary cells is detailed in **Appendix 4.B** when considering a given valid mass fraction $y_\infty \in [0, 1]$ outside. The main conclusions for each scheme are summarized here. If a co-current flow is considered, QRq scheme still satisfies the discrete maximum principle without any condition on the time step. Turning to QG scheme, a slightly different condition on the time step arises for cells sharing a face with the boundary. This condition must be taken into account as it can be the most constraining one. Eventually, QRd scheme still has no limit on the time step. However, the boundary flux used in the QRd scheme must be handled carefully on an outlet face (with respect to the mixture mass flux $q_{i\infty} > 0$ defined in Figure 4.2). When the mass fraction is unknown on this face, the expression of the flux should be modified or another boundary flux should be preferred (such as the one used in QRq Scheme).

4.3 Numerical verification of the schemes

In this section, regular analytical solutions from Section 3.1 are used to compare the accuracy of each scheme, for steady-state solutions and one unsteady case. First, the spatial consistency and convergence rate of the schemes for two steady state solutions are studied. If the scheme is stable and conservative (for steady-state conditions), consistency ensures that the finite volume scheme converges to the correct solution as the mesh size tends to zero [38]. The order of convergence indicates how fast the solution approaches the exact solution as the mesh size decreases. The proposed schemes are analytically first-order accurate. Subsequently, a database of 600 constant-parameter cases (solution from Section 3.1.1) is launched for each scheme to determine the performance of the schemes as a function of relative velocity magnitude, enabling an initial estimate of the accuracy in the ranges of the Chexal-Lellouche and the Bestion correlations. A second database of 200 cases is launched for an analytical solution with the approximated Bestion correlation (solution from Section 3.1.2), in order to take a closer look at the behavior of schemes with this correlation. Finally, the transient solution from Section 3.1.3 is considered in order to estimate the performance of the QG Scheme to simulate unsteady cases. **Appendix 4.C** explains why the QRd and QRq Schemes are not suitable for unsteady simulations.

For each case, 1D configurations of the disequilibrium equation are considered on domain $\Omega = [0, 1]$. For steady-state solutions, the solution is not explicit and must be computed. A dichotomy is used with an accuracy of 10^{-12} .

4.3.1 Consistency and convergence rate of the schemes for steady state solutions

In this section, the regular steady-state analytical solution from Section 3.1.1 is considered for Equation (4.1). The parameters $(\rho, u, \bar{y}, \tau, u_r)$ are considered constant and uniform on the domain. They are noted with the subscript 0. A Dirichlet boundary condition on the left of the domain $y(x = 0) = 0$ is applied. Two test cases corresponding to two configurations representative of a nuclear reactor core are considered. They are noted Test Case 1 and Test Case 2. Table 4.1 gives the parameters of the two test cases. Test Case 1 approaches the behavior of a reactor core flow under accidental conditions, whereas Test Case 2 simulates the nominal flow during normal operation at the bottom of the core, with the occurrence of subcooled boiling. Steady-state numerical solutions are obtained for each scheme with an unsteady simulation using a fixed time step of

$$\Delta t = 0.01 \text{ s.} \quad (4.40)$$

Quantity	Test case 1	Test case 2	Unit
ρ_0	500	700	$kg.m^{-3}$
$\bar{y}_0$	0.3	0.0304	—
u_0	7	5	$m.s^{-1}$
u_{r0}	1	-4.3	$m.s^{-1}$
τ_0	1.5	0.01	s

TABLE 4.1

Parameters for the test cases representative of reactor core conditions.

The steady-state numerical solution y_{num}^{grd} for QRd scheme and y_{num}^{grq} for QRq scheme as well as the exact solution y_{ex} are computed, using a mesh size $\Delta x = 10^{-3}$ m. Results are given in Figure 4.3 for Test Case 1 and in Figure 4.4 for Test Case 2 for a number of cells $n_x = 1000$. The steady-state solution for QG scheme is not represented here as the solution is very similar to the one obtained

with QRq scheme. QRd and QRq schemes seem to give similar results for Test Case 1. For Test Case 2 where the solution adopts a logarithmic form, QRd scheme is less accurate than QRq scheme.

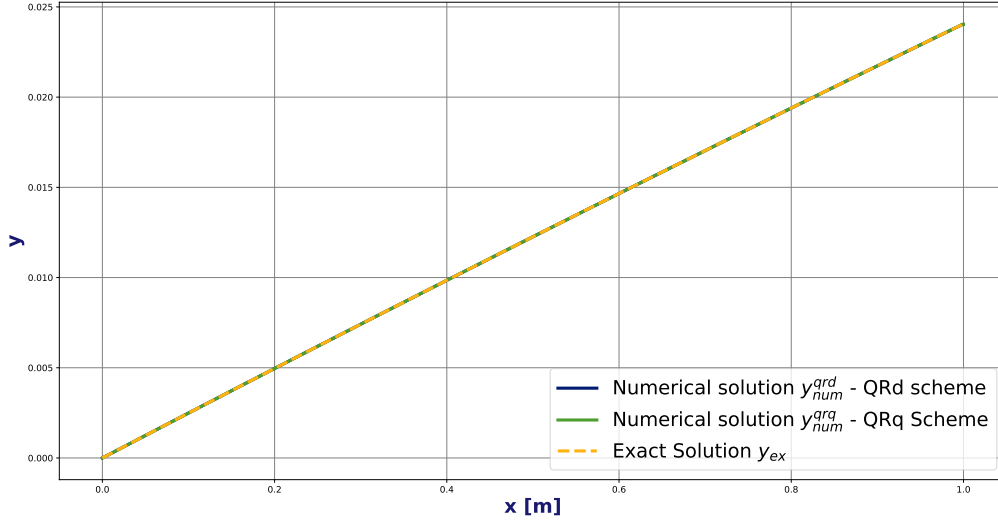


FIGURE 4.3

Test Case 1 - Mass fraction y as a function of x - QRd scheme (blue line), QRq scheme (green line) and exact (dotted orange) solutions.

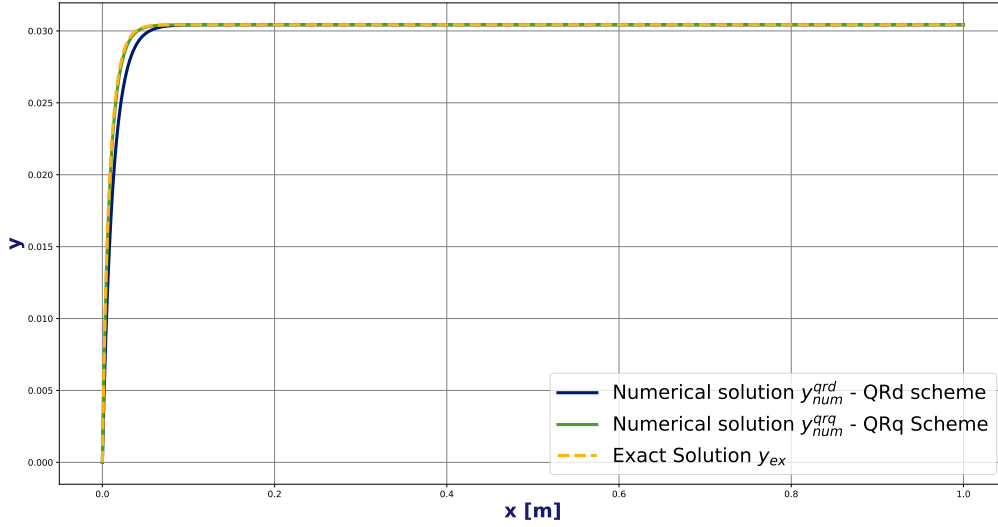


FIGURE 4.4

Test Case 2 - Mass fraction y as a function of x - QRd scheme (blue line), QRq scheme (green line) and exact (dotted orange) solutions.

Figure 4.5 (resp. 4.6) shows the convergence rate with the three schemes for Test Case 1 (resp. Test Case 2) using different mesh sizes: $\Delta x \in [10^{-5}, 10^{-1}]$ m. The error between the numerical solution and the exact solution is called L_2 -norm. It is computed for each mesh size and defined by

$$\text{err}_2 = \frac{\|y_{\text{num}} - y_{\text{ex}}\|_2}{\|y_{\text{ex}}\|_2} \text{ with } \|x\|_2 = \sqrt{\frac{1}{n_x} \sum_{i=1}^{n_x} x_i^2}, \quad n_x = \left\lfloor \frac{L}{\Delta x} \right\rfloor. \quad (4.41)$$

As expected, the three schemes comply with the convergence rate of first order in space. For a given mesh size, QRq scheme and QG scheme have similar accuracy for Test Case 1 and Test Case 2. While QRd scheme is as accurate as the others for Test Case 1, it is less accurate for Test Case 2.

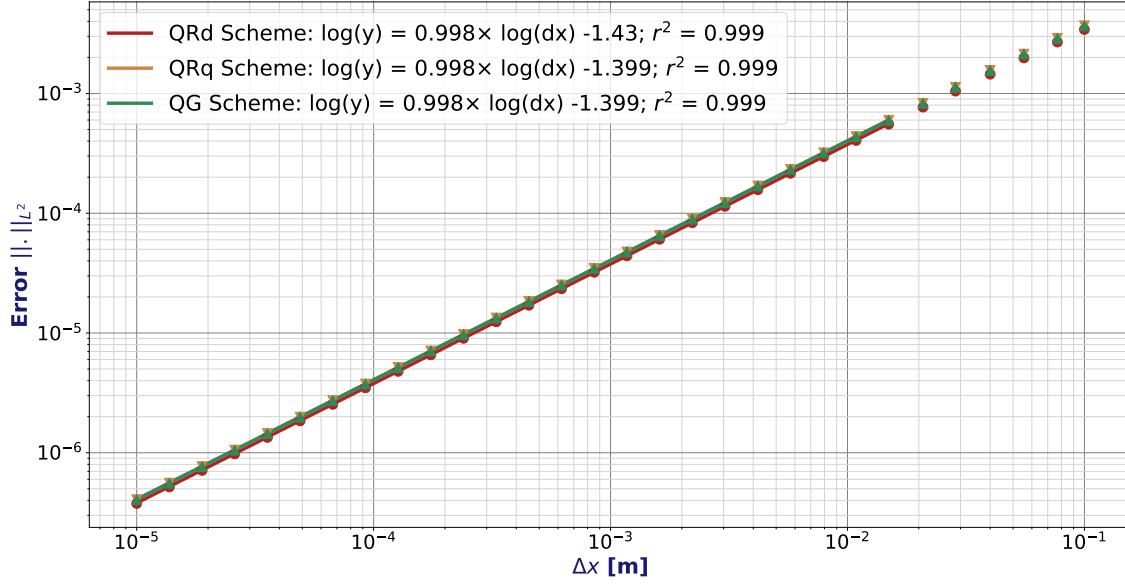


FIGURE 4.5

Test Case 1 - L_2 -norm of the error as a function of the mesh size for QG (green), QRd (red) and QRq (orange) schemes.

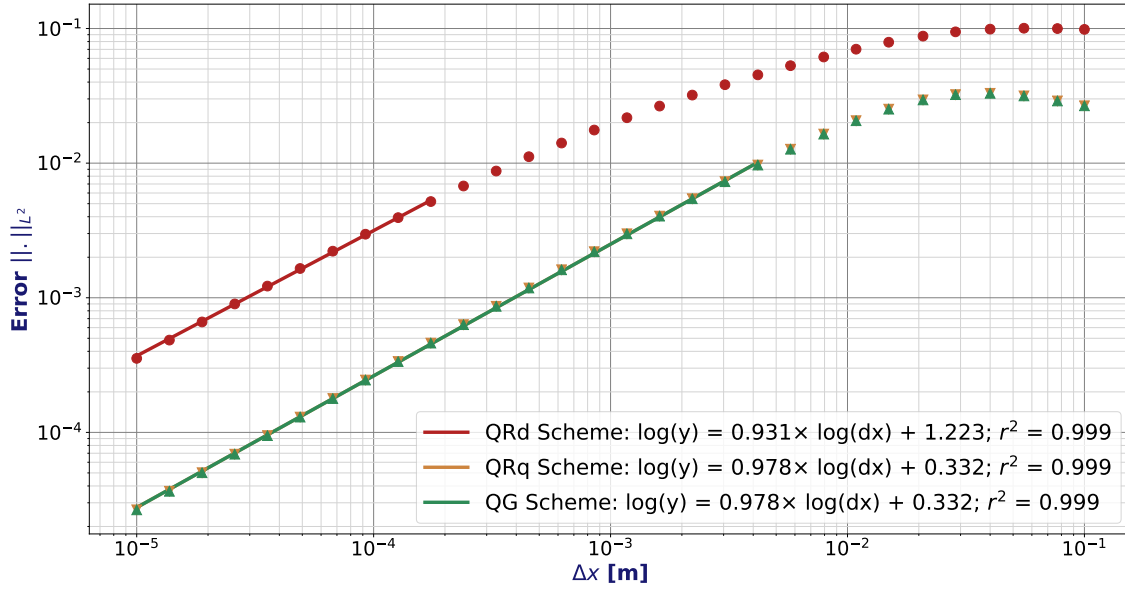


FIGURE 4.6

Test Case 2 - L_2 -norm of the error as a function of the mesh size for QG (green), QRd (red) and QRq (orange) schemes.

4.3.2 Comparison for steady-state simulations with constant parameters

The two previous test cases verified the consistency and the expected convergence rate of 1 for steady-state solutions. However, different accuracies can be observed, particularly for Test Case 2. The aim is to extend this work to a larger number of cases in order to compare the accuracy of the schemes. To this end, the constant-parameter solution of Section 3.1.1 is still considered.

$$\begin{cases} u_0 = 5 \text{ m.s}^{-1}, \\ \rho_0 = 700 \text{ kg.m}^{-3}, \\ y_0 = 0. \end{cases} \quad (4.42)$$

Considering 10 logarithmically spaced values for $\bar{y}_0$ in the interval $[0.01, 0.5]$, the accuracy of the schemes is evaluated by scanning 20 values for the ratio λ_r in the interval $[-0.9, 0.5]$. This methodology is repeated for 3 different values of relaxation time $\tau \in \{0.01, 1, 100\}$. This database of 600 cases approaching the characteristics of a reactor core is used to test the accuracy of each scheme. Considering the two correlations presented in Section 1.4.2, the ratio λ_r obtained for reactor conditions are $\lambda_r \in [0.0, 0.05]$ for the Bestion correlation and $\lambda_r \in [-0.9, -0.5]$ for the Chexal-Lellouche one. This method is a way to study the approximate behavior of the scheme for each correlation on regular steady-state solutions.

For numerical simulations, the number of cells in the mesh is 1000. The time step used to reach the steady-state solution is $\Delta t = 0.01$ s and the stopping criterion is set to 10^{-13} . For each scheme and each case, the L_2 -norm of the error is computed. Figure 4.7 represents the L_2 -norm (averaged over the different $\bar{y}_0$) as a function of the λ_r ratio for the three relaxation times considered. The standard deviation is shown in dotted lines. The λ_r domain corresponding to each relative velocity correlation is shown on the figure, in orange for Bestion and blue for Chexal-Lellouche. Whatever the domain, the accuracy increases with increasing relaxation time. In the Bestion domain, the accuracy of all 3 schemes is very similar. This is also the case for all positive relative velocities. In this domain, accuracy depends very little on the equilibrium mass fraction. As the relative velocity becomes negative, accuracies deteriorate. In the Chexal-Lellouche domain, the QG and QRq schemes have the same accuracy, which increases slightly with decreasing relative velocity. The QRd scheme is less accurate than the other two for negative λ_r . Whatever the scheme, accuracy varies slightly as a function of the equilibrium mass fraction, even more so as relaxation time increases.

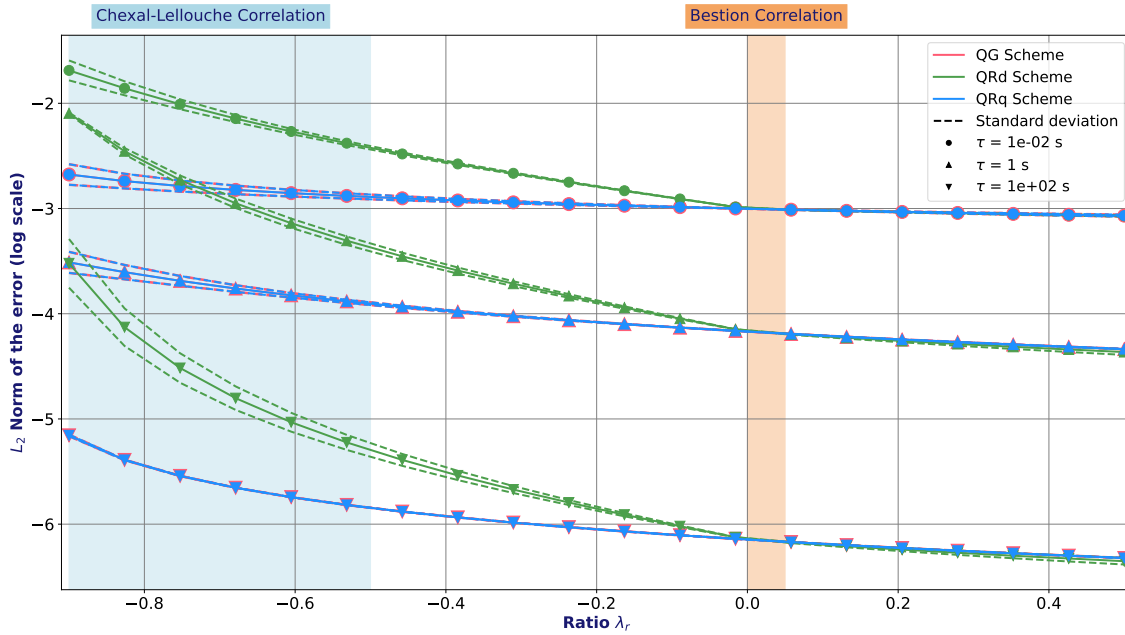


FIGURE 4.7

L_2 Norm of the error as a function of the ratio λ_r (averaged over 10 values of $\bar{y}_0$) for each scheme and three different values of τ_0 (No difference between the three schemes in the Bestion interval).

For steady-state solutions with a relative velocity magnitude of the order of that obtained with the Chexal-Lellouche correlation, QG and QRq schemes are more accurate than QRd scheme. The QG and QRq schemes are therefore recommended when a steady-state case is simulated with Chexal-Lellouche. This study does not draw any conclusion for the Bestion correlation, where all three schemes appear to be equally accurate.

4.3.3 Comparison for steady-states with the approximated Bestion correlation

In order to better study the behavior of the schemes when applying the Bestion correlation, the analytical solution from Section 3.1.2 is investigated. The hydraulic diameter considered is $d_h = 0.011185$ m, a characteristic value for reactor cores. The relative velocity of the approximated Bestion correlation (see **Appendix 1.C**) is

$$u_r = \frac{1 + (\delta - 1)y}{1 - y} u_b(\delta). \quad (4.43)$$

With the exception of the relative velocity, all parameters are assumed constant and uniform, with the same mixture velocity and density as the previous cases (see Equation (4.42)). Only one relaxation time $\tau_0 = 1$ s is simulated here. The interval of the density ratio δ used in Section 1.C is used once again with 20 values taken from this interval. For each value of ratio δ , 10 values for the equilibrium mass fraction are once again used. The resulting database contains 200 test cases. For numerical simulations, the **same numerical parameters** are used as in the previous section. Figure 4.8 represents the L_2 -norm of the error (averaged over $\bar{y}_0$) as a function of the ratio δ for each scheme. The standard deviation is shown in dotted lines. Whatever the scheme, accuracy increases as the ratio δ increases. Equilibrium mass fraction plays an increasing role on accuracy as the ratio δ increases. The three schemes have similar accuracies. However, QRd is slightly more accurate than the other two, QRq and QG.

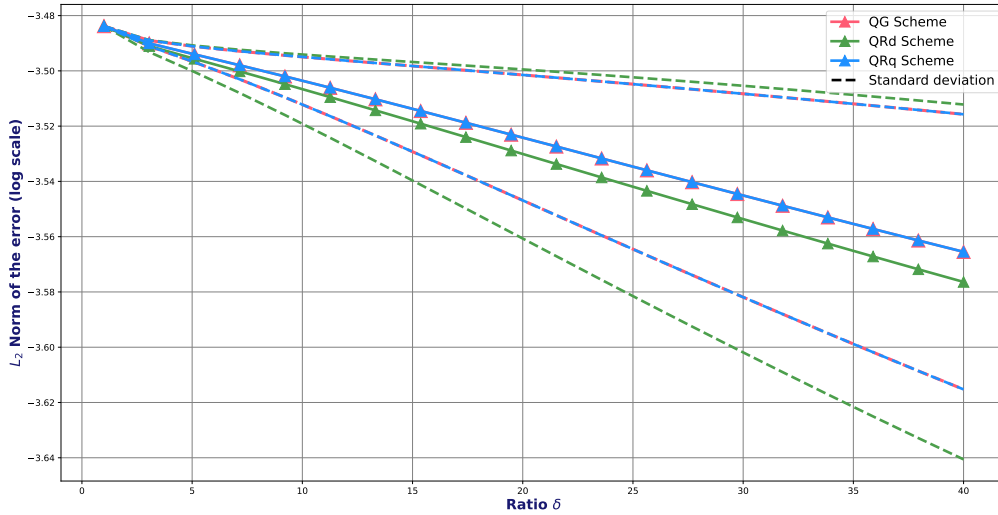


FIGURE 4.8

L_2 -norm of the error as a function of the ratio δ (averaged over 10 values of $\bar{y}_0$) for each scheme - $\tau_0 = 1$ s.

4.3.4 Comparison of the schemes for unsteady simulations

In this section, the transient auto-similar analytical solution of Section 3.1.3 is used to determine the accuracy of the QG Scheme for unsteady simulations. The QRd and QRq Schemes are not conservative in time for unsteady conditions. **Appendix 4.C** provides an explanation of why these schemes cannot be used to study unsteady solutions.

The solution considered here is obtained with the approximated Bestion correlation and the following parameters are considered:

$$\begin{cases} \delta = 10, \\ y_0 = 0.2, \\ d_h = 0.011185 \text{ m}. \end{cases} \quad (4.44)$$

The spatial domain studied is $\Omega = [0, 1]$. Using the variable $\xi = \frac{x}{u_0(t+t_0)}$, for $t_0 = 1$ s, the solution y , the mass fraction at equilibrium $\bar{y}$ and the relaxation time τ are given by

$$\begin{cases} y(\xi) = \left(1 + \frac{1}{2} \sin(\chi_0 \xi)\right) y_0, \\ \tau(x, t) = \frac{t + t_0}{\chi_0(1 + \lambda_b + \alpha_c y(\xi) - \xi)}, \\ \bar{y}(\xi) = \left(1 + \frac{1}{2} \cos(\chi_0 \xi) + \frac{1}{2} \sin(\chi_0 \xi)\right) y_0, \end{cases} \quad (4.45)$$

with $\chi_0 = 20$, $\alpha_c = 0.67$ and $\lambda_b = 0.04$. The test case is computed on the time interval $[t_1, t_2]$ with $t_1 = 0$ s and $t_2 = 0.1$ s. Using $u_0 = 5.0 \text{ m.s}^{-1}$, the initial time t_1 satisfies the constraint

$$t_1 + t_0 > \frac{L}{u_0(1 + \lambda_b)} = 0.193 \text{ s}. \quad (4.46)$$

The solution obtained is studied at time t_2 , computed with a CFL condition satisfying

$$\text{CFL} = \frac{u_0 \Delta t}{\Delta x} = 0.5. \quad (4.47)$$

Figure 4.9 represents, at instant t_2 , the analytical and numerical solutions obtained with the QG Scheme for a mesh of $n_x = 1000$ cells. The solution for the QG Scheme seems to capture the right form of the analytical solution.

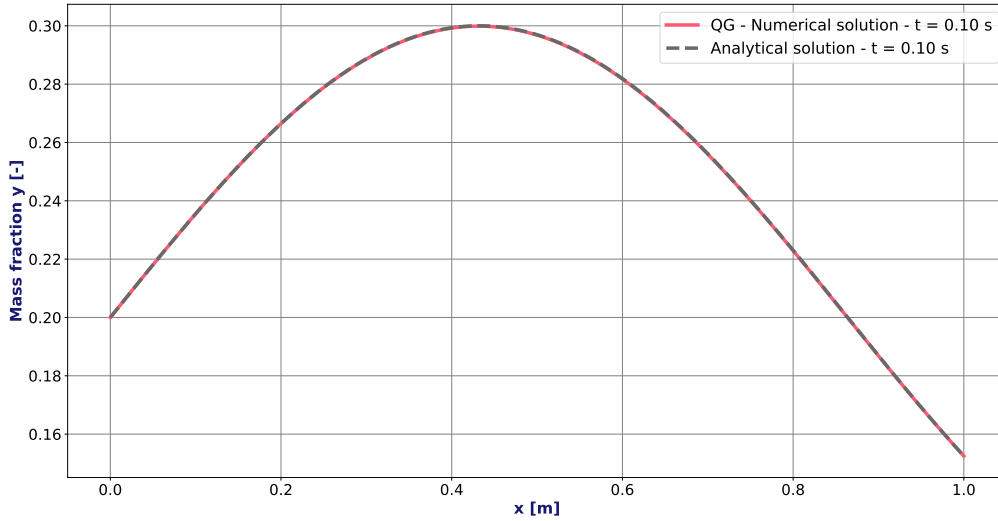


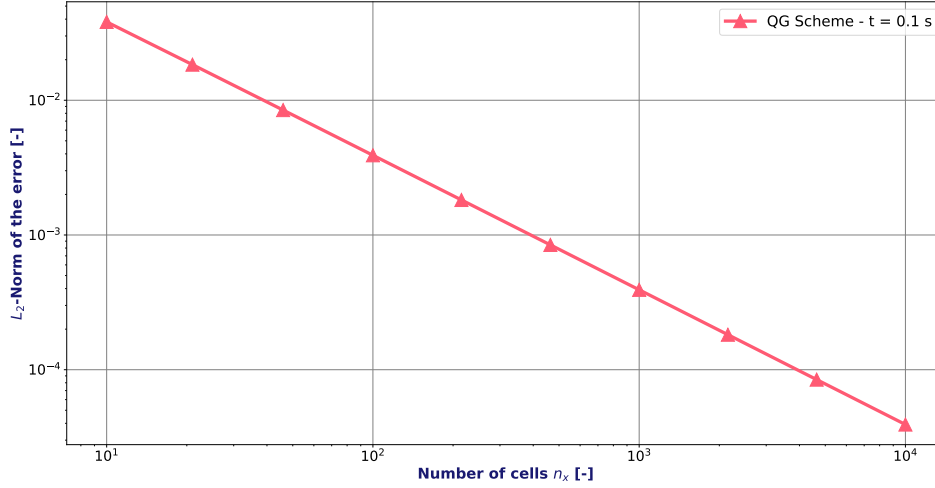
FIGURE 4.9

Analytical and numerical solutions for the QG Scheme at time $t_2 = 0.1$ s - $n_x = 1000$.

A convergence study is done with a mesh size Δx varying in $[10^{-4}, 10^{-1}]$ m, corresponding to a number of cells n_x varying in $[10, 10000]$. The simulations are still computed with

$$\text{CFL} = 0.5. \quad (4.48)$$

In Figure 4.10, the L_2 -norm error is represented as a function of the number of cells for the QG Scheme. Table 4.2 represents the numerical values of the convergence rate computed for QG Scheme. It can be observed that QG scheme converges to the correct solution with a convergence rate that tends to 1, as expected [38]. The QG Scheme can be used for unsteady simulations.

**FIGURE 4.10**

L_2 -norm of the error as a function of the number of cells n_x for the QG Scheme - $t_2 = 0.1$ s.

Number of cells	Convergence rate
10	
21	0.9843
46	0.9903
100	0.9949
215	0.9975
464	0.9988
1000	0.9994
2154	0.9997
4641	0.9998
10000	0.9999

TABLE 4.2

Spatial convergence rate for the QG Scheme, computed at $t_2 = 0.1$ s at constant CFL.

4.3.5 Conclusion

These initial tests on simple 1D solutions enable to assess the approximate performance of the various schemes. Despite a slight constraint on the time step, the QG scheme seems more suitable for all applications. Indeed, it shows a convergence order of 1 in both space and time, is the most accurate of the three schemes, and can be used for unsteady simulations. If the time-step condition of QG scheme becomes limiting for steady-state applications, the QRd and QRq Schemes can be used. In that case, the QRq Scheme would then be recommended when using the Chexal-Lellouche correlation (or other correlations leading to a negative relative velocity). However, care must be taken in cases where the two-phase flow is no longer co-current. For counter-current flows where the time-step constraint of the QG Scheme is limiting, the QRd Scheme can be used. For unsteady simulations, the QRd and QRq Schemes cannot be used, only the QG Scheme is valid.

All the results obtained here are obtained with 1D simulations and with most parameters taken constant and uniform. This is not necessarily representative of a 3D case where the mass fraction equation is coupled to the complete system and to a complex mixture equation of state. However, these verification cases allow for the identification of global trends which are confirmed in **Chapter 5**, where the industrial code THYC-coeur is used on 3D cases of reactor cores.

4.A Continuous maximum principle

The studied equations are

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{q} &= 0, \\ \frac{\partial(\rho y)}{\partial t} + \nabla \cdot (y \mathbf{q}) + \nabla \cdot (y(1-y) \mathbf{q}_r) &= \rho \frac{\bar{y} - y}{\tau}. \end{aligned} \quad (4.49)$$

The notation $\xi = y(1-y)$ is used to study the maximum principle for y . Indeed, if $\xi \geq 0$ then

$$y(1-y) \geq 0 \Leftrightarrow y \in [0, 1]. \quad (4.50)$$

The governing equation for ξ can be obtained from the non-conservative equations of y and $1-y$ as described below:

$$\begin{aligned} & \left(\rho \partial_t y + \mathbf{q} \cdot \nabla y + \nabla \cdot (\xi \mathbf{q}_r) = \rho \frac{\bar{y} - y}{\tau} \right) \times (1-y) \\ & + \left(\rho \partial_t (1-y) + \mathbf{q} \cdot \nabla (1-y) - \nabla \cdot (\xi \mathbf{q}_r) = \rho \frac{(1-\bar{y}) - (1-y)}{\tau} \right) \times y. \end{aligned} \quad (4.51)$$

Then

$$\rho \partial_t \xi + \mathbf{q} \cdot \nabla \xi + (1-2y) \nabla \cdot (\xi \mathbf{q}_r) = \rho \frac{S - 2\xi}{\tau} \text{ with } S = y(1-\bar{y}) + \bar{y}(1-y). \quad (4.52)$$

Using the standard notations $\xi_+ = \max(\xi, 0)$ and $\xi_- = -\min(\xi, 0)$, equation (4.52) is multiplied by $(-\xi_-)$

$$\left(\rho \partial_t \left(\frac{\xi_-^2}{2} \right) + \mathbf{q} \cdot \nabla \left(\frac{\xi_-^2}{2} \right) \right) + (1-2y) \left(\xi_-^2 \nabla \cdot \mathbf{q}_r - \xi_- \mathbf{q}_r \cdot \nabla \xi \right) = \rho \frac{-S \xi_- - 2\xi_-^2}{\tau}, \quad (4.53)$$

$$\begin{aligned} & \partial_t \left(\rho \frac{\xi_-^2}{2} \right) + \nabla \cdot \left(\mathbf{q} \frac{\xi_-^2}{2} \right) + \nabla \cdot \left((1-2y) \mathbf{q}_r \frac{\xi_-^2}{2} \right) \\ & + \xi_-^2 \left\{ (1-2y) \nabla \cdot \mathbf{q}_r - \frac{1}{2} \nabla \cdot \left((1-2y) \mathbf{q}_r \right) \right\} = -\frac{\rho(S + 2\xi_-) \xi_-}{\tau}. \end{aligned} \quad (4.54)$$

Finally, the equation obtained is

$$\partial_t \left(\rho \frac{\xi_-^2}{2} \right) + \nabla \cdot \left([\mathbf{q} + (1-2y) \mathbf{q}_r] \frac{\xi_-^2}{2} \right) + \frac{\xi_-^2}{2} \left\{ (1-2y) \nabla \cdot \mathbf{q}_r - \mathbf{q}_r \cdot \nabla (1-2y) \right\} = -\frac{\rho(S + 2\xi_-) \xi_-}{\tau}. \quad (4.55)$$

We define the quantity $E(t) = \int_{\Omega} \rho \frac{\xi_-^2}{2} d\Omega$ and split the outside surface domain according to the sign of the mass flux $\mathbf{q}_\xi = \mathbf{q} + (1-2y) \mathbf{q}_r$:

$$\begin{cases} \Gamma_w^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma = 0\}, \\ \Gamma_+^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma < 0\}, \\ \Gamma_-^\xi = \{\mathbf{x} \in \Gamma, \mathbf{q}_\xi \cdot \mathbf{n}_\Gamma > 0\}. \end{cases} \quad (4.56)$$

Then, the quantity $E(t)$ satisfies

$$\begin{aligned} \frac{dE(t)}{dt} = & - \int_{\Gamma} \left[(\mathbf{q} + (1-2y)\mathbf{q}_r) \frac{\xi_-^2}{2} \right] \cdot \mathbf{n}_{\Gamma} d\Gamma - \int_{\Omega} \frac{\xi_-^2}{2} \left\{ (1-2y)\nabla \cdot \mathbf{q}_r - \mathbf{q}_r \cdot \nabla(1-2y) \right\} d\Omega \\ & - \int_{\Omega} \frac{\rho(S+2\xi_-)}{\tau} \xi_-. \end{aligned} \quad (4.57)$$

Assumptions:

1. Relevant mass fraction on the boundary conditions: $\xi_-(\mathbf{x} \in \Gamma_+^{\xi}, t) = 0$.
2. Mass fraction for the initial condition such that: $\xi_-(\mathbf{x} \in \Omega, t = 0) = 0$ and thus $E(t = 0) = 0$.
3. $\frac{1}{\rho} \left[(1-2y)\nabla \cdot \mathbf{q}_r - \mathbf{q}_r \cdot \nabla(1-2y) \right] \in \mathcal{L}^{\infty}(\Omega, [0, T])$.
4. Equilibrium mass fraction such that: $\bar{y} \in [0, 1]$ implying $(S+2\xi_-) \geq 0$ (proof of lemma below).
5. Positive relaxation time scale $\tau > 0$ and density $\rho > 0$.

Thus we have

$$\frac{dE(t)}{dt} \leq - \int_{\Omega} \rho \frac{\xi_-^2}{2} \left[\frac{1}{\rho} \left((1-2y)\nabla \cdot \mathbf{q}_r - \mathbf{q}_r \cdot \nabla(1-2y) \right) \right] d\Omega. \quad (4.58)$$

Assuming the assumptions previously presented,

$$\frac{dE(t)}{dt} \leq \left\| \frac{1}{\rho} \left[(1-2y)\nabla \cdot \mathbf{q}_r - \mathbf{q}_r \cdot \nabla(1-2y) \right] \right\|_{\infty} E(t). \quad (4.59)$$

Using the Grönwall's inequality [64] and $E(0) = 0$, it enables to conclude that $E(t) = 0$ on $[0, T]$.

Lemma: If $\bar{y} \in [0, 1]$ then $S + 2\xi_- \geq 0$ with $S = \bar{y}(1-y) + y(1-\bar{y})$

PROOF Let's assume that $\bar{y} \in [0, 1]$.

Case $\xi_- = 0$

If $\xi_- = 0$, then $y \in [0, 1]$ implying that $S \geq 0$.

As a consequence, we have $S + 2\xi_- \geq 0$

Case $\xi_- \neq 0$

$$\xi = -\xi_- < 0, \text{ thus } \begin{cases} y < 0, \\ \text{or } y > 1. \end{cases} \quad (4.60)$$

If $y < 0$, using $S = \bar{y} + y(1-2\bar{y})$, the following inequality is satisfied:

$$\begin{aligned} \Rightarrow \quad & \begin{array}{llll} y & \leq & S & \leq 1-y \\ Sy & \geq & y(1-y) = \xi & \\ \text{and} & S(1-y) & \geq & y(1-y) = \xi. \\ \text{So} & S(y+1-y) = S & \geq & 2\xi = -2\xi_-. \end{array} \end{aligned} \quad (4.61)$$

If $y > 1$, the same proof can be used in a symmetric way. Finally, we have $S + 2\xi_- \geq 0$ for $y \in \mathbb{R}$. $\square$

4.B Extension of the schemes to boundaries conditions

As shown in Figure 4.2, we consider a cell i containing several faces shared with $j \in v(i)$ neighboring cells, a wall face (noted w) and an inlet/outlet face (noted with an index $i\infty$). The inlet/outlet face has a surface $S_{i\infty}$, a given valid mass fraction $y_\infty \in [0, 1]$ outside and an outward unit normal $\mathbf{n}_\infty$. The mass fluxes on this face are noted

$$\begin{aligned} q_{i\infty} &= \mathbf{q}_{i\infty} \cdot \mathbf{n}_\infty, \\ (q_r)_{i\infty} &= (\mathbf{q}_r)_{i\infty} \cdot \mathbf{n}_\infty, \\ (q_g)_{i\infty} &= q_{i\infty} + (1 - y_{i\infty})(q_r)_{i\infty}, \end{aligned} \tag{4.62}$$

where

$$y_{i\infty} = \begin{cases} y_i^n & \text{if } q_{i\infty} \geq 0, \\ y_\infty & \text{if } q_{i\infty} < 0. \end{cases} \tag{4.63}$$

Only one inlet/outlet face is considered here but the discussion is valid for several ones. For quantities on the inlet/outlet boundary face, the instants considered are not specified here for the sake of readability. These values are given as data, hence any consistent formula can be used, with $y_\infty \in [y_\infty^n, y_\infty^{n+1}]$. Conditions (4.27), (4.31) and (4.39) are computed in order to satisfy the discrete maximum principle for cell i . Blue terms are terms added due to the inlet/outlet boundary face.

4.B.1 Boundary condition for QG scheme

For cell i from Figure 4.2, QG scheme is written

$$\begin{aligned} \rho_i^n \omega_i (y_i^{n+1} - y_i^n) - \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* y_i^{n+1} - \Delta t^n S_{i\infty} q_{i\infty} y_i^{n+1} \\ + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_g)_{ij}^* \left\{ \text{sg}_{ij}^g y_i^{n+1} + (1 - \text{sg}_{ij}^g) y_j^{n+1} \right\} \\ + \Delta t^n S_{i\infty} (q_g)_{i\infty} \left\{ \text{sg}_{i\infty}^g y_i^{n+1} + (1 - \text{sg}_{i\infty}^g) y_\infty \right\} \\ = \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}. \end{aligned} \tag{4.64}$$

Using this scheme, coefficients arising from conditions (4.27) are

$$\begin{aligned}
 a_{ii} &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* \text{sg}_{ij}^g - q_{ij}^*) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} ((q_g)_{i\infty} \text{sg}_{i\infty}^g - q_{i\infty}), \\
 a_{ij} &= \Delta t^n \frac{S_{ij}}{\omega_i} \frac{(q_g)_{ij}^*}{\rho_i^n} (1 - \text{sg}_{ij}^g), \\
 b_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n - \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_g)_{i\infty} (1 - \text{sg}_{i\infty}^g) y_\infty, \\
 \hat{b}_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* - q_{ij}^*) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} ((q_g)_{i\infty} (y_\infty + (1 - y_\infty) \text{sg}_{i\infty}^g) - q_{i\infty}).
 \end{aligned} \tag{4.65}$$

Assuming that $a_{ii} > 0$,

$$\begin{aligned}
 \Lambda_i(\mathbf{A}) &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} ((q_g)_{ij}^* - q_{ij}^*) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} ((q_g)_{i\infty} \text{sg}_{i\infty}^g - q_{i\infty}).
 \end{aligned}$$

The three conditions $a_{ii} > 0$, $\Lambda_i > 0$ and $\hat{b}_i \geq 0$ are not always fulfilled. Using the formula

$$(q_g)_{i\infty} (y_\infty + (1 - y_\infty) \text{sg}_{i\infty}^g) \leq \text{sg}_{i\infty}^g (q_g)_{i\infty}, \tag{4.66}$$

it can be stated that $a_{ii} \geq \Lambda_i \geq \hat{b}_i$. The only remaining condition on the time step is once again on $\hat{b}_i$. This condition, which is slightly different from (4.22), should be monitored as it can become the most constraining one.

4.B.2 Boundary condition for QRd scheme

For cell i from Figure 4.2, QRd scheme is written

$$\begin{aligned}
 &\rho_i^n \omega_i (y_i^{n+1} - y_i^n) + \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* \left\{ (1 - \text{sg}_{ij}^g) (y_j^{n+1} - y_i^{n+1}) \right\} \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} S_{i\infty} q_{i\infty} (1 - \text{sg}_{i\infty}^g) (y_\infty - y_i^{n+1}) \\
 &\quad + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_r)_{ij}^* \left\{ \text{sg}_{ij}^r y_i^{n+1} (1 - y_j^n) + (1 - \text{sg}_{ij}^r) y_j^n (1 - y_i^{n+1}) \right\} \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} S_{i\infty} (q_r)_{i\infty} \left\{ \text{sg}_{i\infty}^r y_i^{n+1} (1 - y_\infty) + (1 - \text{sg}_{i\infty}^r) y_\infty (1 - y_i^{n+1}) \right\} \\
 &= \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}.
 \end{aligned} \tag{4.67}$$

Using this scheme, coefficients arising from conditions (4.31) are

$$\begin{aligned}
 (a_d)_{ii} &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} q_{ij}^* (\text{sg}_{ij} - 1) + \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - y_j^n) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (\text{sg}_{i\infty} - 1) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} (\text{sg}_{i\infty}^r - y_\infty), \\
 (a_d)_{ij} &= \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} q_{ij}^* (1 - \text{sg}_{ij}), \\
 (b_d)_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - 1) y_j^n \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (\text{sg}_{i\infty} - 1) y_\infty + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} (\text{sg}_{i\infty}^r - 1) y_\infty, \\
 (\hat{b}_d)_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* \text{sg}_{ij}^r (1 - y_j^n) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (\text{sg}_{i\infty} - 1) (1 - y_\infty) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} \text{sg}_{i\infty}^r (1 - y_\infty), \\
 \Lambda_i(\mathbf{A}_d) &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (\text{sg}_{ij}^r - y_j^n) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (\text{sg}_{i\infty} - 1) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} (\text{sg}_{i\infty}^r - y_\infty).
 \end{aligned} \tag{4.68}$$

Once more, no condition on the time step arises for the QRd scheme. However, it should be handled carefully for an outlet boundary (with respect to mixture mass flux $q_{i\infty} > 0$). Indeed, whatever the sign of the relative mass flux $(q_r)_{i\infty}$, the flux requires a given value of y_∞ . If the value of the outlet mass fraction y_∞ is unknown, a different consistent formula should be used to approximate $y_\infty \in [0, 1]$ (for instance $y_\infty = y_i^n$). Otherwise the flux should be modified.

4.B.3 Boundary condition for QRq scheme

For cell i from Figure 4.2, QRq scheme is written

$$\begin{aligned}
 &\rho_i^n \omega_i (y_i^{n+1} - y_i^n) + \Delta t^n \sum_{j \in v(i)} S_{ij} q_{ij}^* \left\{ (1 - \text{sg}_{ij}) (y_j^{n+1} - y_i^{n+1}) \right\} \\
 &\quad + \Delta t^n S_{i\infty} q_{i\infty} (1 - \text{sg}_{i\infty}) (y_\infty - y_i^{n+1}) \\
 &\quad + \Delta t^n \sum_{j \in v(i)} S_{ij} (q_r)_{ij}^* \left[\begin{aligned} &\text{sg}_{ij} \left\{ y_i^n (1 - y_i^{n+1}) (1 - \text{sg}_{ij}^r) + y_i^{n+1} (1 - y_i^n) \text{sg}_{ij}^r \right\} \\ &+ (1 - \text{sg}_{ij}) \left\{ y_j^n (1 - y_j^{n+1}) (1 - \text{sg}_{ij}^r) + y_j^{n+1} (1 - y_j^n) \text{sg}_{ij}^r \right\} \end{aligned} \right] \\
 &\quad + \Delta t^n S_{i\infty} (q_r)_{i\infty} \left[\begin{aligned} &\text{sg}_{i\infty} \left\{ y_i^n (1 - y_i^{n+1}) (1 - \text{sg}_{i\infty}^r) + y_i^{n+1} (1 - y_i^n) \text{sg}_{i\infty}^r \right\} \\ &+ (1 - \text{sg}_{i\infty}) \left\{ y_\infty (1 - y_\infty) \right\} \end{aligned} \right] \\
 &= \Delta t^n \rho_i^n \omega_i \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n}.
 \end{aligned} \tag{4.69}$$

Using this scheme, coefficients arising from conditions (4.39) are

$$\begin{aligned}
 (a_q)_{ii} &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} q_{ij}^* (sg_{ij} - 1) + \sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* sg_{ij} (sg_{ij}^r - y_i^n) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (sg_{i\infty} - 1) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} sg_{i\infty} (sg_{i\infty}^r - y_i^n), \\
 (a_q)_{ij} &= \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} q_{ij}^* (1 - sg_{ij}) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (1 - sg_{ij}) (sg_{ij}^r - y_j^n), \\
 (b_q)_i &= y_i^n + \frac{\Delta t^n}{\tau_i^n} \bar{y}_i^n + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (sg_{ij}^r - 1) (sg_{ij} y_i^n + (1 - sg_{ij}) y_j^n) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (sg_{i\infty} - 1) (q_{i\infty} + (1 - y_\infty) (q_r)_{i\infty}) y_\infty \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} sg_{i\infty} (sg_{i\infty}^r - 1) y_i^n, \\
 (\hat{b}_q)_i &= (1 - y_i^n) + \frac{\Delta t^n}{\tau_i^n} (1 - \bar{y}_i^n) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* sg_{ij}^r (1 - [sg_{ij} y_i^n + (1 - sg_{ij}) y_j^n]) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (sg_{i\infty} - 1) (q_{i\infty} - y_\infty (q_r)_{i\infty}) (1 - y_\infty) \\
 &\quad + \Delta t^n \frac{S_{i\infty}}{\omega_i} \frac{(q_r)_{i\infty}}{\rho_i^n} sg_{i\infty} sg_{i\infty}^r (1 - y_i^n).
 \end{aligned} \tag{4.70}$$

Assuming that $\forall j \in v(i), (a_q)_{ij} \leq 0$,

$$\begin{aligned}
 \Lambda_i(\mathbf{A}_q) &= 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n} \left(\sum_{j \in v(i)} \frac{S_{ij}}{\omega_i} (q_r)_{ij}^* (sg_{ij}^r - [sg_{ij} y_i^n + (1 - sg_{ij}) y_j^n]) \right) \\
 &\quad + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} q_{i\infty} (sg_{i\infty} - 1) + \frac{\Delta t^n}{\rho_i^n} \frac{S_{i\infty}}{\omega_i} (q_r)_{i\infty} sg_{i\infty} (sg_{i\infty}^r - y_i^n).
 \end{aligned}$$

The same conditions as (4.37) and (4.38) on the mass fluxes for neighboring inner cells arise again from condition $a_{ij}^q \leq 0$. Similar conditions appear for the boundary face from condition $b_i^q \geq 0$ and $\hat{b}_i^q \geq 0$. When $q_{i\infty} < 0$, the inlet boundary mass fluxes must satisfy

$$\begin{aligned}
 q_{i\infty} + (1 - y_\infty) (q_r)_{i\infty} &\leq 0, \\
 q_{i\infty} - (q_r)_{i\infty} y_\infty &\leq 0.
 \end{aligned} \tag{4.71}$$

When (4.71) is satisfied, no condition on the time step appears. Conditions (4.71) are automatically satisfied when a co-current flow is considered, because in this case, the mixture mass flux $q_{i\infty}$, the gas mass flux $(q_g)_{i\infty} = q_{i\infty} + (1 - y_\infty) (q_r)_{i\infty}$ and the liquid mass flux $(q_l)_{i\infty} = q_{i\infty} - y_\infty (q_r)_{i\infty}$ have the same sign. When (4.71) is not satisfied, a condition on the time step appears from condition $b_i^q \geq 0$ or from condition $\hat{b}_i^q \geq 0$.

4.C Why do the QRd Scheme and the QRq Scheme not converge for unsteady simulations?

Similarly to what is done for the QG Scheme in Section 4.3.4, a convergence study can be achieved for the QRd and QRq Schemes, using the same regular solution and numerical parameters. The

CFL value is kept unchanged such that

$$\text{CFL} = 0.5. \quad (4.72)$$

For $n_x \in [10, 10000]$, the error as a function of the number of cells is shown in Figure 4.11 for each scheme, by comparing the numerical solutions to the analytical solution at time $t_2 = 0.1$ s. While the QG Scheme exhibited the expected first-order convergence, the QRd and QRq Schemes quickly reach a plateau (the green and blue curves overlap). This indicates that these schemes do not converge to the correct solution.

In Figure 4.12, the error as a function of number of cells is shown for simulations performed at different CFL values using the QRd Scheme. As the CFL increases, the error observed on the plateau also increases. The objective of this appendix is to provide an explanation for these observations. The Lax-Wendroff theorem guarantees convergence for a conservative, stable, and consistent finite volume scheme [38]. However, this is not an equivalence. Therefore, it does not directly justify why the non-conservative schemes fail to converge to the correct solution here. An explanation is proposed below.

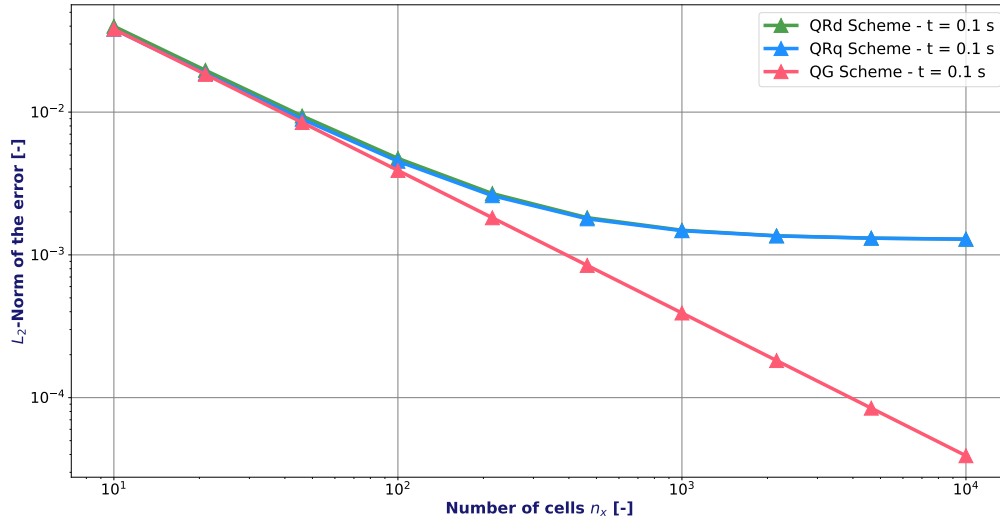


FIGURE 4.11

L_2 -norm of the error as a function of the number of cells n_x for each scheme - $t_2 = 0.1$ s (blue and green curves are overlapped).

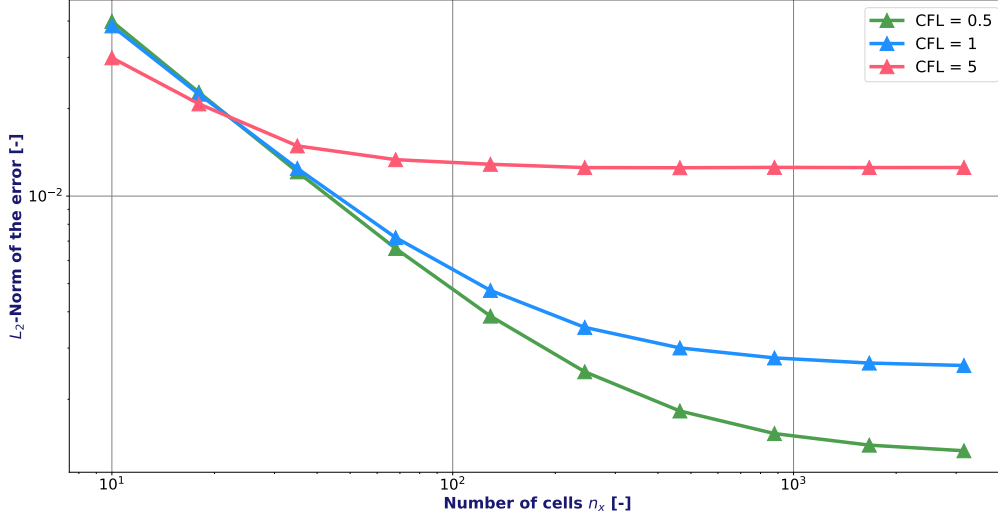


FIGURE 4.12

L_2 -norm of the error as a function of the number of cells n_x for different CFL values - QRd Scheme - $t_2 = 0.1$ s.

The continuous fourth Equation is considered on the spatial domain $[0, L]$ for $t \in [t_1, t_2]$. The mass fraction y satisfies

$$\partial_t(\rho y) + \partial_x F = \rho \frac{\bar{y} - y}{\tau}, \quad (4.73)$$

with $F = qy + q_r y(1 - y)$ the flux. Considering an instant t^n , Equation (4.73) is integrated on the whole domain $[0, L]$ to obtain a global conservation balance for the mass fraction, such that

$$\partial_t W(t^n) + F_L(t^n) - F_0(t^n) = S_L(t^n), \quad (4.74)$$

with

$$\begin{cases} W = \int_0^L (\rho y) dx, \\ S_L = \int_0^L \rho_0 \frac{\bar{y} - y}{\tau} dx, \end{cases} \quad (4.75)$$

and the fluxes F_0 and F_L , which are the inlet and outlet fluxes at the boundaries of the domain.

Now, the discretized equation is considered, using a non-conservative scheme (QRd or QRq). On a cell $i \in [1, n_x]$, it writes

$$\rho_0 \frac{y_i^{n+1} - y_i^n}{\Delta t} \Delta x + F_{i+1/2}^{n+} - F_{i-1/2}^{n-} = \rho_0 \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n} \Delta x, \quad (4.76)$$

with $F_{i+1/2}^{n+}$ the flux at face $i + 1/2$ for cell i . The flux considered for cell $i + 1$ on the same face is written $F_{i+1/2}^{n-}$. Since the scheme is non-conservative in the unsteady case, the fluxes are not equal: $F_{i+1/2}^{n+} \neq F_{i+1/2}^{n-}$. The difference is noted $\Delta F_{i+1/2}^n$ such that

$$\Delta F_{i+1/2}^n = F_{i+1/2}^{n+} - F_{i+1/2}^{n-}. \quad (4.77)$$

Equation (4.76) is summed over the whole mesh. Each term is considered separately. The time derivative term is written as follows

$$\begin{aligned} \rho_0 \sum_{i=1}^{n_x} \frac{y_i^{n+1} - y_i^n}{\Delta t} \Delta x &= \int_0^L \partial_t(\rho y)(t^n) dx + o(\Delta x) + o(\Delta t) \\ &= \partial_t \left(\int_0^L (\rho y) dx \right) (t^n) + o(\Delta x) + o(\Delta t) \\ &= \partial_t W(t^n) + o(\Delta t) + o(\Delta x). \end{aligned} \quad (4.78)$$

Summing the source term gives

$$\begin{aligned} \rho_0 \sum_{i=1}^{n_x} \frac{\bar{y}_i^n - y_i^{n+1}}{\tau_i^n} \Delta x &= \int_0^L \frac{\bar{y} - y}{\tau}(t^n) dx + o(\Delta t) + o(\Delta x) \\ &= S_L(t^n) + o(\Delta x) + o(\Delta t). \end{aligned} \quad (4.79)$$

The summation of the fluxes can be simplified, but a sum remains due to the difference ΔF_i^n , defined in Equation (4.77). It can be expressed as

$$\begin{aligned} \sum_{i=1}^{n_x} (F_{i+1/2}^{n+} - F_{i-1/2}^{n-}) &= F_{n_x+1/2}^{n+} - F_{1/2}^{n-} + \sum_{i=1}^{n_x} \Delta F_{i+1/2}^n \\ &= F_L(t^n) - F_0(t^n) + o(\Delta t) + o(\Delta x) + \sum_{i=1}^{n_x} \Delta F_{i+1/2}^n. \end{aligned} \quad (4.80)$$

Equation (4.76), summed over the domain, can be written

$$\partial_t W(t^n) + F_L(t^n) - F_0(t^n) = S_L(t^n) + R_n + o(\Delta x) + o(\Delta t), \quad (4.81)$$

with

$$R_n = - \sum_{i=1}^{n_x} \Delta F_{i+1/2}^n. \quad (4.82)$$

Equation (4.81) approaches continuous Equation (4.74) at time t^n , with first-order accuracy in both time and space, except for the source term R_n . For a conservative scheme, this source term is strictly zero, and the equations are therefore equivalent. For the non-conservative schemes considered here, the objective is to study this source term. For the QRd scheme, the term is written as follows:

$$\begin{aligned} \Delta F_{i+1/2}^n &= (q_r)_{i+1/2}^n (y_i^{n+1}(1 - y_i^n) - y_i^n(1 - y_i^{n+1})) \\ &= (q_r)_{i+1/2}^n (y_i^{n+1} - y_i^n) \\ &= (q_r)_{i+1/2}^n \frac{\partial y}{\partial t}(t^n, x_i) \Delta t + o(\Delta t). \end{aligned} \quad (4.83)$$

For the QRq Scheme, it is

$$\begin{aligned} \Delta F_{i+1/2}^n &= (q_r)_{i+1/2}^n (y_i^{n+1}(1 - y_{i+1}^n) - y_i^n(1 - y_{i+1}^{n+1})) \\ &= (q_r)_{i+1/2}^n (y_i^{n+1} - y_i^n + y_i^n y_{i+1}^{n+1} - y_i^{n+1} y_{i+1}^n) \\ &= (q_r)_{i+1/2}^n \frac{\partial y}{\partial t}(t^n, x_i) \Delta t + o(\Delta t). \end{aligned} \quad (4.84)$$

At each interface, the error $\Delta F_{i+1/2}^n$ is first-order in time. For unsteady simulation, $(\Delta F_{i+1/2}^n)_{i \in [1, n_x]}$ are non-zero. The summation over the entire domain will yield

$$R_n \sim \sum_{i=1}^{n_x} \Delta t \sim n_x \Delta t \sim \frac{\Delta t}{\Delta x} \sim \text{CFL}. \quad (4.85)$$

Indeed, at each interface, a non-zero error $\Delta F_{i+1/2}^n$ proportional to the time step is accumulated. After summing over the entire domain, the accumulated error becomes proportional to the number of faces, that is, to the inverse of the mesh size Δx . In this case, at constant CFL, an error is introduced and does not decrease with mesh refinement. The discretized Equation (4.76) does not

simulate the continuous Equation (4.73), but a modified equation with an additional source term R_n , added at each time step.

To verify this, the value of the error reached on the plateau is measured as a function of the CFL for $\text{CFL} \in [0.5, 5]$. The case is studied for $n_x = 2500$, which is a sufficient value to reach the plateau when $\text{CFL} = 0.5$. In Figure 4.13, the error obtained on the plateau as a function of the CFL is shown for the QRd and QRq Schemes. As expected, a straight line is obtained: the error is proportional to the CFL.

If the proposed schemes exhibited an error $\Delta F_{i+1/2}^n$ of second order in time, the source term R_n would be of first order in space (at constant CFL). This would eliminate the error made when refining the mesh. However, this is in no way a proof that a non-conservative scheme with second-order error in time necessarily converges to the correct solution for regular solutions. Other reasons may arise that prevent the scheme from converging to the correct solution.

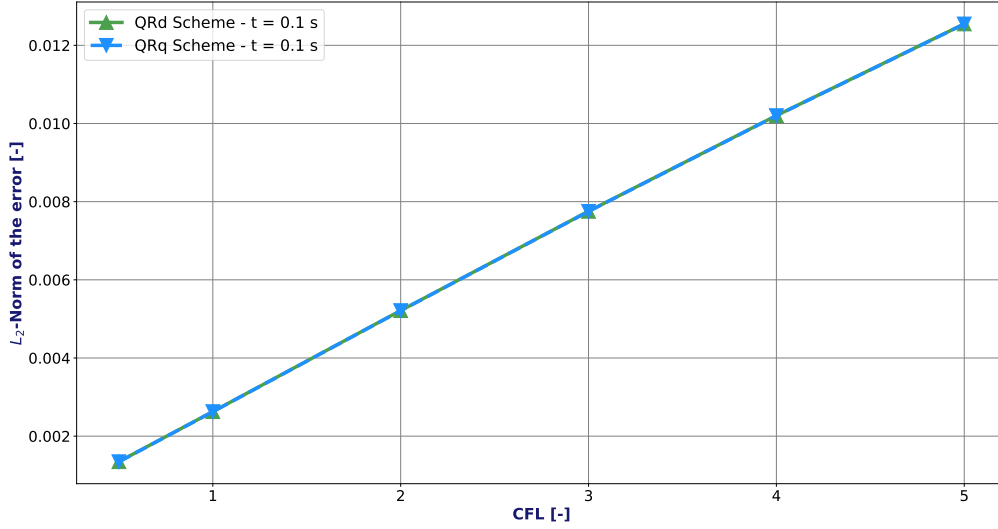


FIGURE 4.13

L_2 -norm of the error as a function of the CFL for the QRd and QRq Schemes - $t_2 = 0.1$ s.

4.D Production control scheme

The production term due to the heat flux is added to the disequilibrium equation with the formulation

$$\Gamma_p = \frac{\chi \phi_s}{L} > 0. \quad (4.86)$$

If the closure law for χ is not chosen wisely (proportional to $1 - y$ for example), this term does not respect the continuous maximum principle for the upper bound 1. In the case where the correlation of χ is arbitrary, a finite volume scheme is proposed in this section to ensure the maximum principle for the discrete solution. In order to propose a general scheme that also works for a negative destruction term that may not respect the mass fraction bound 0, a general source term denoted Γ of any sign is considered in the disequilibrium equation. The disequilibrium equation writes in this case

$$\frac{\partial(\rho y)}{\partial t} + \nabla \cdot (y \mathbf{q}) + \nabla \cdot (y(1 - y) \mathbf{q}_r) = \rho \frac{\bar{y} - y}{\tau} + \Gamma(\mathbf{x}, t). \quad (4.87)$$

A first naive scheme is proposed for this source term. This first discretization scheme does not ensure the maximum principle. It is referred to as GAMi. A second scheme is then proposed to recover the maximum principle at a discrete level, denoted GAMc. This scheme adds a control

that modifies the solution when approaching the boundary (0 or 1) to ensure that the discrete mass fraction remains bounded between 0 and 1. The notations for discretization from Section 4.2.1 are used here.

4.D.1 Finite volume scheme

Direct GAMi Scheme

The first obvious idea is a simple naive scheme, called here GAMi. Its formulation is

$$\int_{\omega_i} \Gamma d\Omega \approx \Gamma_i^* \omega_i, \quad (4.88)$$

where Γ_i^* denotes the value of the source term in cell i for time $t^* \in [t^n, t^{n+1}]$ such that

$$\Gamma_i^* = \Gamma(x_i, t^*). \quad (4.89)$$

This scheme is consistent but may violate the upper bound 1 (resp. the lower bound 0) for the mass fraction when $\Gamma_i^* > 0$ (resp. $\Gamma_i^* < 0$).

Modified GAMc Scheme

Using $sg_i^g = sg(\Gamma_i^*)$, the GAMc scheme is

$$\int_{\omega_i} \Gamma d\Omega \approx \Gamma_i^* \omega_i \cdot \left(sg_i^g \frac{1 - y_i^{n+1}}{\max(1 - y_i^n, \varepsilon_c)} + (1 - sg_i^g) \frac{y_i^{n+1}}{\max(y_i^n, \varepsilon_c)} \right) \text{ with } 0 < \varepsilon_c \ll 1. \quad (4.90)$$

With this scheme, the steady-state solution is modified when the mass fraction is above $(1 - \varepsilon_c)$ (resp. below ε_c) for positive (resp. negative) source term Γ so that the solution remains in $[0, 1]$ at all times. When away from the bounds, GAMc Scheme is consistent for the steady-state solution and it respects the maximum principle for bound 0 and 1.

Property (Maximum principle with the production term): Assuming $\forall i, y_i^n \in [0, 1]$, GAMc Scheme ensures that $y_i^{n+1} \in [0, 1]$.

Proof Using GAMc Scheme, the following contributions are added to the matrix system previously studied for the disequilibrium equation for each scheme (see Equations (4.30), (4.36) or (4.65)):

$$\begin{aligned} (a_g)_{ii} &= sg_i^g \frac{\Gamma_i^*}{\max(1 - y_i^n, \varepsilon_c)} - (1 - sg_i^g) \frac{\Gamma_i^*}{\max(y_i^n, \varepsilon_c)} \geq 0, \\ (a_g)_{ij} &= 0, \\ (b_g)_i &= sg_i^g \frac{\Gamma_i^*}{\max(1 - y_i^n, \varepsilon_c)} \geq 0, \\ (\hat{b}_g)_i &= 0. \end{aligned} \quad (4.91)$$

The contributions respect conditions from Equations (4.27) so GAMc Scheme ensures the maximum principle for the lower bound 0 and for the upper bound 1. $\square$

GAMc Scheme is proposed in a framework for any source term. Turning to applications in THYCCoeur model, production terms considered are such that $\Gamma(\mathbf{x}, t) \geq 0$. As a consequence, the solution is modified only in the region $[1 - \varepsilon_c, 1]$ for the mass fraction. Using $\varepsilon_c = 10^{-9}$, this interval is never reached for steady states in PWR applications.

For the following numerical simulations used to verify GAMc Scheme, the production term is taken implicitly $\Gamma_i^* = \Gamma_i^{n+1}$ as it is a given data. The scheme is

$$\int_{\omega_i} \Gamma d\Omega \approx \Gamma_i^{n+1} \omega_i \frac{1 - y_i^{n+1}}{\max(1 - y_i^n, \varepsilon_c)} \text{ with } \varepsilon_c = 10^{-9}. \quad (4.92)$$

For applications in the industrial code THYC-coeur, the implicit formulation is harder to obtain so the explicit form is considered. In that case, the scheme is

$$\int_{\omega_i} \Gamma d\Omega \approx \Gamma_i^n \omega_i \frac{1 - y_i^{n+1}}{\max(1 - y_i^n, \varepsilon_c)} \text{ with } \varepsilon_c = 10^{-9}. \quad (4.93)$$

The control parameter ε_c is important. It's this parameter that modifies the solution close to the bounds. Without it, the solution would remain stuck at bound 0 or 1 (see Figure 4.14).

4.D.2 Numerical verification of the scheme

Case without flow

The following production term is considered

$$\Gamma(t) = \Gamma_0 \cos(\omega t) e^{-t/\tau_r} + \Gamma_\infty > 0 \text{ with } \begin{cases} \Gamma_0 &= 1.77 \text{ kg.m}^{-3}.\text{s}^{-1}, \\ \Gamma_\infty &= 0.3 \text{ kg.m}^{-3}.\text{s}^{-1}, \\ \omega &= 0.31 \text{ rad/s}, \\ \tau_r &= 5.0 \text{ s}. \end{cases} \quad (4.94)$$

Considering no velocity ($\mathbf{q} = \mathbf{0}$ and $\mathbf{q}_r = \mathbf{0}$) in the disequilibrium equation and a constant uniform equilibrium mass fraction $\bar{y}_0$ and density ρ_0 , the equation becomes a 0D equation with mass fraction depending only on time. It writes

$$\rho_0 \partial_t y = \rho_0 \frac{\bar{y}_0 - y}{\tau_0} + \Gamma(t) \text{ with } \begin{cases} \rho_0 = 1 \text{ kg.m}^{-3}, \\ \tau_0 = 2.0 \text{ s}, \\ \bar{y}^0 = 0.01, \\ y(t=0) = 0.2. \end{cases} \quad (4.95)$$

The mass fractions for each scheme (GAMi and GAMc) as a function of time are represented in Figure 4.14 for a domain $\Omega = [0, 1]$ m. This figure also shows the case where the GAMc scheme is used without a control parameter ε_c . As expected, GAMc Scheme limits the upper bound for the mass fraction to 1. Thanks to the control parameter ε_c , the solution for GAMc Scheme remains in $[1 - \varepsilon_c, 1[$ without never reaching 1, which is not the case without a control ε_c . After a while, the solution for GAMc Scheme exits the control interval $[1 - \varepsilon_c, 1[$ and join the solution of GAMi scheme, reaching the same steady-state solution.

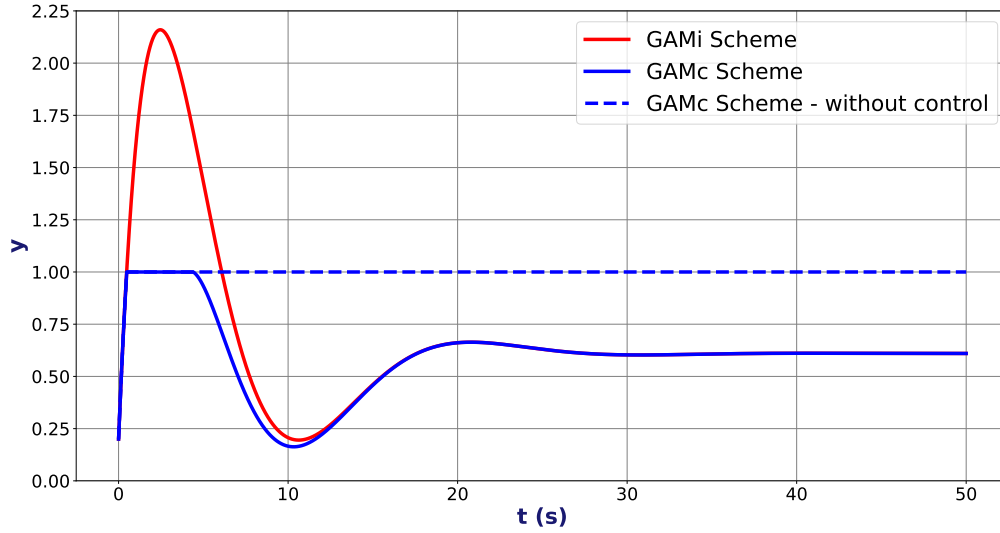


FIGURE 4.14

Mass fraction y as a function of time for GAMc Scheme (with or without control parameter $\varepsilon_c = 10^{-9}$) and GAMi Scheme.

Case with flow

From Equation (4.95), uniform and constant 1D mass fluxes are added $q = q^0 = 1.0 \text{ kg.m}^{-2}.\text{s}^{-1}$; $q_r = q_r^0 = 0.3 \text{ kg.m}^{-2}.\text{s}^{-1}$. Mass fraction at different points of the domain $\Omega = [0, 1] \text{ m}$ as a function of time is represented in Figure 4.15 (for each scheme). The solid line represents the solution with GAMc Scheme, while the dotted one represents the solution with GAMi Scheme. GAMc scheme prevents the mass fraction from exceeding the upper bound 1 while ensuring the right steady-state solution at each point of the domain.

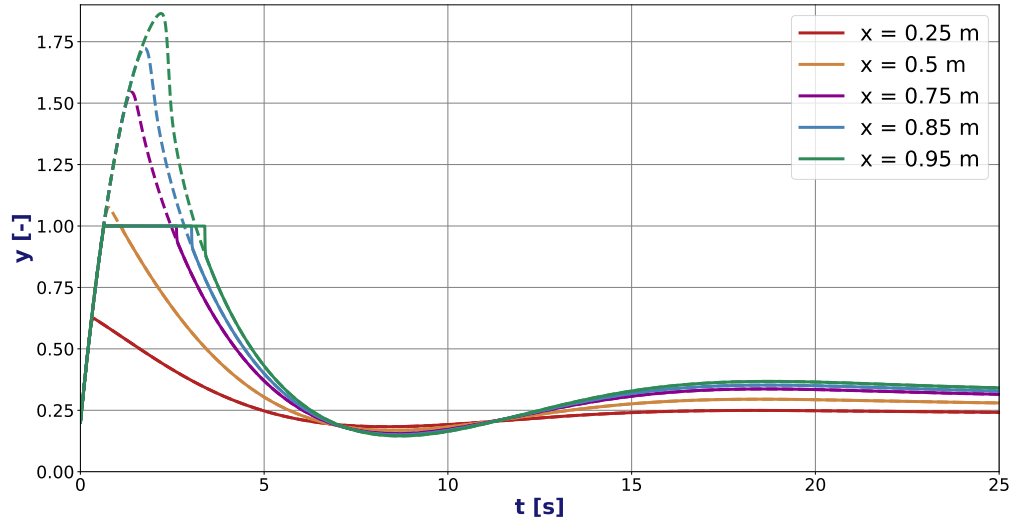


FIGURE 4.15

Mass fraction y for an unsteady 1D case: GAMc scheme (in solid line) and GAMi scheme (in dotted line).

4.D.3 Conclusion

GAMc Scheme is particularly interesting for THYC-coeur application when searching for steady-state solution. For the steady-state solutions considered, mass fraction remain in the bounds $[0, 1]$. However, during the convergence, the time advancement algorithm may cause the mass fraction y to exit the bound one because of the source term Γ_p , particularly when the time step is increased.

In this case, several correlation laws defined for $y \in [0, 1]$ may diverge, and the simulation fails to converge. The GAMc scheme prevents this behavior and allows large time steps while ensuring the right final steady-state solution. However, when considering unsteady solutions, that is better to use the GAMi Scheme.

When GAMc Scheme is used, care must be taken to ensure that the final steady-state solution obtained is sufficiently far from boundary 1, i.e.: $y < 1 - \varepsilon_c$. Otherwise, the solution obtained may be altered by the numerical scheme. Choosing $\varepsilon_c = 10^{-9}$ is sufficient for THYC-coeur applications.

Chapter 5

Benchmark of the numerical schemes in THYC-coeur

This chapter is devoted to the verification of the three finite-volume schemes proposed in **Chapter 4**, which have been implemented in the industrial code THYC-coeur. The objective is to verify the implementation and evaluate the performance of the schemes in terms of accuracy and CPU time. Here, the goal is to compare the new schemes with the reference scheme from THYC-coeur. The comparisons are carried out using industrial numerical parameters (mesh size, time step, etc.) and aim to evaluate the performance of the schemes in predicting the solution of the continuous equation model when the industrial numerical parameters are fixed.

To this end, an industrial database to compute the Departure from Nucleate Boiling Ratio (DNBR, see Introduction - Section 2.2) is used. It is called the Bias Curves Database (BCD). This database contains state points, corresponding to different operating conditions such as pressure, temperature, power and other parameters of a reactor core. The BCD is specific to a nuclear reactor core geometry.

In practice, this database is used to configure one of the protection systems called SPIN (for french *Système de Protection Intégré Numérique*). In this work, the Bias Curves Database is not used for its primary purpose. Indeed, this extensive database can also be used to investigate the impact of numerical and/or physical model modifications on the performance when comparing two versions of THYC-coeur. It is also a way to determine the robustness of the code, the limitations on the time step and to evaluate if some state points diverge.

In this chapter, the complete 4-equation physical model of THYC-coeur is used. For each scheme, performance is evaluated in terms of accuracy (compared to the reference scheme), CPU time, and number of iterations before convergence towards the steady state. The impact on performance is assessed for two different correlations for the relative velocity: the Bestion and the Chexal-Lellouche correlations, which are presented in Section 1.4.2. It was important to evaluate the impact of the new schemes on these two correlations for several reasons. First, during the course of the PhD work, the relative velocity correlation used by default in THYC-coeur changed from the Chexal-Lellouche correlation to the Bestion correlation. The two correlations are therefore currently used in industrial applications. Moreover, these two correlations allow testing solutions for two very different orders of magnitude for relative velocities. The relative velocity for Chexal-Lellouche is negative and almost reaches the mixture velocity in absolute value, whereas that of Bestion is positive and very small compared to the mixture velocity.

The BCD and its associated protection system are presented in Section 5.1. The methodology and the indicators used to assess performance and accuracy on the database are explained in Section 5.2. Then, the BCD is computed to evaluate the impact on performance of the three finite-volume

schemes proposed in **Chapter 4** and to compare them to the reference scheme. Section 5.3 shows the results for the Bestion correlation, and Section 5.4 those for the Chexal-Lellouche correlation.

5.1 Presentation of the SPIN protection system and of the Bias Curves Database

In this section, the operation of the SPIN system is presented. Then, the BCD content is detailed.

5.1.1 SPIN protection system

The system called SPIN (for french *Système de Protection Intégré Numérique*) is an automatic protection system integrated into reactors. It evaluates the maximum vapor mass fraction in the core as well as the margin to the boiling crisis, for a functioning nuclear power plant. It is based on signals measured in the primary circuit. This system is used to quickly evaluate the margin to the safety criteria, with a computational cycle of around 200 ms. It computes, every 200 ms, approximated values for the maximum vapor mass fraction and the minimum of the Departure from Nucleate Boiling Ratio (DNBR - see Section 2.2 of Introduction) in the reactor core. If these quantities reach a threshold value, the system triggers the automatic shutdown of the reactor (noted AAR for *Arrêt Automatique du Réacteur* in French), i.e., the shutdown of the nuclear reaction by the drop of control rods, which are neutron-absorbing elements.

In this work, the Bias Curves Database used for N4-type reactors is employed. The N4 series is the third generation of reactors in the current French nuclear fleet, following the 900 MWe and 1300 MWe series. These reactors are second-generation pressurized water reactors developed by Framatome and EDF. A N4 reactor produces 1450 MWe at nominal power. Four N4 reactors have been built in France, two at the Chooz plant, commissioned in 2000, and two at the Civaux plant, commissioned in 2002. For the N4 series, the Bias Curves Database consists of 36 148 state points.

Figure 5.2 shows a diagram explaining the set up and operation of the SPIN system. First, the SPIN system evaluates the state of the core based on real-time measured data. For this, several sensors measure certain physical quantities in the primary circuit around the nuclear reactor core, where measurements are easier to perform. The locations, where the quantities are measured for the operation of the SPIN system, are shown in Figure 5.1 which represents the primary circuit of a nuclear power plant (*with french notations*). The quantities measured are:

- The **outlet pressure**, which is measured directly in the pressurizer.
- The **inlet mass flux**, which is given by the speed of the Reactor Coolant Pumps (RCP) in the primary circuit. It gives the inlet mass flux for one of the four loops of the system, (so four computations of the SPIN system are done in parallel, one for each loop).
- The **inlet temperature** of the core which is measured by a probe. A second temperature probe at the outlet along with the mass flux of one of the four loops allows obtaining **one fourth of the total core power** through an enthalpy balance (at constant mass flux).

Additionally, neutron detectors, called fission chambers, measure the neutron flux according to the height in the core. This allows to obtain the **normalized axial power shape** in the core. Another necessary quantity is the $F\Delta H$, or **FDH** (from the french *Facteur d'élévation d'enthalpie*). It corresponds to the ratio between the power of the hottest rod in the assembly and the average power of a rod in the core (as a function of the height). Sensors continuously measure the position of the control rods in the reactor core. The position of the control rod enables to evaluate the FDH per core slice according to the height within the core using tabulations. These tabulations, giving the FDH as a function of the control rod positions, are obtained with neutronics computations. The tabulations are updated monthly based on the fuel burn-up, which corresponds to the current

consumption state of the fuel. The total power and the FDH are used to transform the normalized axial power shape into the **hot channel axial power distribution**.

All these quantities enable the evaluation of the one-dimensional **enthalpy** field in the hot assembly as a function of the height in the core (through an enthalpy rise calculation). For this computation, the flow is considered in a single channel. Based on the inlet mass flux and enthalpy, and considering a constant pressure, the **mixture velocity** can also be obtained along the channel.

Thus, the SPIN system provides the approximate 1D vectors for the enthalpy and mass velocity fields in the hottest channel in the core. To improve this simplified calculation, empirical corrections (based on FDH, and other measured parameters) are applied to the local enthalpy and mass velocity evaluated by the single-channel calculation. This allows taking into account the impact of flow redistribution between the assemblies in the core. These functions, called **bias curves**, are applied to the enthalpy and mass velocity. Subsequently, a critical heat flux correlation is used to obtain the approximate value of the DNBR_{min}, i.e., the minimum DNBR value in the reactor core. The bias curves are complex functions that contain many parameters to be adjusted. These parameters are set using the results of the minimum DNBR in the core from a 3D thermohydraulic code such as THYC-coeur, hence the creation of the Bias Curves Database. This adjustment must be redone for any change (change of fuel, core operating parameters, or critical heat flux correlation).

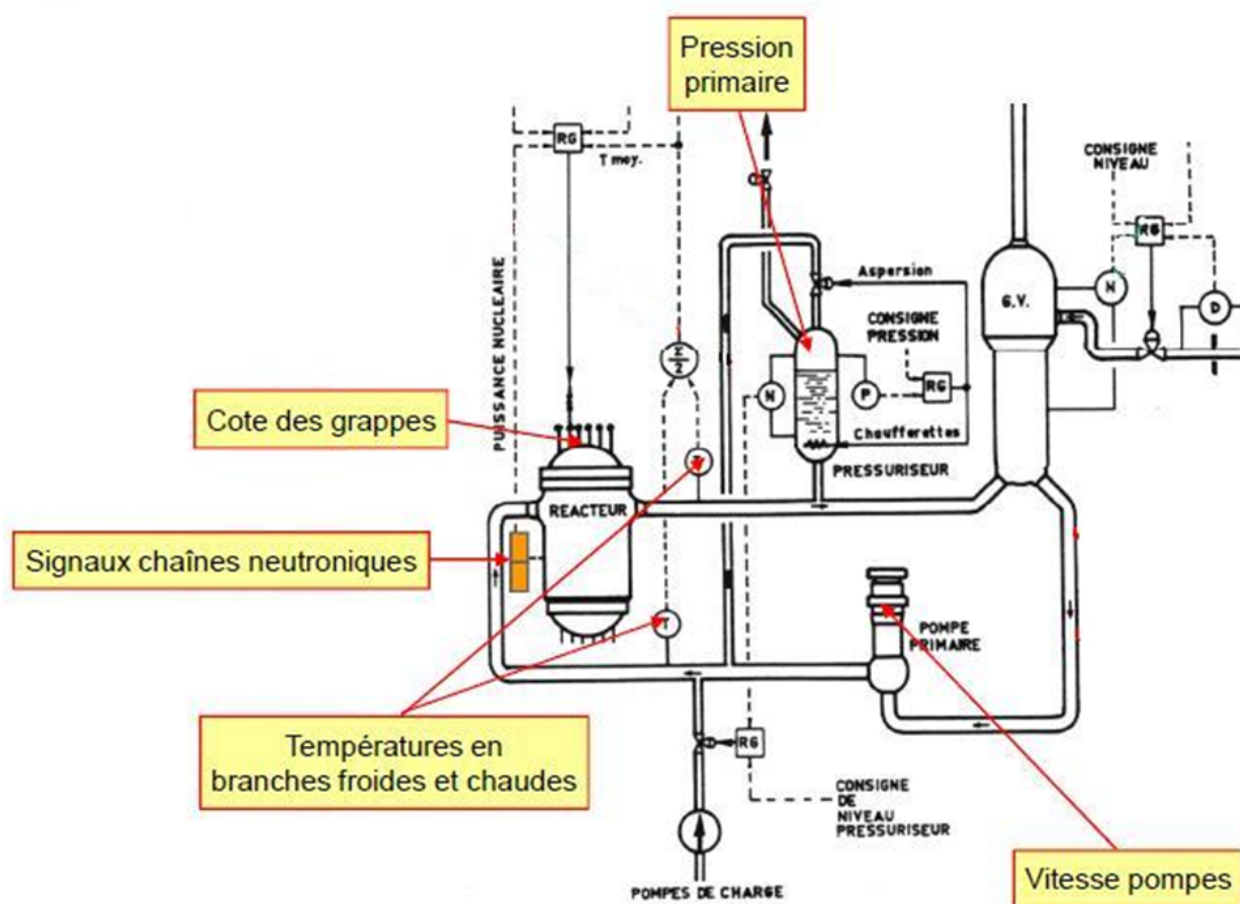


FIGURE 5.1

Scheme of the primary circuit with the measurement points enabling the SPIN system to operate. For simplicity, only one of the four primary loops is pictured.

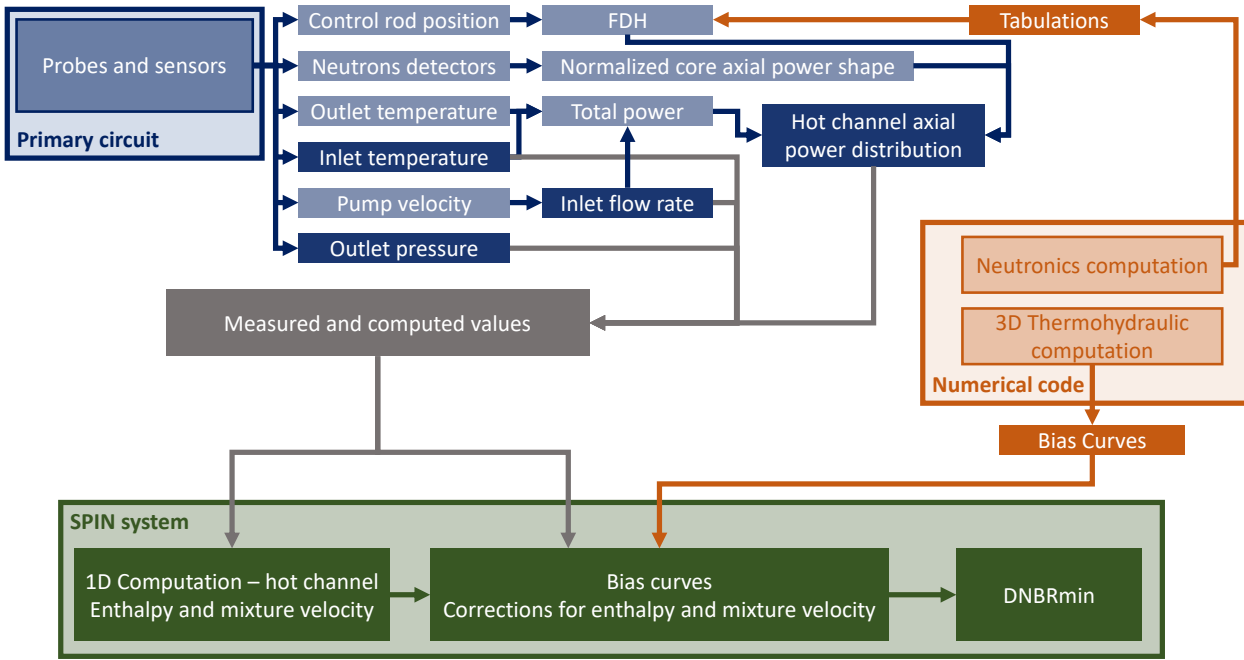


FIGURE 5.2
Operation of the SPIN system.

5.1.2 Bias Curves Database

For a given geometry of a reactor core, the BCD contains the 3D fields characterizing the flow in the core based on six input parameters:

- the inlet mass flux,
- the inlet temperature,
- the outlet pressure,
- the total power,
- the axial power shape,
- the FDH.

The variation of these parameters within predefined ranges leads to the creation of a database of 36 148 points (for N4 reactor with a fuel management called ALCADÉ).

The Bias Curves Database consists of a large number of cases, covering most normal and accidental operating conditions for a reactor core. They are therefore a valuable tool for evaluating the performance in terms of accuracy, CPU time and robustness of numerical changes in THYC-coeur. This makes it possible to verify that modifications (to physical models, numerical schemes, etc.) have no impact, or only a minimal one, on the results obtained with the current THYC-coeur code used in industrial applications. In the following, this database is thus used to assess the impact of the schemes proposed in **Chapter 4** on the DNBR results.

Using the THYC-coeur reference scheme, the state points of the BCD are computed with a fixed time step of $\Delta t = 0.01$ s for industrial applications. By exploiting the symmetries of the reactor core power distribution, the spatial mesh represents only one-eighth of the entire reactor. A cartesian mesh is considered here with an industrial configuration, called semi-refined mesh. The core is meshed with one cell per fuel assembly, except for the central assemblies, which are meshed at the subchannel level (324 cells per assembly). Figure 5.3 shows the semi-refined mesh for a full reactor core, highlighting the eighth-of-core mesh. For this work, we choose to use an axial mesh (along the main flow axis) consisting of cells with a maximum size of 13 cm, over a total height of 4.7945 m. The total mesh contains approximately 15 000 cells.

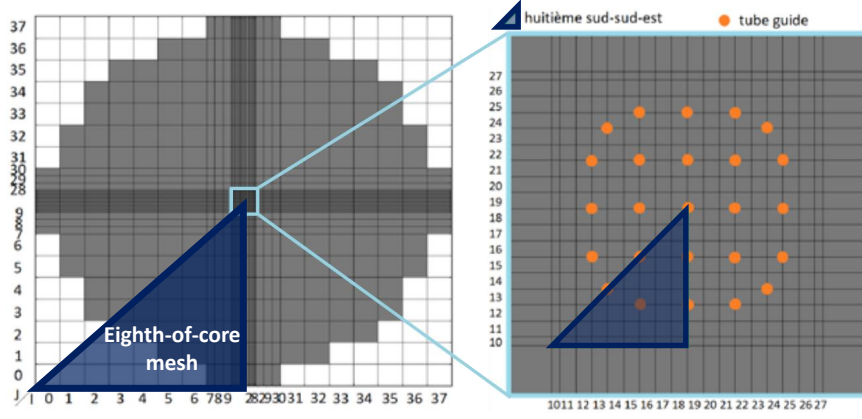


FIGURE 5.3
Semi-refined industrial mesh for a nuclear reactor core.

To compute the DNBR_{min}, the equilibrium quality X is used in the THYC-coeur code. Using notations from **Chapter 1**, it corresponds to

$$X = \frac{yq_g h_g + (1-y)q_l h_l - q_m \bar{h}_l}{q_m \bar{L}}. \quad (5.1)$$

The equilibrium quality serves as an indicator of how strongly the flow is two-phase. It is directly related to the mass fraction y but accounts for the dynamic effects due to the mass flux of each phase. The equilibrium quality is used in the following schemes as the color map for BCD results.

5.2 Definition of the performance indicators

The three finite-volume schemes (see **Chapter 4**) for the disequilibrium equation on the mass fraction have been implemented in THYC-coeur. The schemes for the other equations (total mass, momentum and energy) are not modified. These three new implementations are called QG Scheme, QRd Scheme and QRq Scheme. These schemes are compared to the current scheme of THYC-coeur, which is called here REF Scheme.

In THYC-coeur, the stopping criterion corresponds to the convergence condition used for deciding whether a steady-state has been reached. This criterion is compared to the L_2 -norm of the temporal increments for each variable (entropy, pressure, mass flux and mass fraction), normalized by the L_2 -norm of the variable at the current time. For each relative velocity correlation (Bestion and Chexal-Lellouche), the four schemes are tested for two different stopping criteria (10^{-11} and 10^{-4}).

A criterion of 10^{-11} corresponds to steady-state solutions obtained up to machine precision (for a given mesh size and time step). At this criterion, the new schemes can be compared to the REF Scheme to verify the implementation and compare the accuracy. BCD results with this criterion are noted cvREF, cvQG, cvQRd and cvQRq. The cvREF are the reference results and are used for every comparison in the following.

The stopping criterion of 10^{-4} corresponds to the stopping criterion used in **industrial** applications. It is a compromise between accuracy and CPU time, providing results with acceptable variation of DNBR and reasonable CPU time. BCD results, obtained with this criterion, are noted indREF, indQG, indQRd and indQRq. They are compared to cvREF.

Bias Curves Database is run with a constant time step $\Delta t = 0.01s$, which is the industrial value currently used in THYC-coeur.

The new schemes proposed for the fourth equation make it possible to eliminate constraints on the time step. The time step could be then increased to gain CPU time. However, increasing the time step tends to degrade the steady-state solutions when keeping industrial stopping criteria. The accuracy of the resulting solutions then becomes insufficient, as dispersion in DNBR results compared to the reference exceed the allowed thresholds. There can be various reasons for this degradation in the solution. Increasing the time step worsens the iterative resolution of the matrix systems at each time step, for the coupled momentum-pressure resolution. Moreover, the stopping criterion of THYC-coeur can become sensitive to the time step when it is increased to high values. This is due to nonlinear effects in the computation of the residuals, which uses the increments of the computed variables. This can cause premature termination of steady-state calculations. Time stepping could also be managed with a variable time step, depending on a constraint such as a CFL condition. This possibility is not explored here, as the majority of the industrial applications use a fixed time step. The work is therefore limited here to the constant industrial time step of $\Delta t = 0.01$ s.

In order to evaluate the performance of each BCD computation, several numerical indicators are defined. Mean quantities on the full database (of $N_{BCD} = 36148$ cases) are defined with the notation

$$\langle f \rangle_{BCD} = \frac{1}{N_{BCD}} \sum_{i \in [1, N_{BCD}]} f_i, \quad (5.2)$$

where f_i is the value of quantity f for Case i , with f being the CPU time t_{cpu} , the number of iterations N_{iter} , or the variation of DNBR. To evaluate the performance in accuracy, the absolute and relative variations of DNBR are computed on each case with the following notations

$$\begin{aligned} \Delta \text{DNBR} &= \text{DNBR}^{new} - \text{DNBR}^{ref}, \\ \Delta_r \text{DNBR} &= \frac{\text{DNBR}^{new} - \text{DNBR}^{ref}}{\text{DNBR}^{ref}}, \end{aligned} \quad (5.3)$$

where DNBR^{ref} is the DNBR obtained with cvREF and DNBR^{new} the BCD computation considered (cvQG, cvQRd, cvQRq, indREF, indQG, indQRd, indQRq).

In industrial applications, a variation of DNBR less than $\pm 0.1\%$ or ± 0.001 is considered negligible. If the variation remains less than $\pm 0.5\%$ or ± 0.005 , it is considered acceptable, particularly when the DNBR is high (superior to 2). The number of cases where the relative variation exceeds $\pm 0.1\%$ or $\pm 0.5\%$ are computed when comparing two BCD results. They are noted respectively $N(|\Delta_r| > 0.1\%)$ and $N(|\Delta_r| > 0.5\%)$. The geometric mean of the variation of DNBR is defined by:

$$m(\Delta_r) = -\langle \log |\Delta_r \text{DNBR}| \rangle_{BCD} = -\frac{1}{N_{BCD}} \sum_{i \in [1, N_{BCD}]} \log |\Delta_r \text{DNBR}_i|. \quad (5.4)$$

With this definition, the average variation of DNBR is $10^{-m(\Delta_r)\%}$. As a consequence, the higher the value of $m(\Delta_r)$, the smaller the variation, meaning that the scheme solutions are closer to the cvREF results. To evaluate the performance, the mean CPU time, noted $\langle t_{CPU} \rangle_{BCD}$, and the mean number of iterations before convergence, noted $\langle N_{iter} \rangle_{BCD}$, are computed using Equation (5.2).

For each BCD computation, the evaluated performance indicators in terms of accuracy and CPU time are:

- The geometric mean variation of DNBR $m(\Delta_r)$,
- The number of cases where the relative variation of the DNBR exceeds $\pm 0.1\%$ (noted $N(|\Delta_r| > 0.1\%)$) or $\pm 0.5\%$ (noted $N(|\Delta_r| > 0.5\%)$),
- The mean CPU time $\langle t_{CPU} \rangle_{BCD}$,
- The mean number of iterations before convergence $\langle N_{iter} \rangle_{BCD}$.

5.3 Numerical results of the Bias Curves Database with the Bestion correlation

5.3.1 Verification of the implementation

Using the Bestion correlation from Equation (1.103) for the relative velocity, Bias Curves Database is run with a stopping criterion of 10^{-11} for the four schemes (REF, QG, QRd and QRq). Figure 5.4 (resp. 5.5, 5.6) represents the variation of DNBR for cvQG (resp. for cvQRd, cvQRq) compared with cvREF as a function of the reference DNBR (from cvREF). Different lines represent the industrial limits: the black solid line shows the threshold $\Delta\text{DNBR} = \pm 0.01$, the dotted blue $\Delta\text{DNBR} = \pm 0.005$, the solid red $\Delta_r\text{DNBR} = \pm 1\%$ and the dashed green for $\Delta_r\text{DNBR} = \pm 0.5\%$. The indicators $m(\Delta_r)$ are, evaluated

$$\begin{aligned} \text{for QG Scheme: } m(\Delta_r) &= 11.0 \pm 0.05, \\ \text{for QRd Scheme: } m(\Delta_r) &= 10.8 \pm 1.2, \\ \text{for QRq Scheme: } m(\Delta_r) &= 11.0 \pm 0.06. \end{aligned} \quad (5.5)$$

We conclude that the implementation of the three new schemes is valid and the accuracy obtained is very similar to cvREF as the DNBR variations are far from the criterion of $\pm 0.1\%$.

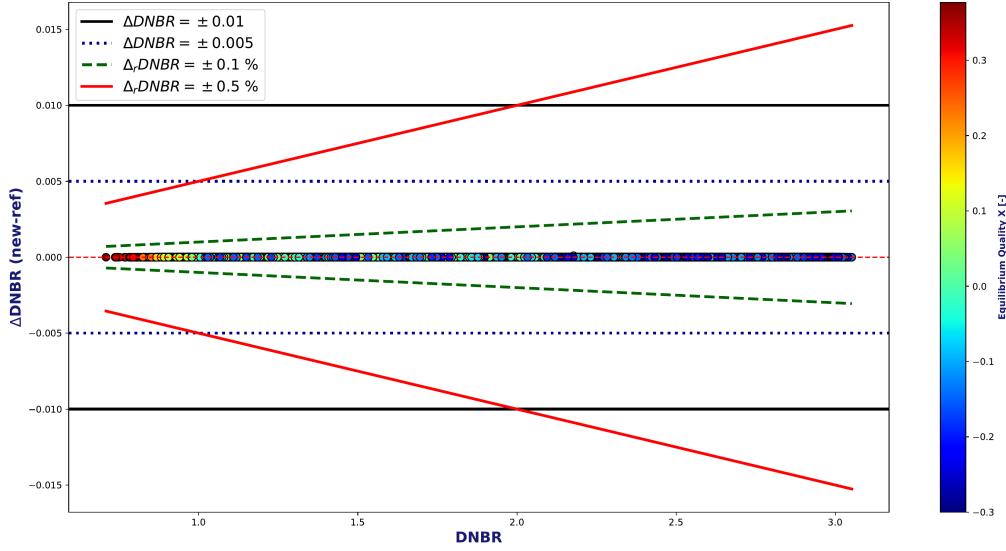


FIGURE 5.4
Relative variation of DNBR for cvQG as a function of the DNBR with the Bestion correlation.

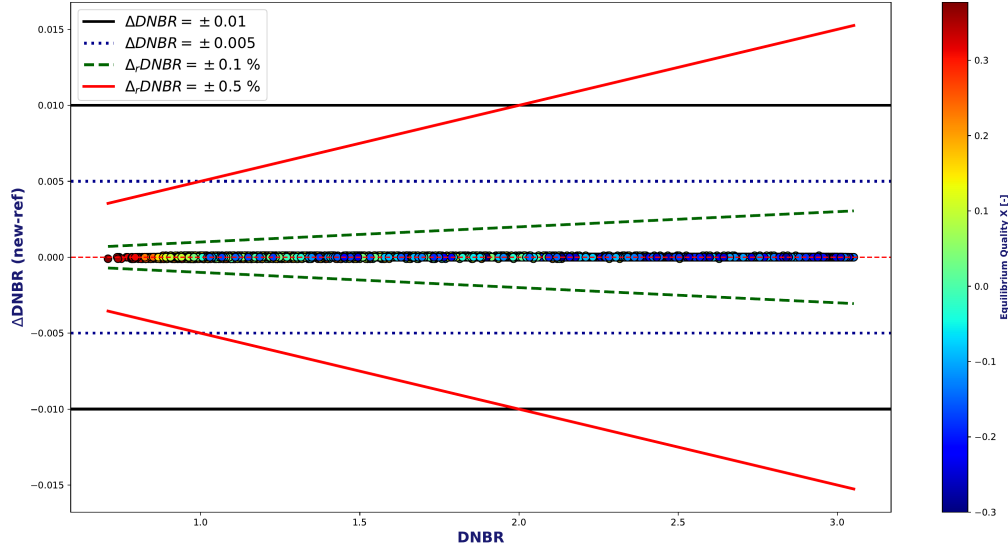


FIGURE 5.5

Relative variation of DNBR for cvQRd as a function of the DNBR with the Bestion correlation.

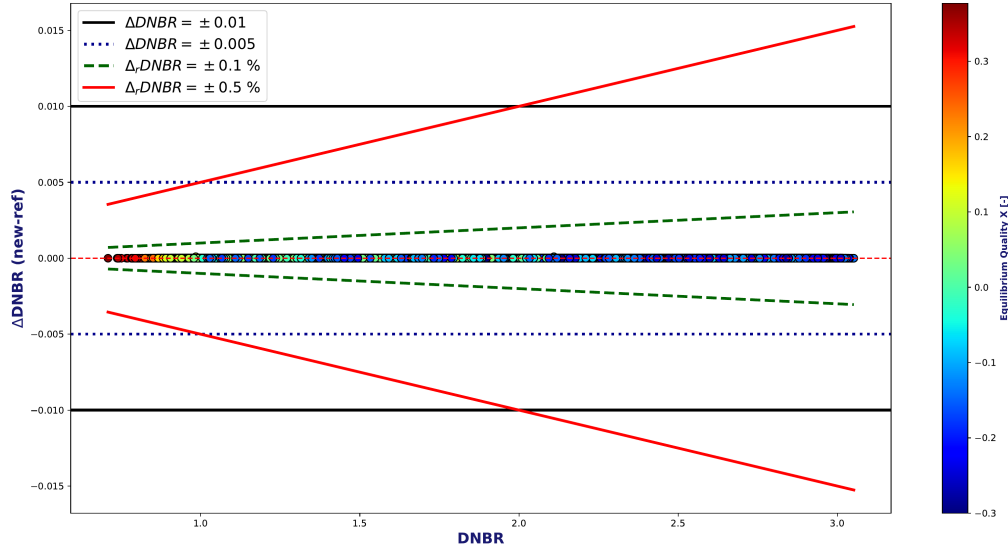


FIGURE 5.6

Relative variation of DNBR for cvQRq as a function of the DNBR with the Bestion correlation.

5.3.2 Impact on the performance at an industrial stopping criterion

Now, the BCD are run with an industrial stopping criterion of 10^{-4} for each scheme. The DNBR variations (compared with cvREF) are represented as a function of the DNBR (of cvREF) in Figure 5.7 for indREF, in Figure 5.8 for indQG, in Figure 5.9 for indQRd and in Figure 5.10 for indQRq. The performance indicators for each scheme are summarized in Table 5.1.

The schemes are similar in performance in terms of accuracy or computing time. For each scheme only a very few state points (4-6) are above the limit of $\pm 0.1\%$ for the variation of DNBR and with a high DNBR of reference (> 2). These slightly larger variations are caused by a threshold effect on DNBR computation when vaporization occurs. The four schemes are considered equivalent for the Bestion correlation and for a stopping criterion of 10^{-4} . This observation does not imply that the resolution of the fourth equation has a small impact on the solution. The differences between the proposed schemes mainly lie in the treatment of the relative velocity term. With the

Bestion correlation, the relative velocity is very small compared to the mixture velocity. Therefore, differences in the scheme have little impact on the solution. This is why the observed performances are similar in terms of both accuracy and CPU time. A more noticeable effect is seen with the Chexal-Lellouche correlation, as shown in the following section.

Scheme	indREF	indQG	indQRd	indQRq
$m(\Delta_r)$	6.43 ± 4.44	6.40 ± 4.44	6.39 ± 4.44	6.40 ± 4.44
$N(\Delta_r > 0.1\%)$	5	6	4	5
$N(\Delta_r > 0.5\%)$	0	0	0	0
$\langle t_{cpu} \rangle_{BCD}$ (s)	7.91 ± 1.02	7.80 ± 1.01	7.86 ± 1.01	7.85 ± 1.01
$\langle N_{iter} \rangle_{BCD}$	67.15 ± 9.40	67.07 ± 9.32	67.04 ± 9.28	67.08 ± 9.30

TABLE 5.1

Performance indicators for each scheme with the Bestion correlation and a stopping criterion of 10^{-4} .

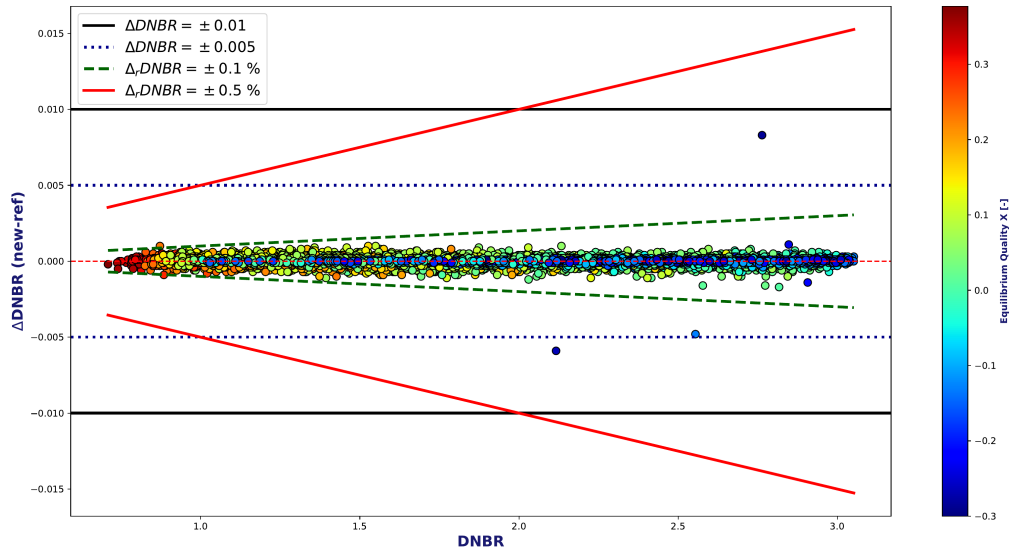


FIGURE 5.7

Relative variation of DNBR for indREF as a function of the DNBR with the Bestion correlation.

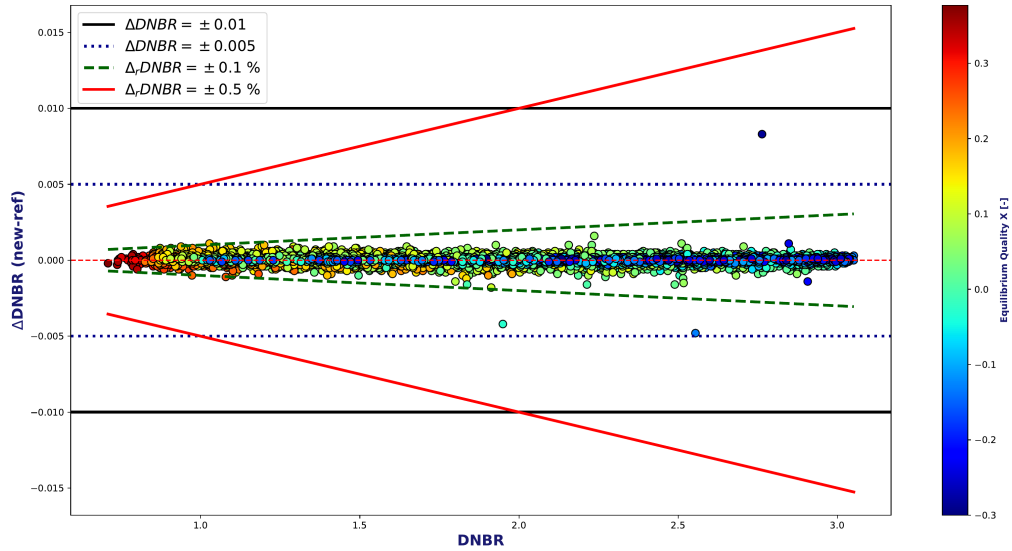


FIGURE 5.8

Relative variation of DNBR for indQG as a function of the DNBR with the Bestion correlation.

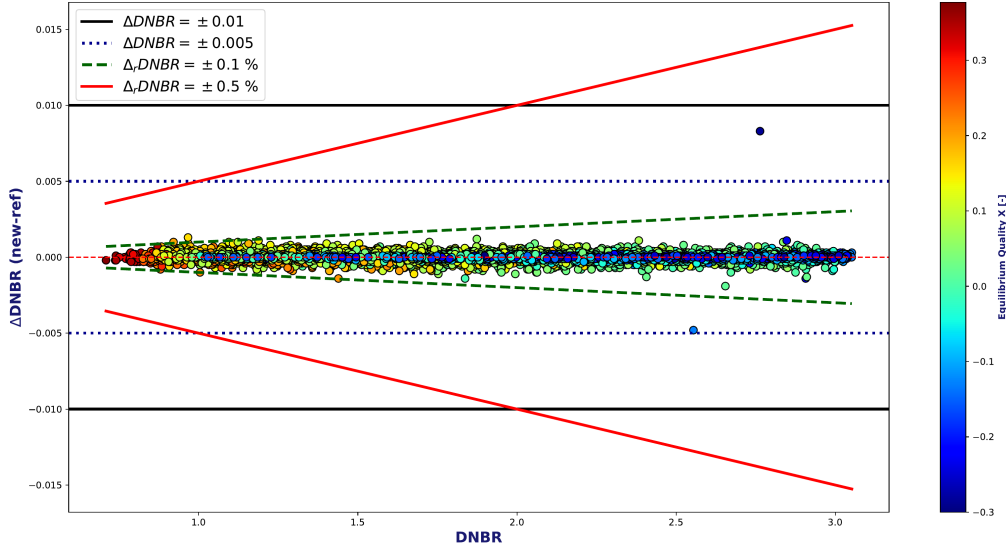


FIGURE 5.9

Relative variation of DNBR for indQRd as a function of the DNBR with the Bestion correlation.

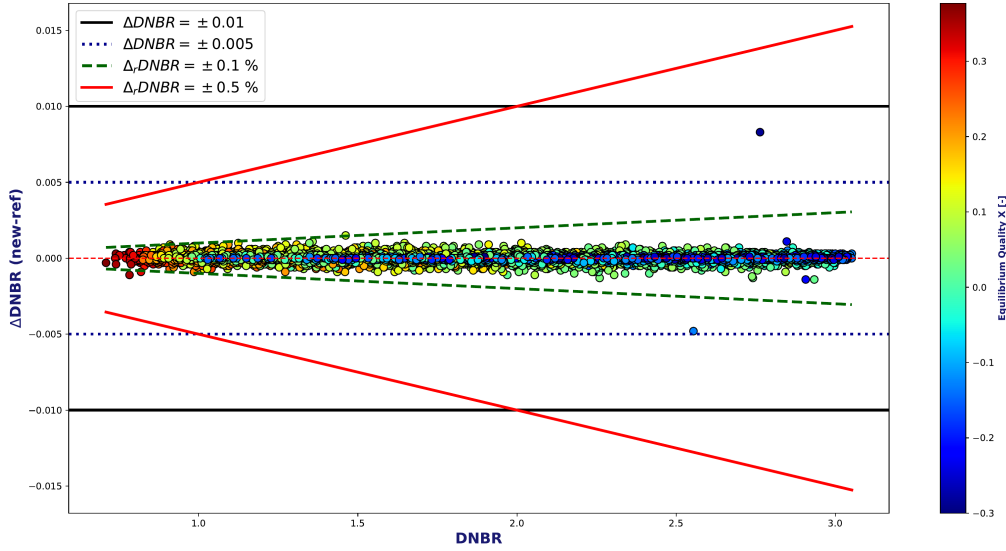


FIGURE 5.10

Relative variation of DNBR for indQRq as a function of the DNBR with the Bestion correlation.

5.4 Numerical results of the Bias Curves Database with the Chexal-Lellouche correlation

5.4.1 Verification of the implementation

The Bias Curves Database is computed for the Chexal-Lellouche correlation. First, the BCD results for each scheme is compared with cvREF for a stopping criterion of 10^{-11} . Figure 5.11 (resp. 5.12 and 5.13) represents the variation of DNBR for cvQG (resp. cvQRd and cvQRq) as a function of the DNBR (for cvREF).

The QRd and REF Scheme have similar results for the stopping criterion of 10^{-11} . For the QG and QRq Scheme, several state points show dispersion of DNBR compared to the reference scheme, as seen in Figure 5.11 and Figure 5.13. This is due to the fact that the spatial discretizations (for

steady-state) of the QG Scheme and the QRq Scheme differ from that of the REF Scheme for the non-linear term involving the relative velocity (see Section 4.3). This leads to different solutions **on the industrial mesh used in THYC-coeur**. This effect is particularly noticeable for the Chexal-Lellouche correlation, where the relative velocity reaches high negative values (almost equal in magnitude to the mixture velocity) as shown in Section 4.3. This effect was not observed with the Bestion correlation because the relative velocity in this case is lower and positive. In Figures 5.11 and 5.13, the two-phase points ($X > 0.2$) are particularly affected, as the non-linear term is larger in these cases (since it takes the form $u_r y(1 - y)$). For these two-phase cases ($X > 0.2$), the variation is negative, which means that the QG and QRq schemes are more penalizing than the reference scheme. However, these variations remain below $\pm 0.5\%$, which is still acceptable. These differences arise because the axial mesh used is not fully converged. The new schemes offer a slight gain in accuracy for an equivalent mesh, as shown in the analyses in Section 4.3. It is therefore recommended to use these new schemes when applying the Chexal-Lellouche correlation.

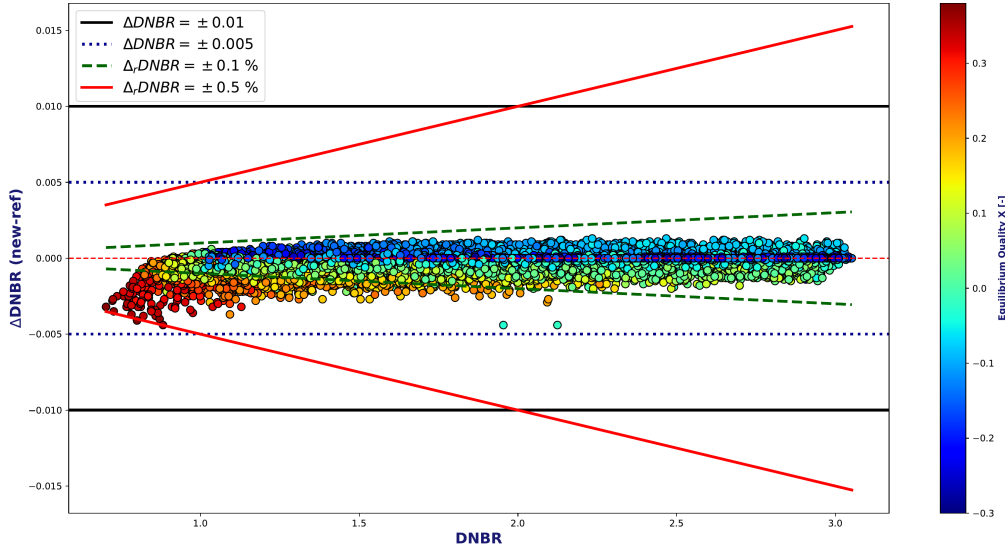


FIGURE 5.11

Relative variation of DNBR for cvQG as a function of the DNBR with the Chexal-Lellouche correlation.

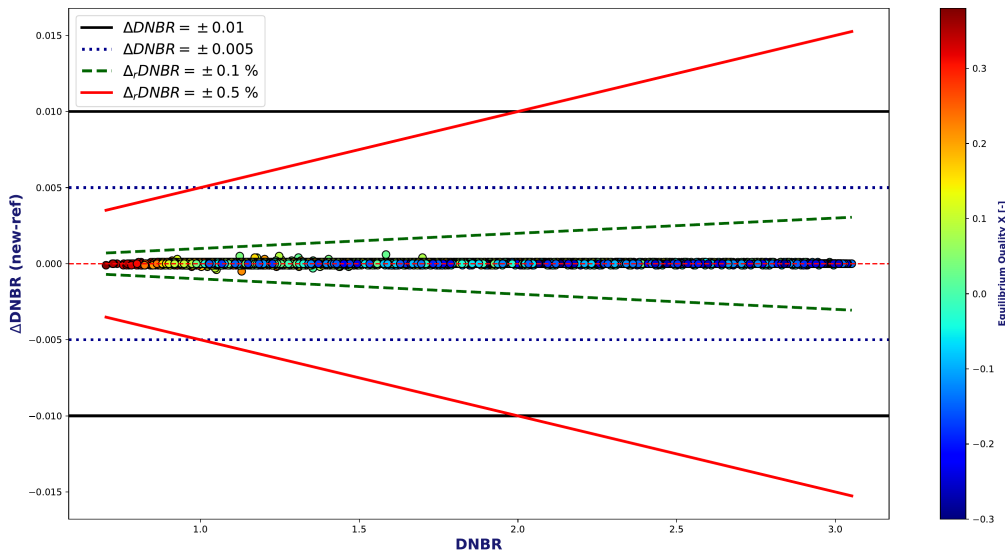


FIGURE 5.12

Relative variation of DNBR for cvQRd as a function of the DNBR with the Chexal-Lellouche correlation.

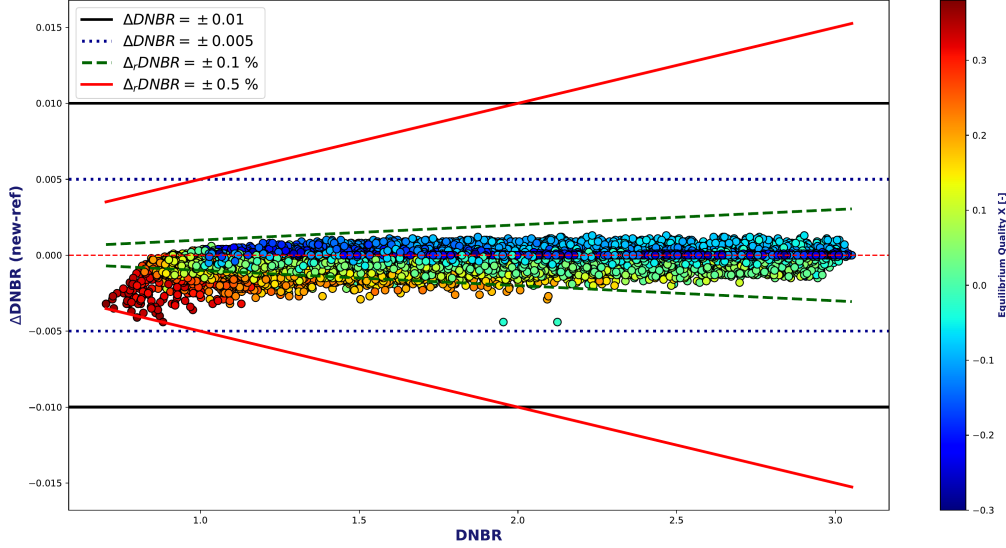


FIGURE 5.13

Relative variation of DNBR for cvQRq as a function of the DNBR with the Chexal-Lellouche correlation.

5.4.2 Impact on the performance at an industrial stopping criterion

For each scheme and with a stopping criterion of 10^{-4} , the DNBR variations compared with cvREF are represented as a function of the DNBR in Figure 5.14 for REF Scheme, in Figure 5.15 for QG Scheme, in Figure 5.16 for QRd Scheme and in Figure 5.17 for QRq Scheme. The performance indicators for each scheme are shown in Table 5.2.

As for the stopping criterion of 10^{-11} , the QG and QRq Scheme show a dispersion of DNBR (when comparing to cvREF) with variations higher than $\pm 0.1\%$. The dispersion is still acceptable, with most variations less than $\pm 0.5\%$. The QRd Scheme and the REF Scheme show a negligible DNBR dispersion. For the Chexal-Lellouche correlation, the three new schemes (QG, QRd and QRq) are faster, by **almost a factor 2** in the number of iterations and in CPU time: the new implementation of the disequilibrium equation allows for a 50% reduction in computation time. As a conclusion, all schemes can be used and the QRq and QG schemes should be prioritized.

Scheme	indREF	indQG	indQRd	indQRq
$m(\Delta_r)$	6.38 ± 4.48	3.68 ± 3.80	6.01 ± 4.44	3.62 ± 3.76
$N(\Delta_r > 0.1\%)$	4	1168	10	1180
$N(\Delta_r > 0.5\%)$	0	1	0	2
$\langle t_{cpu} \rangle_{BCD}$ (s)	14.98 ± 4.06	7.96 ± 1.16	8.36 ± 1.55	8.27 ± 1.37
$\langle N_{iter} \rangle_{BCD}$	134.94 ± 36.78	69.25 ± 9.84	70.24 ± 11.84	72.09 ± 11.82

TABLE 5.2

Performance indicators for scheme with the Chexal-Lellouche correlation and a stopping criterion of 10^{-4} .

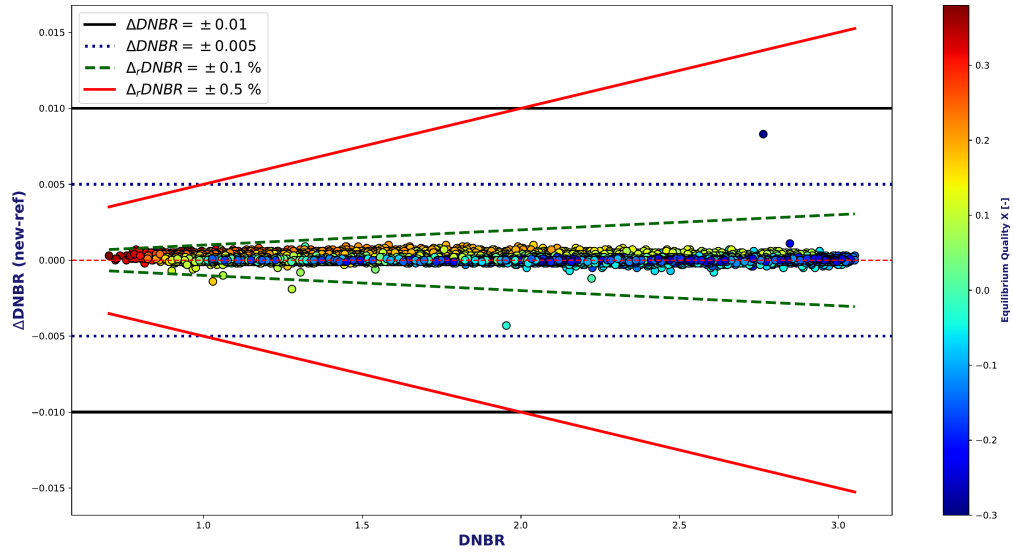


FIGURE 5.14

Relative variation of DNBR for indREF as a function of the DNBR with the Cheval-Lellouche correlation.

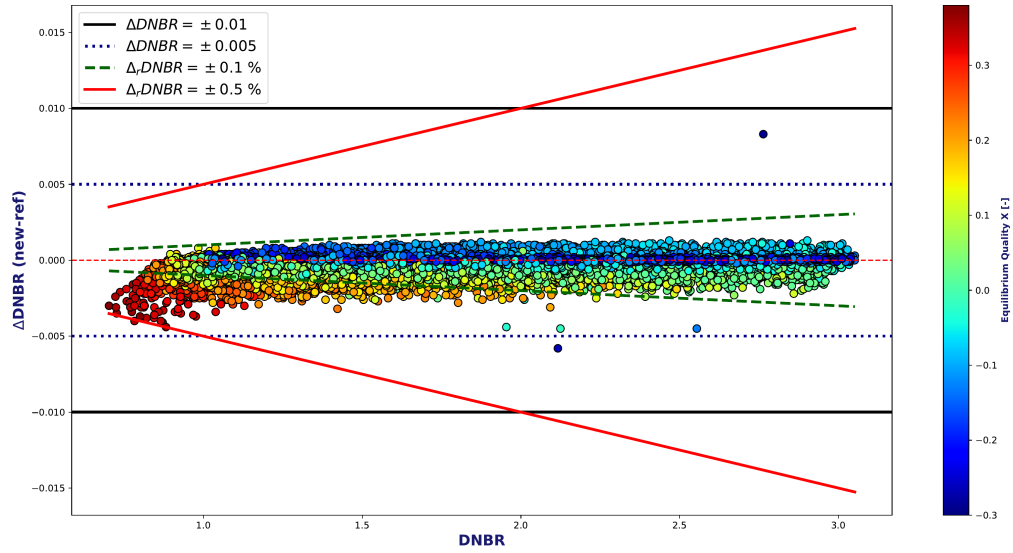


FIGURE 5.15

Relative variation of DNBR for indQG as a function of the DNBR with the Cheval-Lellouche correlation.

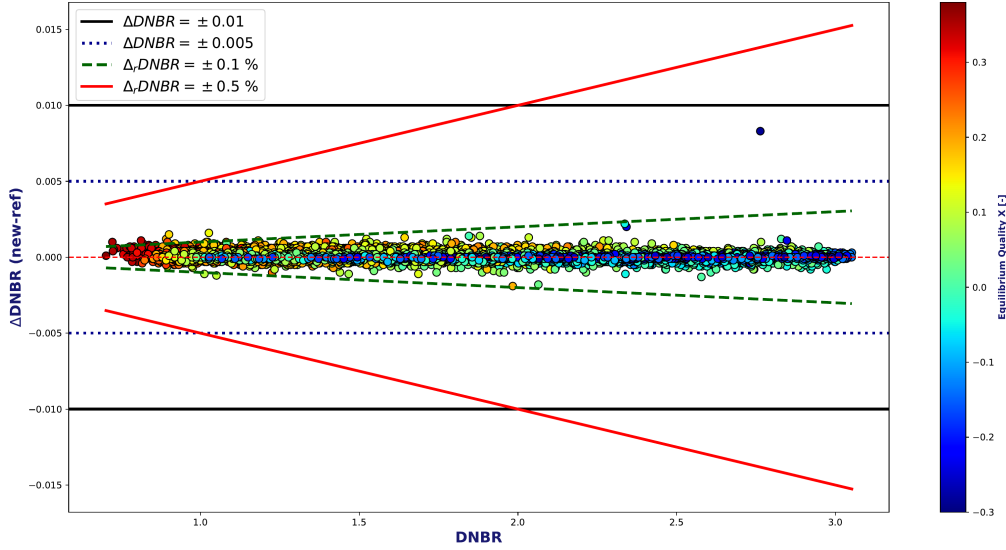


FIGURE 5.16

Relative variation of DNBR for indQRd as a function of the DNBR with the Chexal-Lellouche correlation.

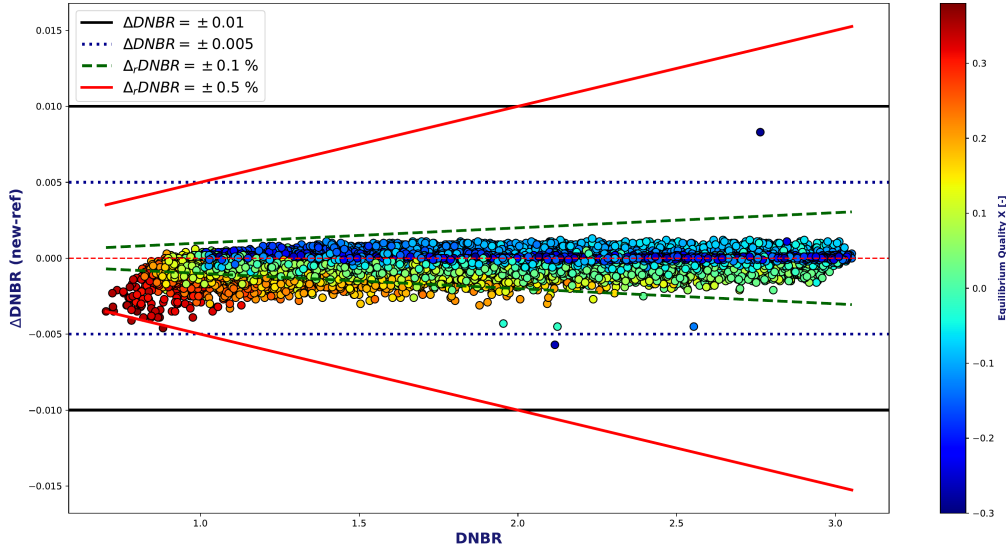


FIGURE 5.17

Relative variation of DNBR for indQRq as a function of the DNBR with the Chexal-Lellouche correlation.

5.5 Schemes comparison

Table 5.3 summarizes the usage recommendations for each scheme based on the simulations performed with THYC-coeur. It leads to the conclusion that the **QG Scheme** should be used **by default** for **steady-state simulations**. If the time step condition becomes limiting in some cases, the QRq scheme is then recommended, except in the case of counter-current flows. In practice, counter-current flows are almost never encountered for the applications concerned here. If, in the future, some counter-current simulations need to be performed and the QG Scheme imposes a too constraining time step, the QRd Scheme could be used instead. For **unsteady simulations**, the **QG Scheme** should also be used. Note that in the case of unsteady simulations, the QRd and QRq schemes are, by construction, not applicable.

The three new finite-volume scheme implementations in THYC-coeur have improved the perfor-

mance for the Chexal-Lellouche correlation, particularly in terms of CPU time. For the Bestion correlation, no performance improvement was observed. However, the new schemes offer better robustness with a control over the time step condition (or even eliminate it), while maintaining the same level of accuracy on industrial meshes.

Scheme	Constraints time step	Other constraints	Steady-state Bestion	Steady-state Chexal-Lellouche	Unsteady simulations
REF	Complex	-	Valid	Less recommended	-
QRd	None	Outlet BC Not limiting	Valid	Less recommended	Not suitable
QRq	None	Co-current flows	Valid	Recommended	Not suitable
QG	Low Not limiting	None	Valid	Recommended	Recommended

TABLE 5.3

Comparison table of schemes characteristics for THYC-coeur applications.

Part III

ThermoTorch 1D without relative velocity

Chapter 6

ThermoTorch: a 1D finite volume code

The ThermoTorch 1D code simulates a one-dimensional two-phase water-steam flow using a finite-volume method. Its initial version was presented in [68]. The current version of ThermoTorch 1D is described in details in this chapter.

This code is a simplified, one-dimensional version of the THYC-coeur code. The objective is to have a prototype that remains close enough to the industrial code to allow testing of numerical schemes and a new method to accelerate the convergence towards the steady-state. It is used in **Chapter 7** to validate the proposed numerical schemes, in both steady-state and unsteady cases, comparing the results with the analytical solutions presented in **Chapter 3**. In **Chapter 8**, a steady-state configuration of the code that is representative of a sub-channel flow in a reactor core, is used to test a Machine Learning-based acceleration method envisioned for the industrial THYC-coeur code.

First, the physical model implemented in the code is presented in Section 6.1, with a focus on the two continuous systems of equations proposed. In Section 6.2, the temporal and spatial discretizations are detailed. Based on this discretization, the numerical schemes implemented in the code are then explained in Section 6.3. Finally, Section 6.4 highlights the different configurations of the code used in the following chapters.

6.1 System of equations

In this section, the two physical models used in the ThermoTorch code are presented. They are simplified versions of the THYC-coeur model. First, the system of conservative equations is detailed for the two models implemented in ThermoTorch. Then, the equations are manipulated to derive the systems implemented numerically.

6.1.1 Conservative balance equations

A one-dimensional spatial domain $\Omega = [x_{\min}, x_{\max}]$ is considered during a time interval $[0, T]$ to represent a two-phase flow of a water-steam mixture. A volumetric heat source, denoted $\phi(x, t)$, can be applied over a portion of the domain. Two models are available in ThermoTorch: the 3-equation model, which describes a thermodynamic equilibrium two-phase flow, and the 4-equation model, which accounts for subcooled boiling (see **Chapter 1**). The simplifications compared to the THYC-coeur code are as follows: the flow is considered in a free medium

$$\varepsilon(x) = 1, \tag{6.1}$$

without second-order differential terms, without gravity and friction due to solid and with no relative velocity

$$u_r = 0. \tag{6.2}$$

With these hypotheses, the two formulations of the energy equation presented in **Chapter 1** - in terms of total energy or in terms of enthalpy - are **equivalent**. The model is expressed in terms of the total energy, in order to obtain a conservative system similar to the one exposed in **Chapter 3**. The resulting **4-equation model** is similar to a Homogeneous Relaxation Model (HRM) [16]:

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2) + \partial_x p = 0, \\ \partial_t(\rho E) + \partial_x(\rho u E) + \partial_x(pu) = \phi(x, t), \\ \partial_t(\rho y) + \partial_x(\rho u y) = \Gamma, \end{cases} \quad (6.3)$$

with

$$\begin{cases} E = e(p, \rho, y) + \frac{u^2}{2}, \\ \Gamma = \frac{\chi \phi(x)}{\bar{L}} + \rho \frac{\bar{y} - y}{\tau}. \end{cases} \quad (6.4)$$

where χ is the fraction used for vaporization, and τ is the relaxation time required to reach thermodynamic equilibrium (see Section 1.4.1 for details). These two quantities are modeled using closure laws as detailed in **Appendix 6.A**. An equation of state provides the internal energy e as a function of pressure p , mixture density ρ and mass fraction y :

$$e(p, \rho, y) = y \bar{e}_g(p) + (1 - y) e_l \left(p, \rho_l = \frac{1 - y}{\frac{1}{\rho} - \frac{y}{\bar{\rho}_g(p)}} \right), \quad (6.5)$$

with $\bar{e}_g$ the internal energy and $\bar{\rho}_g$ the density of the gas phase computed at saturation (see Equation (1.66)).

The 3-equation model consists of the same equations as those in Equations (6.3), except for the fourth equation, which governs the mass fraction y . The equation is not solved; instead, the mass fraction is computed at thermodynamic equilibrium using Equation (1.59), which is recalled here:

$$\bar{y}(p, \rho) = \frac{h(p, \rho) - \bar{h}_l(p)}{\bar{h}_g(p) - \bar{h}_l(p)}, \quad (6.6)$$

with $h(p, \rho) = e(p, \rho) + \frac{p}{\rho}$ the enthalpy. The **3-equation model** is:

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2) + \partial_x p = 0, \\ \partial_t(\rho E) + \partial_x(\rho u E) + \partial_x(pu) = \phi(x), \\ y = \bar{y}(\rho, p). \end{cases} \quad (6.7)$$

with $E = e(p, \rho, \bar{y}) + \frac{u^2}{2}$.

6.1.2 Unknowns considered and implemented equations

Equations (6.3) are expressed in terms of density, velocity, total energy, and mass fraction. The objective is to transform these equations to obtain a final system expressed in terms of pressure p , mixture mass flux defined by $q = \rho u$, mixture enthalpy h , and mass fraction y . The aim of these manipulations is to obtain a system of equations that is easily implementable numerically, particularly by decoupling the discrete equations on mass fraction and on energy from the other equations

in order to solve the unknowns step by step. This section presents the necessary transformations to obtain this final system, which is then discretized using finite volume schemes in Sections 6.2 and 6.3. First, the steps required to derive the enthalpy equation from the total energy equation are presented. However, the obtained equation is still coupled with the pressure. Numerically, this would imply having a coupled system of pressure, velocity, and enthalpy. In this case, the matrix system is complex to solve, which can significantly increase CPU computation time. The objective is therefore to propose a method to decouple energy from the velocity-pressure system. To do this, a thermodynamic function, denoted $\mathfrak{s}$, depending on enthalpy, pressure, and mass fraction, is introduced. It is called the **pseudo-entropy** and noted $\mathfrak{s}$ as it is defined as a mixture quantity from entropy of each phase. For the 3-equation model, the pseudo-entropy is an entropy of the system (see Section 6.3). When considering the 4-equation model, this pseudo-entropy should not be confused with the actual entropy of the system noted s , whose formula is not explicitly known. An approximation on the pseudo-entropy enables to express the energy equation independently of the other equations when it is discretized in time (see Section 6.3).

Using the mixture EoS, the density is expressed as a function of pressure, mass fraction, and the pseudo-entropy $\mathfrak{s}$. This makes it possible to write the total mass conservation equation in the form of a pressure equation. In the momentum equation, a change of variable from velocity u to the mass flux $q = \rho u$ allows the equation to be rewritten in terms of the mass flux.

Transformation of the total energy equation into the enthalpy equation

The total energy equation is rewritten in terms of enthalpy using a standard manipulation. The kinetic energy equation is obtained by multiplying the momentum equation by the velocity (and using the total mass conservation):

$$\partial_t \left(\rho \frac{u^2}{2} \right) + \partial_x \left(\rho u \frac{u^2}{2} \right) + u \partial_x p = 0. \quad (6.8)$$

This equation is subtracted from the total energy equation. Using the mixture enthalpy $h = e + p/\rho$, the energy equation writes

$$\partial_t (\rho h) + \partial_x (\rho u h) = \partial_t p + u \partial_x p + \phi(x). \quad (6.9)$$

Definition of the pseudo-entropy $\mathfrak{s}$

The energy Equation (6.9) is still coupled with the pressure. In what follows, a thermodynamic function called pseudo-entropy and noted $\mathfrak{s}$ is proposed to decouple the discretized energy equation from the pressure equation. It is defined as a mixture quantity, derived from the entropy of each phase:

$$\mathfrak{s}(p, h, y) = y \overline{s_g}(p) + (1 - y) s_l \left(p, h_l = \frac{h - y \overline{h_g}(p)}{1 - y} \right), \quad (6.10)$$

with the gas phase assumed to be at saturation here. For the 3-equation model, the pseudo-entropy corresponds to the entropy of the system $\mathfrak{s} = s$. It satisfies

$$c^2(p, \rho, y) \left(\frac{\partial \mathfrak{s}}{\partial p} \right)_{\rho, y} + \left(\frac{\partial \mathfrak{s}}{\partial \rho} \right)_{p, y} = 0, \quad (6.11)$$

where the mixture speed of sound $c(p, \rho, y)$ is defined by

$$(\rho c)^2 = \left(\frac{\partial e}{\partial p} \right)_{\rho, y}^{-1} \left(p - \rho^2 \left(\frac{\partial e}{\partial \rho} \right)_{p, y} \right). \quad (6.12)$$

The following differential equation can be used

$$T_l ds = dh - v dp, \quad (6.13)$$

with $v = \frac{1}{\rho}$ and T_l the liquid temperature. This expression is valid for both one-phase liquid flow and two-phase flow. It allows Equation (6.9) to be written in terms of entropy, decoupled from the other variables:

$$T_l \left[\partial_t(\rho s) + \partial_x(\rho u s) \right] = \phi(x). \quad (6.14)$$

Equation (6.13) is **not valid for the 4-equation model**, when the mixture is not at thermodynamic equilibrium. In that case, **the two phases do not have the same temperature. The system entropy cannot be easily derived** from the entropies of each phase in this case. The pseudo-entropy $\mathfrak{s}$ of Equation (6.10) is not the entropy of the system. It satisfies

$$T_l d\mathfrak{s} = dh - (v + y(T_l - T_s(p)) \frac{ds_g}{dp}) dp + \mathfrak{s}_y dy, \quad (6.15)$$

with

$$\mathfrak{s}_y = (h_l - \bar{h}_g) - T_l(s_l - \bar{s}_g) = \mu_l(p, T_l) - \bar{\mu}_g(p, T_s) + (T_s - T_l)\bar{s}_g. \quad (6.16)$$

The temperature T_k of phase k is defined in Equation (1.6) and the chemical potential μ_k of phase of phase k in Equation (1.65). As expected, Equation (6.13) can be obtained from Equation (6.15) when the liquid phase is also assumed to be at saturation, since, in that case: $\mu_l = \bar{\mu}_l = \bar{\mu}_g$ and $T_l = T_s(p)$.

The pseudo-entropy $\mathfrak{s}$ is used in the implicit formulation of the numerical schemes within the enthalpy equation for the 4-equation model (see Section 6.3.1) through the following approximation

$$T_l \delta \mathfrak{s} \approx \delta h - v \delta p. \quad (6.17)$$

With this approximation, the discrete energy equation can be decoupled from the other equations, although some terms have been neglected (highlighted in blue in Equation (6.15)). It is detailed in Section 6.3. As it will be shown in Section 6.3, this approximation is used only in the unsteady part of the discrete numerical scheme. In the steady-state balance ($\partial_x(\rho u h) - u \partial_x p + \phi(x)$), the energy equation is still expressed in terms of enthalpy and pressure. When considering steady-state, the unsteady terms in the equations tend toward zero. The terms neglected (blue terms in Equation (6.15)) in the differential of the pseudo-entropy are therefore zero at that point. The steady-state solution of the original continuous model is thus preserved despite this approximation. This approximation only imposes a different numerical path for the computation to reach the steady state. **Thus, the resulting steady-state solutions are not affected by this approximation.** In **Chapter 7**, a smooth unsteady solution of a one-dimensional Riemann problem for the 4-equation model (with a simplified mixture thermodynamics) is used to evaluate the impact of this approximation on unsteady solutions.

Mixture density linearization to rewrite the total mass conservation equation

The objective is to transform the mass conservation equation into an equation for pressure. To do so, the mixture density $\rho(p, \mathfrak{s}, y)$ can be expressed as a function of pressure p , pseudo-entropy $\mathfrak{s}$, and mass fraction y , using the mixture EoS. The linearized form of the density is written as follows

$$d\rho = \alpha dp + \beta d\mathfrak{s} + m dy, \quad (6.18)$$

$$\text{with } \alpha = \left(\frac{\partial \rho}{\partial p} \right)_{\mathfrak{s}, y}, \beta = \left(\frac{\partial \rho}{\partial \mathfrak{s}} \right)_{p, y} \text{ and } m = \left(\frac{\partial \rho}{\partial y} \right)_{p, \mathfrak{s}}.$$

The coefficients α , β and m are given in **Appendix 6.C** for different equations of state available in ThermoTorch. The total mass conservation equation from Equations (6.3) can be rewritten

$$\alpha \partial_t p + \beta \partial_t \mathbf{s} + m \partial_t y + \partial_x q = 0. \quad (6.19)$$

System of equations implemented numerically

The chosen unknowns for the final system are (p, q, h, y) (or (p, q, h) for the 3-equation model). The system for the 4-equation model in ThermoTorch 1D consists of the total mass balance (equation for pressure), the momentum balance (equation for mass flux), the energy balance (equation for enthalpy), and the mass fraction balance (equation for mass fraction):

$$\forall (x, t) \in \Omega \times [0, T], T > 0 \text{ and } \Omega = [x_{min}, x_{max}], \quad \begin{cases} \alpha \partial_t p + \beta \partial_t \mathbf{s} + m \partial_t y + \partial_x q = 0, \\ \partial_t q + \partial_x (uq) + \partial_x p = 0, \\ \partial_t (\rho h) - \partial_t p + \partial_x (qh) - u \partial_x p = \phi(x), \\ \partial_t (\rho y) + \partial_x (qy) = \Gamma_p + \rho \frac{\bar{y} - y}{\tau}. \end{cases} \quad (6.20)$$

The resulting system of equations is easily implementable numerically. However, it is no longer in a conservative form, which means that the resulting numerical model will not be able to simulate solutions with shocks (see Section 7.2). In practice, the industrial code THYC-coeur is not used for applications involving shocks. The 3-equation model is the same model as Equations (6.20), but without the mass fraction equation.

6.2 Discretization framework for finite volume schemes

The continuous equations presented in the previous section are studied on a one-dimensional domain $\Omega = [x_{min}, x_{max}]$ over the time interval $[0, T]$. In this section, the temporal and spatial discretizations are presented. This allows for the transition from a continuous problem to a discrete one, whose solution approximates that of the continuous solution. After discretizing the time interval, the spatial discretization is introduced to enable the integration of the time-discretized equations using finite volume schemes. Here, a staggered grid mesh is used. It enables to avoid odd-even decoupling between pressure and momentum leading to checkerboards patterns. The staggered mesh grid also provides a gain in accuracy, compared to collocated grids. In particular, the pressure gradient in the momentum balance is centered. This choice is also justified by the fact that this type of scheme has good behavior in the low Mach limit (see for example [71]). Finally, the various methods available in ThermoTorch for controlling the time step are presented.

6.2.1 Time discretization

The time interval $[0, T]$ is discretised in N_T intervals $[t^n, t^{n+1}]$, $n \in \llbracket 0, N_T - 1 \rrbracket$ with $\Delta t^n = t^{n+1} - t^n$, $n \in \llbracket 0, N_T - 1 \rrbracket$ such that

$$t^0 = 0 \quad ; \quad \forall n \in \llbracket 0, N_T - 1 \rrbracket, \quad t^{n+1} = t^n + \Delta t^n \text{ and } T = t^{N_T}. \quad (6.21)$$

For any function $f(x, t)$, its value at time t^n is written:

$$\forall n \in [0, N_T], \forall x \in \Omega, \quad f^n(x) = f(x, t^n). \quad (6.22)$$

The time increment of $f(x, t)$ between time t^n and t^{n+1} is defined by

$$\forall x \in \Omega, \quad \delta f(x) = f^{n+1}(x) - f^n(x). \quad (6.23)$$

6.2.2 Spatial discretization with a staggered grid

A regular one-dimensional staggered Cartesian mesh is implemented in ThermoTorch. It is represented in Figure 6.1. This means that scalar variables are stored in the centers of the control volumes while the mass flux (and velocity) variables are defined at the faces of the control volumes, giving the staggered grid. The first mesh, where pressure, energy, and mass fraction are defined, is called **pressure mesh**. The second mesh is centered on the faces of the pressure cells, which defines new control volumes. This mesh is called **flow mesh**. To construct this staggered mesh, the spatial domain $\Omega = [x_{min}, x_{max}]$ is divided into cells of uniform size, which define the control volumes for the pressure mesh. The size of the cells is denoted Δx and satisfies

$$\Delta x = \frac{x_{max} - x_{min}}{n_x - 1/2}, \quad (6.24)$$

with $n_x + 1$ the number of cells in the pressure mesh. The cell centers of the pressure mesh are noted x_i with $i \in [0, n_x]$. They satisfy, $\forall i \in [0, n_x]$,

$$x_i = x_{min} + (i - 1/2)\Delta x. \quad (6.25)$$

The flow mesh is composed of n_x cells of center $x_{i-1/2}$ with $i \in [1, n_x]$. The centers satisfy, $\forall i \in [1, n_x]$,

$$x_{i-1/2} = x_{min} + (i - 1)\Delta x. \quad (6.26)$$

The value of the function $f^n(x), x \in \Omega$ at point x_i of the pressure mesh is noted:

$$\forall n \in [0, N_T], i \in [0, n_x], \quad f_i^n = f^n(x_i) = f(t^n, x_i). \quad (6.27)$$

The value of function $f^n(x), x \in \Omega$ at point $x_{i-1/2}$ of the flow mesh is noted:

$$\forall n \in [0, N_T], i \in [1, n_x], \quad f_{i-1/2}^n = f^n(x_{i-1/2}) = f(t^n, x_{i-1/2}). \quad (6.28)$$

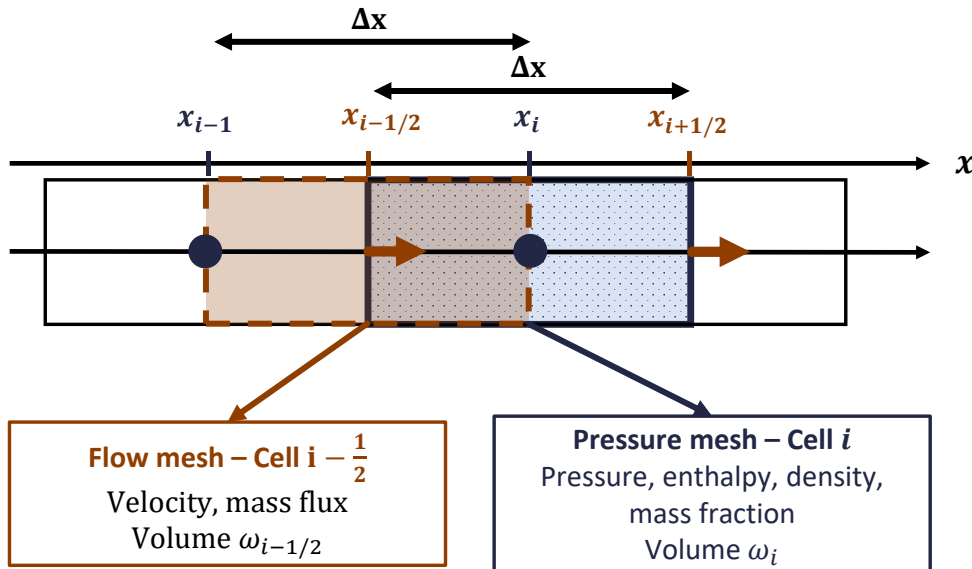


FIGURE 6.1
Schematic representation of the staggered-grid mesh in ThermoTorch 1D.

Starting from Equations (6.20), the pressure, energy, and mass fraction equations are integrated over the control volumes of the pressure mesh, whereas the momentum equation is integrated over

the control volumes of the flow mesh. ω_i denotes the volume of the cell i , which has neighboring cells $i - 1$ and $i + 1$. Similarly, $\omega_{i-1/2}$ denotes the volume of the flow cell $i - 1/2$ (see Figure 6.1), with neighboring cells $i - 3/2$ and $i + 1/2$.

6.2.3 Control of the time step

Time advancement can be handled in different ways in ThermoTorch. First, simulations can be performed with a constant time step Δt_0 chosen by the user, such that $\forall n \in \llbracket 0, N_T - 1 \rrbracket$

$$\Delta t^n = \Delta t_0. \quad (6.29)$$

A CFL can also be imposed by the user, either by considering only the fluid velocity (slow wave) as follows, $\forall n \in \llbracket 0, N_T - 1 \rrbracket$,

$$\Delta t^n = \text{CFL}_u \frac{\Delta x}{\max_{i \in [1, n_x]} |u_{i-1/2}^n|}, \quad (6.30)$$

or by considering the mixture speed of sound (fast wave),

$$\Delta t^n = \text{CFL}_{u+c} \frac{\Delta x}{\max_{i \in [1, n_x]} (|u_{i-1/2}^n| + c_{i-1/2}^n)}. \quad (6.31)$$

6.3 Finite volume numerical scheme

In this section, using the temporal and spatial discretizations from Section 6.2, the numerical scheme used to solve the system of Equations (6.20) with a finite volume scheme is described. A similar scheme is used for the 3-equation model, excluding the scheme for the equation for the mass fraction.

Using Equations (6.20) from Section 6.1.2, the time discretization detailed in Section 6.3.1 is used to describe the time-stepping method applied to these equations. Then, the global finite volume scheme is presented for each equation in Section 6.3.2, by integrating the time-discretized equations over the staggered grid mesh.

For the numerical schemes of each equation, the terms are made as implicit as possible in order to easily increase the time step. However, the terms are made implicit only on the condition that the equations are not recoupled with each other.

6.3.1 Time stepping

Methodology

Equations (6.20) are formulated at time t^{n+1} . An implicit Euler scheme is used for the time derivative, such that for $f \in \{p, q, \mathfrak{s}, y\}$, $\forall x \in \Omega$,

$$\partial_t f(x, t^{n+1}) \approx \frac{f^{n+1}(x) - f^n(x)}{\Delta t^n} = \frac{\delta f(x)}{\Delta t^n}. \quad (6.32)$$

For enthalpy and mass fraction, mass conservation is used and the density is evaluated at time t^n . For $f \in \{h, y\}$, the implicit time-stepping scheme is, $\forall x \in \Omega$,

$$\begin{aligned} \partial_t(\rho f)(x, t^{n+1}) &= \rho \partial_t f(x, t^{n+1}) - f \partial_x q(x, t^{n+1}) \\ &\approx \rho^n \frac{\delta f(x)}{\Delta t^n} - f^{n+1} \partial_x q^n. \end{aligned} \quad (6.33)$$

The time-stepping scheme is taken implicit (at time t^{n+1}), so that most of the terms detailed below are taken implicitly (at time t^{n+1}). Several exceptions should be noted. The density appearing as a factor of the time derivatives is always taken at time t^n , as was done in Equation (6.33). In addition, the transport vector, whether it is the mass flux q or the velocity u , is taken at time t^n , so that the scheme is linear and not coupled with the momentum equation when considering the mass fraction and energy equation. This allows the mass fraction and energy equations to remain decoupled from the other equations, particularly from the momentum equation. As a consequence, for any function $f \in \{q, h, p, y\}$ transported by the mass flux q (or the velocity u), the following scheme is used, $\forall x \in \Omega$,

$$\begin{aligned}\partial_x(qf)(x, t^{n+1}) &\approx \partial_x(q^n f^{n+1}) \\ &\approx \partial_x(q^n(x) \delta f(x)) + \partial_x(q^n(x) f^n(x)).\end{aligned}\tag{6.34}$$

The pressure gradient in the momentum balance and the mass flux gradient in the mass balance are treated implicitly, i.e. evaluated at time t^{n+1} so that, $\forall x \in \Omega$,

$$\begin{cases} \partial_x p(x, t^{n+1}) = \partial_x(\delta p)(x) + \partial_x p^n(x), \\ \partial_x q(x, t^{n+1}) = \partial_x(\delta q)(x) + \partial_x q^n(x). \end{cases}\tag{6.35}$$

Pressure equation

The conservation of the mass was written in pressure, pseudo-entropy $\mathfrak{s}$ and mass fraction. The time derivative are taken implicit and the mass flux is time-discretized with Equation (6.35) so that the mass balance discretized at time t^{n+1} writes:

$$\alpha^n \delta p + \beta^n \delta \mathfrak{s} + m^n \delta y + \Delta t^n \partial_x(\delta q) = -\Delta t^n \partial_x q^n(x).\tag{6.36}$$

In this equation for pressure, the increments of the four unknowns considered appear. It is then coupled with the three other equations.

Momentum equation

Using Equations (6.33), (6.34) and (6.35), the time-stepping scheme for the momentum writes:

$$\delta q + \Delta t^n \partial_x(u^n \delta q) + \Delta t^n \partial_x(\delta p) = -\Delta t^n (\partial_x(u^n q^n) + \partial_x p^n).\tag{6.37}$$

The time-stepping scheme for the momentum equation couples the pressure (due to the gradient $\partial_x p$) and the mass flux. As explained later, the mass fraction and energy equations (with the pseudo-entropy formulation) are written decoupled from pressure and mass flux. This allows the mass fraction and pseudo-entropy increments to be solved independently. What remains is the coupled velocity-pressure system, expressed in terms of pressure increment δp and mass flux increment δq with Equations (6.36) and (6.37).

Energy equation

Using Equations (6.33) and (6.34) for the enthalpy Equation (6.9), the time-stepping scheme for the enthalpy equation is

$$\frac{\rho^n}{\Delta t^n} \left(\frac{\delta h(x) - v^n \delta p(x)}{\Delta t^n} \right) + \partial_x(q^n \delta h) - \partial_x(u^n \delta p) - (\delta h \partial_x q^n - \delta p \partial_x u^n) = -\text{BHM}^n(x),\tag{6.38}$$

with

$$\text{BHM}^n(x) = \partial_x(q^n h^n) - \partial_x(u^n p^n) - h^n \partial_x q^n - p^n \partial_x u^n - \phi^n(x).\tag{6.39}$$

The pressure increment appears in this equation due to the presence of the time derivative of pressure $\partial_t p \approx \frac{\delta p}{\Delta t^n}$ and the pressure transport term $u \partial_x p = \partial_x(u p) - p \partial_x u$ discretized at time t^{n+1} (implicit scheme). Without further manipulation, the enthalpy equation remain coupled with the pressure equation. The **linearized approximation** of the pseudo-entropy $\mathfrak{s}$ from Equation (6.17) is used to write $T_l^n \delta \mathfrak{s} \approx \delta h - v^n \delta p$, so that the time-discretized enthalpy equation becomes

$$\frac{\rho^n}{\Delta t^n} T_l^n \delta \mathfrak{s} + T_l^n \partial_x (q^n \delta \mathfrak{s}) - T_l^n \delta \mathfrak{s} \partial_x q^n = -\text{BHM}^n. \quad (6.40)$$

With this new formulation in pseudo-entropy, the time-stepping scheme for the energy equation is independent of the other equations.

Mass fraction equation

For the disequilibrium equation on the mass fraction, the scheme presented in **Chapter 4** is used (the three proposed schemes QG, QRd, and QRq are equivalent since the relative velocity is zero). For the production term Γ_p , the two schemes GAMi and GAMc are implemented in ThermoTorch (see **Appendix 4.D.1**). Only the GAMi scheme is presented here. The time-stepping scheme of the mass fraction equation from System (6.20) writes:

$$\rho^n \delta y + \Delta t^n (\partial_x (q^n \delta y) - \delta y \partial_x q^n) + \frac{\rho^n \Delta t^n}{\tau^n} \delta y = -\Delta t^n \text{BYM}^n, \quad (6.41)$$

with

$$\text{BYM}^n = \partial_x (q^n y^n) - y^n \partial_x q^n - \rho^n \frac{\bar{y}^n - y^n}{\tau^n} - \Gamma_p^n. \quad (6.42)$$

The equation on mass fraction is independent of other increments. It will be solved independently.

Global time-stepping scheme

Equations (6.20) at time t^{n+1} are approximated by the following time-stepping scheme:

$$\left\{ \begin{array}{l} \alpha^n \delta p + \beta^n \delta \mathfrak{s} + m^n \delta y + \Delta t^n \partial_x (\delta q) = -\Delta t^n \text{BMM}^n(x), \\ \delta q + \Delta t^n \partial_x (u^n \delta q) + \Delta t^n \partial_x (\delta p) = -\Delta t^n \text{BQM}^n(x), \\ \rho^n T_l^n \delta \mathfrak{s} + T_l^n \Delta t^n (\partial_x (q^n \delta \mathfrak{s}) - \delta \mathfrak{s} \partial_x q^n) = -\Delta t^n \text{BHM}^n(x), \\ \rho^n \delta y + \Delta t^n (\partial_x (q^n \delta y) - \delta y \partial_x q^n) + \frac{\rho^n \Delta t^n}{\tau^n} \delta y = -\Delta t^n \text{BYM}^n(x), \end{array} \right. \quad (6.43)$$

with the steady-state balance at time t^n defined by

$$\left\{ \begin{array}{l} \text{BMM}^n(x) = \partial_x q^n, \\ \text{BQM}^n(x) = \partial_x (u^n q^n) + \partial_x p^n, \\ \text{BHM}^n(x) = \partial_x (q^n h^n) - h^n \partial_x q^n - \left(\partial_x (u^n p^n) - p^n \partial_x u^n \right) - \phi^n(x), \\ \text{BYM}^n(x) = \partial_x (q^n y^n) - y^n \partial_x q^n - \rho^n \frac{\bar{y}^n - y^n}{\tau^n} - \Gamma_p^n. \end{array} \right. \quad (6.44)$$

6.3.2 Spatial integration and global numerical scheme

In this section, the time-discretized Equations (6.43) are integrated over the control volumes of the spatial mesh. The energy equation, the mass fraction equation and the pressure equation are integrated on the pressure mesh, for a cell i . The mass flux equation (momentum equation) is integrated on the flow mesh, for a cell $i - 1/2$. Details on the scheme used for the convective terms

is first provided. Then, the interpolation of the velocity on the pressure mesh is presented to use it in the discrete momentum equation. The global schemes for each equation are then given.

Scheme for the convective terms on pressure mesh

Several convective terms are integrated over the pressure mesh. The variables involved are $f \in \{\delta\mathbf{s}, \delta y, h^n, p^n, y^n\}$. As in Section 4.2, the sign of the mass flux $q_{i-1/2}^n$ at cell $i - 1/2$ is denoted

$$\text{sg}_{i-1/2} = \text{sg}(q_{i-1/2}^n) = \begin{cases} 0 & \text{if } q_{i-1/2}^n < 0, \\ 1 & \text{if } q_{i-1/2}^n \geq 0. \end{cases} \quad (6.45)$$

An upwind implicit scheme, depending on the sign of the mass flux, is used for the convective terms, for $f \in \{\delta\mathbf{s}, \delta y, h^n, p^n, y^n\}$,

$$\begin{aligned} \int_{\omega_i} \partial_x(q^n f) d\Omega \approx & \left[q_{i+1/2}^n (\text{sg}_{i+1/2} f_i + (1 - \text{sg}_{i+1/2}) f_{i+1}) \right. \\ & \left. - q_{i-1/2}^n (\text{sg}_{i-1/2} f_{i-1} + (1 - \text{sg}_{i-1/2}) f_i) \right] \frac{\omega_i}{\Delta x}. \end{aligned} \quad (6.46)$$

The terms of the form " $f \partial_x q^n$ ", which appear due to mass conservation, for $f \in \{\delta\mathbf{s}, \delta y, h^n, y^n\}$ are integrated using

$$\int_{\omega_i} f \partial_x q^n d\Omega \approx f_i (q_{i+1/2}^n - q_{i-1/2}^n) \frac{\omega_i}{\Delta x}. \quad (6.47)$$

For the convective term of the pressure in the energy balance, the mass flux q^n is replaced by the velocity u^n in Equations (6.46) and (6.47). To obtain the velocity on the flow mesh nodes, the following interpolation is used, $\forall i \in [1, n_x]$,

$$u_{i-1/2}^n \approx \frac{2q_{i-1/2}^n}{\rho_i^n + \rho_{i-1}^n}. \quad (6.48)$$

Interpolation of the velocity on the pressure mesh and finite volume scheme for the momentum equation

For the mass flux (momentum balance), the integration of convective terms on the flow mesh introduces the velocity on the pressure mesh, denoted u_p . An interpolation is necessary and is given by, $\forall i \in [0, n_x]$,

$$(u_p)_i^n \approx \frac{q_{i-1/2}^n + q_{i+1/2}^n}{2\rho_i^n}. \quad (6.49)$$

A sign function for the velocity on the pressure mesh can be defined by, $\forall i \in [0, n_x]$,

$$\text{sg}_i^u = \text{sg}((u_p)_i^n) = \begin{cases} 0 & \text{if } (u_p)_i^n < 0, \\ 1 & \text{if } (u_p)_i^n \geq 0. \end{cases} \quad (6.50)$$

The finite volume scheme for the convective terms writes, for $f \in \{\delta q, q^n\}$,

$$\begin{aligned} \int_{\omega_{i-1/2}} \partial_x(u^n f) d\Omega \approx & \left[(u_p)_i^n (\text{sg}_i^u f_{i-1/2} + (1 - \text{sg}_i^u) f_{i+1/2}) \right. \\ & \left. - (u_p)_{i-1}^n (\text{sg}_{i-1}^u f_{i-3/2} + (1 - \text{sg}_{i-1}^u) f_{i-1/2}) \right] \frac{\omega_{i-1/2}}{\Delta x}. \end{aligned} \quad (6.51)$$

Energy equation

The finite volume scheme for the energy equation is, $\forall i \in [0, n_x]$,

$$a_{i,i-1}^s \delta \mathfrak{s}_{i-1} + a_{i,i}^s \delta \mathfrak{s}_i + a_{i,i+1}^s \delta \mathfrak{s}_{i+1} = b_i^s, \quad (6.52)$$

with

$$\left\{ \begin{array}{l} a_{i,i-1}^s = -\frac{\Delta t^n}{\rho_i^n \Delta x} q_{i-1/2}^n \text{sg}_{i-1/2}, \\ a_{i,i}^s = 1 + \frac{\Delta t^n}{\rho_i^n \Delta x} \left(q_{i-1/2}^n \text{sg}_{i-1/2} - q_{i+1/2}^n (1 - \text{sg}_{i+1/2}) \right), \\ a_{i,i+1}^s = \frac{\Delta t^n}{\rho_i^n \Delta x} q_{i+1/2}^n (1 - \text{sg}_{i+1/2}), \\ b_i^s = -\frac{\Delta t^n}{\rho_i^n (T_l)_i^n \Delta x} \left(q_{i+1/2}^n (1 - \text{sg}_{i+1/2}) (h_{i+1}^n - h_i^n) + q_{i-1/2}^n \text{sg}_{i-1/2} (h_i^n - h_{i-1}^n) \right) \\ \quad + \frac{\Delta t^n}{\rho_i^n (T_l)_i^n \Delta x} \left(u_{i+1/2}^n (1 - \text{sg}_{i+1/2}) (p_{i+1}^n - p_i^n) + u_{i-1/2}^n \text{sg}_{i-1/2} (p_i^n - p_{i-1}^n) \right) \\ \quad + \frac{\Delta t^n}{\rho_i^n (T_l)_i^n} \phi_i^n. \end{array} \right. \quad (6.53)$$

The matrix $A_{\mathfrak{s},\mathfrak{s}}$ of size $(n_x + 1, n_x + 1)$, built from coefficients $(a_{i,i-1}^s < 0, a_{i,i}^s > 1, a_{i,i+1}^s < 0)$, is an invertible tridiagonal matrix and its inverse matrix have positive coefficients:

$$\forall (i, j) \in [0, n_x]^2, (A_{\mathfrak{s},\mathfrak{s}}^{-1})_{ij} > 0. \quad (6.54)$$

Mass fraction equation

The GAMi Scheme (see **Appendix 4.D.1**) is used for the production term in the mass fraction equation. For the disequilibrium equation, the global scheme is written, $\forall i \in [0, n_x]$,

$$a_{i,i-1}^y \delta y_{i-1} + a_{i,i}^y \delta y_i + a_{i,i+1}^y \delta y_{i+1} = b_i^y, \quad (6.55)$$

with

$$\left\{ \begin{array}{l} a_{i,i-1}^y = -\frac{\Delta t^n}{\rho_i^n \Delta x} q_{i-1/2}^n \text{sg}_{i-1/2}, \\ a_{i,i}^y = 1 + \frac{\Delta t^n}{\tau_i^n} + \frac{\Delta t^n}{\rho_i^n \Delta x} \left(q_{i-1/2}^n \text{sg}_{i-1/2} - q_{i+1/2}^n (1 - \text{sg}_{i+1/2}) \right), \\ a_{i,i+1}^y = \frac{\Delta t^n}{\rho_i^n \Delta x} q_{i+1/2}^n (1 - \text{sg}_{i+1/2}), \\ b_i^y = -\frac{\Delta t^n}{\rho_i^n \Delta x} \left(q_{i+1/2}^n (1 - \text{sg}_{i+1/2}) (y_{i+1}^n - y_i^n) + q_{i-1/2}^n \text{sg}_{i-1/2} (y_i^n - y_{i-1}^n) \right) \\ \quad + \frac{\Delta t^n}{\tau_i^n} (\bar{y}_i^n - y_i^n) \\ \quad + \frac{\Delta t^n}{\rho_i^n} (\Gamma_p)_i^n. \end{array} \right. \quad (6.56)$$

The matrix $A_{y,y}$ of size $(n_x + 1, n_x + 1)$, built from coefficients $(a_{i,i-1}^y < 0, a_{i,i}^y > 1, a_{i,i+1}^y < 0)$ is a invertible tridiagonal matrix and its inverse is a matrix with positive coefficients:

$$\forall (i, j) \in [0, n_x]^2, (A_{y,y}^{-1})_{ij} > 0. \quad (6.57)$$

Coupled system pressure-velocity

The coupled pressure-velocity system is written below with the implicit terms in blue. The pressure equation writes, $\forall i \in [0, n_x]$,

$$\delta p_i + \frac{\Delta t^n}{\alpha_i^n \Delta x} (\delta q_{i+1/2} - \delta q_{i-1/2}) = - \left(\frac{\Delta t^n}{\alpha_i^n \Delta x} (q_{i+1/2}^n - q_{i-1/2}^n) \right) + \frac{\beta_i^n}{\alpha_i^n} \delta \mathfrak{s}_i + \frac{m_i^n}{\alpha_i^n} \delta y_i. \quad (6.58)$$

The global scheme of the momentum equation is, $\forall i \in [1, n_x]$,

$$a_{i,i-1}^q \delta q_{i-3/2} + a_{i,i}^q \delta q_{i-1/2} + a_{i,i+1}^q \delta q_{i+1/2} + \frac{\Delta t^n}{\Delta x} (\delta p_i - \delta p_{i-1}) = b_i^q, \quad (6.59)$$

with

$$\left\{ \begin{array}{l} a_{i,i-1}^q = - \frac{\Delta t^n}{\Delta x} (u_p)_{i-1}^n \text{sg}_{i-1}^u, \\ a_{i,i}^q = \left(1 + \frac{\Delta t^n}{\Delta x} [(u_p)_i^n \text{sg}_i^u - (u_p)_{i-1}^n (1 - \text{sg}_{i-1}^u)] \right), \\ a_{i,i+1}^q = \frac{\Delta t^n}{\Delta x} (u_p)_i^n (1 - \text{sg}_i^u), \\ b_i^q = - \frac{\Delta t^n}{\Delta x} \left[- (u_p)_{i-1}^n \text{sg}_{i-1}^u q_{i-3/2}^n + \left((u_p)_i^n \text{sg}_i^u - (u_p)_{i-1}^n (1 - \text{sg}_{i-1}^u) \right) q_{i-1/2}^n \right. \\ \quad \left. + (u_p)_i^n (1 - \text{sg}_i^u) q_{i+1/2}^n + (p_i^n - p_{i-1}^n) \right]. \end{array} \right. \quad (6.60)$$

6.3.3 Matrix system formulation and variable resolution order

The global finite volume schemes of Equations (6.52), (6.55), (6.58) and (6.59) can be written with a matrix formulation

$$\begin{pmatrix} A_{\mathfrak{s},\mathfrak{s}} & 0 & 0 & 0 \\ 0 & A_{y,y} & 0 & 0 \\ A_{p,\mathfrak{s}} & A_{p,y} & A_{p,p} & A_{p,q} \\ 0 & 0 & A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta \mathcal{S} \\ \delta Y \\ \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_{\mathfrak{s}} \\ B_y \\ B_p \\ B_q \end{pmatrix}, \quad (6.61)$$

where $\delta \mathcal{S}$, δY , δP and δQ are the vector solutions:

$$\left\{ \begin{array}{l} \delta \mathcal{S} = (\delta \mathfrak{s}_i)_{i \in [0, n_x]}, \\ \delta Y = (\delta y_i)_{i \in [0, n_x]}, \\ \delta P = (\delta p_i)_{i \in [0, n_x]}, \\ \delta Q = (\delta q_{i-1/2})_{i \in [1, n_x]}. \end{array} \right. \quad (6.62)$$

The choice of pseudo-entropy and the scheme for the mass fraction allows for obtaining a block matrix system with the desired form, that is, with an equation on the mass fraction and on the pseudo-entropy, independent of the other variables. This makes it possible to separate the resolution of a time step into several independent steps that are easier to solve than the full system. First, the pseudo-entropy equation can be solved, then the one for the mass fraction. Once the increments $\delta \mathcal{S}$ and δY are determined, the coupled pressure-velocity system is then solved. The vector $\delta \mathcal{S}$ is obtained using

$$\delta \mathcal{S} = A_{\mathfrak{s},\mathfrak{s}}^{-1} B_{\mathfrak{s}}, \quad (6.63)$$

and the vector δY using

$$\delta Y = A_{y,y}^{-1} B_y. \quad (6.64)$$

The matrices $A_{s,s}$ and $A_{y,y}$ are tridiagonal in the one-dimensional framework considered here. Their inversion is performed using an LU decomposition. Once the vectors $\delta\mathcal{S}$ and δY are determined, the matrices $A_{p,s}$ and $A_{p,y}$ are moved to the right-hand side of the equation to formulate a coupled pressure-velocity system. The pressure-velocity system remains coupled and is written as

$$\begin{pmatrix} A_{p,p} & A_{p,q} \\ A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_p - A_{p,s}\delta\mathcal{S} - A_{p,y}\delta Y \\ B_q \end{pmatrix}. \quad (6.65)$$

Numerically, a sparse matrix is used to represent this block matrix, using the format `scipy.sparse.csc_matrix`. The vectors δP and δQ are obtained by inverting the sparse matrix, using the Python method `scipy.sparse.linalg.spsolve`.

6.3.4 Steady-state computations

To compute steady state solutions, the ThermoTorch code starts from an initial state and advances in time until the steady state is reached. The initial state is defined by uniform fields based on the boundary conditions. To determine whether the steady state has been achieved, residuals based on increments are computed at each time step for each solved variable. The residuals for variables defined on the pressure mesh (s , y and p) are

$$\text{res}_v^n = \frac{1}{\Delta t^n} \sqrt{\frac{\sum_{i=0}^{n_x} \delta v_i^2}{n_x}}, \text{ with } v \in \{s, y, p\}, \quad (6.66)$$

and the residuals for the mass flux on the flow mesh are

$$\text{res}_q^n = \frac{1}{\Delta t^n} \sqrt{\frac{\sum_{i=1}^{n_x} \delta q_{i-1/2}^2}{n_x}}, \quad (6.67)$$

The user provides a stopping criterion for each variable, denoted as crit_v (for variable v). The computation stops when the residuals from the last three time steps are below the stopping criterion for each variable, which corresponds to

$$\max(\text{res}_v^{n-2}, \text{res}_v^{n-1}, \text{res}_v^n) < \text{crit}_v, \text{ with } v \in \{s, y, p, q\}. \quad (6.68)$$

6.4 Configurations considered with the code

In this section, the various configurations implemented in ThermoTorch are described. These correspond to test cases simulated in **Chapter 7** and **Chapter 8**. For each configuration, the assumptions made and the EoS used are detailed.

6.4.1 Steady-state Heated Channel Configuration

The Heated Channel Configuration (HCC) is shown in Figure 6.2. The objective is to approximate the flow in a sub-channel of a nuclear reactor core. A one-dimensional heated pipe of length $L = 4.16$ m is considered. A flow of liquid water enters the domain. A volumetric heat input heats the liquid water, possibly up to saturation, where the liquid vaporizes. The flow is characterized by several boundary conditions:

- the inlet temperature T_{in} ,
- the inlet mass flux q_{in} ,
- the outlet pressure p_{out} ,
- the volumetric heat input ϕ_0 , uniform and constant for $x \in [L/8, 7L/8]$.

The 3- and 4-equation models can be considered for this configuration where only steady-states are studied. Liquid water is entering the domain so that, when the 4-equation model is used, the additional boundary condition for the mass fraction is always $y = 0$ at the inlet. For the 3-equation model, an analytical solution can be obtained (see **Appendix 3.A**). It is used in **Chapter 7** to verify the scheme of ThermoTorch in steady-state. In **Chapter 8**, the HCC is used to evaluate the performance of an acceleration method using machine learning to converge faster towards the steady-state.

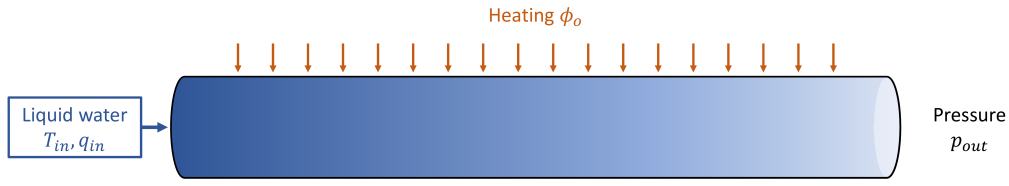


FIGURE 6.2
Scheme of the configuration for the Heated Channel Configuration

Equation of State for the Heated Channel Configuration

For this configuration, a mixture of stiffened gases is considered to obtain the mixture EoS. Each phase (liquid water and vapor) is described by a stiffened gas EoS (see [42, 95, 96]) with both phases assumed to be at the same mixing pressure p :

$$e_k(p, \rho_k) = \frac{p + \gamma_k p_{\infty k}}{\rho_k(\gamma_k - 1)} + h_{0k}, k \in \{g, l\}, \quad (6.69)$$

with the following constants to be set:

- the polytropic index γ_k ,
- the reference enthalpy h_{0k} ,
- the minimum pressure $p_{\infty k}$.

The temperature T_k of phase k is defined by

$$T_k(p, \rho_k) = \frac{p + p_{\infty, k}}{\rho_k c_{vk}(\gamma_k - 1)}, \quad (6.70)$$

with c_{vk} the constant specific volumetric heat capacity of phase k . The entropy of phase k writes

$$s_k(e_k, \rho_k) = s_{0k} + c_{vk} \ln((e_k - h_{0k} - p_{\infty k} \rho_k^{-1}) \rho_k^{1-\gamma_k}), \quad (6.71)$$

with the reference entropy s_{0k} , a constant to be set. The EoS considered for phase k has five parameters to set: $p_{\infty k}$, γ_k , c_{vk} , h_{0k} and s_{0k} . It gives ten settable parameters to reproduce the behavior of liquid and vapor water under conditions of pressure and temperature of a nuclear reactor core. To achieve this, two points (of different temperature) are considered: one at a pressure of 155 bar and one at 150 bar, as follows:

- Point (a): Liquid water at $T_l = 310^\circ\text{C}$, $p_a = 155$ bar,

- Point (b): Vapor at $T_g = 350^\circ\text{C}$, $p_b = 155$ bar,
- Point (c): Saturated mixture at $p_c = 150$ bar.

Point (a) is used to set the parameters of the liquid. Point (b) is used to set three of the gas parameters. The last two gas parameters, the reference entropy and enthalpy, are set in order to match the saturation temperature at pressure of Point (a) and Point (c). The exact equations used to set each parameter, as well as the numerical values obtained for each phase, are described in **Appendix 6.B**. Using the Equation of State for each phase, the mixture EoS is defined as explained in Section 1.3.3, in particular Equation (1.68).

Numerical boundary conditions for heated channel configuration

The boundary conditions are the inlet mass flux q_{in} , the inlet temperature T_{in} , and the outlet pressure p_{out} . The inlet mass flux q_{in} is set by a Dirichlet condition on the inlet cell of the flow mesh:

$$\forall n, \quad q_{i=1/2}^n = q_{in}. \quad (6.72)$$

To ensure the desired temperature at the inlet, a Dirichlet condition is applied to the function $\mathfrak{s}$ on the inlet cell in the pressure mesh domain. For this, the entropy of the liquid phase is evaluated at the pressure calculated at the inlet cell:

$$\forall n, \quad \mathfrak{s}_{i=0}^n = s_l(p = p_{i=0}^n, T_l = T_{in}). \quad (6.73)$$

The pressure boundary condition p_{out} is enforced by a Dirichlet condition on the outlet cell of the pressure mesh.

$$\forall n, \quad p_{i=n_x}^n = p_{out}. \quad (6.74)$$

Finally, a liquid water flow is assumed at the inlet. The inlet mass fraction is therefore set to zero for the 4-equation model

$$\forall n, \quad y_{i=0}^n = 0. \quad (6.75)$$

To determine the number of boundary conditions at the inlet and the outlet, we consider the results of [36, 37]. When considering the 4-equation model, the system admit four real eigenvalues ($u - c, u, u, u + c$) (see Section 2.4.1). Subsonic cases with $u_{in} > 0$ and $u_{out} < 0$ are considered here. At the inlet, only one wave exit the domain, so three (four unknowns minus one wave) boundary conditions are necessary (q_{in}, T_{in}, y_{in} here). At the outlet, three waves exit the domain, only one (four unknowns minus three waves) boundary condition is necessary (p_{out} here). For the 3-equation model with eigenvalues ($u - c, u, u + c$), the results are: two boundary conditions for the inlet (q_{in}, T_{in} here) and one for the outlet (p_{out} here).

6.4.2 Unsteady configuration for Riemann problems

In **Chapter 7**, unsteady analytical solutions of Riemann problems are tested to verify the numerical schemes. These analytical solutions are detailed in **Appendix 3.B** for the 3-equation model and in **Appendix 3.C** for the 4-equation model. In this context, no heating or phase change is taken into account

$$\begin{cases} \phi = 0, \\ \Gamma = 0, \\ \tau \rightarrow +\infty. \end{cases} \quad (6.76)$$

The equation for mass fraction is, in that configuration,

$$\partial_t(\rho y) + \partial_x(\rho u y) = 0. \quad (6.77)$$

A simplified equation of state is used for the mixture. It depends on the model considered.

3-equation model EoS

For the solutions of the 3-equation model, the mixture EoS is a perfect gas EoS:

$$e(p, \rho) = \frac{p}{\rho(\gamma - 1)}, \quad (6.78)$$

with $\gamma = 1.4$.

4-equation model EoS

When considering the 4-equation model for Riemann Problems, the mixture EoS defined in **Appendix 3.C.1** is used. It is an extension of the perfect gas EoS with a reference energy h_0 that depends upon the mass fraction y . It corresponds to the mixture of two perfect gas EoS with the same polytropic index, but different reference enthalpies. For a phase k , the EoS writes:

$$e_k(p, \rho_k) = \frac{p}{\rho_k(\gamma - 1)} + h_{0k}. \quad (6.79)$$

The mixture EoS writes

$$e(p, \rho, y) = \frac{p}{\rho(\gamma - 1)} + h_0(y), \quad (6.80)$$

with

$$h_0(y) = yh_{0g} + (1 - y)h_{0l}. \quad (6.81)$$

6.A Closure laws

In this section, the closure laws used in Thermotorch for the fraction χ used for vaporization, and the relaxation time τ , are detailed. These laws appear in the disequilibrium equation. The hydraulic diameter d_h and the liquid conductivity λ_l are two constant parameters involved in the closure laws. They are set to

$$\begin{cases} d_h = 0.011185 \text{ m}, \\ \lambda_l = 0.5 \text{ W.m}^{-1}.\text{K}^{-1}. \end{cases} \quad (6.82)$$

6.A.1 Fraction used for vaporization

The closure law for the fraction χ used for vaporization is taken from *Saha-Zuber* [124]. This function is used to model subcooled boiling, by triggering the vaporization of liquid even when the saturation temperature is not reached. This phenomenon occurs only when the liquid temperature is close enough to saturation. The function χ must therefore depend on the liquid temperature T_l and should only activate above a temperature called the **subcooled boiling temperature**, denoted T_d . Subcooled boiling can occur when the liquid is not, on average, at saturation, but the wall heat flux is high enough to instantly vaporize the liquid at the wall. The temperature T_d must therefore decrease as the heat input ϕ_0 increases. The temperature T_d is obtained with

$$T_d = T_{sat} - C_1 \phi_0, \quad (6.83)$$

where the constant C_1 is defined according to the value of the Peclet number Pe

$$Pe = \frac{q c_{p,l} d_h}{\lambda_l}. \quad (6.84)$$

If $Pe < 7 \times 10^4$,

$$C_1 = 0.022 \frac{d_h}{\lambda_l}, \quad (6.85)$$

and else

$$C_1 = \frac{154}{|q| c_{p,l}}. \quad (6.86)$$

Once subcooled boiling begins, the fraction χ increases progressively from 0 to 1. It reaches 1 when saturation is achieved, i.e., when $T_l = T_{sat}$. The complete correlation for the fraction χ is

$$\chi(T_l) = \begin{cases} 0 & \text{if } T_l < T_d, \\ \frac{(T_l - T_d)^2 (3T_{sat} - 2T_l - T_d)}{(T_{sat} - T_d)^2} & \text{if } T_l \in [T_d, T_{sat}], \\ 1 & \text{if } T_l \geq T_{sat}. \end{cases} \quad (6.87)$$

6.A.2 Relaxation time

In the disequilibrium equation, subcooled boiling allows the mass fraction y to deviate from the equilibrium mass fraction $\bar{y}$. A return-to-equilibrium term is added to account for certain physical phenomena, such as recondensation. The relaxation time therefore represents the time required to return to thermodynamic equilibrium, which is not instantaneous. It is calculated using a correlation known as the bubble diameter tau. This correlation is based on the characteristic bubble diameter in the flow, $d_b = 10^{-4}$ m. Its formula is deduced from the methodology of [122].

The function used to compute the relaxation time τ depends on whether the liquid is far from or near saturation:

$$\tau = \begin{cases} \frac{\rho(y - \bar{y})(h_g - h_l)}{F(T_l - T_{sat})} & \text{if } |T_{sat} - T_l| > 10^{-3}K, \\ \frac{\rho(1 - y)c_{p,l}}{F} & \text{else.} \end{cases} \quad (6.88)$$

In this correlation, the relaxation time τ depends on a quantity F defined by

$$F = \sqrt{C_2} \frac{6\alpha}{d_b}, \quad (6.89)$$

with the void fraction α defined by

$$\alpha = \frac{y\rho}{\rho_g} \in [0, 1], \quad (6.90)$$

and the quantity C_2 defined by

$$C_2 = \frac{4u_r^{DF} \rho_l \lambda_l c_{p,l}}{\pi d_b}, \quad (6.91)$$

where u_r^{DF} is the relative velocity between phases, obtained with a correlation. In order to use this closure law, a non-zero relative velocity must be considered. Despite the fact that the model assumes zero relative velocity here, the Bestion correlation (see Section 1.4.2) is used to estimate the value of the quantity τ :

$$u_r^{DF} = \frac{0.188}{1 - \alpha} \sqrt{9.81 d_h \left(\frac{\rho_l}{\rho_g} - 1 \right)}. \quad (6.92)$$

6.B Adjustement of thermodynamic parameters from water-steam data under reactor conditions for the EoS of the heated channel configuration

The physical properties of vapor and liquid water are obtained using the Python module `iapws`¹ [139]. This module provides the thermodynamic properties for water and steam. It is developed by the International Association for the Properties of Water and Steam (IAPWS). The version IAPWS97 is used here, corresponding to the industrial formulation. The parameters of the EoS are determined for Points (a), (b) and (c) defined in Section 6.4.1 and used to set the parameters of the stiffened gas EoS for each phase. The parameters for the liquid EoS are determined using Point (a).

- The heat capacity is set to

$$c_{vl} = c_v^{(a)}. \quad (6.93)$$

- Parameters $(\gamma_l, p_{\infty l})$ are chosen to ensure the speed of sound $c_s^{(a)}$ and the density $\rho^{(a)}$ at point (a), which gives

$$\begin{cases} p_{\infty l} = \frac{\rho^{(a)} \left(c_s^{(a)} \right)^2}{\gamma_l} - p^{(a)}, \\ \gamma_l = 1 + \frac{p_{\infty l}}{\rho^{(a)} c_{vl} T^{(a)}}. \end{cases} \quad (6.94)$$

¹<https://iapws.readthedocs.io/en/latest/modules.html>

- The reference enthalpy is set to satisfy

$$h_{0l} = h^{(a)} - \gamma_l c_{vl} T^{(a)}. \quad (6.95)$$

- The reference entropy ensures the entropy at Point (a) such that

$$s_{0l} = s^{(a)} - c_{vl} \ln \left(\frac{p^{(a)} + p_{\infty l}}{\gamma_l - 1} \left(\rho^{(a)} \right)^{-\gamma_l} \right). \quad (6.96)$$

The parameters $p_{\infty g}$, γ_g and c_{vg} for the gas phase are fixed using Point (b) in a similar way as for the liquid parameters:

- The heat capacity is set to

$$c_{vg} = c_v^{(b)}. \quad (6.97)$$

- Parameters $(\gamma_g, p_{\infty g})$ are chosen to impose the speed of sound $c_s^{(b)}$ and the density $\rho^{(b)}$ at point (b), such that

$$\begin{cases} p_{\infty g} = \frac{\rho^{(b)} \left(c_s^{(b)} \right)^2}{\gamma_g} - p^{(b)}, \\ \gamma_g = 1 + \frac{p_{\infty g}}{\rho^{(b)} c_{vg} T^{(b)}}. \end{cases} \quad (6.98)$$

For the parameters h_{0g} and s_{0g} , they are chosen so that the physical saturation temperature is reached for pressure $p^{(a)}$ and for pressure $p^{(c)}$ such that

$$\begin{cases} T_s(p^{(a)}, h_{0g}, s_{0g}) = T_s^{(a)}, \\ T_s(p^{(c)}, h_{0g}, s_{0g}) = T_s^{(c)}. \end{cases} \quad (6.99)$$

The resulting numerical values for the parameters are listed in Table 6.1.

Parameter	Liquid	Vapor	Unit
$p_{\infty k}$	433027888.73886645	2265618.76	Pa
γ_k	1.347721005	1.092548	-
c_{vk}	3030.1475144213396	3273.937158624049	J/kg/K
h_{0k}	-987900.1770384915	377353.1095	J/kg
s_{0k}	-33462.74606510723	-41040.96603	J/kg/K

TABLE 6.1
Stiffened gas EoS parameters for each phase.

6.C Thermodynamic coefficients for the density linearization

The density ρ is linearized as a function of pressure p , the function $\mathfrak{s}$ and mass fraction y :

$$d\rho = \alpha dp + \beta d\mathfrak{s} + m dy. \quad (6.100)$$

In this appendix, the formulas for the coefficients α , β , and m for a general mixture EoS are given. The formulas for the various EoS considered are then provided.

6.C.1 General case

The differential of the density of phase k can be written

$$d\rho_k(p, s_k) = \alpha_k dp + \beta_k ds_k. \quad (6.101)$$

The case in which the gas phase is at saturation is considered. In that case, the entropy of the gas, s_g , is at saturation and depends only on pressure. It is denoted $\bar{s}_g(p)$. The liquid density is a function of pressure, the pseudo-entropy $\mathfrak{s}$ and mass fraction

$$\rho_l(p, \mathfrak{s}, y) = \rho_l \left(p, s_l = \frac{\mathfrak{s} - y s_g(p)}{1 - y} \right). \quad (6.102)$$

Using Equation (6.101) for each phase, the coefficients α , β and m write

$$\begin{aligned} \alpha &= \rho^2 \left(y v_g^2 \alpha_g + (1 - y) v_l^2 \alpha_l + y \frac{ds_g}{dp} (v_g^2 \beta_g - v_l^2 \beta_l) \right), \\ \beta &= \left(\frac{\rho}{\rho_l} \right)^2 \beta_l, \\ m &= \begin{cases} \rho^2 \left(v_l - v_g + \frac{(s_l - s_g)}{\rho_l^2} \beta_l \right) & \text{if } y \in]0, 1[, \\ 0 & \text{else.} \end{cases} \end{aligned} \quad (6.103)$$

If it is the liquid phase is considered to be at saturation, the coefficients can be written symmetrically by inverting the indices g and l . In the following, the coefficients are provided for each EoS considered in ThermoTorch.

6.C.2 Configuration for Riemann problems

Perfect gas EoS

The perfect gas EoS is considered for the Riemann problems with the 3-equation model. When considering the perfect gas EoS for the mixture, the coefficients write

$$\begin{cases} \alpha = \frac{\rho}{\gamma p}, \\ \beta = -\frac{\rho}{\gamma c_v}, \\ m = 0. \end{cases} \quad (6.104)$$

4-equation model EoS

When using the 4-equation model, the mixture EoS given in Equation (6.80) (detailed in Section 3.C.1) is used.

The pseudo-entropy $\mathfrak{s}$ is used because, for a general thermodynamic framework, the entropy of the non-equilibrium system is not known. The resulting coefficients for this EoS are:

$$\begin{cases} \alpha = \frac{\rho}{\gamma p} + \frac{\rho^2 y}{\gamma} \left(\frac{v_l}{c_{vl}} - \frac{v_g}{c_{vg}} \right) \frac{ds_g}{dp} = \frac{\rho}{\gamma p} + \frac{\rho y}{\gamma c_v(y)} \frac{T_l - T_g}{T} \frac{ds_g}{dp}, \\ \beta = -\left(\frac{\rho}{\rho_l} \right)^2 \frac{\rho_l}{\gamma c_{vl}} = -\frac{\rho}{\gamma c_v(y)} \frac{\rho c_v}{\rho_l c_{vl}} = -\frac{\rho}{\gamma c_v(y)} \frac{T_l}{T}, \\ m = \rho^2 \left((v_l - v_g) - \frac{s_l - s_g}{\gamma \rho_l c_{vl}} \right). \end{cases} \quad (6.105)$$

In that case, the entropy of the mixture is known for the EoS considered. If it is used instead of the pseudo-entropy $\mathfrak{s}$, the resulting coefficients are given by:

$$\begin{cases} \alpha^s = \frac{\rho}{\gamma p}, \\ \beta^s = -\frac{\rho}{\gamma c_v(y)}, \\ m^s = \frac{\rho}{\gamma c_v(y)} \left(\frac{ds_0}{dy}(y) + \frac{dc_v}{dy}(y) \frac{s - s_0(y)}{c_v(y)} \right). \end{cases} \quad (6.106)$$

In **Chapter 7**, both methods (with the pseudo-entropy $\mathfrak{s}$ and the entropy s) are explored to study the impact of using the pseudo-entropy $\mathfrak{s}$ on unsteady solutions compared to the entropy s , in particular the use of the approximation in Equation (6.17).

6.C.3 Heated Channel Configuration

For each phase, the density ρ_k can be expressed as a function of the pressure and the entropy s_k of phase k using the stiffened gas law as follows:

$$\rho_k(p, s_k) = \left(\frac{p + p_{\infty,k}}{\gamma_k - 1} \right)^{\frac{1}{\gamma_k}} \exp \left(-\frac{s_k - s_{0k}}{\gamma_k c_{vk}} \right). \quad (6.107)$$

Coefficients α_k and β_k of phase k are given by:

$$\begin{cases} \alpha_k = \frac{\rho_k}{\gamma_k(p + p_{\infty,k})}, \\ \beta_k = -\frac{\rho_k}{\gamma_k c_{vk}}. \end{cases} \quad (6.108)$$

The coefficients from Equations (6.108) are used in Equations (6.103) to obtain α , β and m .

Chapter 7

Verification of finite volume schemes in ThermoTorch

In this chapter, several analytical solutions from **Chapter 3** are used to verify the schemes of the ThermoTorch code presented in **Chapter 6**. The objective is to verify the implementation of the schemes and to evaluate the convergence rate of the schemes in space for steady-state solutions and the convergence rate of the scheme in space and time (for a constant CFL number) for unsteady solutions. First, the steady-state solution of the heated channel configuration (see Section 6.4 for the configuration and **Appendix 3.A** for the solution) is used to verify the steady-state schemes for the 3-equation model of ThermoTorch in Section 7.1. Then, one-dimensional Riemann problem solutions for the 3-equation model (see **Appendix 3.B**) are tested in Section 7.2. These solutions enable to study the behavior of ThermoTorch's schemes for unsteady solutions, either smooth or with shocks. Finally, Section 7.3 presents the behavior of the schemes when the disequilibrium equation is activated, using an analytical solution of a Riemann problem for the 4-equation model (see **Appendix 3.C**). In particular, this enables to evaluate the impact of using the pseudo-entropy $\mathfrak{s}$ in ThermoTorch schemes (described in Section 6.1.2) on the accuracy.

7.1 Verification of ThermoTorch for steady-state solutions

In this section, the analytical solution from **Appendix 3.A** is used as the solution of the 3-equation model for the Heated Channel Configuration presented in Section 6.4.1. Two different cases are studied. The first one is to a single-phase case where the liquid is heated as it flows through the channel without reaching saturation. It is referred to as the liquid case. The second case corresponds to a two-phase case where the applied heat flux is high enough to vaporize part of the liquid; it is referred to as the two-phase case. The boundary conditions for these two cases are shown in Table 7.1. The two test cases are implemented in ThermoTorch to verify the implementation and consistency (for steady states) of the numerical schemes of ThermoTorch. In particular, the spatial convergence rate of the scheme is studied. To do so, the analytical solution from **Appendix 3.A** is used. As a reminder, the solution is obtained by integrating between two points (a) and (b). The solution at point (b) is given from the state at point (a). The solution is therefore computed step by step from the inlet conditions. To do this, a Newton method is used, with an accuracy of 10^{-8} . This analytical solution is compared to the numerical solution obtained with ThermoTorch. For all the calculations performed in this section, the 3-equation model is considered and the following numerical parameters are used. The stopping criteria are set to 10^{-8} for all variables (pressure, mass flux, entropy). A CFL_u condition, based on the mixture velocity, is used for time-stepping such that

$$\text{CFL}_u = 5. \tag{7.1}$$

Parameter	Liquid case	Two-phase case	Unit
q_{in}	3500	1500	$kg.m^{-2}.s^{-1}$
T_{in}	273.15	273.15	$^{\circ}C$
p_{out}	155	155	bar
ϕ_0	10^8	10^8	$W.m^{-3}$

TABLE 7.1
Boundary conditions for the liquid and the two-phase cases.

7.1.1 Solutions of ThermoTorch for two steady-state cases

Both cases are first computed using a mesh composed of $n_x = 50$ cells. It corresponds to the approximate size of the industrial mesh used in THYC-coeur along the axial direction (main direction of the flow in the core). Figures 7.1 and 7.2 show the residuals (defined in Section 6.3.4) as a function of physical time for the liquid case and the two-phase case, respectively. The residuals clearly decrease over time until reaching steady state, achieved when the residuals are below 10^{-8} for each variable (which is lower than simple precision). For the two-phase case, a significant variation in the pressure residuals is observed during the initial iterations (for a physical time less than 1s). This variation seems to correspond to the transition of the flow into the two-phase regime. Indeed, the initial constant initialization corresponds to a single-phase liquid flow, and the transition to two-phase causes a significant pressure variation (see the slope break in the numerical solutions below in Figure 7.5).

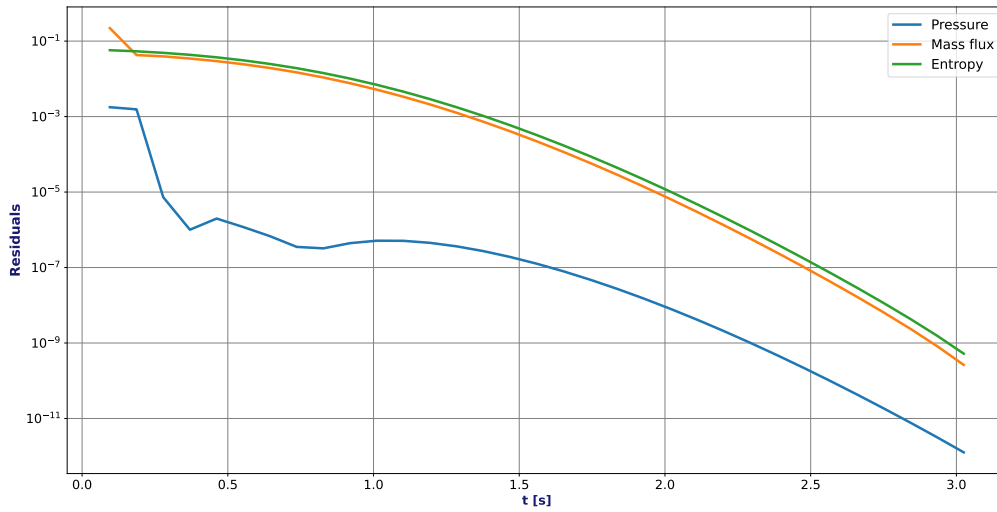


FIGURE 7.1
Residuals for pressure, mass flux and entropy as a function of time for the liquid case with $n_x = 50$.

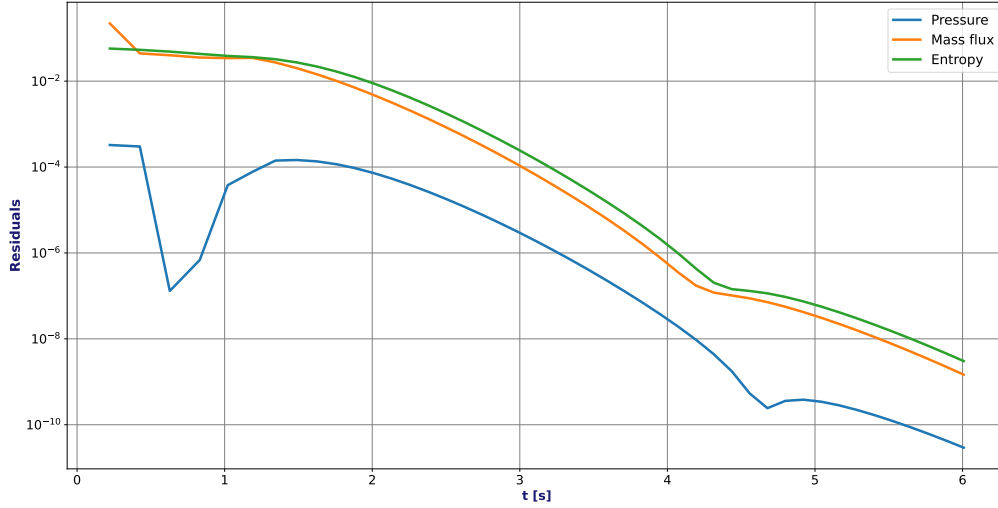


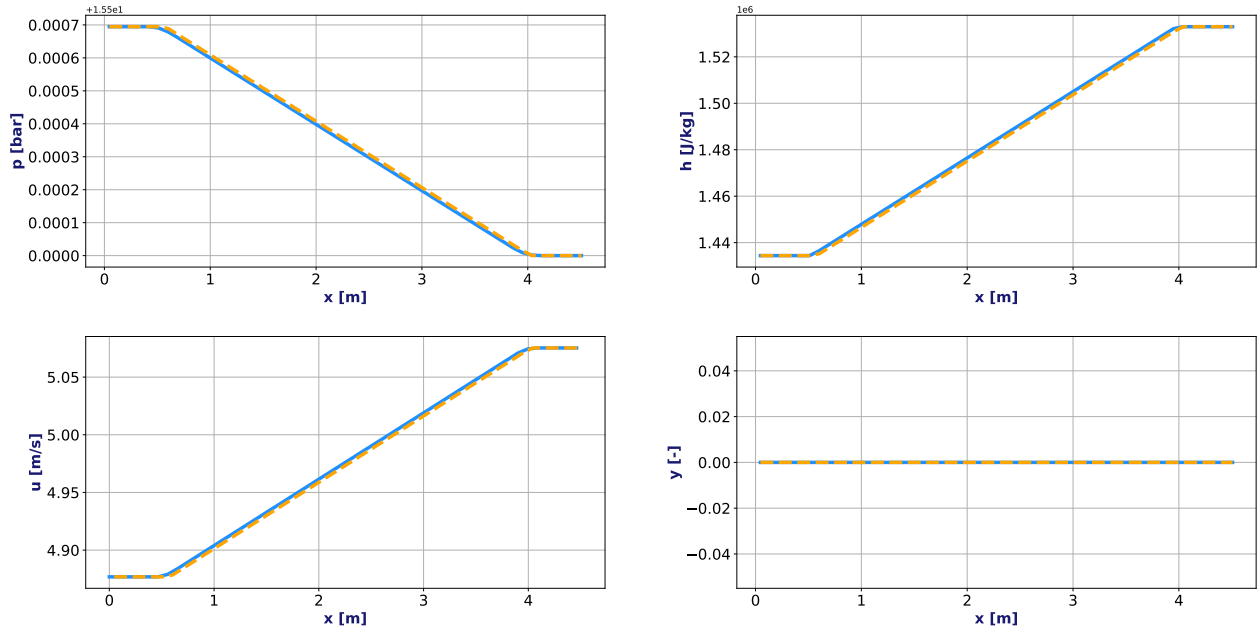
FIGURE 7.2

Residuals for pressure, mass flux and entropy as a function of time for the two-phase case with $n_x = 50$.

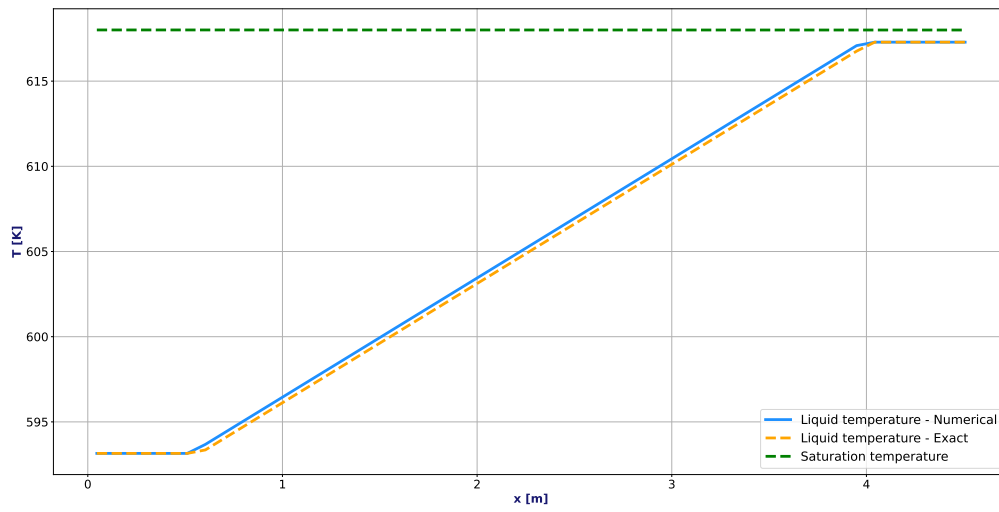
Figure 7.3 (resp. Figure 7.5) shows the analytical solution and the numerical solution from ThermoTorch (with $n_x = 50$ cells) for the liquid (resp. two-phase) case. Pressure, enthalpy, velocity and mass fraction are plotted. The exact and numerical temperatures are shown in Figure 7.4 for the liquid case and in Figure 7.6 for the two-phase case. The saturation temperature is also shown on these figures. The configuration is one-dimensional, so the mass flux is uniform in the steady-state solution. Heating is applied starting from $x = L/8$, which causes a progressive increase in enthalpy from that point until $7L/8$, where the heating is turned off.

In the liquid case, the liquid heats up but does not reach saturation. The flow therefore remains in the liquid phase, and the mass fraction of vapor remains equal to zero. In this situation, the increase in liquid temperature leads to a decrease in density due to thermal dilatation, which causes an increase in velocity. The pressure slightly decreases throughout the domain.

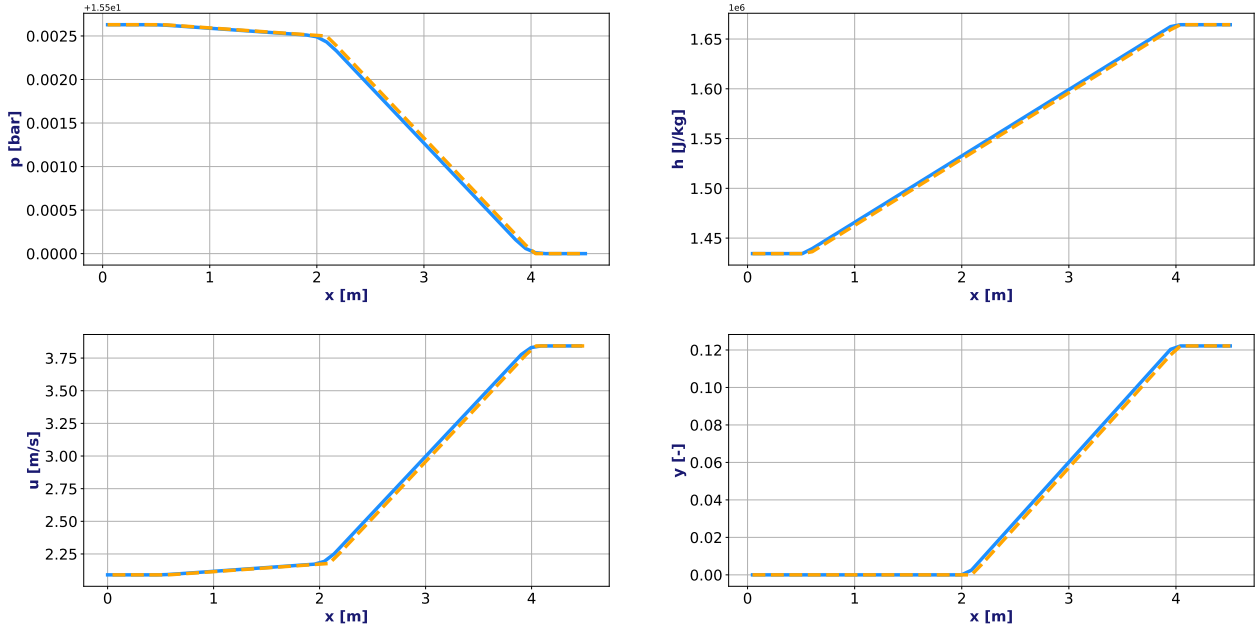
For the two-phase case, the mass flux is lower, and the liquid is more heated before being convected. It eventually reaches saturation. From that point (approximately $x = 2.2$ m), the temperature remains at saturation while the enthalpy continues to increase gradually. From this location, vapor begins to appear, and then the vapor mass fraction increases progressively. The appearance of the vapor phase causes a slope break in the velocity and pressure curves. Vapor is much lighter than liquid water. The effect of liquid density expansion is replaced by the phase change effect, which causes a faster decrease in density, hence the change in slope.

**FIGURE 7.3**

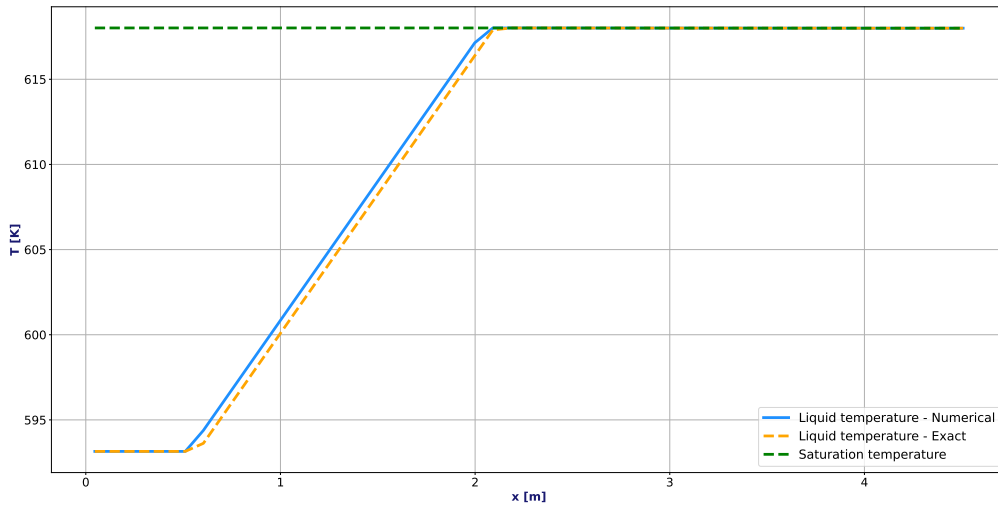
Exact (dotted orange) and numerical (blue solid line, $n_x = 50$) solutions with the 3-equation model for the HCC liquid case.

**FIGURE 7.4**

Exact (dotted orange) and numerical (blue solid line, $n_x = 50$) temperatures with the 3-equation model for the HCC liquid case.

**FIGURE 7.5**

Exact (dotted orange) and numerical (blue solid line, $n_x = 50$) solutions with the 3-equation model for the HCC two-phase case.

**FIGURE 7.6**

Exact (dotted orange) and numerical (blue solid line, $n_x = 50$) temperatures with the 3-equation model for the HCC two-phase case.

Despite a very coarse mesh, the numerical results obtained are already very close to the exact solution. The two-phase case is also simulated with a finer mesh of $n_x = 500$ cells. The resulting numerical solution is shown in Figure 7.7, and the temperature in Figure 7.8. The numerical solution approaches the exact solution more closely. To further investigate the consistency of the schemes, a convergence study is conducted below.

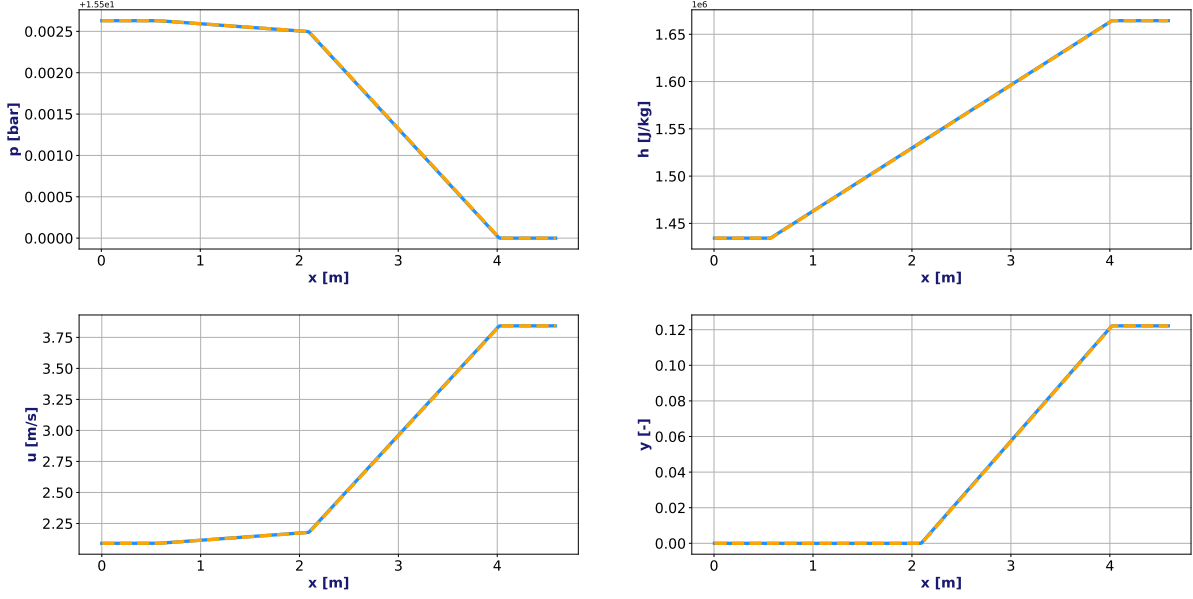


FIGURE 7.7

Exact (dotted orange) and numerical (blue solid line, $n_x = 500$) solutions with the 3-equation model for the HCC two-phase case.

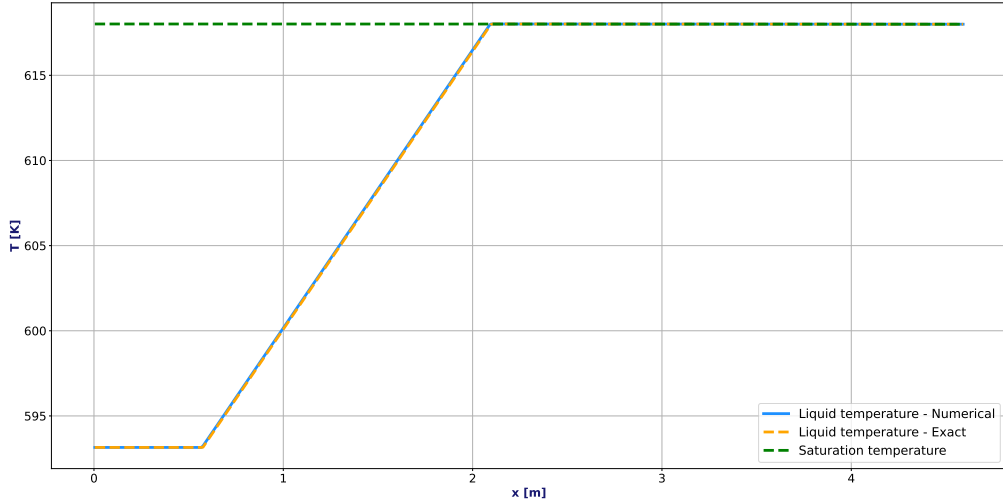


FIGURE 7.8

Exact (dotted orange) and numerical (blue solid line, $n_x = 500$) temperatures with the 3-equation model for the HCC two-phase case.

7.1.2 Consistency and convergence rate of ThermoTorch scheme for steady-state

For each test case, a convergence study is conducted for steady-state solutions. The number of cells varies between 10 and 10^4 cells. An error using the L_2 -norm is computed for each variable between the analytical solution and the numerical solution. The resulting error in pressure, enthalpy, mass flux and velocity is shown as a function of the number of cells in the mesh in Figure 7.9 for the liquid case and in Figure 7.10 for the two-phase case. The convergence rates computed between two mesh sizes are summarized in Table 7.2 for the liquid case and in Table 7.3 for the two-phase case. Regardless of the case, for pressure and enthalpy, a slope close to one is quickly reached. For the mass flux, the error is constant throughout the domain at a value of around 10^{-11} . It means that the solution is already the exact solution up to machine precision. A slope greater than one is

observed on the coarser meshes for velocity. It oscillates depending on the mesh sizes. The average observed slope is around $1.4 - 1.5$, which is higher than first-order accuracy. This high order may be explained by the use of a staggered grid mesh with uniform cells. This allows for second-order spatial accuracy with the centered scheme for the pressure gradient in the momentum equation, which could explain the higher order observed for the velocity.

As expected, a spatial convergence rate close to one is observed for pressure and enthalpy for both test cases. The ThermoTorch schemes are therefore consistent at steady state for regular solutions of the 3-equation model.

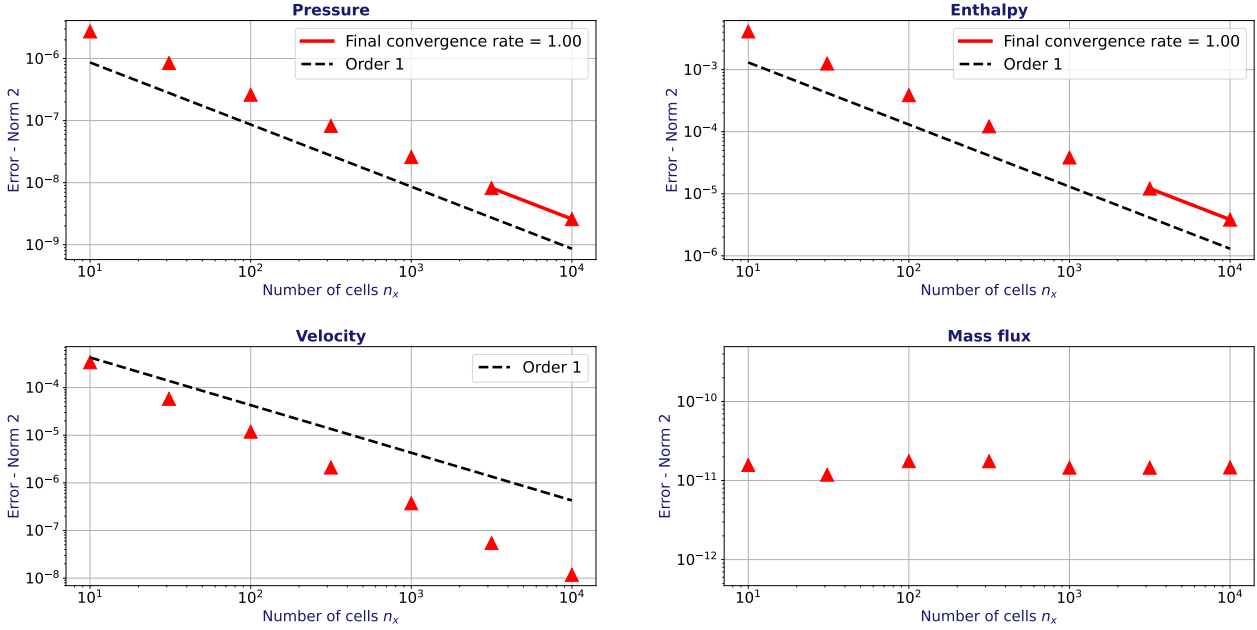


FIGURE 7.9

L_2 -norm of the error as a function of the number of cells for the liquid case.

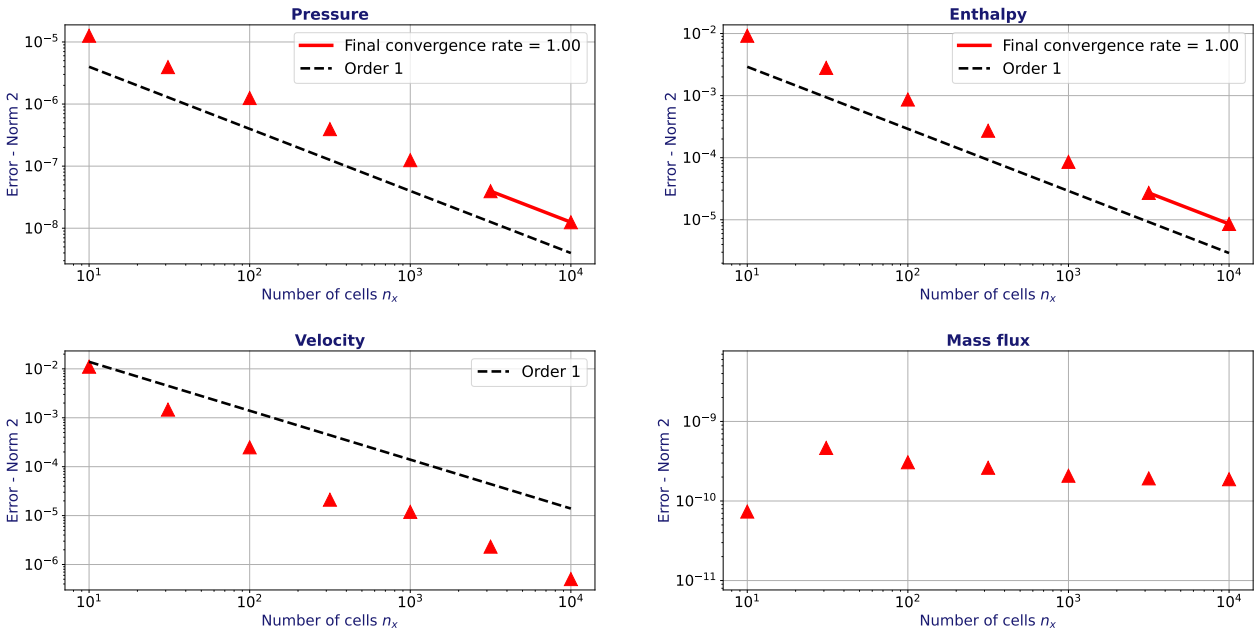


FIGURE 7.10

L_2 -norm of the error as a function of the number of cells for the two-phase case.

Mesh size	p	h	u	q
10				
31	1.04	1.05	1.56	0.25
100	1.00	1.00	1.36	-0.34
316	1.01	1.01	1.51	0.00
1000	1.00	1.00	1.50	0.16
3162	1.00	1.00	1.67	0.00
10000	1.00	1.00	1.33	-0.01

TABLE 7.2

Convergence rate for pressure, enthalpy, velocity and mass flux for the liquid case.

Mesh size	p	h	u	q
10				
31	1.03	1.05	1.78	-1.63
100	0.98	1.00	1.52	0.35
316	1.00	1.01	2.14	0.14
1000	1.00	1.00	0.50	0.20
3162	1.00	1.00	1.42	0.07
10000	1.00	1.00	1.33	0.02

TABLE 7.3

Convergence rate for pressure, enthalpy, velocity and mass flux for the two-phase case.

7.2 Verification of ThermoTorch for unsteady one-dimensional Riemann problems with the 3-equation model

In this section, one-dimensional Riemann problems for the 3-equation model are used to verify the ThermoTorch code for unsteady simulations. The three test cases considered are the solutions developed in **Appendix 3.B**: the symmetric two-rarefaction wave, the symmetric two-shock wave, and the Sod shock tube test case. They are used to evaluate of the behavior of ThermoTorch's numerical schemes for both smooth solutions and solutions with shocks. In all these cases, a perfect gas law is used for the mixture (see Section 6.4.2) with a polytropic index of 1.4. Unless otherwise specified, the simulations are carried out at constant CFL, considering the following value of the CFL_{u+c} associated with the fast waves (described in Section 6.2.3):

$$CFL_{u+c} = 0.5. \quad (7.2)$$

7.2.1 Symmetric double rarefaction wave

The symmetric double rarefaction wave test case is considered here. It corresponds to the one-dimensional Riemann problem described in **Appendix 3.B.1**. The analytical solution obtained is a smooth unsteady solution. Table 7.4 describes the initial conditions for the test case. We note this configuration **SDR** (Symmetric Double Rarefaction wave). Two test cases are considered with two different initial velocity conditions u_0 . The first test case (resp. second test case) is simulated with $u_0 = 2 \text{ m.s}^{-1}$ (resp. $u_0 = 10 \text{ m.s}^{-1}$) and is noted SDR-2 (resp. SDR-10).

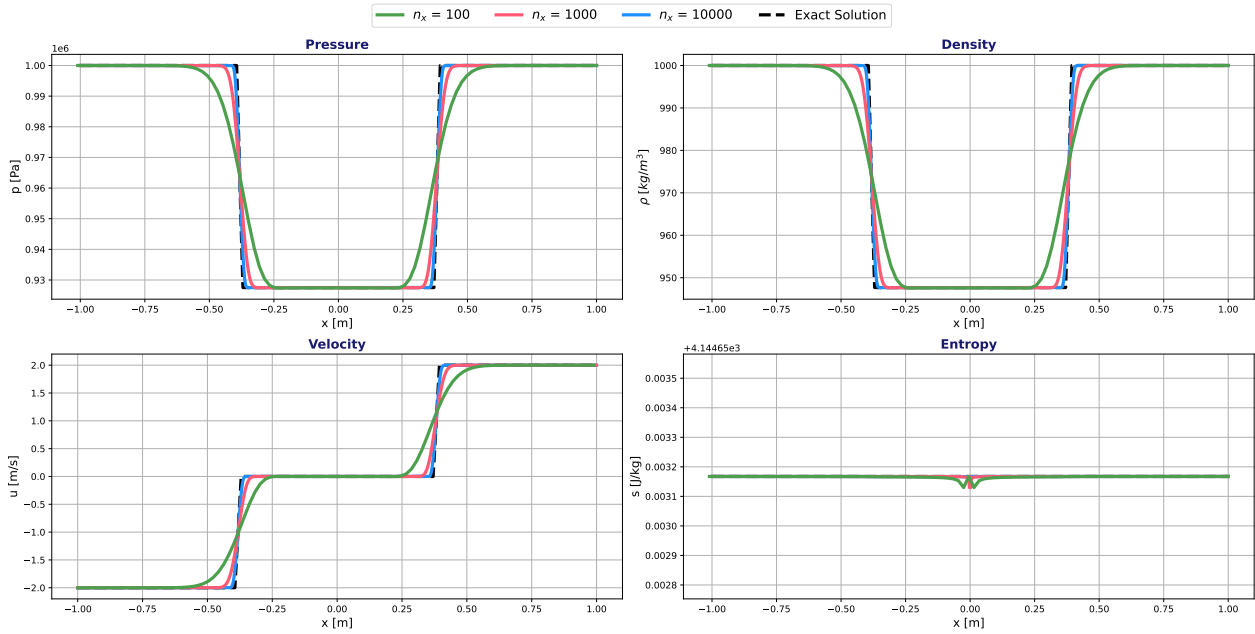
To study the ThermoTorch scheme, the numerical results for SDR-2 and SDR-10 cases are compared with the analytical solutions at a time $t_{max} = 10 \text{ ms}$. For this time, the solution is fully developed and the rarefaction waves are far enough from the boundaries.

Left state	Right state	Unit
$\rho_L = 1000$	$\rho_R = 1000$	$kg.m^{-3}$
$u_L = -u_0$	$u_R = u_0 > 0$	$m.s^{-1}$
$p_L = 10^6$	$p_R = 10^6$	Pa

TABLE 7.4

Initial conditions for symmetric double rarefaction wave test case.

For a number of cells $n_x = \{10^2, 10^3, 10^4\}$, the numerical and analytical solutions are shown at time t_{max} in Figure 7.11 for SDR-2 and in Figure 7.12 for SDR-10. The solutions are smooth solutions. Despite a sharp variation in physical quantities for the SDR-2 case, the slope is not infinite and corresponds to the expansion wave. At first glance, the solutions of each case appear to converge to the analytical solution with mesh refinement. The scheme accurately captures the rarefaction waves, and the numerical plateaus seem to match the analytical solution. Apart from some perturbations at the origin, a constant solution for entropy is observed throughout the entire domain.

**FIGURE 7.11**Symmetric double rarefaction wave (SDR-2) - Analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$.

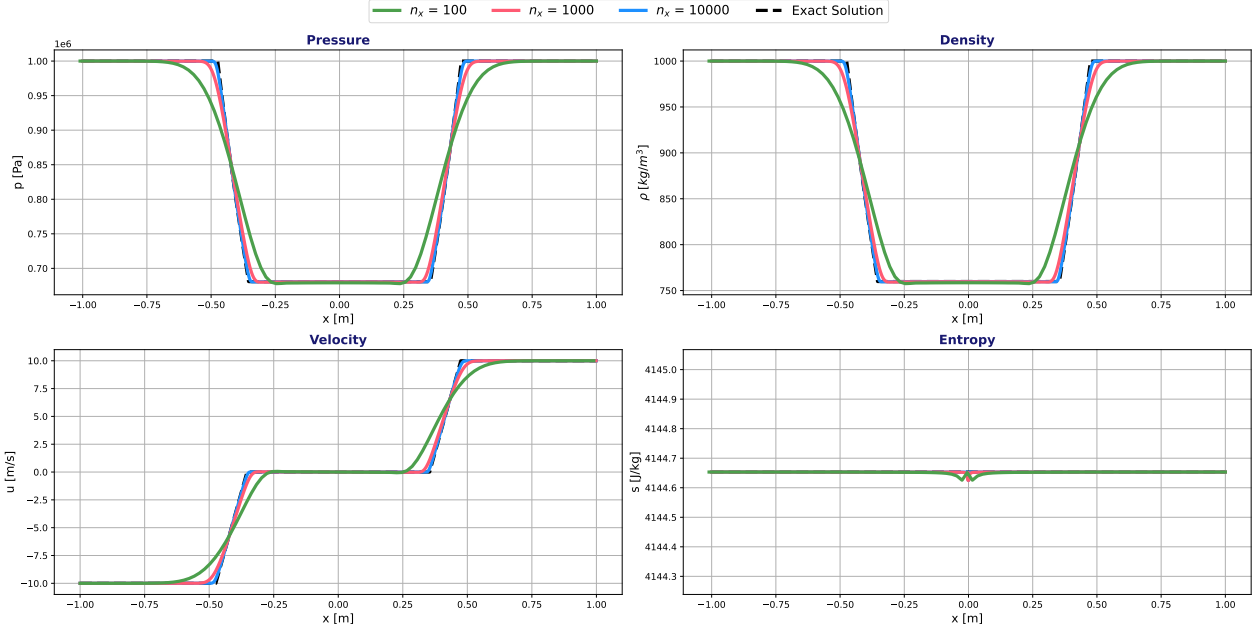
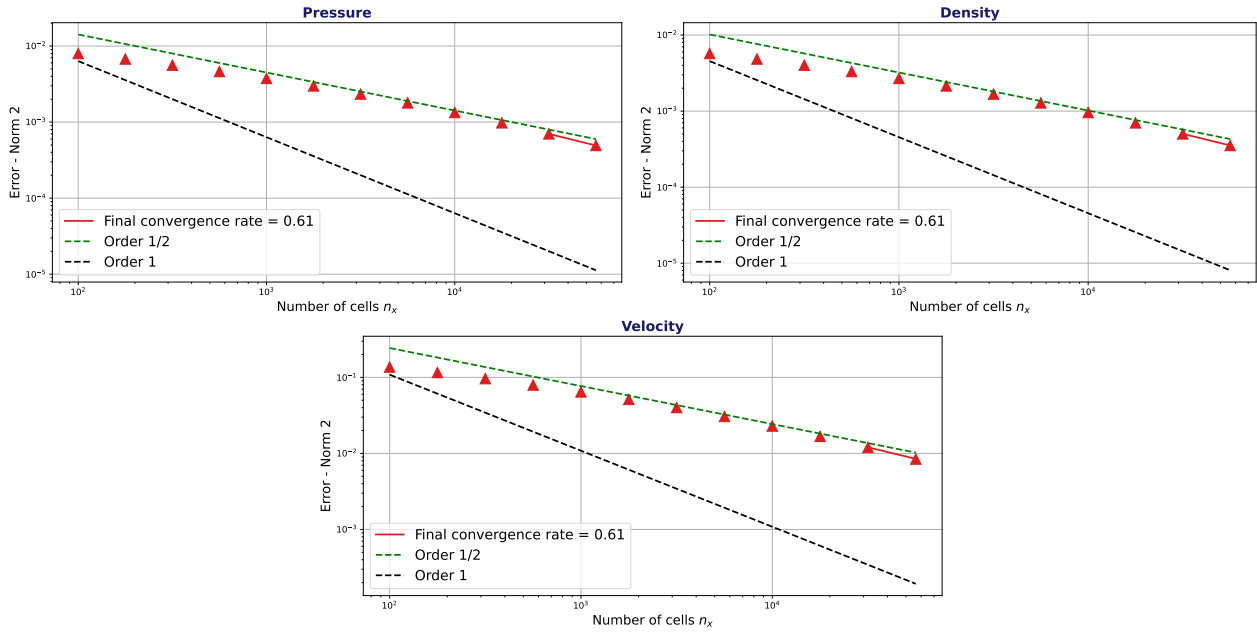


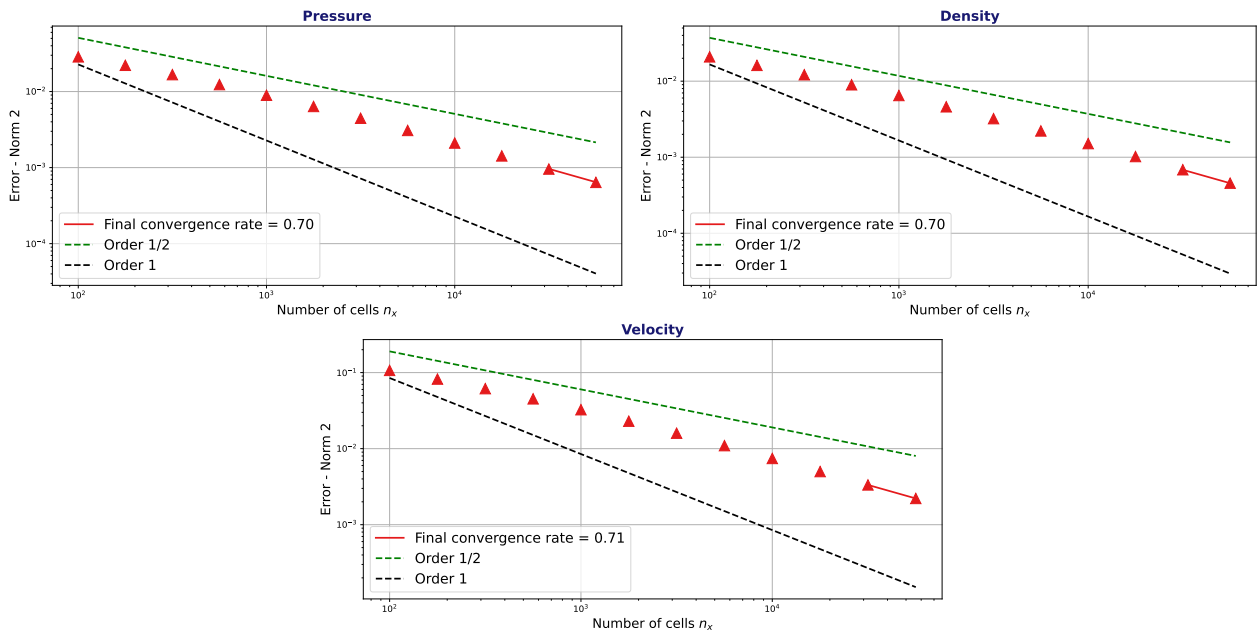
FIGURE 7.12

Symmetric double rarefaction wave (SDR-10) - Analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$.

The errors (in L_2 -norm) between the numerical and the analytical solutions for variables (p, ρ, u) are shown as function of the mesh size for $n_x \in [31, 56234]$ in Figure 7.13 for SDR-2 and in Figure 7.14 for SDR-10. Table 7.5 (resp. Table 7.6) represents the numerical values of the convergence rate for SDR-2 Case (resp. SDR-10 Case). The approximate solution converges towards the exact solution for each case as expected (see for example [52]). The ThermoTorch schemes are consistent for smooth unsteady solutions using the 3-equation model. As observed in Figures 7.13 and 7.14, the convergence rate increases as the number of cells increases, and both test cases show the same behavior. The convergence rate increases with monotonic growth to reach $0.6 - 0.7$ for fine meshes, as expected (see for example [52]). For both test cases, the slope increases by 0.3 between the coarsest and finest mesh. The convergence rate has not yet reached a stabilized value for the meshes considered. The expected convergence rate is 1^- for the rarefaction waves and $1/2$ for the contact wave at the center. Since the case is symmetric, it is possible that the $1/2$ -order error is compensated and vanishes, which explains the higher order obtained.

**FIGURE 7.13**

Symmetric double rarefaction wave (SDR-2) - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ) with a velocity of 2 m.s^{-1} .

**FIGURE 7.14**

Symmetric double rarefaction wave (SDR-10) - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ) with a velocity of 10 m.s^{-1} .

Number of cells	Pressure	Velocity	Density
100			
177	0.29	0.29	0.29
316	0.32	0.32	0.32
562	0.33	0.34	0.34
1000	0.36	0.36	0.36
1778	0.39	0.39	0.39
3162	0.43	0.43	0.43
5623	0.47	0.47	0.47
10000	0.51	0.51	0.51
17782	0.54	0.54	0.54
31622	0.58	0.58	0.58
56234	0.61	0.61	0.61

TABLE 7.5

Symmetric double rarefaction wave (SDR-2) - Convergence rate for pressure, velocity and density as a function of the number of cells with a velocity of 2 m.s^{-1} .

Number of cells	Pressure	Velocity	Density
100			
177	0.45	0.46	0.45
316	0.49	0.50	0.49
562	0.52	0.54	0.53
1000	0.56	0.57	0.56
1778	0.59	0.60	0.59
3162	0.62	0.63	0.62
5623	0.64	0.66	0.65
10000	0.66	0.67	0.67
17782	0.68	0.69	0.68
31622	0.69	0.70	0.69
56234	0.70	0.71	0.70

TABLE 7.6

Symmetric double rarefaction wave (SDR-10) - Convergence rate for pressure, velocity and density as a function of the number of cells with a velocity of 10 m.s^{-1} .

7.2.2 Symmetric double shock wave

The symmetric double shock wave test case from **Appendix 3.B.2** is considered. The analytical solution is a solution with two shock waves propagating toward the exit, symmetrically in the domain. Table 7.7 describes the initial conditions. We note this configuration **SDS** (Symmetric Double Shock wave). Three different velocity conditions are considered $u_0 \in \{2, 10, 20\} \text{ m.s}^{-1}$. The three test cases are respectively noted SDS-2, SDS-10 and SDS-20.

Left state	Right state	Unit
$\rho_L = 1000$	$\rho_R = 1000$	kg.m^{-3}
$u_L = u_0 > 0$	$u_R = -u_0$	m.s^{-1}
$p_L = 10^6$	$p_R = 10^6$	Pa

TABLE 7.7

Initial conditions for symmetric double shock wave test case.

The numerical results for SDS-2, SDS-10 and SDS-20 Cases with the ThermoTorch scheme are compared with the analytical solutions at a time $t_{max} = 10 \text{ ms}$, such that the shock waves are fully developed but far enough from the boundaries. For $n_x = \{10^2, 10^3, 10^4\}$, the numerical and analytical solutions are shown at time t_{max} in Figure 7.15 for SDS-2, in Figure 7.16 for SDS-10 and in Figure 7.18 for SDS-20. While one might think that the ThermoTorch schemes converge to the correct solution for the SDS-2 case, a consistency error seems to emerge when the kinetic energy $\rho u^2/2$ increases for SDS-10 and SDS-20 cases. Indeed, the plateau values of pressure and density obtained numerically do not seem to match the analytical values, even for the refined case ($n_x = 10^4$ cells). This is visible in Figure 7.17, where a zoom on the density plateau is shown for SDS-10. Moreover, the shock speed estimated numerically does not appear to match that of the analytical solution. This consistency error, which is expected for the scalar case [76] and for the system case of Euler equations [51], is confirmed by a convergence study presented below.

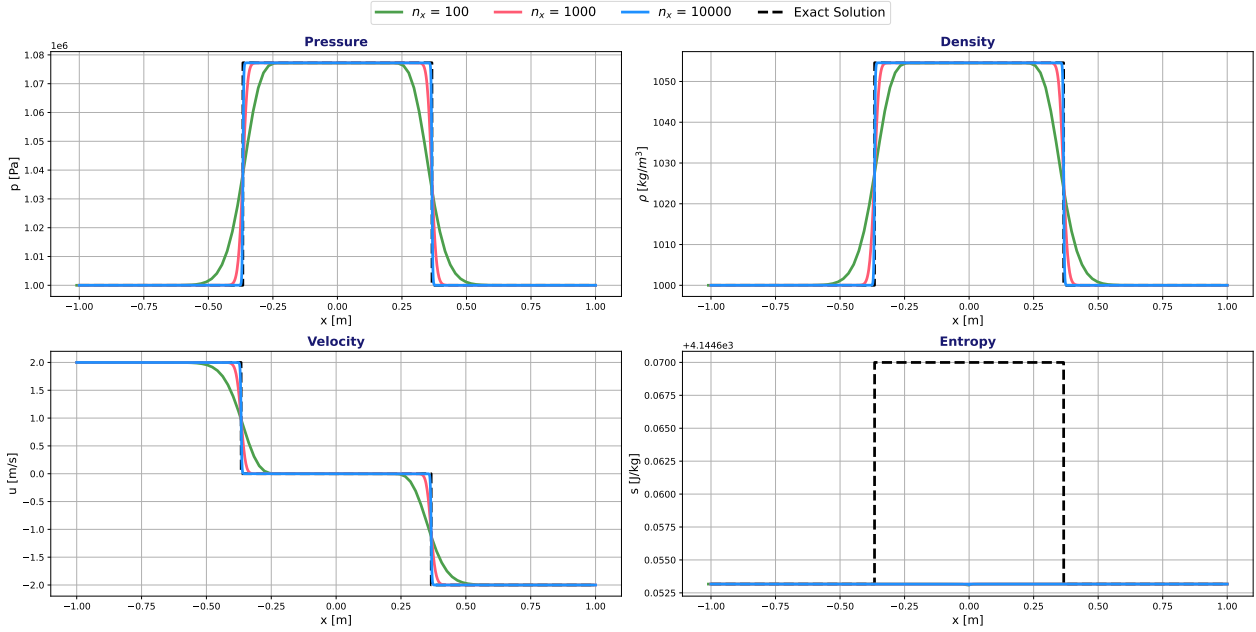


FIGURE 7.15

Symmetric double shock wave (SDS-2) - Analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$.

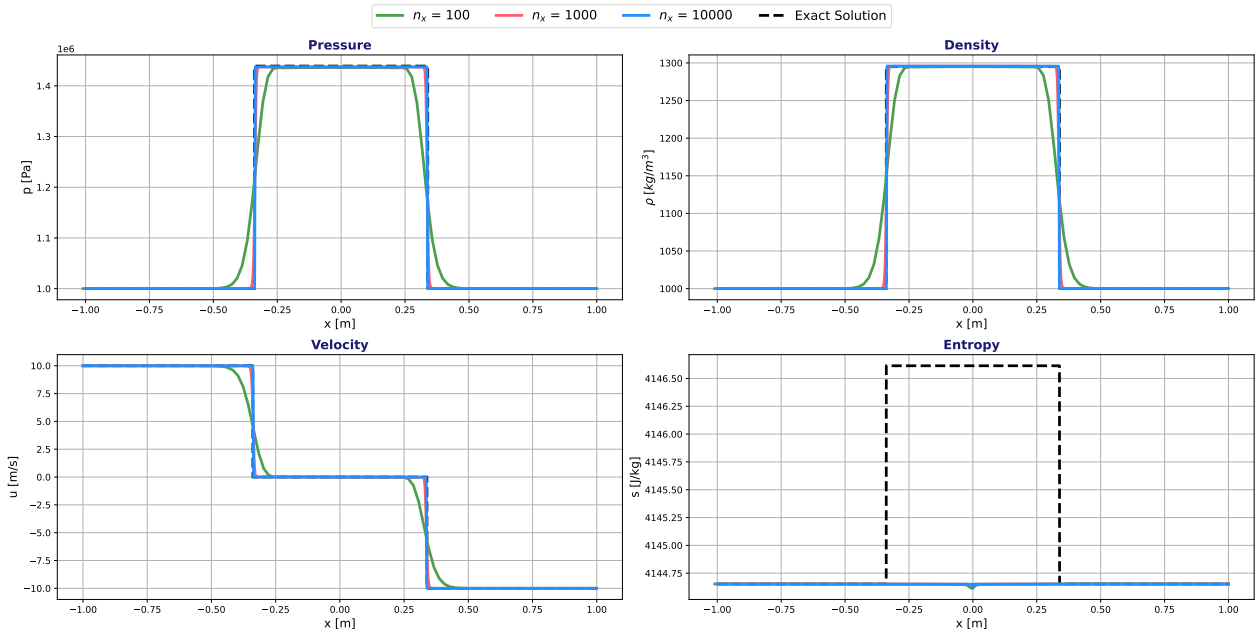


FIGURE 7.16

Symmetric double shock wave (SDS-10) - Analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$.

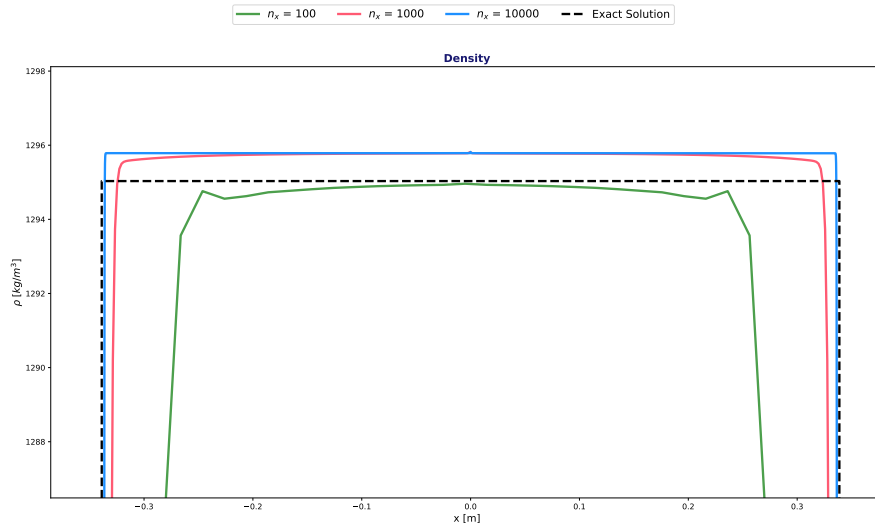


FIGURE 7.17

Symmetric double shock wave (SDS-10) - Zoom on the analytical and numerical solutions for density with $n_x \in \{10^2, 10^3, 10^4\}$.

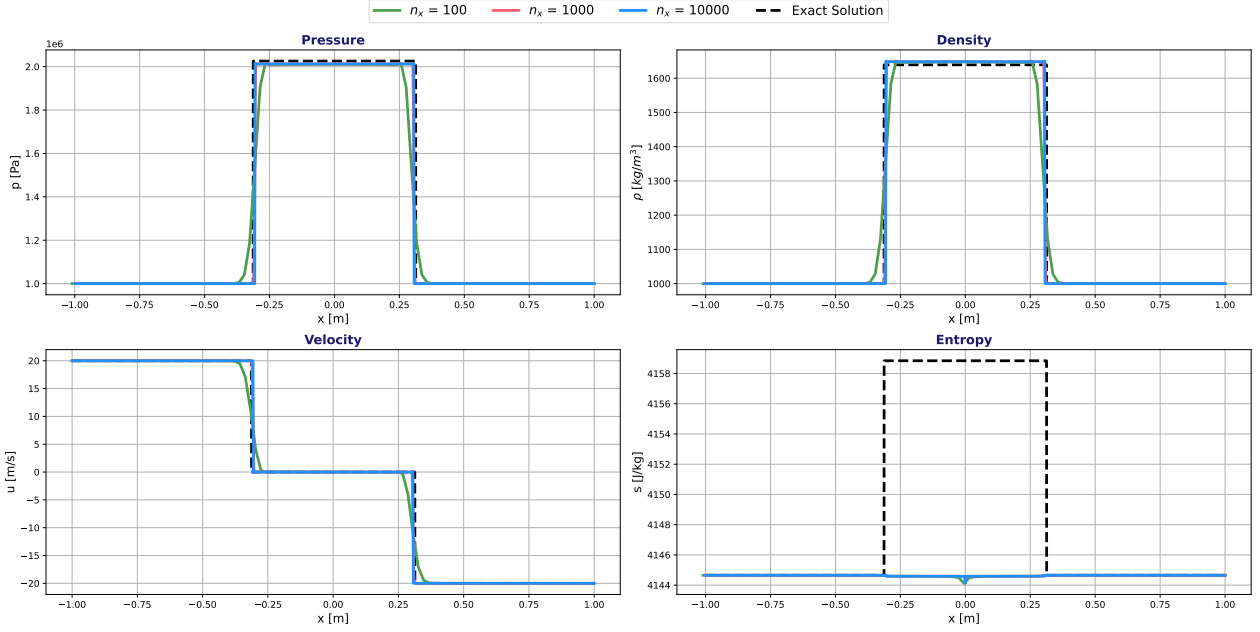
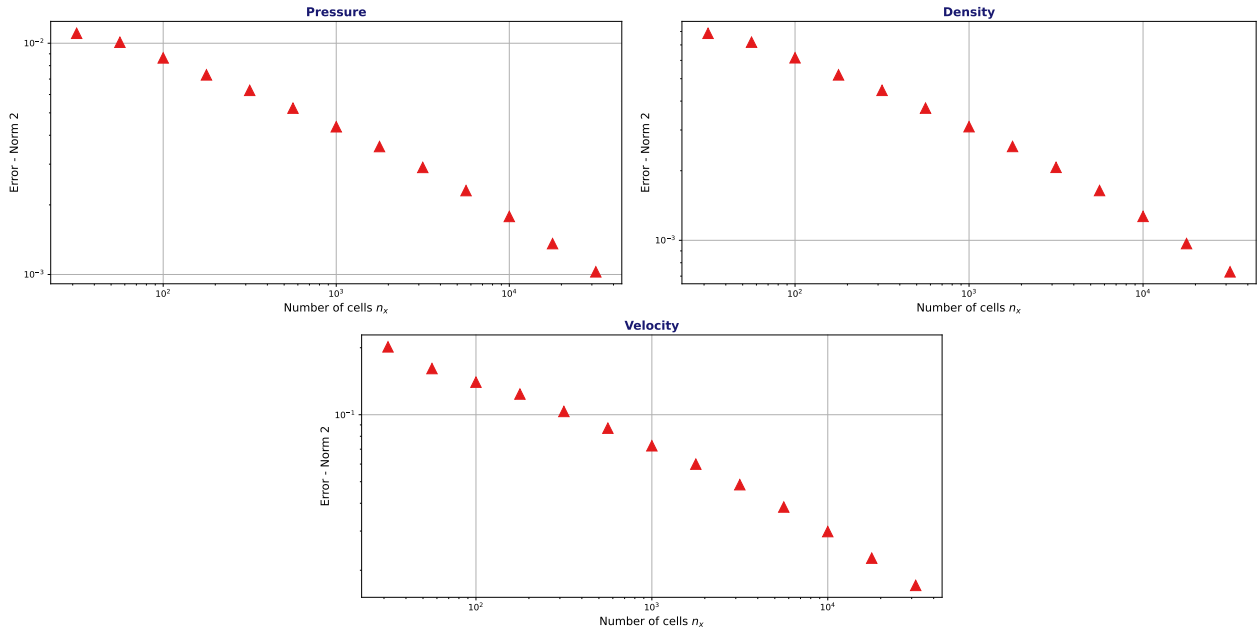


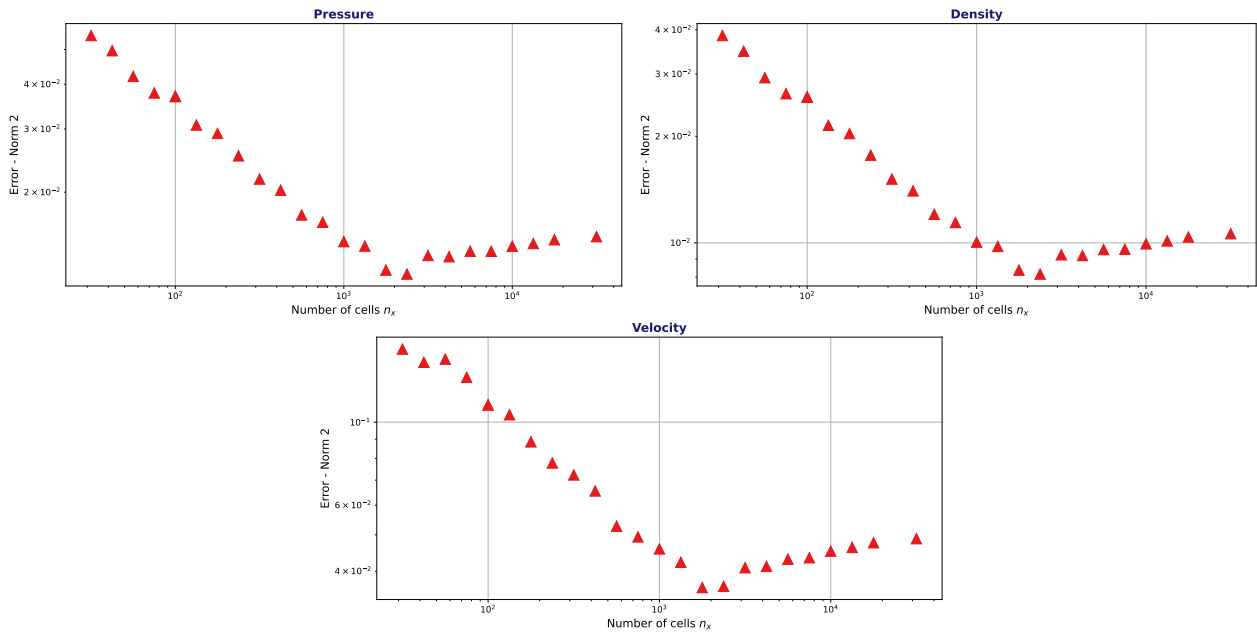
FIGURE 7.18

Symmetric double shock wave (SDS-20) - Analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$.

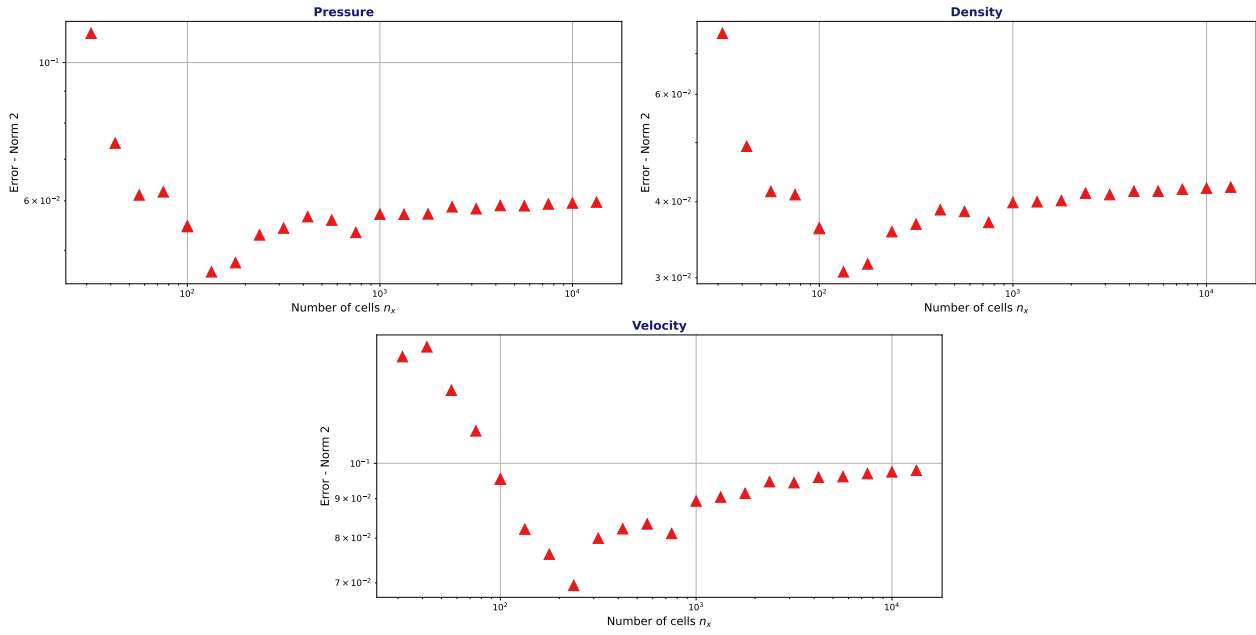
The L_2 -norm of the errors between the numerical and the analytical solutions for variables (p, ρ, u) are shown as a function of the number of cells for $n_x \in [31, 31622]$ in Figure 7.19 for SDS-2, in Figure 7.20 for SDS-10 and in Figure 7.21 for SDS-20. Table 7.8 (resp. Table 7.9) represents for SDR-2 Case (resp. SDR-10 Case) the numerical values for the convergence rates computed. The convergence rate for pressure, density and velocity increases as the number of cells increases for SDS-2 Case and SDS-10 Case (only for low number of cells in the latter case). For SDS-10, when $n_x > 1778$, the errors reach a plateau showing that the scheme converges to another solution as shown in Table 7.9. For SDS-2 Case, the mesh sizes considered are not fine enough to reach the plateau. The SDS-20 Case displays a similar behavior to SDS-10 case but the plateau is reached for a lower number of cells ($n_x \approx 200$). The errors observed on the plateau are in the order of $3\text{--}6 \times 10^{-2}$. This clearly corresponds to a consistency error rather than machine precision, which would be significantly lower. The numerical schemes do not converge to the analytical solution but rather to a solution that exhibits different value plateaus and shock propagation speeds. This consistency error is expected (see for instance [76] for the scalar case, and [51] for the Euler case) when the discretized equations are derived from non-conservative continuous equations, instead of the straightforward finite volume formulation associated with the true *conservation* laws. In the following, the objective is to evaluate the magnitude of the error and compare it to other discretization-induced errors, such as those introduced by using an industrial spatial mesh.


FIGURE 7.19

Symmetric double shock wave (SDS-2) - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ) with a velocity of 2 m.s^{-1} .


FIGURE 7.20

Symmetric double shock wave (SDS-10) - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ) with a velocity of 10 m.s^{-1} .

**FIGURE 7.21**

Symmetric double shock wave (SDS-20) - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ) with a velocity of 20 m.s^{-1} .

Number of cells	Pressure	Velocity	Density
31			
56	0.15	0.38	0.15
100	0.27	0.24	0.27
177	0.30	0.22	0.30
316	0.27	0.31	0.27
562	0.31	0.30	0.31
1000	0.32	0.32	0.32
1778	0.34	0.33	0.34
3162	0.36	0.37	0.36
5623	0.40	0.40	0.40
10000	0.45	0.44	0.45
17782	0.47	0.48	0.47
31622	0.49	0.49	0.49

TABLE 7.8

Symmetric double shock wave (SDS-2) - Convergence rate for pressure, velocity and density as a function of the number of cells with a velocity of 2 m.s^{-1} .

Number of cells	Pressure	Velocity	Density
31			
42	0.32	0.27	0.34
56	0.57	-0.07	0.60
74	0.38	0.40	0.37
100	0.08	0.56	0.08
133	0.65	0.21	0.65
177	0.19	0.58	0.19
237	0.49	0.45	0.48
316	0.52	0.26	0.54
421	0.25	0.34	0.27
562	0.55	0.75	0.53
749	0.16	0.23	0.19
1000	0.43	0.26	0.44
1333	0.10	0.29	0.10
1778	0.53	0.54	0.54
2371	0.09	-0.03	0.09
3162	-0.41	-0.40	-0.44
4216	0.03	-0.03	0.02
5623	-0.12	-0.15	-0.13
7498	0.00	-0.03	-0.01
10000	-0.11	-0.14	-0.12
13335	-0.06	-0.08	-0.07
17782	-0.08	-0.10	-0.09
31622	-0.03	-0.04	-0.04

TABLE 7.9

Symmetric double shock wave (SDS-10) - Convergence rate for pressure, velocity and density as a function of the number of cells with a velocity of 10 m.s^{-1} .

7.2.3 Numerical evaluation of the consistency error for solutions with shocks

In order to determine the magnitude of the consistency error made on the pressure and density plateaus, the test cases are studied with a fine mesh of size $n_x = 10^4$. The relative error compared to the analytical values is calculated on the central plateaus for pressure and density, considering velocities in $\{2, 10, 20\} \text{ m.s}^{-1}$. The formula used for the relative error is

$$\text{err}_r(f) = \frac{|f_{num}^{n_x=10^4} - f_{exact}|}{f_{exact}} \times 100 \quad [\%], \quad (7.3)$$

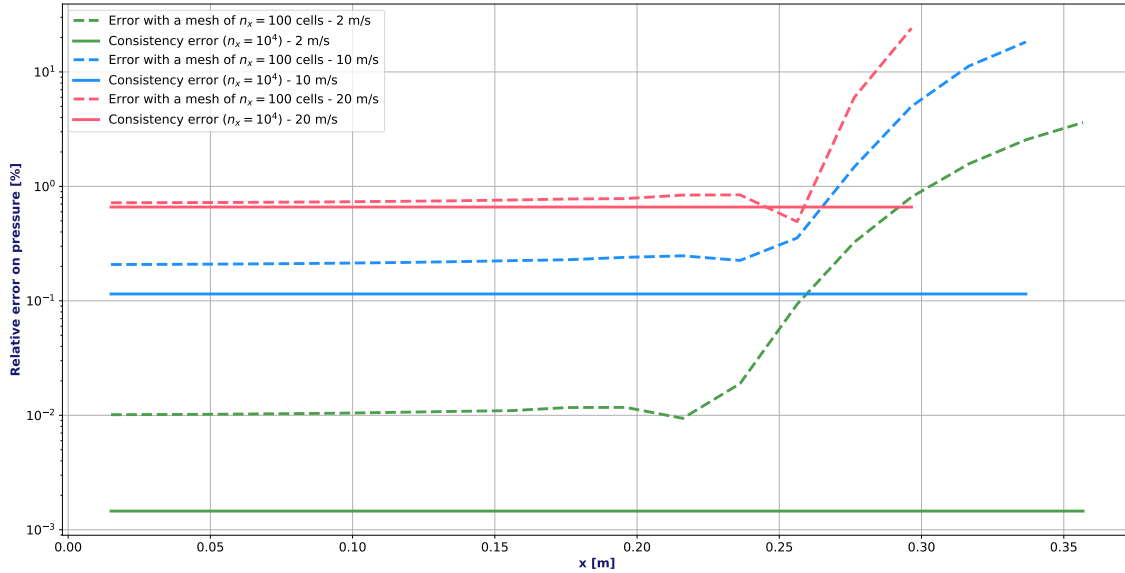
with $f \in \{p, \rho\}$. Table 7.10 represents the error obtained for each test case (for the density and the pressure). The relative error increases as the velocity of the test case increases. For a velocity of $u_0 = 2 \text{ m.s}^{-1}$, the error is well below 0.01%. It increases to around 0.1% for the case $u_0 = 10 \text{ m.s}^{-1}$. For a high velocity of 20 m.s^{-1} , the error is in the order of 0.5%. The errors made on pressure and density are therefore quite small.

Test case	u_0 [$m.s^{-1}$]	Consistency error for pressure [%]	Consistency error for density [%]
SDS-2	2	1.5×10^{-3}	2×10^{-4}
SDS-10	10	0.11	0.06
SDS-20	20	0.66	0.55

TABLE 7.10

Relative consistency error between the numerical and analytical values obtained on the central plateau for pressure and density with a mesh of 10^4 cells.

The consistency error can be compared to the error caused by the use of industrial meshes. For each velocity considered (2, 10 and 20 $m.s^{-1}$), the consistency error and the numerical error due to the use of an industrial mesh are plotted as a function of the position on the central plateau (at time $t_{max} = 10$ ms) in Figure 7.22 for pressure and in Figure 7.23 for density. The numerical error due to an industrial mesh corresponds to the error between the analytical solution and the numerical solution with a mesh of 100 cells, corresponding to an industrial mesh. Relative errors are shown in these two figures. Since the solution is symmetric, only positive abscissas are shown. With an industrial mesh, the shock is poorly resolved. The error on the plateau tends to increase as it approaches the shock, particularly for $x > 0.25$ m. Regardless of the velocity, the error on pressure and density reaches around 10%, a value, which is much higher than the consistency error. The consistency error is all the more small compared to the numerical error due to the industrial mesh when the velocity is low. If we consider the configurations encountered in nuclear reactor cores, i.e. the application of THYC-coeur, the maximum velocities encountered are in the order of 4 to 5 $m.s^{-1}$. For these velocities and with an industrial mesh, the consistency error due to the numerical schemes is therefore negligible.

**FIGURE 7.22**

Consistency error and numerical error due to industrial mesh as a function of the position x on the plateau of pressure for velocities in $\{2, 10, 20\}$ $m.s^{-1}$.

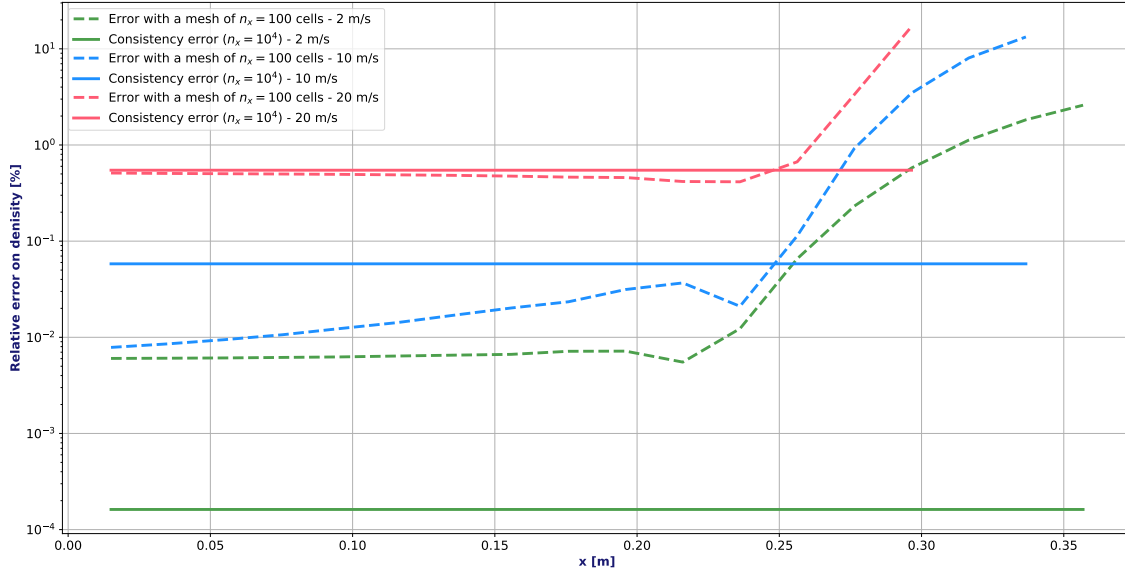


FIGURE 7.23

Consistency error and numerical error due to an industrial mesh as a function of the position x on the plateau of density for velocities in $\{2, 10, 20\} \text{ m.s}^{-1}$.

7.2.4 Impact of the CFL value on the shock profile

ThermoTorch scheme is implicit so that the CFL_{u+c} number can be increased to values greater than 1. Figure 7.24 shows the numerical solutions of SDS-10 Test Case for $\text{CFL}_{u+c} \in \{0.1, 0.5, 2.5\}$ with a number of cells $n_x = 200$. When a low CFL is used ($\text{CFL}_{u+c} = 0.1$ here), significant though stable oscillations appear near the shocks. These oscillations disappear when the CFL is increased. However, the higher the CFL, the less accurately the shock is resolved (the slope is less steep), with an increasingly significant diffusion. The use of implicit schemes therefore allows for an increase in CFL, but this is not recommended if shocks are present in the case considered. Optimal values are of order $\text{CFL}_{u+c} \sim 1$.

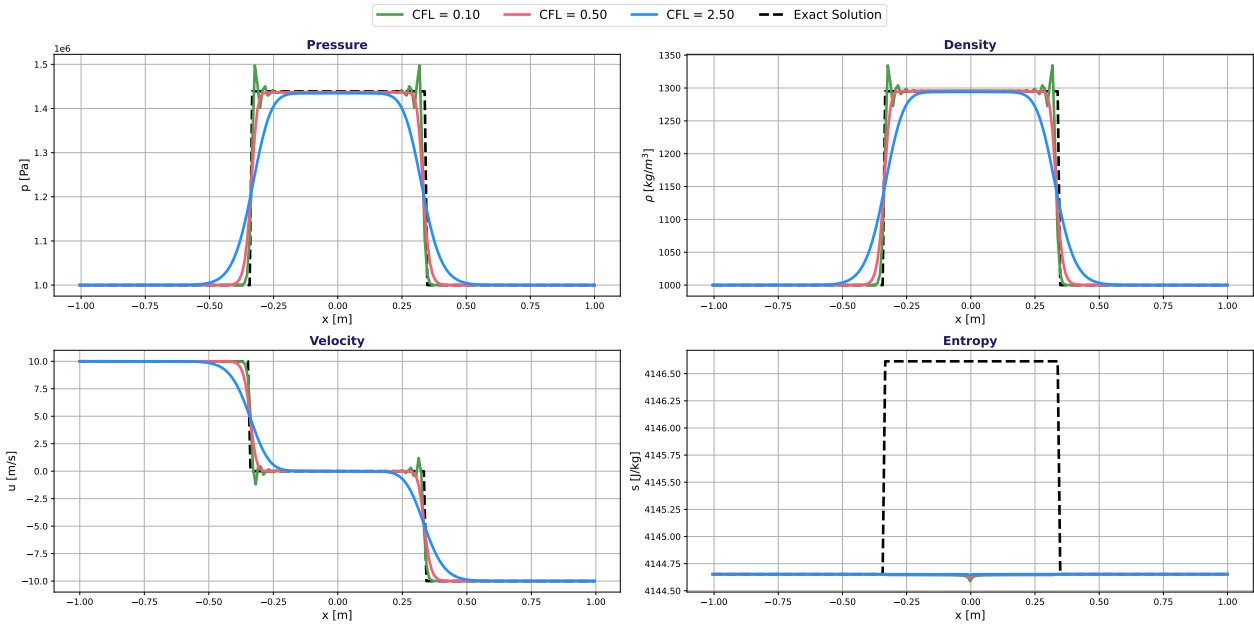


FIGURE 7.24

Symmetric double shock wave (SDS-10) - Numerical solution with $n_x = 200$ for three different $\text{CFL} \in \{0.1, 0.5, 2.5\}$ - Velocity 10 m.s^{-1} .

7.2.5 Sod shock tube

The Sod Shock Tube is a classical test case from [131]. Unlike the two previous tests, the initial conditions, summarized in Table 7.11, are not symmetrical.

Left state	Right state
$\rho_L = 1$	$\rho_R = 0.125$
$u_L = 0$	$u_R = 0$
$p_L = 10^5$	$p_R = 10^4$

TABLE 7.11
Initial conditions for Sod shock tube test case.

The analytical solution is compared with the results of numerical simulations with the scheme of ThermoTorch at a time $t_{max} = 0.3 \text{ ms}$. The numerical results are computed for number of cells $n_x \in \{10^2, 10^3, 10^4\}$. The results are shown in Figure 7.25 at time t_{max} . A zoom on the 2-contact and 3-shock wave for density is shown in Figure 7.26. The solution seems to be well predicted for the 1-rarefaction wave but the constant states (ρ^*, p^*, u^*) and (ρ^{**}, p^*, u^*) converge to another solution when the number of cells increases (see the zoom in Figure 7.26), as expected (see for instance [51, 76]). It can also be noted in Figure 7.26 that the speed of the 3-shock wave for the numerical solution is not the same as the analytical one. The same behavior than for the symmetric double shock wave is observable for consistency errors due to the presence of a shock.

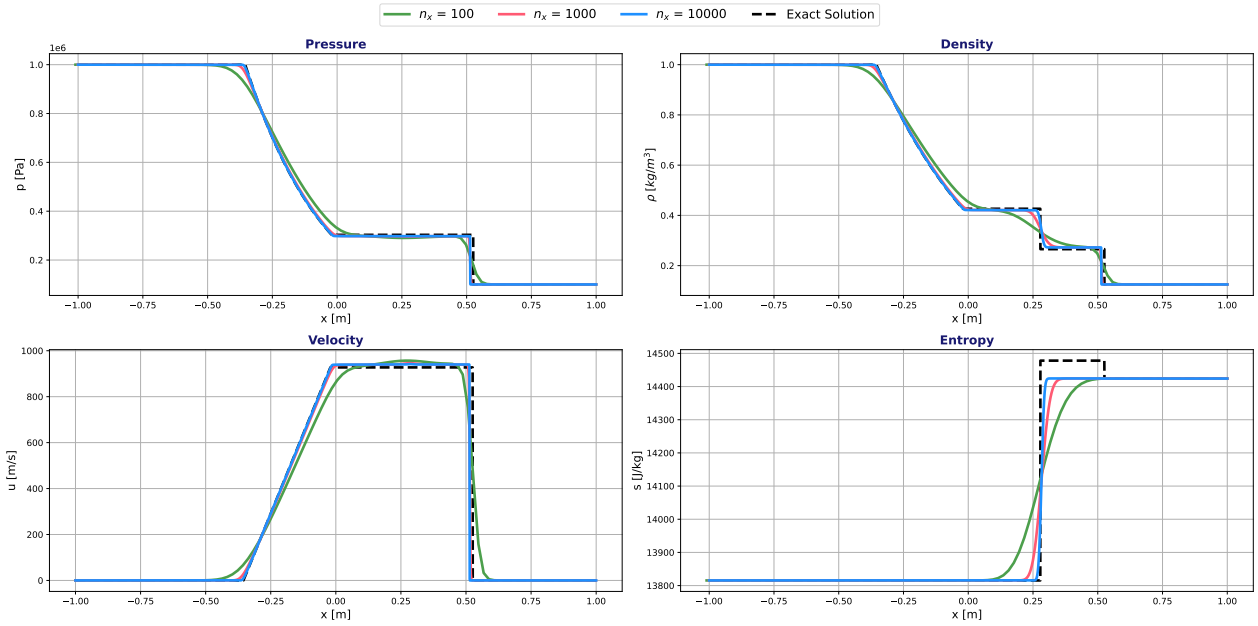


FIGURE 7.25
Sod shock tube - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$.

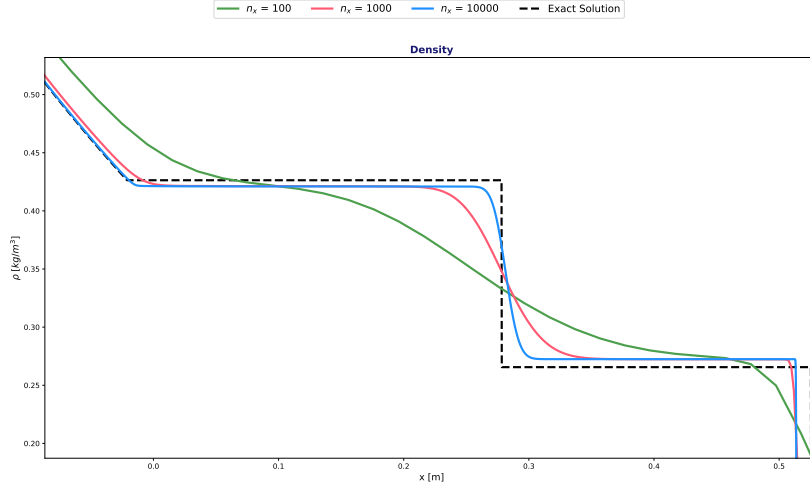


FIGURE 7.26

Sod shock tube - Analytical and numerical solutions for the density (zoomed) with $n_x \in \{10^2, 10^3, 10^4\}$.

In Figure 7.27, the L_2 -norm of the errors is represented as a function of the number of cells for the pressure, the velocity and the density. The presence of plateaus for $n_x > 10^3$ confirms that the numerical solution converges to another solution than the analytical one (again, see [76] for scalar case and [51] for Euler Equations).

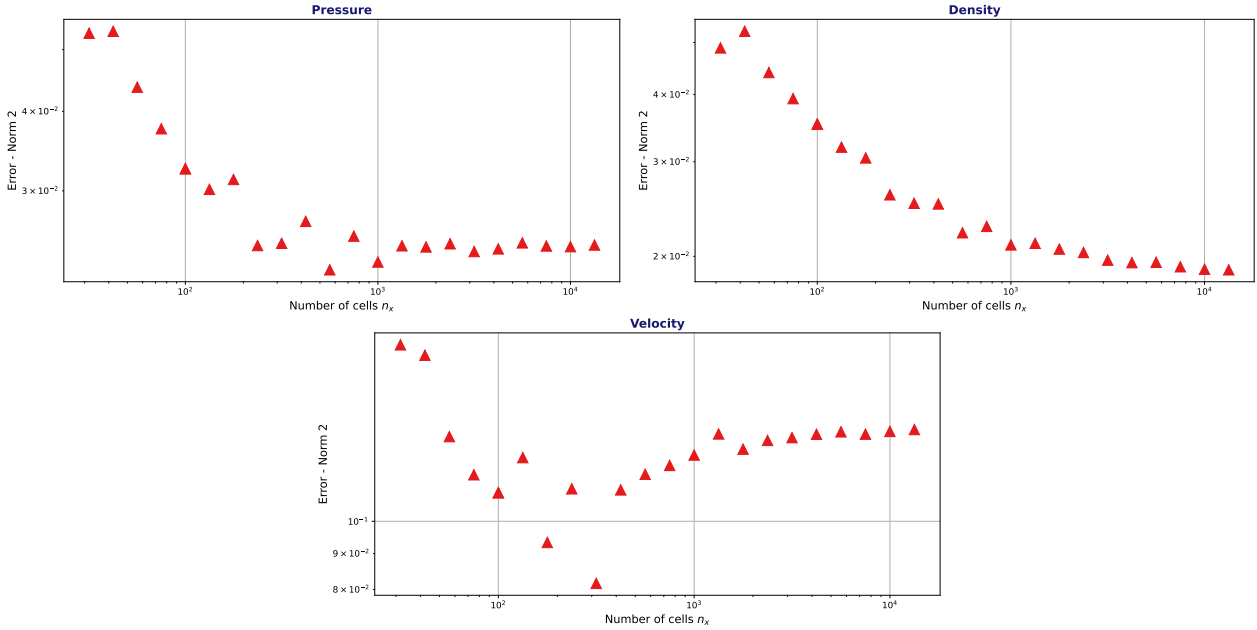


FIGURE 7.27

Sod shock tube - L_2 -norm of the error as a function of the number of cells for variables (p, u, ρ) .

7.3 Verification of ThermoTorch for a smooth one-dimensional Riemann problem with the 4-equation model

In this section, a solution to a Riemann problem (from **Appendix 3.C**) is used to verify the unsteady numerical methods of ThermoTorch with the 4-equation model. Here, the objective is to determine the consistency error introduced by the use of the pseudo-entropy approximation in

the unsteady part of the energy equation scheme (see Section 6.1.2) of ThermoTorch. Smooth solutions with rarefaction waves are considered, since Section 7.2 has already shown a consistency error for shock solutions with the 3-equation model, which would also appear for the 4-equation model. The analytical solution from **Appendix 3.C** is obtained using a simple mixture equation of state, for which the entropy formula of the system is known (unlike the general case). Thus, the analytical results can be compared with the results obtained with ThermoTorch when either entropy or pseudo-entropy is used.

First, the analytical solution with two-rarefaction waves is presented in Section 7.3.1. The pseudo-entropy approximation causes a consistency error due to two neglected terms: one term from the time variation of pressure and another from the time variation of the mass fraction. To decouple these two effects and first study the consistency error due to pressure, a case where the mass fraction remains uniform is studied. Section 7.3.2 compares the results of this case obtained with entropy (without consistency error) and pseudo-entropy (with consistency error on pressure). Section 7.3.3 proposes to numerically evaluate the magnitude of the error caused by the consistency error in pressure. In particular, the impact is evaluated as a function of the initial velocity conditions and the deviation from thermodynamic equilibrium. **Appendix 7.D** explores a case where the mass fraction is also variable.

7.3.1 Analytical solution and methodology

Two-rarefaction wave Riemann problem

The analytical solution with its mixture equation of state from **Appendix 3.C** is used. The initial conditions at $t = 0$ s are described in Figure 7.28. Several values for the velocities $u_R > 0$ and $u_L < 0$ are considered, always under subsonic conditions. The configurations considered are chosen so that two rarefaction waves develop. The structure of the solution is shown in Figure 7.29. Two rarefaction waves propagate through the medium at speeds $u - c$ and $u + c$. A double contact wave propagates at speed u in the center. The intermediate states are denoted with indices $*$ and $**$ as indicated in Figure 7.29. The pressure and the velocity are the same for the states $*$ and $**$. The detailed formulas to obtain the $*$ and $**$ states can be found in **Appendix 7.A**. They are noted respectively p_* and u_* , as shown in Figure 7.29.

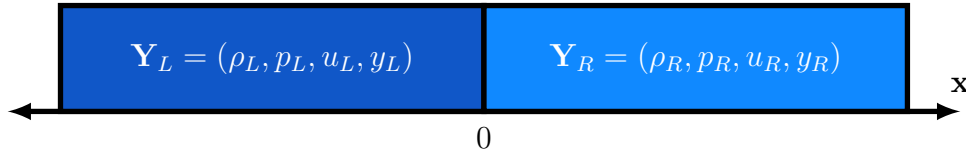
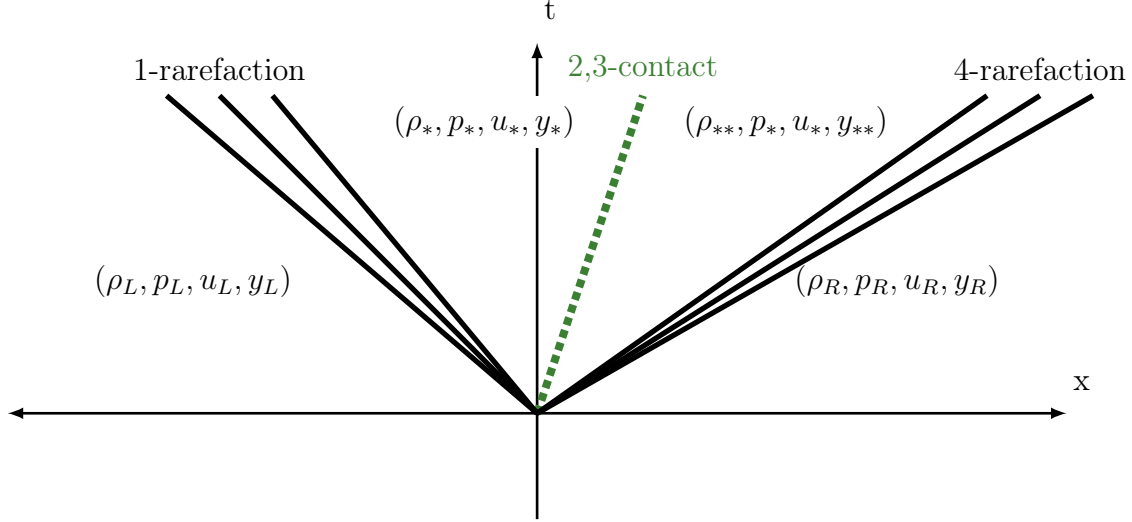


FIGURE 7.28
Initial configuration for a 1D Riemann problem with the 4-equation model.


FIGURE 7.29

Scheme of the solution for the two-rarefaction waves case for the 4-equation model.

Regardless of the initialization side $X \in \{L, R\}$, the mass fraction y_X must not be too far from the equilibrium mass fraction $\bar{y}(p_X, \rho_X)$. Otherwise, the code becomes unstable for the equation of state considered. Indeed, this can lead to negative thermodynamic quantities (temperature, density) for the liquid. For each configuration in the following, the mass fraction at equilibrium is obtained with the boundary conditions and the deviation from equilibrium is defined by:

$$\Delta y_X^{eq} = y_X - \bar{y}(p_X, \rho_X). \quad (7.4)$$

The flow is locally at thermodynamic equilibrium if $\Delta y_X^{eq} = 0$. If the deviation from equilibrium is positive, the flow is in subcooled boiling $T_l < T_{sat}$, with T_{sat} the saturation temperature and T_l the liquid temperature. If the deviation is negative, then the liquid is superheated ($T_l > T_{sat}$).

Pseudo-entropy and entropy computations

The configuration presented above is simulated with ThermoTorch. Two numerical methods are used. First, the method using pseudo-entropy is applied. It corresponds to a numerical method presented in **Chapter 6**. In this numerical scheme with pseudo-entropy, the blue and green terms in the following linearization have been neglected for the unsteady terms of the energy equation scheme:

$$T_l d\mathfrak{s} = dh - (v + y(T_l - T_s(p)) \frac{ds_g}{dp}) dp + \mathfrak{s}_y dy, \quad (7.5)$$

with

$$\mathfrak{s}_y = (h_l - \bar{h}_g) - T_l(s_l - \bar{s}_g) = \mu_l(p, T_l) - \bar{\mu}_g(p, T_s) + (T_s - T_l)\bar{s}_g. \quad (7.6)$$

For the approximate unsteady solutions computed with this scheme, this approximation introduces two consistency errors, one due to the variation of pressure (in green in Equation (7.5)) and the other due to the variation of mass fraction (in blue in Equation (7.5)). For the simplified mixture equation of state used in ThermoTorch, the entropy of the mixture is known. With p the pressure, ρ the mixture density, and y the mass fraction of vapor, the mixture entropy writes:

$$s(p, \rho, y) = s_0(y) + c_v(y) \log \left(\frac{p \rho^{-\gamma}}{\gamma - 1} \right). \quad (7.7)$$

The equation in entropy writes:

$$\rho \partial_t s + \partial_x (qs) - s \partial_x q = 0. \quad (7.8)$$

In ThermoTorch, a scheme similar to the one used for the mass fraction (see Equations (6.41) and (6.42) without source terms) is implemented for Equation (7.8) in entropy. The analytical solution of the problem can be compared with the solution obtained using the system entropy (without consistency error) or with the pseudo-entropy (with a consistency error). First (in Section 7.3.2 and Section 7.3.3), solutions with constant mass fraction y are considered. This allows evaluating only the impact of the consistency error due to pressure (in blue in Equation (7.5)), independently of the error due to the mass fraction.

All the simulations in this section are carried out with a constant CFL value, based on the fast waves (see Section 6.2.3):

$$\text{CFL}_{u+c} = 0.5. \quad (7.9)$$

We denote t_{max} as the maximum simulation time, i.e. the time at which one of the expansion waves reaches the boundary of the domain. All solutions are compared at the time $t_{comparison}$ defined as:

$$t_{comparison} = 0.3 t_{max}. \quad (7.10)$$

This allows obtaining fully developed solutions without the appearance of boundary effects.

7.3.2 Two-rarefaction waves solution with a constant mass fraction

Configuration

In this section, a symmetric configuration is considered:

$$\begin{cases} p_R = p_L = p_0, \\ y_R = y_L = y_0, \\ u_R = -u_L = u_0 > 0, \\ \rho_R = \rho_L = \rho_0. \end{cases} \quad (7.11)$$

It gives a symmetric initialization for the mass fraction at equilibrium:

$$\bar{y}(p_R, \rho_R) = \bar{y}(p_L, \rho_L) = \bar{y}_0. \quad (7.12)$$

The deviation from equilibrium is noted

$$\Delta y_0^{eq} = y_0 - \bar{y}_0. \quad (7.13)$$

This configuration corresponds to the simulation of a boundary condition at a wall. In this configuration, two rarefaction waves propagate on either side of the origin ($x = 0$). The wavefront of the 1-rarefaction wave propagates to the left of the domain at the speed $-u_0 - c(p_0, \rho_0)$ and the wavefront of the 4-rarefaction wave propagates to the right of the domain at the symmetric speed $u_0 + c(p_0, \rho_0)$. Due to symmetry, the 2,3-contact wave is steady ($u_* = 0$). Since the mass fraction y and entropy s are Riemann invariants in rarefaction waves and the configuration is symmetric, they remain uniformly constant, regardless of the simulated time. The contact is therefore a stationary ghost wave at $x = 0$ as the jump in mass fraction and entropy is null.

Data set of test cases

The objective is to select parameters that are representative of the conditions encountered in reactor core flows. A first reference case is chosen to be representative of a core reactor flow with moderate mass fraction. It is called **Test Case 0**. A typical reactor core mixture pressure and velocity are selected: $p_0 = 155$ bar and $u_0 = 5 \text{ m.s}^{-1}$. The mass fraction at equilibrium is $\bar{y}_0 = 0.01$. This case simulates a subcooled two-phase flow, with a deviation from thermodynamic equilibrium of $\Delta y_0^{eq} = 0.01$, so that $y_0 = 0.02$. The density is computed from these conditions, it is $\rho_0 = 769 \text{ kg.m}^{-3}$. Starting from this reference case, eleven test cases are considered to study the behavior of the scheme on this regular symmetric solution for conditions encountered in nominal operations but also for accidental cases. For u_0 , arbitrary values of 10 and 30 m.s^{-1} are considered. For $\bar{y}_0^0$ and Δy_0^{eq} , representative values have been selected, based on steady state THYC-calculations (see details in **Appendix 7.B**). The parameters considered for each case are shown in Table 7.12.

First, Test Case 0 is used to study the shape of the solutions obtained with pseudo-entropy and with entropy in Section 7.3.2. It is also used to study the order of convergence of the numerical scheme for this solution (also in Section 7.3.2). Subsequently, the other cases are used to evaluate the evolution of the consistency error due to pressure as a function of the variation of parameters (velocity, mass fraction at equilibrium, deviation from equilibrium) in Section 7.3.3. For each case, only one parameter varies, allowing a sensitivity analysis for each variable such as:

- Cases 1-2: Sensitivity analysis on velocity u_0 ,
- Cases 3-6: Sensitivity analysis on the mass fraction at equilibrium $\bar{y}_0$ with a constant deviation Δy_0^{eq} .
- Cases 7-11: Sensitivity analysis on the deviation from equilibrium Δy_0^{eq} with a constant mass fraction at equilibrium $\bar{y}_0$,

Case	$u_0 > 0$ [m.s^{-1}]	p_0 [bar]	$\bar{y}_0$ [-]	Δy_0^{eq} [-]	y_0 [-]	ρ_0 [kg.m^{-3}]
0	5	155	0.01	0.01	0.02	769
1	10	155	0.01	0.01	0.02	769
2	30	155	0.01	0.01	0.02	769
3	5	155	0.05	0.01	0.06	602
4	5	155	0.1	0.01	0.11	473
5	5	155	0.2	0.01	0.21	331
6	5	155	0.3	0.01	0.31	255
7	5	155	0.01	-0.005	0.005	575
8	5	155	0.01	0	0.01	627
9	5	155	0.01	0.005	0.015	691
10	5	155	0.01	0.02	0.03	993
11	5	155	0.01	0.03	0.04	1400

TABLE 7.12

Initial conditions for the test cases of symmetric two-rarefaction waves.

Comparison of the solutions obtained with entropy and pseudo-entropy

Test Case 0 is first simulated with pseudo-entropy and with entropy. The numerical solution obtained for $n_x \in \{10^2, 10^3, 10^4\}$ and the analytical solution are shown in Figure 7.30 when entropy is used, and in Figure 7.31 when pseudo-entropy is used. A zoom on the central plateaus for pressure and density are represented in Figure 7.32 when using entropy and in Figure 7.33 when using pseudo-entropy. As expected, the solution is symmetric, with two rarefaction waves propagating

at the same speed in opposite directions outward from the domain. Since the observed time is $t_{comparison} = 0.3t_{max}$, the waves are approximately one-third of the way through the domain. The velocity u_0 considered here is 5 m.s^{-1} , which is low compared to the speed of sound in the domain (in the order of 250 m.s^{-1} with the equation of state and thermodynamic parameters considered here). As a result, the rarefaction waves are very steep, but they are indeed rarefaction waves and not shocks.

When entropy is used in ThermoTorch, the numerical solution appears to approach the analytical solution as the mesh is refined. The pressure and density plateau seems to converge toward the analytical values. When pseudo-entropy is used, the numerical solution no longer appears to converge to the analytical solution but rather to a different one, with a notably different pressure and density plateau. Since the central pressure and density values are incorrect, this also affects the position of the rarefaction waves. A consistency error in the unsteady solutions of the four-equation model thus arises due to the pressure term neglected in the energy equation (see Section 6.1.2). This consistency error is visually noticeable, and the computed difference in pressure between the plateau of the numerical solution with the finest mesh and the analytical solution is in the order of 0.07%. A convergence study confirms these observations below. Whether using entropy or pseudo-entropy, the numerical solution does produce a 2,3-contact wave that is a ghost wave. The observed solution for the mass fraction is indeed constant.

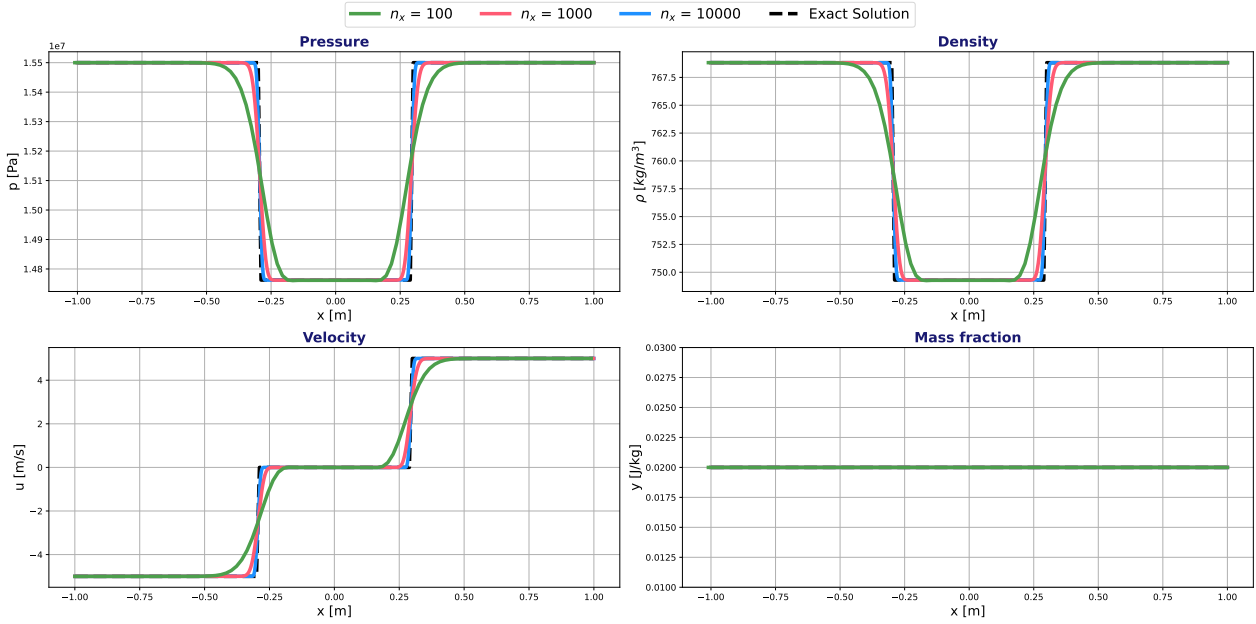


FIGURE 7.30

Test Case 0 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using entropy.

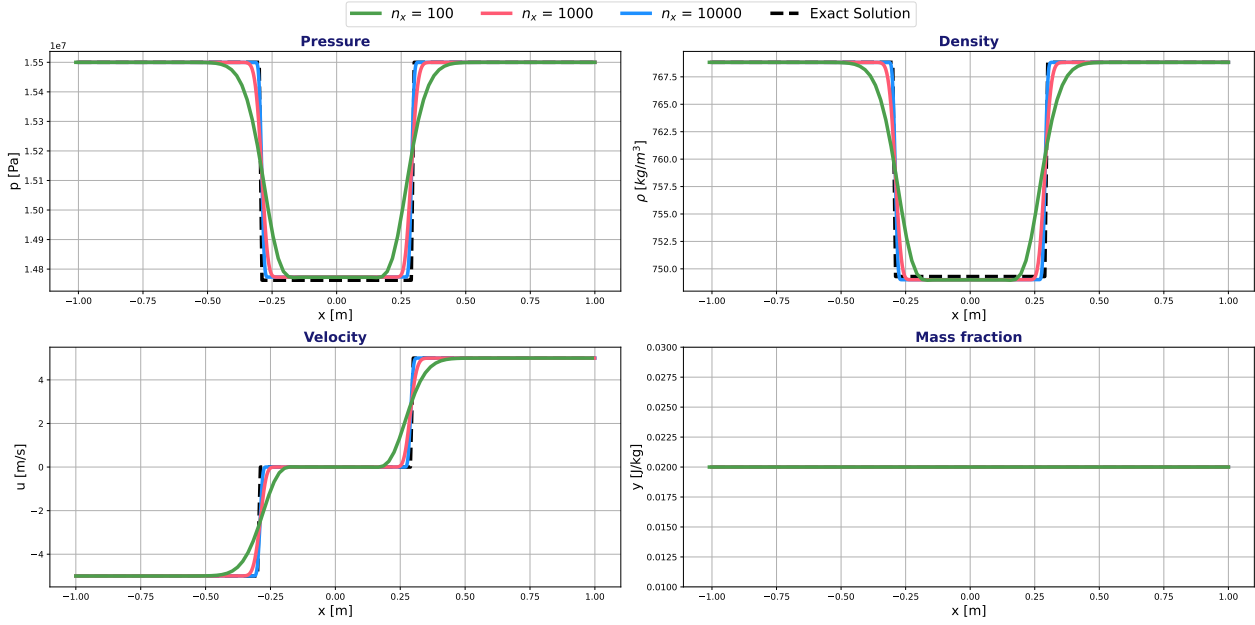


FIGURE 7.31

Test Case 0 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy.

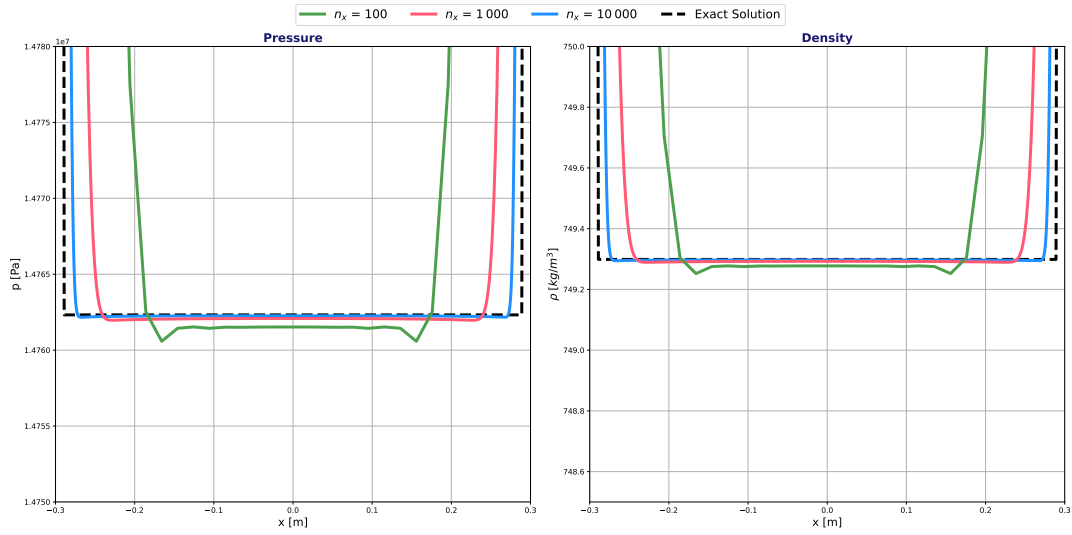


FIGURE 7.32

Test Case 0 - Analytical and numerical pressure and density fields with $n_x \in \{10^2, 10^3, 10^4\}$ when using entropy - zoom on the central plateaus.

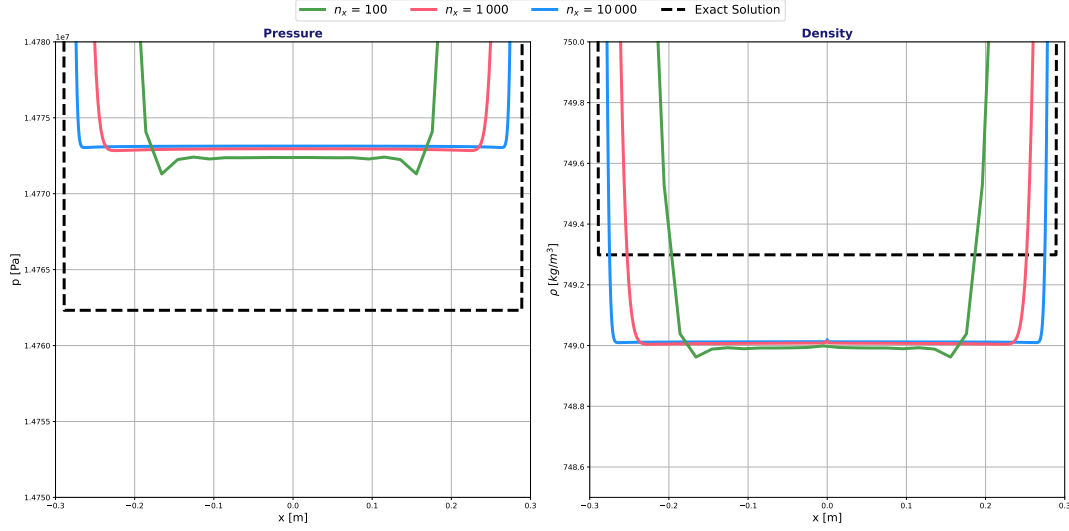


FIGURE 7.33

Test Case 0 - Analytical and numerical pressure and density fields with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy - zoom on the central plateaus.

Study of the convergence rate

The numerical schemes used in ThermoTorch are first-order in both time and space. Therefore, a convergence rate of 1 is expected for the rarefaction waves and 1/2 at the contact discontinuity. The contact is thus the limiting factor, and the overall convergence rate should be around 1/2.

For each case (with entropy and with pseudo-entropy), a convergence study is carried out by evaluating the L_2 -norm error on all variables between the analytical solution and the numerical solution obtained at time $t_{comparison}$. The errors (in L_2 -norm) between the numerical and analytical solutions for the variables (p, ρ, u, y) are shown as a function of the number of cells for $n_x \in [10^2, 10^{4.75}]$ in Figure 7.34 when entropy is used and for $n_x \in [10^2, 10^{4.5}]$ in Figure 7.35 when pseudo-entropy is used. Table 7.13 (resp. Table 7.14) presents the numerical values for the convergence rates computed when entropy (resp. pseudo-entropy) is considered.

The mass fraction is constant in the analytical solution. This behavior is well reproduced by both methods (with entropy or pseudo-entropy). The error on the mass fraction is therefore close to round-of error (in the order of 10^{-16}). As a result, the convergence rate for mass fraction is not observable with this Test Case where mass fraction is constant. This is why the convergence rate for mass fraction is not shown in Tables 7.13 and 7.14. In Figure 7.34, the error decreases for pressure, velocity, and density, with a convergence rate that increases gradually and exceeds slightly the value of 0.5. As expected, the ThermoTorch scheme with entropy converges to the correct solution, with a convergence rate of 1/2.

When pseudo-entropy is used, as already observed in Figure 7.36, the numerical scheme does not converge to the analytical solution but to another solution due to a consistency error in the energy equation. The order of convergence therefore gradually decreases as the mesh is refined. A plateau is reached for the error in pressure, density, and velocity in Figure 7.35. This plateau is reached starting from a mesh of 17782 cells.

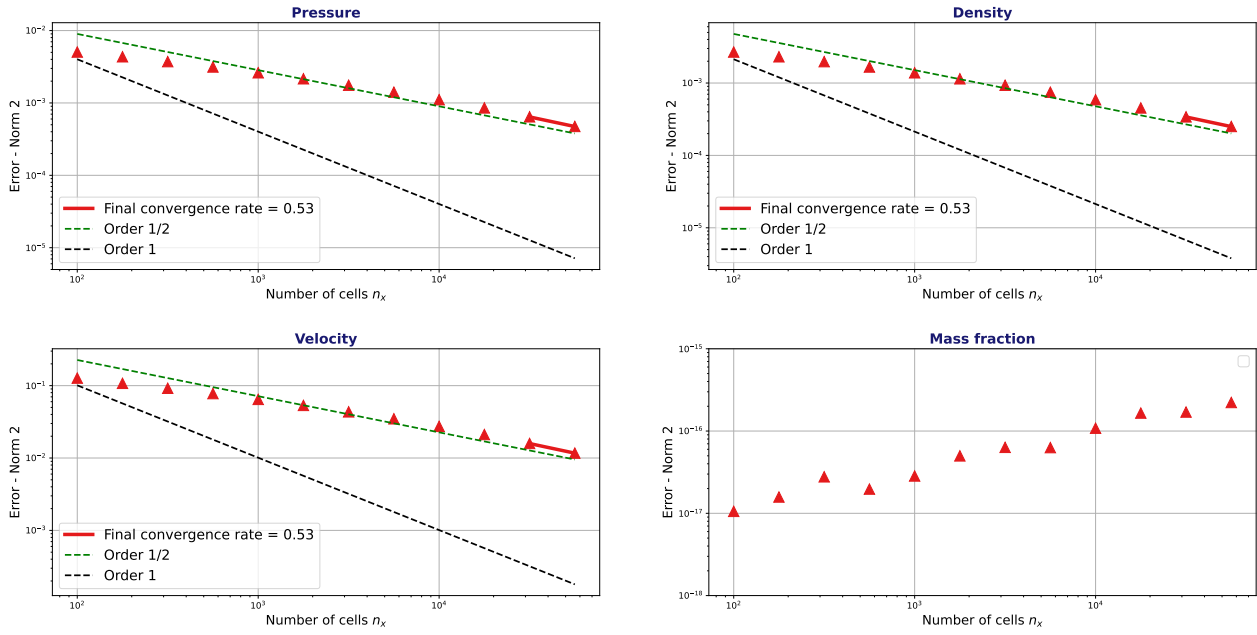


FIGURE 7.34

Test Case 0 - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ, y) when the entropy is considered.

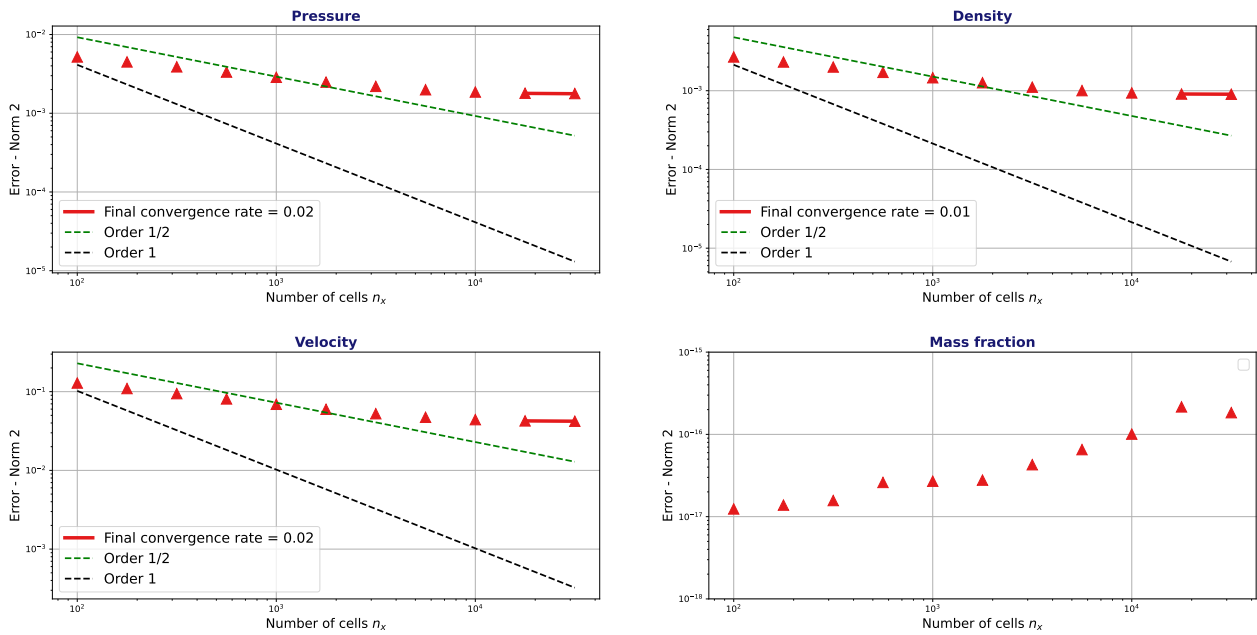


FIGURE 7.35

Test Case 0 - L_2 -norm of the error as a function of the number of cells n_x for variables (p, u, ρ, y) when the pseudo-entropy $\bar{s}$ is considered.

Number of cells	Pressure	Velocity	Density
100			
177	0.26	0.29	0.27
316	0.27	0.27	0.27
562	0.30	0.29	0.30
1000	0.32	0.32	0.32
1778	0.33	0.33	0.33
3162	0.36	0.36	0.36
5623	0.38	0.38	0.38
10000	0.42	0.42	0.42
17782	0.45	0.45	0.45
31622	0.49	0.49	0.49
56234	0.53	0.53	0.53

TABLE 7.13

Test Case 0 - Convergence rate for pressure, velocity and density as a function of the number of cells when the entropy is considered.

Number of cells	Pressure	Velocity	Density
100			
177	0.25	0.28	0.26
316	0.25	0.26	0.25
562	0.26	0.27	0.27
1000	0.27	0.27	0.27
1778	0.24	0.25	0.25
3162	0.22	0.23	0.22
5623	0.17	0.18	0.18
10000	0.12	0.12	0.12
17782	0.06	0.06	0.06
31622	0.02	0.02	0.01

TABLE 7.14

Test Case 0 - Convergence rate for pressure, velocity and density as a function of the number of cells when the pseudo-entropy $\mathfrak{s}$ is considered.

7.3.3 Numerical evaluation of the consistency error due to the pseudo-entropy approximation

As observed in Section 7.3.2, the ThermoTorch numerical scheme with pseudo-entropy gives an approximate solution that does not converge to the analytical solution as the mesh is refined. The objective of this section is to evaluate the order of magnitude of the consistency error due to pressure (see green term in Equation (7.5)). In particular, sensitivity analyses are carried out by varying the velocity u_0 , the equilibrium mass fraction $\bar{y}_0$, and the deviation from equilibrium Δy_0^{eq} . To do this, the consistency error due to pressure is computed using the quantity e_c^r . This quantity corresponds to the relative error made on the pressure plateau with the pseudo-entropy scheme compared to the analytical solution. Let $p_s^{n_x=3000}$ denote the value of the pressure plateau obtained with pseudo-entropy for a sufficiently fine mesh ($n_x = 3000$), and p_{exact} the analytical value of the pressure plateau. The consistency error e_c^r is then given by:

$$e_c^r = \frac{|p_s^{n_x=3000} - p_{exact}|}{p_{exact}}. \quad (7.14)$$

This consistency error is compared with the error made on the pressure plateau due to the use of an industrial mesh. To do this, the solution with entropy is considered on an industrial mesh ($n_x = 50$). On the pressure plateau, denoted $\mathcal{P}$, the pressure computed with the industrial mesh is noted $p_s^{n_x=50}(x_{\mathcal{P}})$. It depends on the position $x_{\mathcal{P}}$ on the plateau $\mathcal{P}$ at the comparison time $t_{comparison}$. The relative error $e_{im}^r(\mathcal{P})$ on the pressure plateau due to the use of an industrial mesh is thus evaluated with:

$$\forall x_{\mathcal{P}} \in \mathcal{P}, e_{im}^r(x_{\mathcal{P}}) = \frac{|p_s^{n_x=50}(x_{\mathcal{P}}) - p_{exact}|}{p_{exact}}. \quad (7.15)$$

The average value over the entire plateau, denoted $\langle e_{im}^r \rangle$, is used to compare with the consistency error. It is calculated with:

$$\langle e_{im}^r(x_{\mathcal{P}}) \rangle = \int_{\mathcal{P}} e_{im}^r(x_{\mathcal{P}}) dx_{\mathcal{P}} \approx \frac{1}{N_{\mathcal{P}}} \sum_{i=1}^{N_{\mathcal{P}}} e_{im}^r((x_{\mathcal{P}})_i), \quad (7.16)$$

where $(x_{\mathcal{P}})_i$ are the $N_{\mathcal{P}}$ points of the industrial mesh located on the pressure plateau. In the following, the errors e_c^r and $\langle e_{im}^r \rangle$ are computed for comparison. The objective is to verify that this consistency error is negligible compared to the error caused by the use of the industrial mesh.

In the case considered here, the mass fraction is constant. The consistency error is caused by a pressure term neglected in the linearization of the pseudo-entropy. This term is noted f_p and is given by

$$f_p(p, s, y) = y(T_g(p) - T_l(p, s, y)) \frac{ds_g}{dp}(p). \quad (7.17)$$

The consistency error should therefore be more significant when the mass fraction is high, and when the deviation from equilibrium, represented by the factor $T_g - T_l$ in Equation (7.17) is large. Sensitivity analyses are conducted to study the impact of variations in velocity, mass fraction, and deviation from equilibrium on the consistency error.

Consistency error as a function of the velocity

In this section, the results obtained for Test Cases 0, 1, and 2 with pseudo-entropy are compared. The velocity u_0 varies for these Test Cases from 5 $m.s^{-1}$ to 30 $m.s^{-1}$. Figure 7.31 (resp. 7.36) shows the analytical and numerical solutions for $n_x \in \{10^2, 10^3, 10^4\}$ of Test Case 1 (resp. Test Case 2), for which the initial velocity is $u_0 = 10 \text{ m.s}^{-1}$ (resp. $u_0 = 30 \text{ m.s}^{-1}$). These solutions can be compared with Figure 7.31, where Test Case 0 is shown (velocity $u_0 = 5 \text{ m.s}^{-1}$). The rarefaction waves become less steep as the velocity increases, as expected. The error on the plateau appears to increase slightly with velocity. Table 7.15 shows the errors e_c^r and $\langle e_{im}^r \rangle$ as a function of the test cases. Figures 7.38, 7.39, and 7.40 show the error $e_{im}^r(x_{\mathcal{P}})$ due to the industrial mesh as a function of the position on the plateau $x_{\mathcal{P}}$ (green solid line) for Test Cases 0, 1, and 2. The values of $\langle e_{im}^r \rangle$ and e_c^r are also shown with the green dashed and pink solid lines.

The use of an industrial mesh causes significant errors at the edges of the plateau because the rarefaction waves are poorly resolved with so few cells. An increase in the consistency error with the flow velocity is observed. The error due to the use of the industrial mesh also increases. As shown in Table 7.15, for Test Case 0, the consistency error of 0.07% is much lower (by a factor of 7) than the error due to the industrial mesh (0.45%). As the velocity increases, the consistency error increases more rapidly than the error due to the industrial mesh. It remains negligible (factor of 6) for Test Case 1 where $u_0 = 10 \text{ m.s}^{-1}$. For Test Case 2, the consistency error is still more than three times lower than the error due to the industrial mesh, but it becomes less negligible.

This sensitivity study shows that the consistency error is negligible for Test Case 0 when an industrial mesh is used. This error increases with the mixture velocity u_0 . It is shown that the consistency

error is proportional to velocity in Section 7.3.4. It is also negligible for $u_0 = 10 \text{ m.s}^{-1}$. It is no longer completely negligible for high velocities ($u_0 = 30 \text{ m.s}^{-1}$) even if the consistency error remains below the error due to industrial mesh, but such velocities are not encountered in nuclear reactor core conditions.

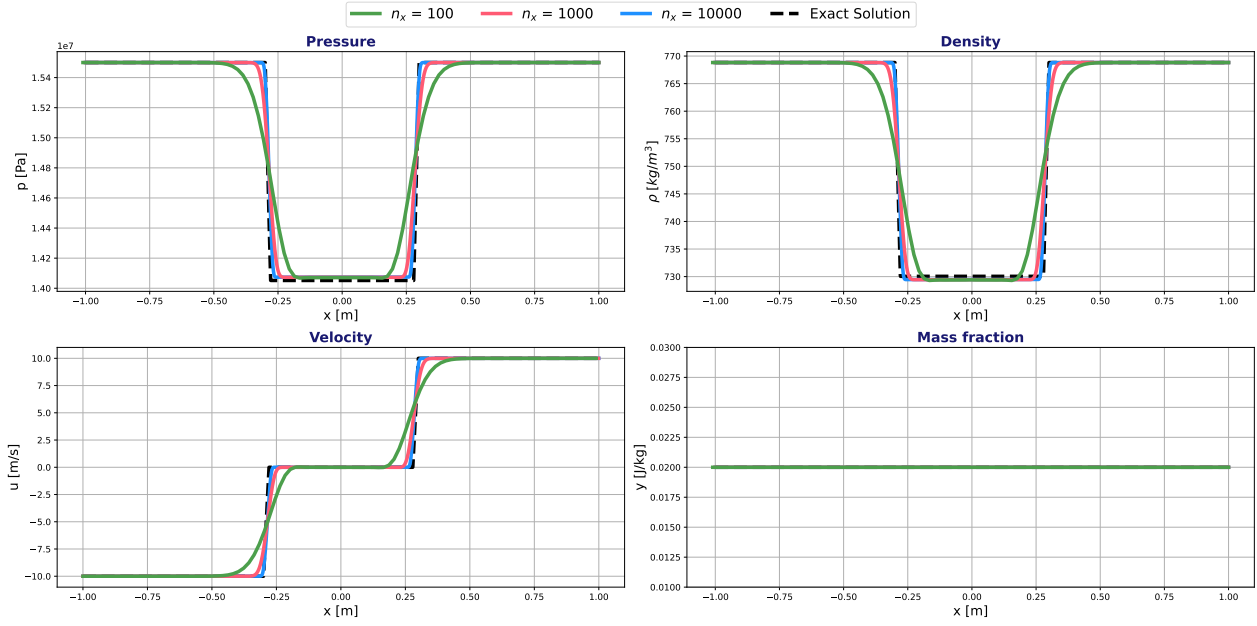


FIGURE 7.36

Test case 1 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy.

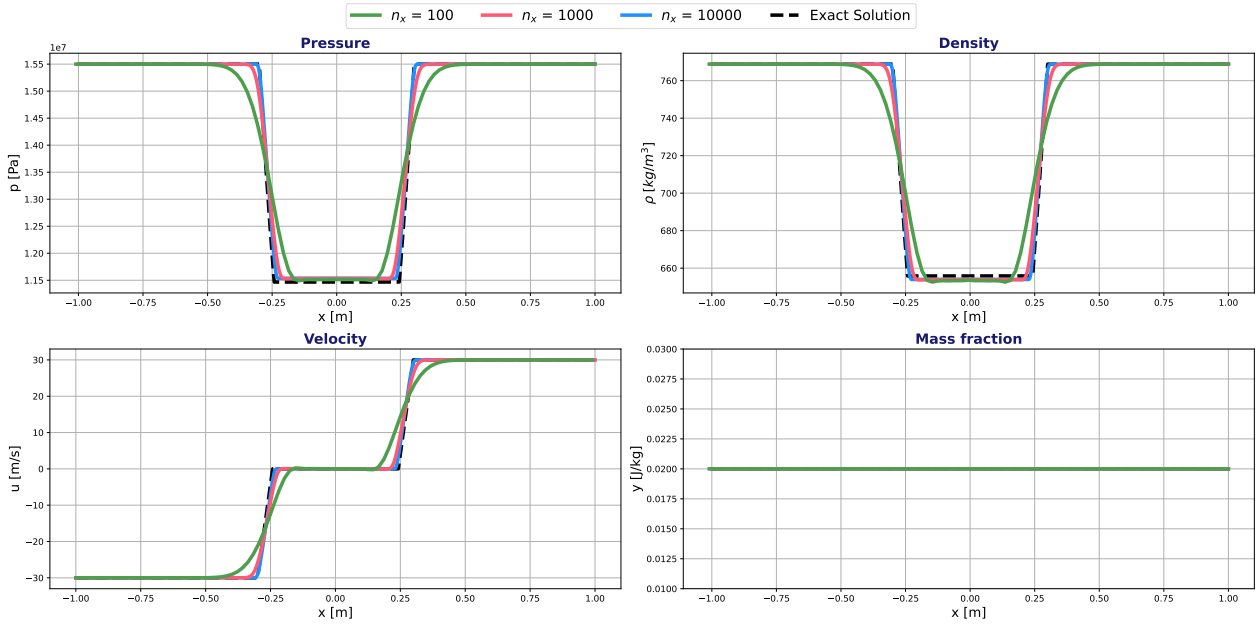


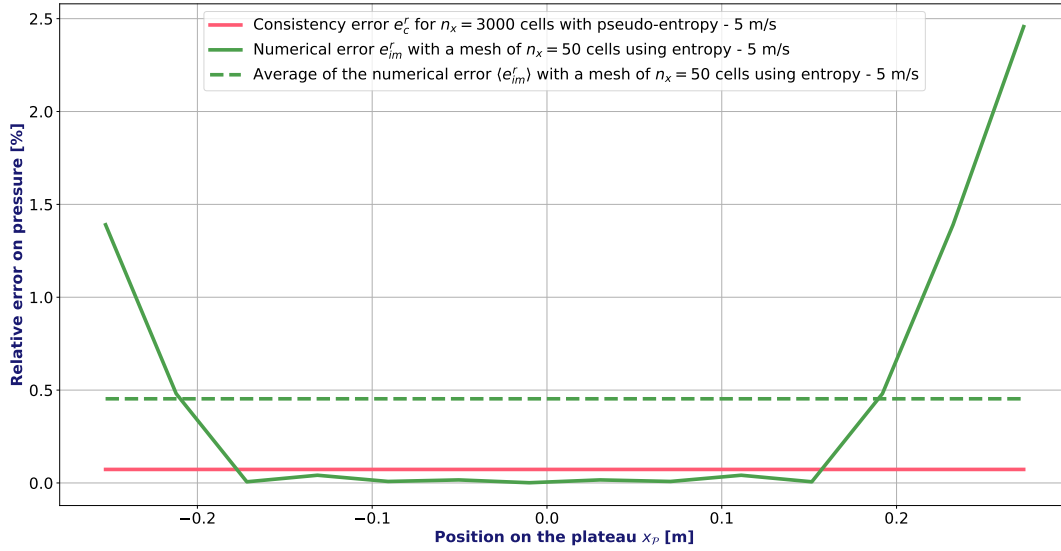
FIGURE 7.37

Test case 2 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy.

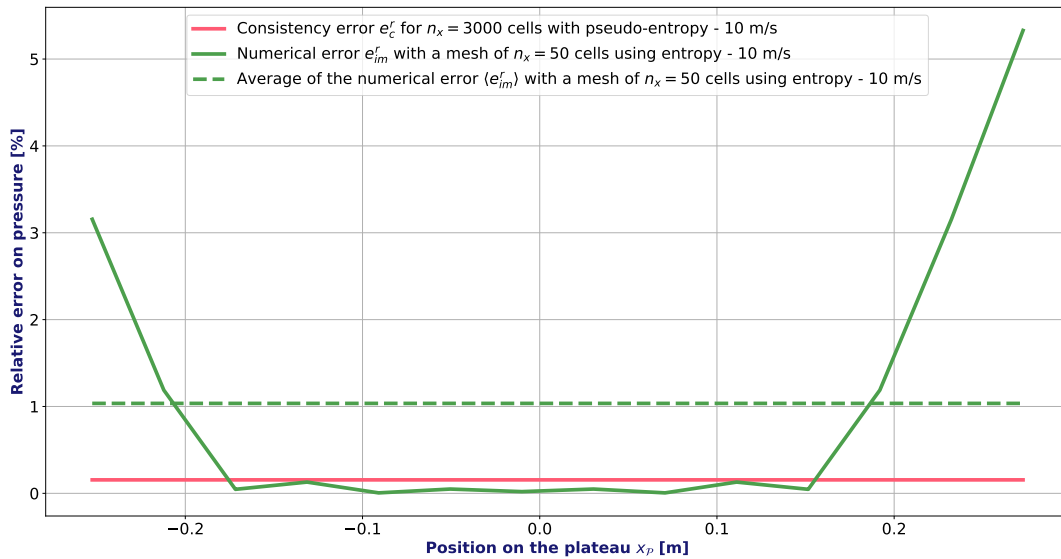
Case	$u_0 [m.s^{-1}]$	$e_c^r [\%]$	$\langle e_{im}^r \rangle [\%]$
0	5	0.07	0.45
1	10	0.16	1.04
2	30	0.60	2.86

TABLE 7.15

Relative error on the pressure plateau as a function of the velocity of the test case.

**FIGURE 7.38**

Test case 0 - Relative error e_{im}^r (and its average value $\langle e_{im}^r \rangle$ in dotted green) as a function of the position on the plateau x_p . The consistency error e_c^r is also represented in pink.

**FIGURE 7.39**

Test case 1 - Relative error e_{im}^r (and its average value $\langle e_{im}^r \rangle$ in dotted green) as a function of the position on the plateau x_p . The consistency error e_c^r is also represented in pink.

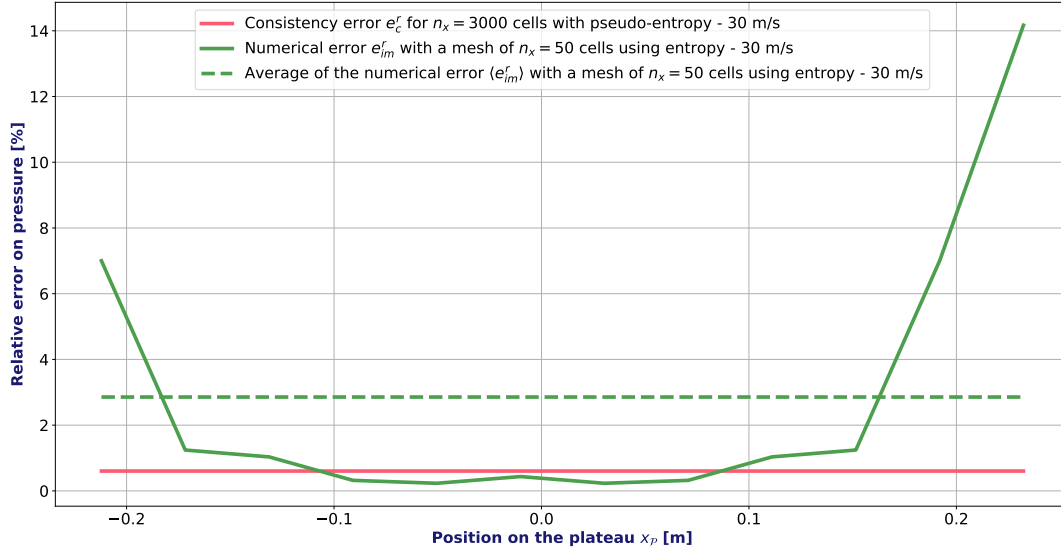


FIGURE 7.40

Test case 2 - Relative error e_{im}^r (and its average value $\langle e_{im}^r \rangle$ in dotted green) as a function of the position on the plateau x_p . The consistency error e_c^r is also represented in pink.

Consistency error as a function of the mass fraction at constant deviation from equilibrium

Test Cases 3-6 are computed, gradually increasing the equilibrium mass fraction y_0 while maintaining the deviation from equilibrium Δy_0^{eq} , so that the mass fraction y also increases. Table 7.16 shows the errors e_c^r and $\langle e_{im}^r \rangle$ as a function of the equilibrium mass fraction $\bar{y}_0$. The increase in mass fraction causes an increase in the consistency error. The neglected term $y(T_l - T_g) \frac{ds_g}{dp}$ is proportional to the mass fraction y , which explains the increase in consistency error with the mass fraction. From a mass fraction of 0.1, the error is no longer negligible compared to the error due to the industrial mesh. However, when the mass fraction increases above $y = 0.1$, the deviation from equilibrium is well below the 0.01 for nuclear reactor core conditions. This is discussed in **Appendix 7.C** in order to determine the pairs $(\bar{y}_0, \Delta y_0^{eq})$ for which the consistency error is negligible.

Case	$\bar{y}_0$	e_c^r [%]	$\langle e_{im}^r \rangle$ [%]
0	0.01	0.07	0.45
3	0.05	0.15	0.39
4	0.1	0.20	0.35
5	0.2	0.24	0.28
6	0.3	0.27	0.32

TABLE 7.16

Relative error on the pressure plateau as a function of the mass fraction at equilibrium.

Consistency error as a function of the deviation from the equilibrium at constant mass fraction at equilibrium

Test Cases 7-11 are computed. This allows the behavior of the consistency error in pressure to be studied as a function of the deviation from thermodynamic equilibrium Δy_0^{eq} . Test Case 7 is an initial configuration where the liquid is supersaturated ($y_0 < \bar{y}_0$). Test Case 8 corresponds to an initial configuration at thermodynamic equilibrium ($y_0 = \bar{y}_0$). Test Cases 9–11 are cases with subcooled boiling ($y_0 > \bar{y}_0$).

Table 7.17 shows the errors e_c^r and $\langle e_{im}^r \rangle$ as a function of the deviation from equilibrium Δy_0^{eq} . The

further the flow is from thermodynamic equilibrium, the greater the consistency error, whether the deviation is positive or negative. This can be explained by analyzing the neglected term, which is proportional to $y(T_l - T_g) \frac{ds_g}{dp}$. The greater the deviation from thermodynamic equilibrium (regardless of its sign), the greater the temperature difference between the two phases $T_l - T_g$, which increases the consistency error. An error is still observed for the case $\Delta y_0^{eq} = 0$ because the initial state is at equilibrium, but the intermediate state $*$ is no longer at equilibrium, which introduces a consistency error.

While the error is negligible at thermodynamic equilibrium compared to the error due to the use of an industrial mesh, it gradually increases as the flow deviates from equilibrium. It remains negligible compared to the error due to the industrial mesh up to $\Delta y_0^{eq} = 0.01$. From $\Delta y_0^{eq} = 0.02$, the error is no longer negligible. This sensitivity analysis makes it possible to evaluate the behavior of the consistency error as a function of the deviation from equilibrium. However, this analysis was carried out for constant $\bar{y}_0$. The approximate values from which the consistency error is negligible depends not only on the deviation from equilibrium but on the pair $(\bar{y}_0, \Delta y_0^{eq})$. A study on these pairs is carried out in **Appendix 7.C** in order to more precisely determine the combinations for which the error is negligible.

Case	$\Delta y_0^{eq} [-]$	$e_c^r [\%]$	$\langle e_{im}^r \rangle [\%]$
7	-0.005	0.006	0.38
8	0	1.4e-04	0.40
9	0.005	0.02	0.43
0	0.01	0.07	0.45
10	0.02	0.30	0.52
11	0.03	0.90	0.64

TABLE 7.17

Relative error on the pressure plateau as a function of the deviation from equilibrium.

7.3.4 Conclusion on the consistency error

An analytical estimate for the pressure consistency error is detailed in **Appendix 7.C**. This made it possible to quickly scan the $(\bar{y}_0, \Delta y_0^{eq})$ space to estimate when the consistency error is negligible compared to the error due to the industrial mesh, when $u_0 = 5 \text{ m.s}^{-1}$. The sensitivity studies and Figure 7.44 (in **Appendix 7.C**) have shown that the consistency error (due to the neglected pressure term) is negligible compared to the error caused by using an industrial mesh, when:

- the velocity is under 10 m.s^{-1} ,
- the equilibrium mass fraction and the deviation from equilibrium are in the green zone of Figure 7.44.

The ThermoTorch scheme with pseudo-entropy allows for obtaining good results for steady-state solutions very quickly thanks to the pseudo-entropy approximation, which decouples the energy equation from the other equations. However, this approximation introduces a consistency error when computing unsteady solutions using the 4-equation model. According to these results, the consistency error may no longer be considered negligible when extreme accidental conditions are considered.

To study fast transients with the 4-equation model, particularly for highly out-of-equilibrium flows, a new numerical method should be used to eliminate the consistency error due to pressure. Several methods could be proposed in this case. First, the development of a triple coupled pressure-momentum-pseudo-entropy system with iterative resolution could solve this problem. As discussed in **Appendix 7.C**, this method is likely to be CPU time-consuming. Another method could be to

propose a new numerical scheme for energy, possibly based on the internal energy of the mixture (see [18, 19] and all the work related to the co-authors of these articles).

It is important to recall here that the results were obtained with a one-dimensional model with the simplified physical model of ThermoTorch and for a single Riemann problem case at constant mass fraction. The obtained conclusions should be consolidated with other test cases.

This first study focuses only on the consistency error caused by pressure variation (see the green term in Equation (7.5)). The consistency error due to variation in the mass fraction (blue term in Equation (7.5)) was not taken into account. **Appendix 7.D** presents a convergence study on a case with variable mass fraction when entropy and when pseudo-entropy are used. It is more difficult to determine the magnitude of this error because it is inherently coupled with the error due to pressure variation. However, it is not necessarily important to determine whether this error is negligible. The consistency error due to the mass fraction (in green in Equation (7.5)) could be easily corrected by considering the resolution system of ThermoTorch (see **Chapter 6**). Indeed, the numerical scheme for the disequilibrium equation on the mass fraction is decoupled from all other variables in the system (see Section 6.3). It can be solved first, and thus the variations in the mass fraction y would be known. The green term in Equation (7.5) could then be added to the energy equation.

7.A Derivation details of the solution for the unsteady Riemann problem with the 4-equations model

7.A.1 General case

The solution and the equation of state from **Appendix 3.C** are used. The mixture entropy is

$$s(p, \rho, y) = s_0(y) + c_v(y) \log \left(\frac{p \rho^{-\gamma}}{\gamma - 1} \right). \quad (7.18)$$

The speed of sound of the mixture is

$$c(p, \rho) = \sqrt{\frac{\gamma p}{\rho}}. \quad (7.19)$$

The solution is a self-similar function of the variable $\xi = \frac{x}{t}$, whose structure is described in Figure 7.29. Two rarefaction waves propagate on either side of the domain: the 1-rarefaction between state L and $*$, and the 4-rarefaction between state $**$ and R . A double contact wave, called the 2,3-contact wave, propagates at the center, between the two rarefaction waves. According to **Appendix 3.C**, the Riemann invariants for each wave are:

$$\begin{cases} \phi_1 = \{y, s, u + \frac{2c}{\gamma - 1}\}, \\ \phi_{2,3} = \{u, p\}, \\ \phi_4 = \{y, s, u - \frac{2c}{\gamma - 1}\}. \end{cases} \quad (7.20)$$

The following four states are considered:

- Known left state: $Y_L = (\rho_L, p_L, u_L, y_L)$,
- Left intermediate state to be determined: $Y_* = (\rho_*, p_*, u_*, y_*)$,
- Right intermediate state to be determined: $Y_{**} = (\rho_{**}, p_{**}, u_{**}, y_{**})$,
- Known right state: $Y_R = (\rho_R, p_R, u_R, y_R)$.

There are therefore eight unknowns (four for each intermediate state) to be determined. According to the invariants at the 2,3-contact wave, the states $*$ and $**$ have the same velocity and the same pressure such that:

$$\begin{cases} u_{**} = u_*, \\ p_{**} = p_*. \end{cases} \quad (7.21)$$

In each rarefaction wave, the mass fraction is conserved such that:

$$\begin{cases} y_* = y_L, \\ y_{**} = y_R. \end{cases} \quad (7.22)$$

The following notations are adopted:

$$\begin{cases} c_L = c(\rho_L, p_L), \\ R_L^+ = u_L + \frac{2c_L}{\gamma - 1}, \\ c_R = c(\rho_R, p_R), \\ R_R^- = u_R - \frac{2c_R}{\gamma - 1}. \end{cases} \quad (7.23)$$

The system to be solved now contains only four unknowns: $p_*, u_*, \rho_*, \rho_{**}$. The remaining four relations (two for each rarefaction) are written as:

$$\begin{cases} s(p_*, \rho_*, y_L) = s(p_L, \rho_L, y_L), \\ u_* + \frac{2c(p_*, \rho_*)}{\gamma - 1} = R_L^+, \\ u_* - \frac{2c(p_*, \rho_{**})}{\gamma - 1} = R_R^-, \\ s(p_*, \rho_{**}, y_R) = s(p_R, \rho_R, y_R). \end{cases} \quad (7.24)$$

Since the mass fraction is conserved in the rarefaction waves, the entropy equalities can be rewritten as:

$$\begin{cases} p_* \rho_*^{-\gamma} = p_L \rho_L^{-\gamma}, \\ p_* \rho_{**}^{-\gamma} = p_R \rho_R^{-\gamma}. \end{cases} \quad (7.25)$$

The system is then solved using the following relations:

$$\begin{aligned} r &= \frac{\rho_R}{\rho_L} (p_R p_L)^{-1/\gamma}, \\ u_* &= \frac{1}{1 + \sqrt{r}} (R_L^+ + \sqrt{r} R_R^-), \\ c_* &= (R_L^+ - u_*) \frac{\gamma - 1}{2}, \\ \rho_* &= \left(c_*^2 \frac{\rho_L^\gamma}{\gamma p_L} \right)^{\frac{1}{\gamma-1}}, \\ \rho_{**} &= r \rho_*, \\ p_* &= p_L \left(\frac{\rho_*}{\rho_L} \right)^\gamma. \end{aligned} \quad (7.26)$$

7.A.2 Symmetric configuration

When considering a symmetric configuration such that

$$\begin{cases} p_R = p_L = p_0, \\ \rho_R = \rho_L = \rho_0, \\ y_R = y_L = y_0, \\ u_R = -u_L = u_0 > 0. \end{cases} \quad (7.27)$$

The intermediate state $*$ and $**$ simplified. The 2,3-contact wave becomes a ghost wave:

$$\begin{cases} \rho_* = \rho_{**}, \\ p_* = p_{**}, \\ u_* = u_{**} = 0, \\ y_* = y_{**} = y_0. \end{cases} \quad (7.28)$$

The speed of sound c_* , the pressure p_* and the density ρ_* of the intermediate state can be written

using Equations (7.26)

$$\begin{cases} c_* = c_L - \frac{\gamma - 1}{2} u_0, \\ \rho_* = \left(c_*^2 \frac{\rho_0^\gamma}{\gamma p_0} \right)^{\frac{1}{\gamma-1}}, \\ p_* = p_0 \rho_0^{-\gamma} \left(c_*^2 \frac{\rho_0^\gamma}{\gamma p_0} \right)^{\frac{\gamma}{\gamma-1}} = f(p_0, \rho_0) c_*^{\frac{2\gamma}{\gamma-1}}. \end{cases} \quad (7.29)$$

7.B Analysis of two THYC-coeur computations in accidental conditions.

In this appendix, the objective is to propose boundary conditions for the Riemann problem that correspond to values encountered in reactor core flows. To consider realistic conditions in a reactor core, two THYC-coeur simulations are run. Two configurations are simulated, one under typical accidental conditions (called accidental case) and one in extreme accidental conditions (called severe accidental case). The mass fraction along the hottest assembly (where under-saturated boiling is most significant) is considered in order to examine the most penalizing conditions, i.e., those furthest from equilibrium. The considered cases are taken from the Bias Curve Database (see Section 5.1), among the most penalizing ones for the consistency error, i.e., with a high mass fraction and significant subcooled boiling (see Section 7.3.3).

The left part of Figure 7.41 (resp. Figure 7.42) shows the mass fraction and the equilibrium mass fraction as a function of height z in the core for the accidental case (resp. the severe accidental case). The right part of Figure 7.41 shows the deviation from equilibrium as a function of the equilibrium mass fraction for the accidental case, and Figure 7.42 does so for the severe accidental case. For both cases, it is observed that the axial flow in the core can be divided into three regions. The description is given for the accidental case. First, in the lower region of the core ($z \in [0, 2.8]$ m), the flow is single-phase liquid. Both the mass fraction and the equilibrium mass fraction are zero, and the deviation from equilibrium is also zero. The liquid is heated by the heat flux from the fuel.

The second region, defined by $z \in [2.8, 3.3]$ m, is the subcooled boiling zone. The liquid begins to vaporize even though it is not, on average, at saturation. The mass fraction therefore increases while the equilibrium mass fraction remains zero. This is where the deviation from equilibrium reaches its maximum value ($\Delta y_0^{eq} \sim 0.017$).

Eventually, the mixture reaches saturation and the equilibrium mass fraction begins to increase (for $z > 3.3$ m). This is the third region ($z \in [3.3, 4.8]$ m), where the deviation from equilibrium gradually decreases towards zero.

In the severe accidental case, these three regions also exist but occur at different heights. The third region is much wider. For $y > 0.1$, the deviation from equilibrium is near zero. While in the accidental case the maximum mass fraction is 0.07 and the maximum deviation from equilibrium is 0.017, in the severe accidental case these values are higher, with a maximum mass fraction of 0.35 and a maximum deviation of 0.03.

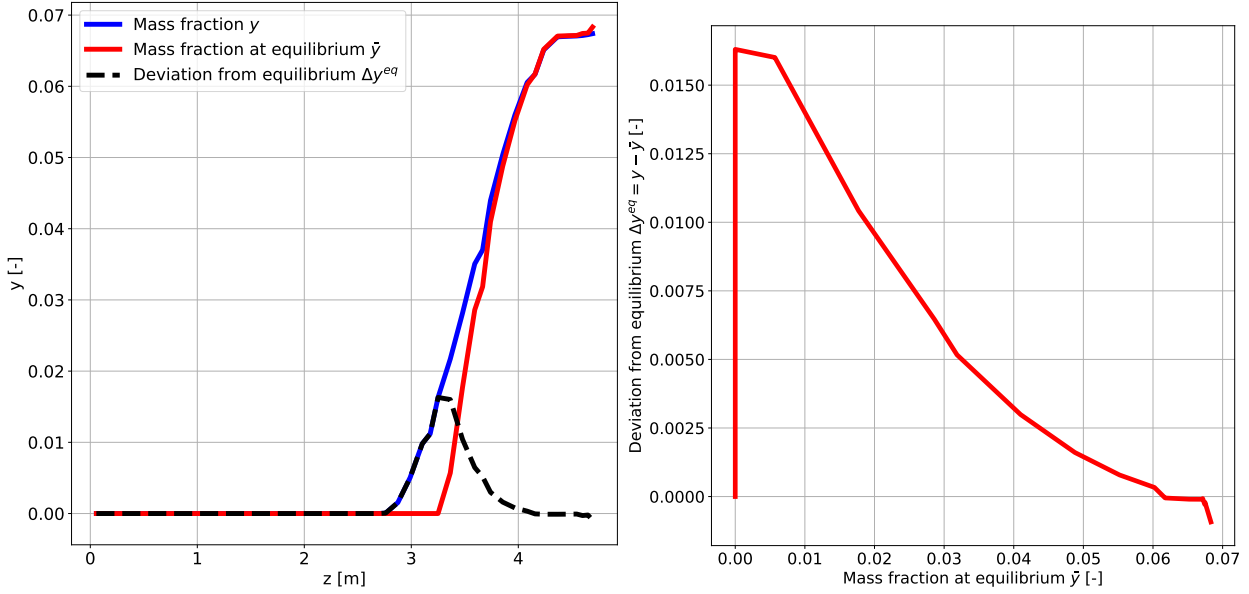


FIGURE 7.41

Mass fraction and mass fraction at equilibrium as a function of height (left) and deviation from equilibrium as a function of the mass fraction at equilibrium (right) - Accidental conditions of a nuclear reactor core.

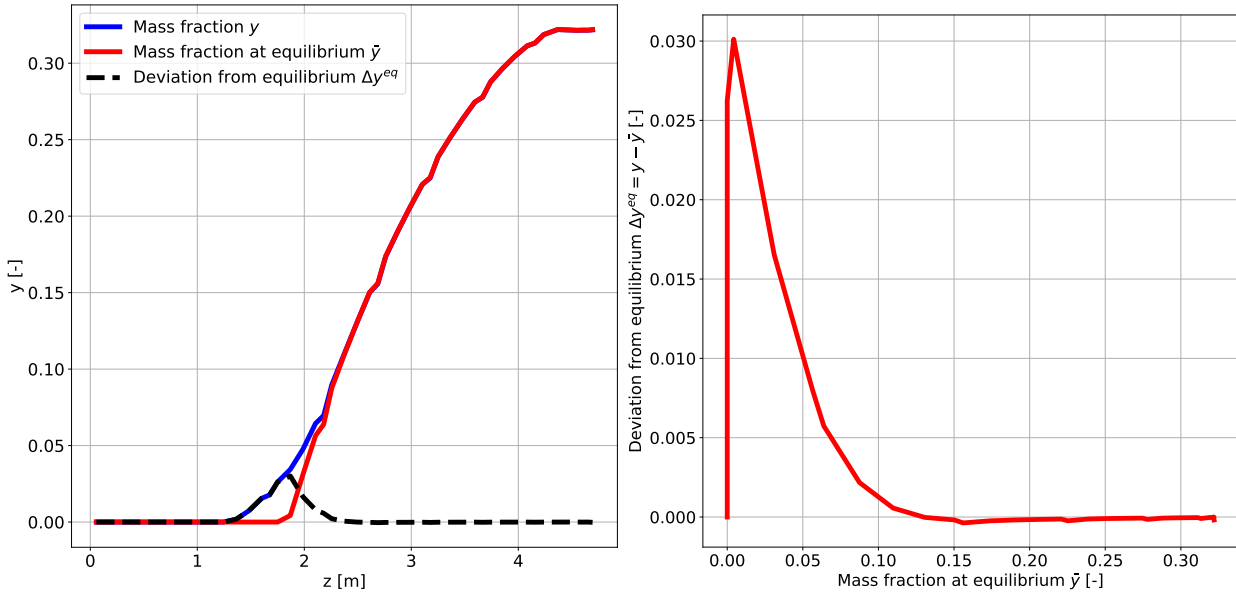


FIGURE 7.42

Mass fraction and mass fraction at equilibrium as a function of height (left) and deviation from equilibrium as a function of the mass fraction at equilibrium (right) - Severe accidental conditions of a nuclear reactor core.

7.C Estimation of the consistency error due to pressure

In this appendix, the consistency error e_c^r is analytically evaluated to show that the error is proportional to the neglected function f_p from Equation (7.17). Subsequently, the results obtained from the various sensitivity studies are summarized. In particular, this section details under which conditions the consistency error is negligible compared to the error due to the industrial mesh.

The consistency error is due to the term $f_p(p, s, y) = y(T_g - T_l) \frac{ds_g}{dp}$, which is neglected in the linearization of the pseudo-entropy. When the pseudo-entropy is used, the resolved energy equation

is

$$\rho T_l (\partial_t \mathbf{s} + u \partial_x \mathbf{s}) = 0. \quad (7.30)$$

The calculation to obtain the solution of the Riemann problem carried out in **Appendix 3.C** can be performed again. This time, the linearization of the density is expressed differently from the case with entropy, using Equation (6.105). The quantities f calculated for the case with pseudo-entropy are denoted $\tilde{f}$, and those calculated with entropy are written without a tilde. Then, using Equation (6.105):

$$\tilde{\alpha} = \alpha(1 + \lambda), \quad (7.31)$$

with $\alpha = \left(\frac{\partial \rho}{\partial p} \right)_{s,y}$, $\tilde{\alpha} = \left(\frac{\partial \rho}{\partial p} \right)_{\tilde{s},y}$ and the function λ defined from the neglected function f_p such that:

$$\lambda(p, s, y) = \rho(\gamma - 1) f_p(p, s, y). \quad (7.32)$$

The speed of sound computed with the pseudo-entropy is written $\tilde{c}$, such that

$$\tilde{c} = \frac{1}{\sqrt{\tilde{\alpha}}} = \frac{c}{\sqrt{1 + \lambda}}, \quad (7.33)$$

with c the speed of sound computed with entropy, i.e. $c = \sqrt{\frac{\gamma p}{\rho}}$. The objective is to derive an approximate formula for the consistency error in pressure. It is assumed that the neglected function f_p (see Equation (7.17)) in the energy equation when using pseudo-entropy is small compared to the specific volume of the mixture:

$$f_p = y(T_g - T_l) \frac{ds_g}{dp} \ll v. \quad (7.34)$$

This implies a slight modification of the speed of sound when using the pseudo-entropy compared to the actual speed of sound c (when entropy is used). It implies

$$\lambda(p, s, y) = \rho(\gamma - 1) f_p(p, s, y) \ll 1. \quad (7.35)$$

We can, as a first approximation, use the solution from **Appendix 7.A**. The Riemann invariant in the 4-rarefaction wave can be rewritten as:

$$u_0 - \frac{2c(p_0, \rho_0)}{\sqrt{1 + \lambda}(\gamma - 1)} \approx - \frac{2c(\tilde{p}_*, \tilde{\rho}_*)}{\sqrt{1 + \lambda}(\gamma - 1)}. \quad (7.36)$$

The modified speed of sound $c_*(\tilde{p}_*, \tilde{\rho}_*)$ obtained on the plateau using the pseudo-entropy can therefore be rewritten in terms of the speed of sound c_* obtained on the plateau using entropy:

$$c(\tilde{p}_*, \tilde{\rho}_*) \approx c_* \left(1 - \frac{\gamma - 1}{2} \frac{u_0}{c_*} \left(\sqrt{1 + \lambda} - 1 \right) \right), \quad (7.37)$$

where

$$c_* = \sqrt{\frac{\gamma p_*}{\rho_*}}. \quad (7.38)$$

The analytical solution from **Appendix 7.A**, using Equations (7.29) for a symmetric case, ensures that the pressure obtained on the plateau is written as:

$$p_* = f(p_0, \rho_0) c_*^{\frac{2\gamma}{\gamma-1}}. \quad (7.39)$$

Assuming that this equation remains valid when the pseudo-entropy is used, the pressure obtained on the plateau is:

$$\begin{aligned}\tilde{p}_* &\approx f(p_0, \rho_0) c_*^{\frac{2\gamma}{\gamma-1}} \times \left(1 - \frac{\gamma-1}{2} \frac{u_0}{c_*} \left(\sqrt{1+\lambda} - 1\right)\right)^{\frac{2\gamma}{\gamma-1}} \\ &= p_* \left(1 - \frac{\gamma-1}{2} \frac{u_0}{c_*} \left(\sqrt{1+\lambda} - 1\right)\right)^{\frac{2\gamma}{\gamma-1}}.\end{aligned}\quad (7.40)$$

The value of the consistency error can then be approximately estimated. This approximation will be denoted as $\tilde{e}_c^r$ and is obtained with

$$\tilde{e}_c^r = \frac{|\tilde{p}_* - p_*|}{p_*} \approx \left| \left(1 - \frac{\gamma-1}{2} \frac{u_0}{c_*} \left(\sqrt{1+\lambda} - 1\right)\right)^{\frac{2\gamma}{\gamma-1}} - 1 \right|. \quad (7.41)$$

As $\lambda \ll 1$ and considering $\frac{u_0}{c_*} \ll 1$, the approximate consistency error can be simplified to

$$\tilde{e}_c^r \approx \frac{\gamma u_0 \lambda}{2 c_*}, \text{ with } \lambda = \rho(\gamma-1)y(T_l - T_g) \frac{ds_g}{dp}. \quad (7.42)$$

This estimated consistency error $\tilde{e}_c^r$ is compared to the consistency error measured in ThermoTorch in Figure 7.43. This analytical approximation appears to be accurate compared to the numerically measured value for most of the Test Cases. The points for Test Case 2 and 11 are less accurate because the assumptions made are no longer valid: for Test Case 2, the fluid velocity $u_0 = 30 \text{ m.s}^{-1}$ is no longer negligible compared to the speed of sound on the plateau ($u_0/c_* \sim 0.16$), and for Test Case 11, it is the function λ that is no longer negligible compared to 1, since $\lambda \sim 0.32$ in that case. When the assumptions are valid, the consistency error is indeed proportional to the neglected term $f_p = y(T_l - T_g) \frac{ds_g}{dp}$, but also to the initial velocity u_0 , which explains the observations made in the various sensitivity studies in Section 7.3.3.

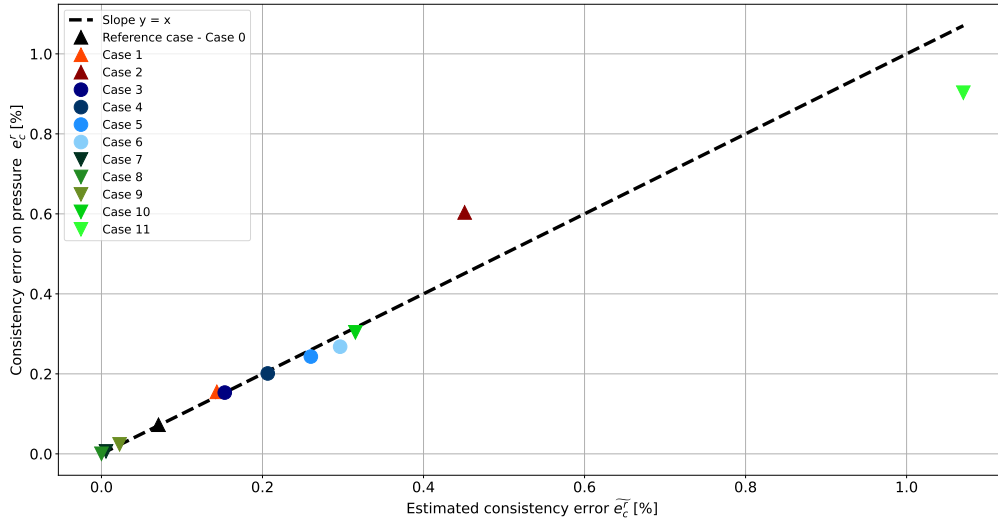


FIGURE 7.43

Consistency error e_c^r as a function of the estimated consistency error $\tilde{e}_c^r$.

The estimated consistency error is used to scan the $(\bar{y}_0, \Delta y_0^{eq})$ plane and compare the obtained consistency error with the industrial mesh error e_{im}^r for a constant velocity $u_0 = 5 \text{ m.s}^{-1}$. The considered cases are taken within the range $(\bar{y}_0, \Delta y_0^{eq}) \in [0.0, 0.5] \times [-0.01, 0.04]$. A grid with

100 values for the mass fraction at equilibrium $\bar{y}_0$ interval and 60 values for the deviation from equilibrium Δy_0^{eq} interval is considered. Only physically valid cases are retained, i.e., cases where the mass fraction is positive and the deviation from equilibrium is low enough for the equation of state to yield a liquid phase with positive temperature and density. The final data set includes 4451 cases for which the ratio $\tilde{e}_r^r / \langle e_{im}^r \rangle$ is computed. Figure 7.44 shows this ratio as a function of the mass fraction at equilibrium and the deviation from equilibrium. The values for the accidental and severe accidental cases from Figures 7.41 and 7.42 are also shown. For accidental and severe accidental conditions, the most penalizing cases were considered here. It is observed that for the accidental case, regardless of the point considered in the core, the consistency error on pressure is negligible. For the severe accidental case, the subcooled boiling zone extends beyond the region where the consistency error is negligible ($\tilde{e}_r^r / \langle e_{im}^r \rangle > 1$). The consistency error is not always negligible in the subcooled zone. A slope break is observable at the center of the domain in Figure 7.44. The approximate consistency error is continuous, but the error due to the use of an industrial mesh shows a discontinuity at that point. This is due to the calculation of the average error over the plateau. With so few cells used for averaging, a sudden variation in the error on the plateau causes a sharp change in the average error. This is the case at the location of this slope break.

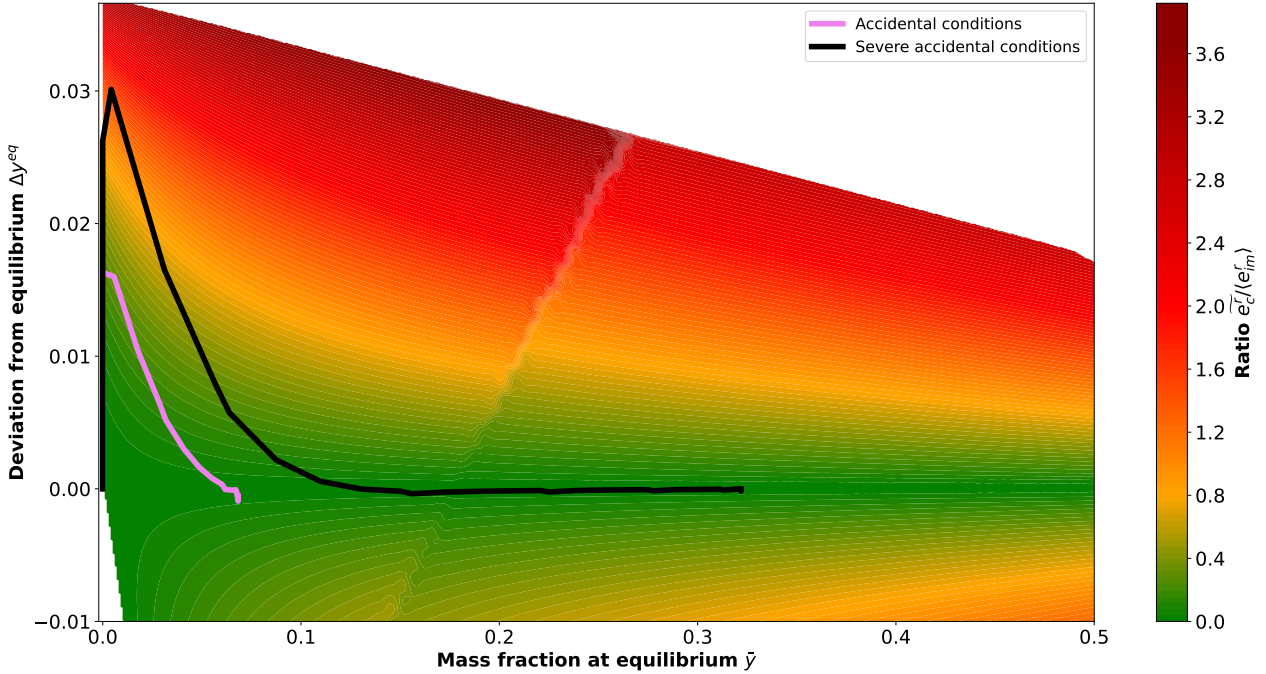


FIGURE 7.44

Ratio $\tilde{e}_c^r / \langle e_{im}^r \rangle$ as a function of deviation from equilibrium Δy_0^{eq} and mass fraction y_0 . The values for the accidental and severe accidental cases of Figures 7.41 and 7.42 are also represented.

7.D Two rarefaction waves solution with a mass fraction jump

In this appendix, the solution to the Riemann problem found in **Appendix 7.A** is used again. The pressure in the left and right states is the same, and the velocities still have the same magnitude but opposite directions:

$$\begin{cases} p_R = p_L = p_0, \\ u_R = -u_L = u_0 > 0. \end{cases} \quad (7.43)$$

However, the left and right states do not have the same initial mass fraction $y_R \neq y_L$. The mass

fraction at equilibrium is initialized uniformly throughout the domain at $t = 0$ such that

$$\bar{y}(p_0, \rho_R, y_R) = \bar{y}(p_0, \rho_L, y_L) = \bar{y}_0, \quad (7.44)$$

which sets the density for state L and R.

The test case is numbered Test Case 12. The boundary conditions for Test Case 12 are shown in Table 7.18. The analytical solution to this Riemann problem includes two rarefaction waves propagating in opposite directions. A double contact wave separates the two intermediate states * and **. These intermediate states share the same velocity u_* and pressure p_* . The contact propagates at velocity u_* . At the contact, a jump in mass fraction and density is observed.

Case	u_0 [$m.s^{-1}$]	p_0 [bar]	$\bar{y}_0$ [-]	y_L [-]	ρ_L [$kg.m^{-3}$]	y_R [-]	ρ_R [$kg.m^{-3}$]
12	5	155	0.01	0.015	691	0.02	769

TABLE 7.18

Initial conditions for the test case of two-rarefaction waves with a mass fraction jump.

Test Case 12 is simulated with pseudo-entropy and with entropy. The numerical solution obtained for $n_x \in \{10^2, 10^3, 10^4\}$ are alongside the analytical solution in Figure 7.45 when entropy is used, and in Figure 7.46 when pseudo-entropy is used. As expected, the analytical solution have two rarefaction waves propagating in opposite directions and a contact wave propagating at a velocity $u_* \approx 0.12 m.s^{-1}$. At this contact wave, a mass fraction jump as well as a density jump are observed. Since the observed time is $t_{comparison} = 0.3 t_{max}$, the rarefaction waves are approximately one-third of the way through the domain. As the velocity u_0 considered is once again low compared to the speed of sound, the rarefaction waves are very steep.

When entropy is used in ThermoTorch, the numerical solution seems to approach the analytical solution as the mesh is refined. A zoom on the plateaus for each variable is represented in Figure 7.47. For the velocity, the numerical solution provides two different values for the velocity plateaus in states * and **, with both plateaus located on either side of the analytical value. However, they both appear to converge toward the analytical value as the mesh is refined. The jumps in density and mass fraction are well captured by the scheme, but the scheme seems to struggle to converge rapidly to the analytical solution. A singularity is also noticeable at the contact in the velocity profile. Since the numerical schemes in ThermoTorch are first-order, the expected order of convergence for this solution is once again $1/2$, due to the contact.

When pseudo-entropy is used, the numerical solution no longer appears to converge to the analytical solution but rather to a different one, with a different pressure, velocity and density plateau. This effect is more visible in Figure 7.48. As the solutions obtained on the plateaus are not the analytical ones, the rarefaction waves are also badly resolved. As expected, a consistency error appears for unsteady solution when using the pseudo-entropy. For this solutions, the two neglected terms of Equation (7.5) (the green term in pressure and the blue term in mass fraction) intervene in the consistency error.

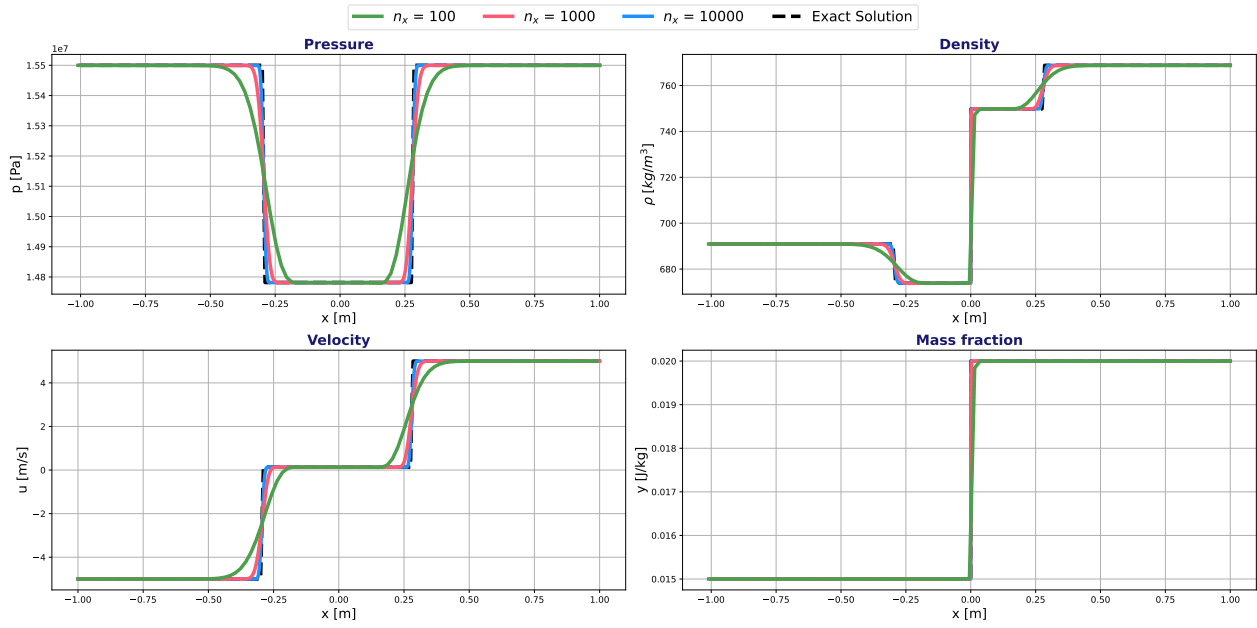


FIGURE 7.45

Test Case 12 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using entropy.

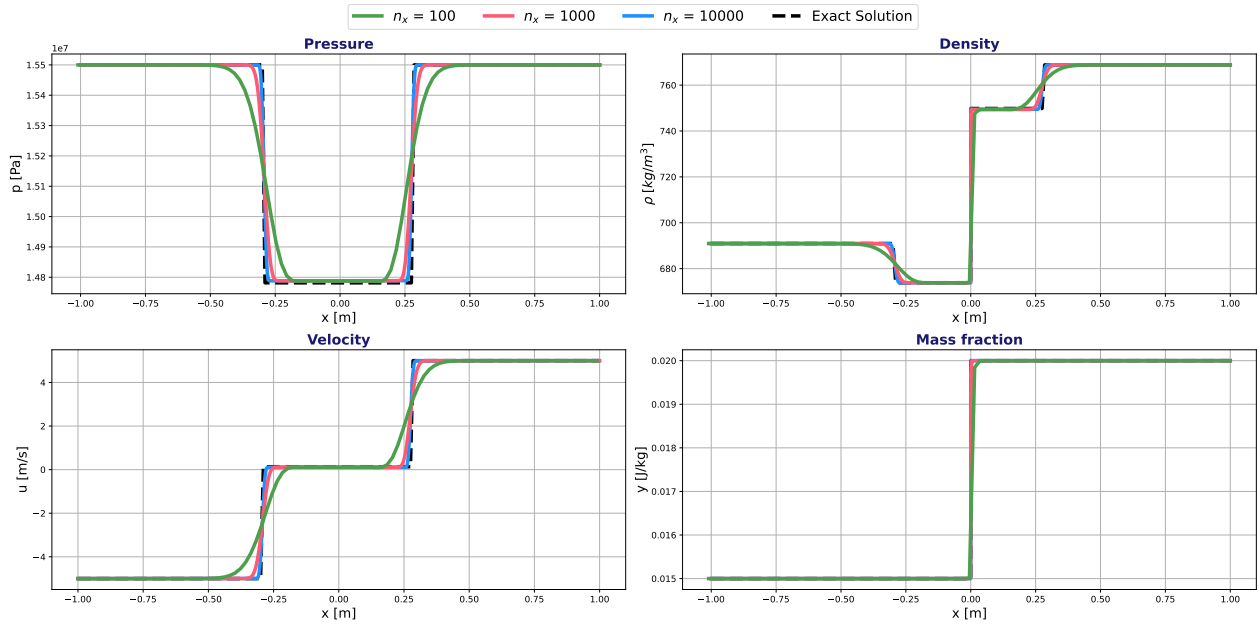


FIGURE 7.46

Test Case 12 - Analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy.

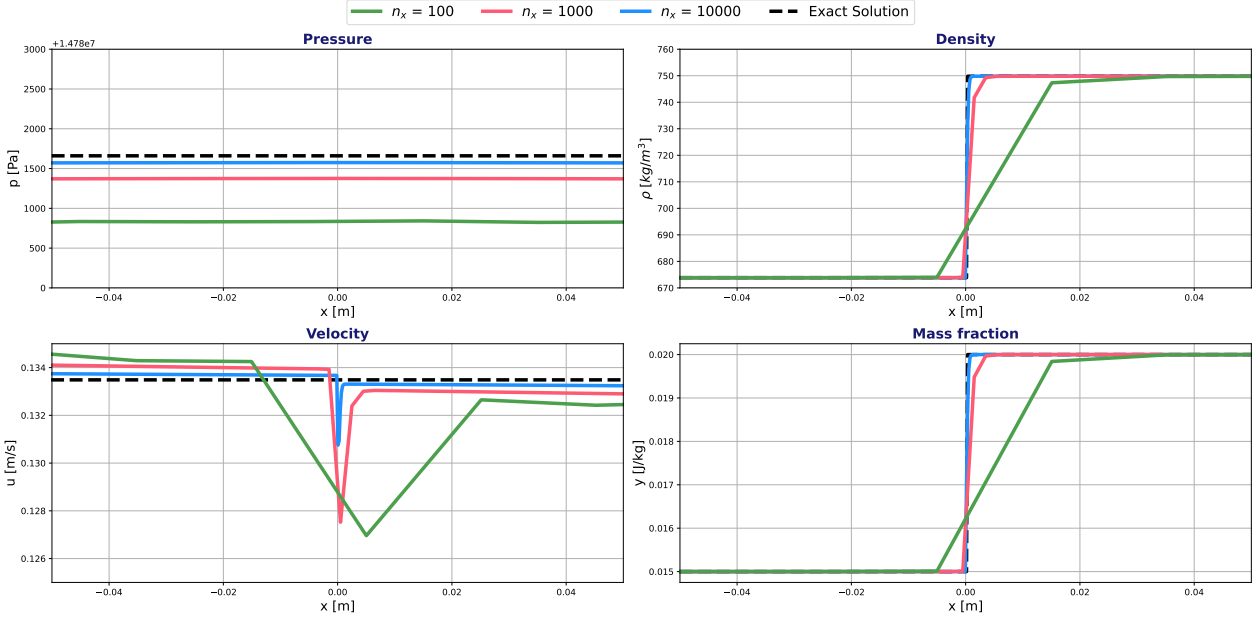


FIGURE 7.47

Test Case 12 - Zoom on analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using entropy.

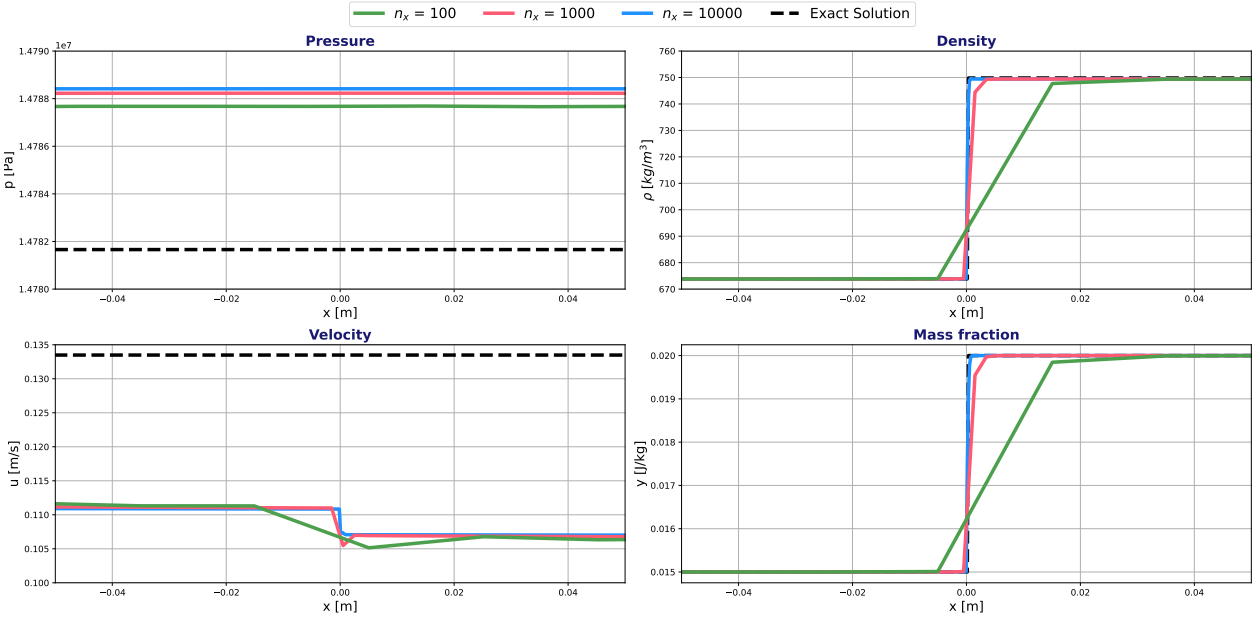


FIGURE 7.48

Test Case 12 - Zoom on analytical and numerical solutions with $n_x \in \{10^2, 10^3, 10^4\}$ when using the pseudo-entropy.

A convergence study is conducted for each method (with entropy and with pseudo-entropy) to confirm the observations made in Figures 7.45 and 7.46. The errors (in L_2 -norm) between the numerical and analytical solutions for the variables (p, ρ, u, y) at time $t_{comparison}$ are shown as a function of the number of cells for $n_x \in [10^2, 10^{4.75}]$ in Figure 7.49 when entropy is used, and for $n_x \in [10^2, 10^{4.5}]$ in Figure 7.50 when pseudo-entropy is used. Table 7.19 (resp. Table 7.20) presents the numerical values for the convergence rates computed when entropy (resp. pseudo-entropy) is considered.

When entropy is used, a convergence rate approaching 1/2 is observed for pressure and velocity.

This is the expected value with first-order spatial schemes for a solution involving a contact wave. For density and mass fraction, the convergence rates fluctuate and appears to increase monotonically over the last two meshes size. The actual convergence rate for mass fraction and density has not yet been reached for the meshes considered. However, it is expected that the convergence rate will stabilize around $1/2$ as the mesh is refined.

When pseudo-entropy is used, the errors for pressure and velocity appear to reach a plateau. The convergence rate in pressure and velocity decreases starting from a mesh of 1000 cells and reaches a plateau for a mesh with 31622 cells. The scheme converges toward the wrong solution, as expected due to the consistency error introduced in the energy equation. The errors due to pressure variation (blue term in Equation (7.5)) and mass fraction variation (green term in Equation (7.5)) accumulate in this case.

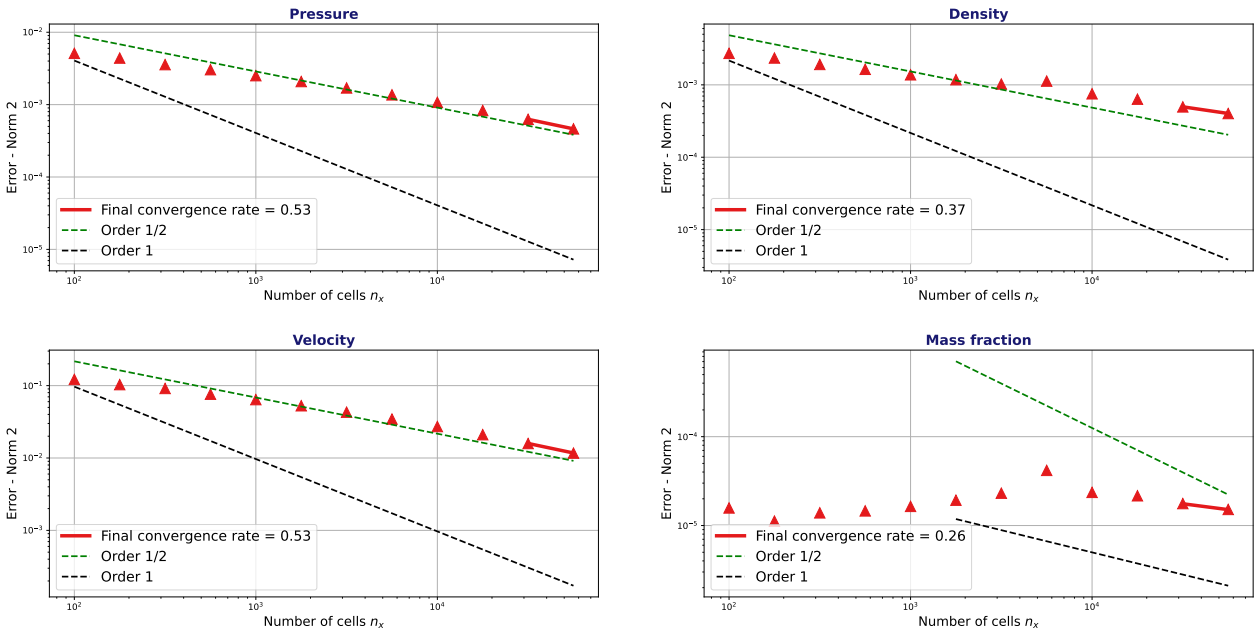
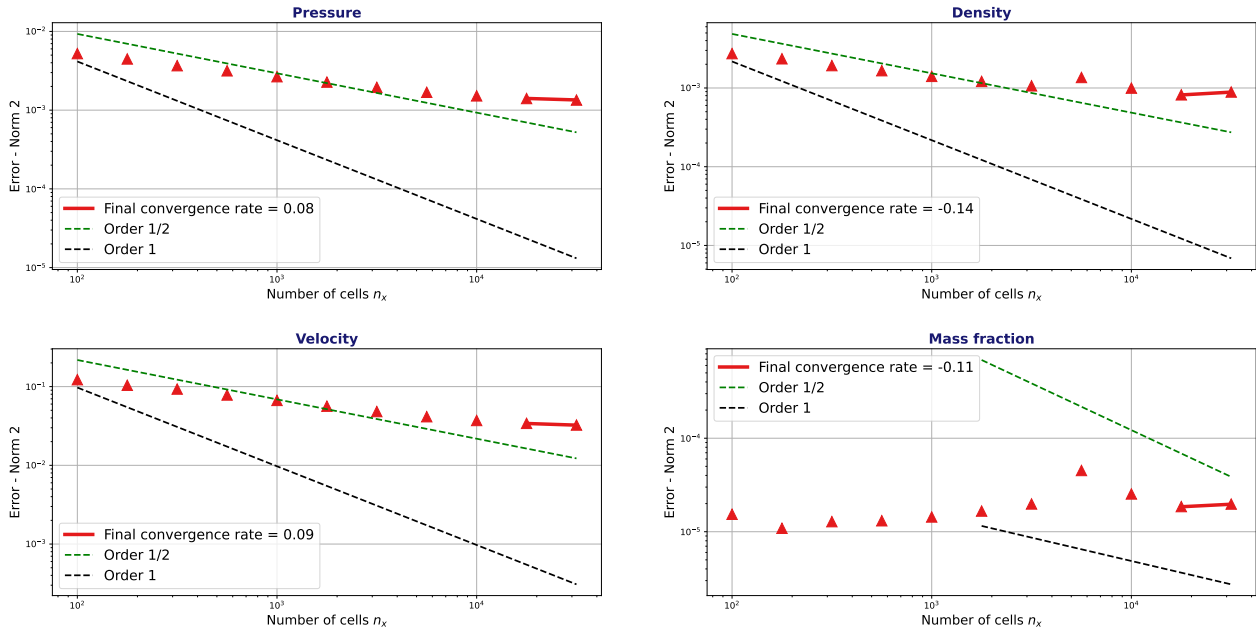


FIGURE 7.49

Test Case 12 - Error (L_2 -norm) as a function of the number of cells n_x for variables (p, u, ρ, y) when the entropy is considered.

**FIGURE 7.50**

Test Case 12 - Error (L_2 -norm) as a function of the number of cells n_x for variables (p, u, ρ, y) when the pseudo-entropy $\mathfrak{s}$ is considered.

Number of cells	Pressure	Velocity	Density	Mass fraction
100				
177	0.26	0.29	0.26	0.60
316	0.36	0.21	0.35	-0.37
562	0.28	0.32	0.27	-0.09
1000	0.33	0.30	0.30	-0.20
1778	0.33	0.34	0.27	-0.28
3162	0.35	0.35	0.24	-0.32
5623	0.38	0.38	-0.16	-1.03
10000	0.41	0.41	0.70	0.99
17782	0.45	0.45	0.30	0.15
31622	0.49	0.49	0.43	0.35
56234	0.53	0.53	0.37	0.26

TABLE 7.19

Test Case 12 - Convergence rate for pressure, velocity, density and mass fraction as a function of the number of cells when the entropy is considered.

Number of cells	Pressure	Velocity	Density	Mass fraction
100				
177	0.27	0.28	0.26	0.60
316	0.34	0.19	0.34	-0.28
562	0.26	0.30	0.26	-0.04
1000	0.30	0.27	0.29	-0.16
1778	0.27	0.29	0.25	-0.25
3162	0.27	0.27	0.23	-0.30
5623	0.25	0.26	-0.42	-1.43
10000	0.18	0.19	0.54	1.00
17782	0.14	0.15	0.34	0.54
31622	0.08	0.08	-0.14	-0.11

TABLE 7.20

Test Case 12 - Convergence rate for pressure, velocity, density and mass fraction as a function of the number of cells when the pseudo-entropy $\mathfrak{s}$ is considered.

Chapter 8

Steady-state convergence acceleration using initialization from a neural network

The development of this method has first appeared in the article [68] published during the PhD. This chapter continues the research presented at NURETH-21 [92] and updated it with the latest results.

This work proposes a Machine Learning-based method to accelerate convergence of THYC-coeur when looking for steady-state solutions. To evaluate the potential of the ML approach in a 1D configuration, the simplified 1D code ThermoTorch [68] is used (see **Chapter 6**). A Deep Neural Network (DNN) is developed to predict the numerical solutions generated by the ThermoTorch code. These predicted solutions are used to initialize the computation, aiming to reduce the number of external iterations. This code simulates water flow in a heated channel with phase change (vaporization). The use of a Neural Network (NN) is particularly justified here, as extensive datasets of results are easily obtained.

Machine Learning is increasingly widespread across various fields. Numerous applications of Machine Learning in fluid mechanics have been developed [20, 138], particularly for improving CFD performance, such as accelerating the resolution of a Poisson problem [1]. When dealing with complex partial differential equation (PDE) systems commonly encountered in this domain, additional information and constraints can be incorporated into standard neural network architectures. One prominent method, known as Physics-Informed Neural Networks (PINNs), integrates physical principles directly into the loss function (the function to be optimized) of the neural network [84, 86, 117, 129]. Another approach involves adapting the structure of the network itself [15, 32] to add physical properties, such as ensuring divergence-free behavior [106], and the conservation of the energy [62] or boundary conditions [90]. Incorporating physical constraints allows for more robust neural networks able to generalize to domains beyond their training datasets.

For the application concerned here, using such complex networks would significantly increase the time of prediction without necessarily providing a major gain in acceleration. Here, the aim is not to supplant the code. The goal is to design and train a predictive NN that is not overly complex, ensuring that its predictions are computationally fast. In this work, standard Multi-Layer Perceptrons (MLPs) [59] are used which take boundary conditions of ThermoTorch as input and try to predict the outputs fields of the code. As the results presented here show, the accuracy of simple DNNs has been proved to be sufficient to significantly accelerate the code and allows for rapid predictions (negligible time compared to the finite volume computation). A preliminary Deep Feedforward Network was proposed in [68], and initial tests showed a 20% acceleration on the

3-equation ThermoTorch model.

Improvements of the initial NN are described here, including normalization, better data generation, and hyperparameter optimization. Additionally, the application is extended to both physical models of ThermoTorch (3-equation and 4-equation). Inspired by PINNs, which incorporate physical constraints into the loss function, a new loss function is proposed to minimize those error in frequencies that slow down convergence.

8.1 Methodology

The methodology is first presented, which involves using predictions from a Neural Network (NN) as initializations of unsteady simulations to accelerate the convergence toward the steady state. Next, the generation process of the different datasets and their normalizations are described. The structure of the NN used is also detailed. Finally, a quick proof of concept is presented, demonstrating that initializing with a solution close to the steady-state effectively reduces the iterations needed for convergence.

8.1.1 Training a neural network to predict steady state solution

In this section, the ThermoTorch 1D code presented in **Chapter 6** is used with the Heated Channel Configuration. The code requires four boundary conditions:

- the liquid inlet mass flux q_{in} ,
- the inlet temperature T_{in} of the liquid flow,
- the volumetric heat input ϕ_0 ,
- the outlet pressure p_{out} .

These four scalars are the input data. At the steady-state, the ThermoTorch provides three (or four) solutions fields on a one dimensional mesh of n_x cells:

- the entropy s ,
- the pressure p ,
- the mass flux q ,
- the mass fraction y (only for the 4-equation model).

The configuration is one dimensional, the mass flux remains constant along the mesh so it does not need to be predicted by the NN. The output data consists of the two (resp. three) arrays of size n_x composed of the pressure, the entropy (resp. the pressure, the entropy and the mass fraction) for the 3-equation model (resp. for the 4-equation model). The number of cells is kept constant for all this work with $n_x = 50$. With this choice, the output data is composed of 100 (resp. 150) scalar values for the 3-equation model (resp. 4-equation model). Convergence is achieved through an unsteady simulation with a constant time step Δt .

The objective is to train a neural network to predict the steady state from the input data (T_{in} , q_{in} , p_{out} , ϕ_0). The predicted fields are denoted as $(\hat{p}, \hat{s}, \hat{y})$. To achieve this, a dataset composed of input data and corresponding steady-state solution fields (output data) must first be generated. During the learning phase, this dataset is used to train the NN. The idea is to use the NN's predictions as initializations of computations to converge more quickly. For this method to be effective, a simple NN must be proposed to quickly predict approximate solution fields that are sufficiently accurate to accelerate the convergence. This methodology is described in Figure 8.1. To evaluate the NN performance, the indicator considered is the relative gain in the number of iterations before

convergence, denoted as g_r . The relative gain compares the number of iterations to converge from constant fields, denoted as N_{cst} , and the number of iterations to converge from the NN predicted fields, denoted as N_{ml} such that

$$g_r = \frac{N_{cst} - N_{ml}}{N_{cst}}. \quad (8.1)$$

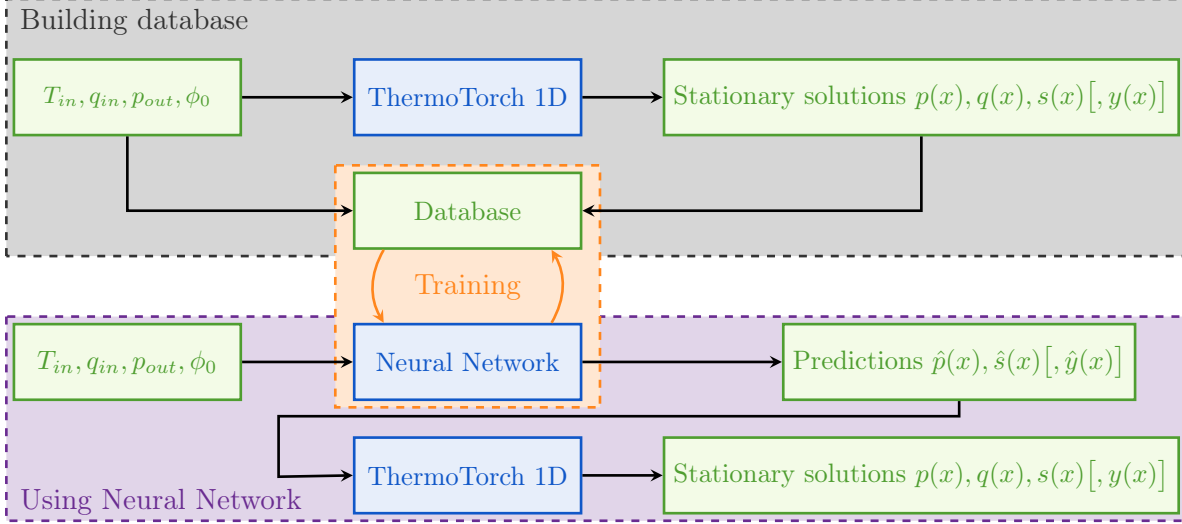


FIGURE 8.1

Construction of the database used for training and acceleration of ThermoTorch 1D with a neural network.

8.1.2 Characteristics of the Neural Network

In this work, Multi-Layer Perceptrons (MLPs) are used, which are NNs with multiple successive layers, each layer being linearly connected to all neurons in the previous layer and then activated with a non-linear function [59]. The input layer represents the input data of ThermoTorch, i.e., four scalars (inlet mass flux, inlet temperature, outlet pressure and volumetric heat input), and the output layer represents the output values, either 100 or 150 scalars depending on the model considered. The layers between these input and output layers are the hidden layers. The number of layers and neurons per layer for the hidden layers will be discussed later. At each layer (except the output), an activation function is applied to introduce non-linearity into the network. This enables the Universal Function Approximation Theorem for Deep Neural Networks (DNNs) [31, 75, 115], which guarantees that a sufficiently large NN can approximate any function. The activation functions used are Rectified Linear Units (ReLU) [48, 49, 56]. This type of activation function is a standard choice for DNNs and has demonstrated excellent performance, despite not being differentiable at 0.

To train the neural network, the defined convergence criterion is the loss function, simply called the loss hereafter. In the first part of this work, a classic Mean Squared Error (MSE) Loss was chosen. In Section 8.3, an evolution of this loss to improve the results is discussed. Using a steady state solution of a field $\mathbf{x}$ and the NN predicted field $\hat{\mathbf{x}}$, the MSE for one field (see the extension to several fields in (8.7)) writes

$$\text{MSE}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 = \frac{1}{n_x} \sum_{i=1}^{n_x} (x_i - \hat{x}_i)^2. \quad (8.2)$$

The fields are normalized before computing the loss (see Section 8.1.3). After initializing the weights of the NN, an iterative process is launched to optimize these weights so that the loss decreases. This is the learning phase, during which, at each iteration called an epoch, the gradient of the

loss is evaluated with a backpropagation algorithm [123] and the gradient descent enables the NN training. The algorithm used for gradient descent is the Adam algorithm [89]. The PyTorch (<https://pytorch.org/>) module was used to build the neural network. During the training, the number of cases given at once to optimize the loss, called the batch size, is 50. Its value is discussed in Section 8.2.3.

8.1.3 Data generation and normalization

To train the NN, a dataset is generated. For this, a range of values for each input was chosen to correspond to the applications of the industrial code THYC-coeur as follows:

$$\begin{cases} p_{out} \in [140, 170] \text{ bar}, \\ q_{in} \in [1000, 5000] \text{ kg.m}^{-2}.\text{s}^{-1}, \\ T_{in} \in [600, \min(618, T_{sat}(p_{out}))] \text{ K}, \\ \phi_0 \in [10^7, \min(5 \cdot 10^8, \phi_{vap}(p_{out}, T_{in}, q_{in}))] \text{ W.m}^{-3}, \end{cases} \quad (8.3)$$

with $T_{sat}(p_{out})$ the saturation temperature so that the inlet flow is a subcooled (or saturated) liquid flow and with ϕ_{vap} the heat input that allows for the complete vaporization of the liquid flow. The dataset of input conditions is generated randomly, while respecting the above conditions. The output fields for each of these state points are obtained through a ThermoTorch 1D computation. This dataset is divided into two parts: the training dataset, which consists of the cases provided to the network for training, and the validation dataset, for which the loss is calculated at each epoch of the training to detect when the network has converged and started to overfit, which can prevent it from generalizing effectively. Additionally, a third dataset, called the test dataset, is generated independently to compute the relative gain to determine the NN performance once trained. The pressure variations are small along the pipe (no pressure losses are considered in this simplified model), i.e. $p_{in} - p_{out} \ll p_{out}$. To learn effectively, the output fields considered are Δs and Δp , such that $\forall i \in [1, n_x]$

$$\begin{cases} \Delta p_i = p_i - p_{out}, \\ \Delta s_i = s_i - s_l(T_{in}, p_{out}). \end{cases} \quad (8.4)$$

For the 4-equation model, the vapor mass fraction is directly given as an output field. The neural network considers each output scalar independently, whether it is a value of entropy, pressure, or vapor mass fraction. To optimize the network's learning, the fields are normalized. Unlike the results presented in [68], which used min-max normalization, the normalization considered here is median normalization [59]. It has been observed that the neural network is more effective with this normalization for the considered application. For a case noted j in the training dataset composed of n_c cases, the variable $\mathbf{x}$ at cell i is normalized as follows

$$\tilde{x}_{i,j} = \frac{x_{i,j} - \langle x_i \rangle_{n_c}}{\sigma_{n_c}(x_i)}, \quad (8.5)$$

where $\langle x_i \rangle_{n_c}$ is the mean value and $\sigma_{n_c}(x_i)$ the standard deviation for the field $\mathbf{x}$ at cell i on the entire training dataset (n_c cases) defined by

$$\begin{cases} \langle x_i \rangle_{n_c} = \frac{1}{n_c} \sum_{j=1}^{n_c} x_{i,j}, \\ \sigma_{n_c}(x_i) = \sqrt{\frac{1}{n_c} \sum_{j=1}^{n_c} |x_{i,j} - \langle x_i \rangle_{n_c}|^2}. \end{cases} \quad (8.6)$$

The neural network developed is sufficiently accurate to ensure valid values for the pressure and entropy, which guarantees the proper functioning of ThermoTorch 1D. When using the 4-equation model, the predicted vapor mass fraction must be filtered after denormalization to ensure $y \in [0, 1]$. It often happens that the prediction provides a negative vapor mass fraction when the steady state value is close to zero.

For the cost function, the considered loss is the sum of the MSE Loss on each normalized field. Noting $\mathbf{V} = (\widetilde{\Delta \mathbf{s}}, \widetilde{\Delta \mathbf{p}}, \widetilde{\mathbf{y}})$ the normalized steady-state solution and $\widehat{\mathbf{V}} = (\widetilde{\Delta \widehat{\mathbf{s}}}, \widetilde{\Delta \widehat{\mathbf{p}}}, \widetilde{\widehat{\mathbf{y}}})$ the normalized prediction, the loss function is

$$\text{Loss}(\mathbf{V}, \widehat{\mathbf{V}}) = \sum_{\mathbf{x} \in \{\widetilde{\Delta \mathbf{s}}, \widetilde{\Delta \mathbf{p}}, \widetilde{\mathbf{y}}\}} \text{MSE}(\mathbf{x}, \widehat{\mathbf{x}}). \quad (8.7)$$

8.1.4 Proof of concept

To evaluate the performance of the method, a proof of concept was carried out with the 3-equation model. A test dataset of 1000 different cases (chosen randomly) is considered and their steady states are determined. After normalizing the fields, a random perturbation of amplitude ϵ is added, representing the error made by the neural network with its prediction. The number of iterations required to return to the steady state is then observed. This process is repeated $n_r = 5$ times for each case to generate perturbations of different forms. The entropy field and the pressure field are perturbed as follows

$$\hat{x}_{i,j,k} = x_{i,j,k} + \epsilon P_{i,j,k}, P_{i,j,k} \in U(-1, 1), i \in [1, n_x], j \in [1, n_c], k \in [1, n_r], \quad (8.8)$$

with $\mathbf{x} \in \{\widetilde{\Delta \mathbf{p}}, \widetilde{\Delta \mathbf{s}}\}$ and $U(-1, 1)$ a continuous uniform distribution between -1 and 1 . On Figure 8.2, the average value of the relative gain (averaged over the number of repetitions n_r and the number of cases n_c in the test dataset) and the associated standard deviation is shown as a function of the amplitude of the applied perturbation (noted ϵ). As expected, when the amplitude of the perturbation decreases, the relative gain increases. Initializing with a solution close to the steady state solution thus effectively accelerates convergence. High accuracy ($\epsilon < 10^{-4}$) must be achieved to reach the maximum gain. For less precise perturbations, the acceleration is still interesting, but the slope is steep, and the standard deviation is large, indicating that the gain can vary over a wide range. Therefore, it can be expected that neural networks will have variable performance depending on the cases considered.

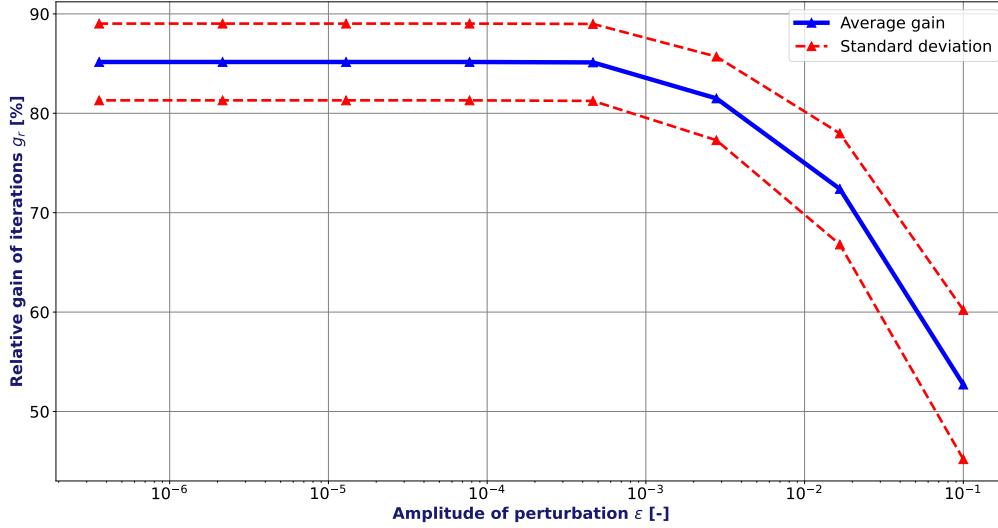


FIGURE 8.2

Proof of concept - Relative gain as a function of the amplitude of the random perturbations - Averaged on 1000 cases.

8.2 Sensitivity analysis and performance obtained with the MSE Loss

In this section, the performance of the method is evaluated using the MSE Loss introduced above. To this end, datasets for training, validation, and testing are generated randomly, respecting the ranges conditions of Equations (8.3). The required size of these datasets is discussed in Section 8.2.1. Once the datasets are defined, the influence of ThermoTorch's numerical parameters on performance is analyzed in Section 8.2.2. Next, a sensitivity study is carried out on various NN parameters, known as hyperparameters, in Section 8.2.3. The results obtained with the optimized NN are then presented in Section 8.2.4 for both the 3-equation and the 4-equation models.

To study the impact of a specific parameter, all other parameters are set to their default values. The NN and ThermoTorch parameters, as well as the dataset sizes, are listed in Table 8.1. This table provides both the default values and the ranges of variation used for the sensitivity analysis. Some parameters remain fixed, in which case no range is specified.

The performance metric for these sensitivity analyses is the average relative gain on a test set of 1000 cases. The relative gain varies significantly depending on the cases and the NN's predictions. As a result, the gain has a high standard deviation, which can make the sensitivity analysis more difficult to interpret.

Parameter	Default Value	Range sensitivity analysis
Datasets size		
Training and validation dataset	10000	100 - 20000
Proportion used for validation	10%	-
Test dataset	1000	10-2000
Hyperparameters of the NN		
Number of epochs	500	1-1000
Batch size	50	2-9000
Neurons per layer	200	10-500
Number of layers	2	1-4
Loss	MSE	-
Optimizer	Adam	-
Initial learning rate	0.001	-
Numerical parameters of ThermoTorch		
Time step (s)	0.05	0.05-0.5
Stopping criterion	10^{-4}	$10^{-2} - 10^{-8}$

TABLE 8.1

Default values and ranges of variation for parameters for the datasets used, the NN and ThermoTorch's numerical parameters.

8.2.1 Size of the training, validation and test datasets

The impact of dataset sizes on performance is studied, both for the test dataset and the training dataset, to evaluate the sizes needed to train and test the NN. Sensitivity analyses are detailed in **Appendix 8.B**. The main results are summarized here.

The size of the training dataset should be sufficient to reach good and stable acceleration. Without enough cases to train on, the neural network is not able to generalize well, and the accuracy of its predictions can be poor. On the other hand, the time required to generate the dataset and to train the neural network increases proportionally with its size. Various sizes of training (and validation) datasets are evaluated, ranging from 100 to 20000 cases (chosen randomly). The results presented in **Appendix 8.B.1** indicate that a dataset size of 10000 cases is sufficient for training, whether for the 3-equation or the 4-equation model. The same base of 10000 cases is used for both models (3-equations and 4-equations) throughout this work to allow performance comparisons. To verify that convergence is reached, 10% of the dataset is used as the validation dataset.

The size of the test dataset is a major factor in evaluating the NN's performance. It must be large enough to ensure that the results obtained are representative of the method's overall performance. However, increasing the test dataset size significantly increases computation time. A method is developed in **Appendix 8.B.2** to test different sizes, ranging from 10 to 2000. A test dataset size of 1000 cases is found to be sufficient, providing performance results within a 2% margin of error. This test base is used to evaluate the performance of all trained networks throughout this study, regardless of the model used.

8.2.2 Impact of the numerical parameters of ThermoTorch on the performance

The number of iterations before convergence varies significantly depending on two numerical parameters in ThermoTorch: the fixed time step Δt used, and the value of the stopping criterion (see **Chapter 6**).

The sensitivity analyses are detailed in **Appendix 8.C**, where the performance of the method is evaluated for different time steps and stopping criteria. These numerical parameters, and especially

the stopping criterion, have a significant impact on the observed gain. Therefore, the performance of the method should be evaluated based on fixed stopping criterion and a fixed time step. To align as closely as possible with industrial applications, a time step of $\Delta t = 0.05$ s and stopping criterion of 10^{-4} (to obtain the same order of number of iterations to reach convergence as THYC-coeur) are selected to evaluate the method's performance.

8.2.3 Optimization of the hyperparameters

The structure of the proposed neural network allows for the adjustment of numerous hyperparameters to optimize the network's performance. A sensitivity analysis was conducted for each parameter listed in Table 8.1 for both the 3-equation and the 4-equation models. In order to identify the individual impact of each hyperparameter on performance, the other hyperparameters are kept fixed at their default values, as shown in Table 8.1. The analyses are detailed in **Appendix 8.D** and the main results are summarized below.

- **Number of epochs:** By monitoring the loss of the validation dataset, it is possible to determine when the neural network is overfitting, as the validation loss starts to increase or stagnate. The application considered here is not particularly prone to overfitting. After testing several numbers of epochs between 1 and 2000, the value of 500 epochs ensures that convergence is reached, i.e. the average gain no longer varies significantly after this epoch, regardless of the model used (3-equation or 4-equation).
- **Number of hidden neural layers and number of neurons per layer:** When the number of hidden layers (and number of neurons per layer) increase, the neural network becomes more precise, but it may also lead to increased overfitting and longer prediction time. Several NNs with two hidden layers have been tested, with the number of neurons per layer ranging from 10 to 500. Starting from 100 neurons per layer, the average gain no longer increases. A configuration with 200 neurons per layer is therefore chosen. Similarly, with 200 neurons per layer, networks with 1, 2, 3, or 4 layers were tested, and the impact on the gain was found to be very low. A network with two hidden layers is retained.
- **Batch size:** The batch size corresponds to the size of the data packets used to optimize the loss. The batch sizes can have an impact on performance. In [88], it is shown that the use of large batch size can lead to poor generalization. Reducing the batch size increases the training time and can lead to oscillations of the loss for the case considered here, unlike large batch sizes, where the loss is smoother. Batch sizes, ranging from 2 to 9 000 have been tested in **Appendix 8.D**. Except for a batch size of 2, the performance does not vary significantly with the batch size for the application considered. The default value used is 50.

We observed that the training process is highly resistant to overfitting for this application. The number of epochs is optimized primarily to minimize computation time while ensuring that training converges as quickly as possible. This is also why dropout [74, 133] and L1/L2 regularization [111] are not implemented.

8.2.4 Acceleration results with optimized neural network using MSE Loss

The optimized neural network was tested to evaluate the average relative gain on the test dataset of 1000 cases. The distribution of relative gains is shown in Figure 8.3, on the left for the 3-equation model and on the right for the 4-equation model. The liquid cases, where the flow remains completely liquid, are separated from the two-phase cases, where the vapor mass fraction is positive. First, the method accelerates every case, regardless of the model. The average gain obtained is $70 \pm 19\%$ for the 3-equation model and $48 \pm 21\%$ for the 4-equation model. The network is slightly less effective for the 4-equation model, which appears to be more difficult to predict, particularly because the vapor mass fraction field must also be predicted.

The histogram of the relative gain on the test dataset is represented in Figure 8.3 for both models. For the 3-equation model (Figure 8.3 - left), many cases are already accelerated almost to the maximum (80% – 100% range). Apart from these cases, most other cases have a relative gain between 30% and 70%. There is no notable difference between the liquid and two-phase cases. For the 4-equation model (Figure 8.3 - right), few cases are accelerated to the maximum. Unlike the 3-equation model, there is a significant difference between the liquid and two-phase cases. The liquid cases are very well accelerated, while most two-phase cases have a relative gain of between 20% and 50%. The difference of performance between the 3-equation and 4-equation models is due to the fourth equation, that slows the convergence. In **Appendix 8.F**, the neural network trained with the 3-equation model is used to initialize the 4-equation model. Despite differences in results between the two models, the same performance are obtained than the ones obtained with the neural network trained on the 4-equation model. Therefore, the handling of the mass fraction in the prediction should be improved in order to achieve similar accelerations observed with the 3-equation model.

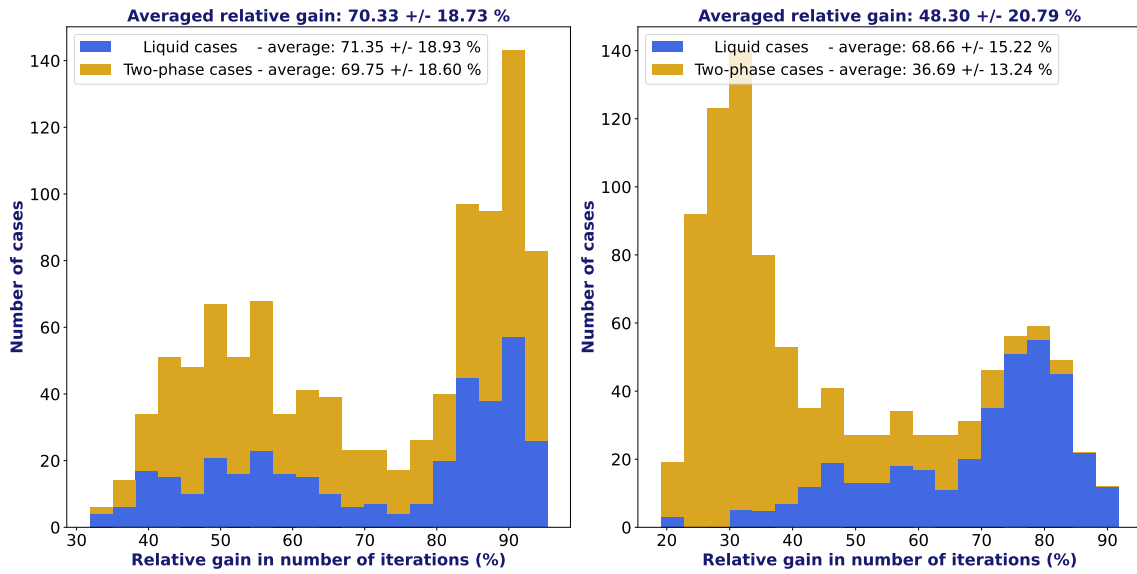


FIGURE 8.3

Histogram of relative gain on the test dataset of 1000 cases - MSE Loss - 3-equation (left) and 4-equation model (right).

8.3 Optimization with regularization on normal modes

In this section, a spatial frequency analysis is performed to study the impact of the shape of the fields on acceleration. A new loss function is proposed and tested.

8.3.1 Correlation between prediction accuracy and acceleration performance

The MSE Loss was a good initial indicator already showing interesting accelerations. The aim is to better understand the mechanisms governing convergence. For each case in the test dataset, the relative gain as a function of the MSE Loss (computed for each variable independently) is plotted, separating the liquid cases from the two-phase cases. Figure 8.4 represents the results (for entropy and pressure) for the 3-equation model, and Figure 8.5 for the 4-equation model (for entropy, pressure, and mass fraction). The points are colored according to the maximum mass fraction reached in the pipe, which allows distinguishing between single-phase and two-phase cases.

The loss in pressure is not related to the gain. Regardless of the case, the network is sufficiently

accurate for pressure, so that the error in pressure is not a limiting factor. For the 4-equation model, pressure accuracy is better for most single-phase cases, and this appears to enable significantly higher gains for these liquid cases.

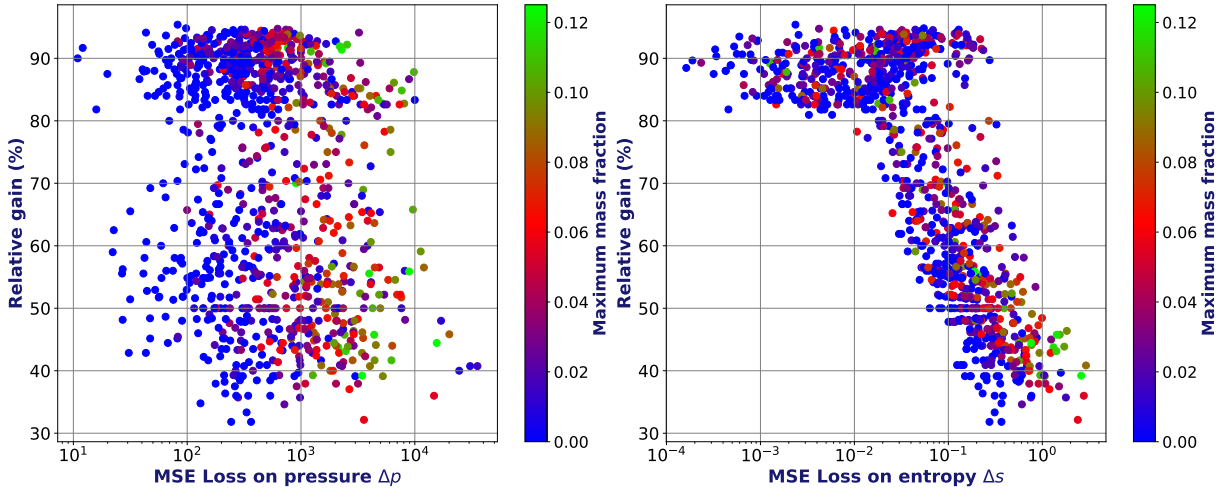


FIGURE 8.4

Relative gain as a function of the MSE loss for pressure (left) and entropy (right) - 3-equation model.

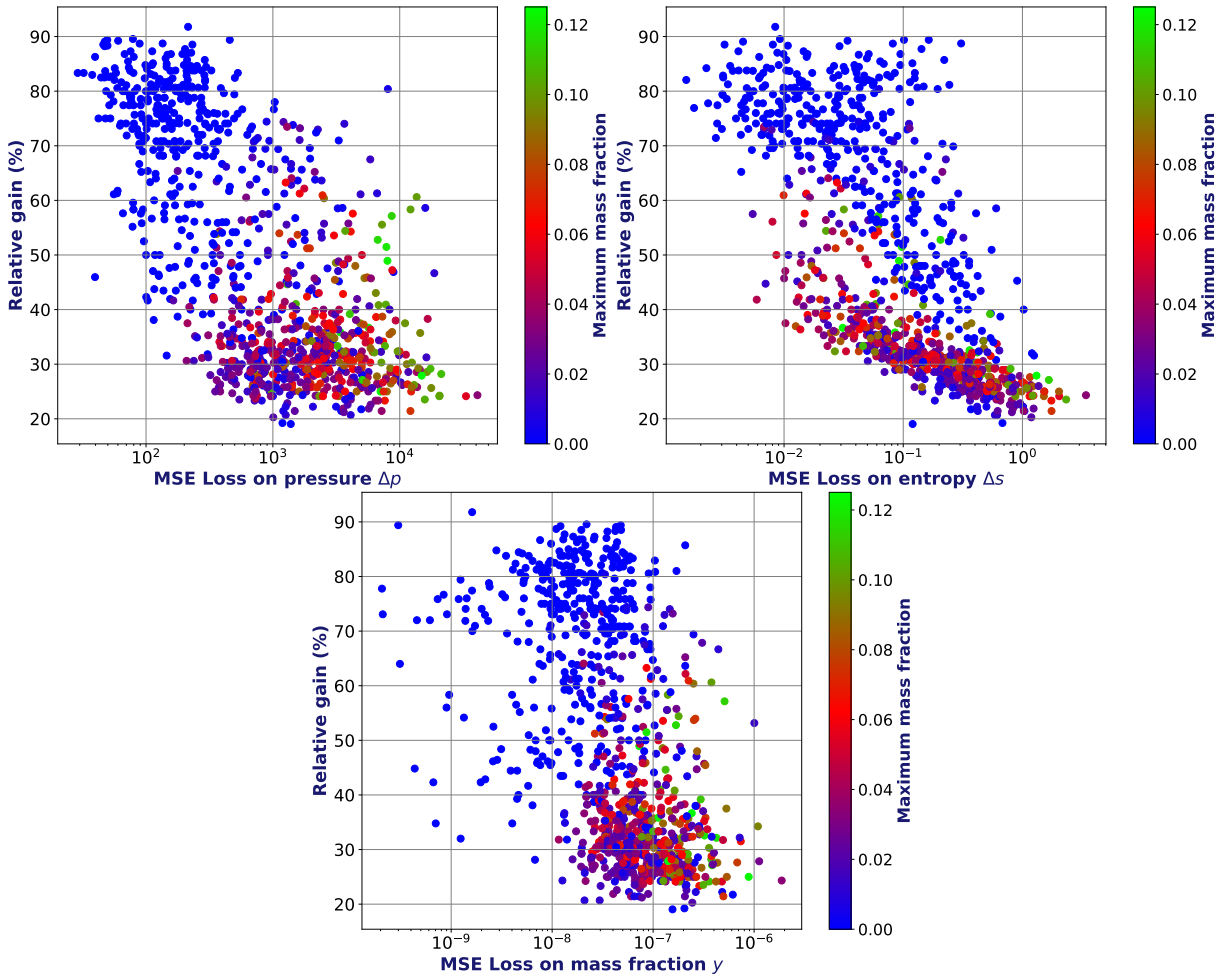


FIGURE 8.5

Relative gain as a function of the loss for pressure (upper left), entropy (upper right) and mass fraction (lower center) - 4-equation model.

For the 3-equation model, entropy is correlated with the relative gain. The error in entropy seems to drive the gain in iterations but the gain in iterations varies greatly, especially for cases where the loss exceeds a value of 10^{-2} . We observe that two cases with the same error in entropy can be accelerated very differently. For example, cases are accelerated from 40% to 90% for the same entropy loss of 10^{-1} . Concerning the neural network's accuracy, acceleration is therefore not driven only by accuracy but by another factor yet to be determined.

For the 4-equation model, the two-phase cases are predicted with much less accuracy. A correlation between the error in entropy and relative gain in iterations is still observed. Unlike the 3-equation model, the behavior differs depending on whether the cases are single-phase or two-phase: the correlation slope is not the same, with the slope for two-phase cases being significantly lower. Therefore, much higher accuracy in entropy is required for two-phase cases to achieve significant acceleration. In all cases, a spread in gains similar to that observed with the 3-equation model appears. The same conclusion applies as for the 3-equation model: another factor besides the accuracy of the solutions seems to influence the gain. There does not appear to be any correlation between the gain and the loss in mass fraction.

In conclusion, entropy appears to be the key to increasing acceleration. To further enhance the network's performance, two issues can be addressed. First, improving the accuracy of entropy enable to move toward the left of the figure on the gain-loss correlation, particularly for the two-phase cases in the 4-equation model. The second issue is the spread of the correlation. As mentioned earlier, the same accuracy can result in very different gains. The goal in the continuation of this work is to understand the origin of this spread in the correlation and try to limit it.

8.3.2 Normal mode analysis of the model

Given that the same loss can give two very different results, the shape of the spatial perturbations is studied. It corresponds to the shape of the difference between the predicted solution and the corresponding steady state. The spatial frequency domain, also called normal modes, is used for analyzing the perturbation. Under steady-state conditions, only one variable at a time - entropy, pressure, or mass fraction - is perturbed using a sinusoidal function with frequency f_p , corresponding to the eigenmodes of the considered mesh, such that

$$f_p = \frac{i_p}{n_x \cdot \Delta x}, i_p \in [0, \frac{n_x}{2}]. \quad (8.9)$$

This perturbation is applied to the non-normalized fields and has an amplitude ϵ corresponding to the characteristic amplitude of the error made by the neural network on the non-normalized fields: $\epsilon_p = 10^3$ for pressure, $\epsilon_s = 10^{-1}$ for entropy, and $\epsilon_y = 10^{-7}$ for mass fraction. For a frequency f_p , the field used as initialization is, $\forall i \in [1, n_x]$,

$$\hat{h}_i = h_i + \epsilon_h \sin(2\pi f_p x_i), \quad \mathbf{h} \in \{\Delta \mathbf{p}, \Delta \mathbf{s}, \mathbf{y}\}. \quad (8.10)$$

The number of iterations required to return to the steady state is studied for each perturbation. Figure 8.6 represents the results obtained for the 3-equation model (left) and for the 4-equation model (right) averaged over the test dataset. The perturbation in pressure has no impact on convergence as cases converge in three iterations, confirming the observation made in Section 8.3.1. A slight increase in iterations is observed for very low-frequency perturbations in pressure, but it remains negligible compared to the other variables. For entropy, the behavior is very different. Low-frequency perturbations in entropy have a very strong impact on convergence, regardless of the model considered. Similarly, in the 4-equation model, low frequencies in mass fraction perturbation force the system to perform several iterations before converging. Moreover, the number of iterations remains high for high-frequency perturbations because the filtering ($y \in [0, 1]$) used before initializing with the mass fraction reintroduces low frequencies. To conclude, given the current accuracy of

the neural network, low frequencies in entropy and mass fraction perturbations must be excluded. Moreover, it is observed that regardless of the frequency, the limit on the number of iterations for entropy and mass fraction remains high (superior to 10 iterations for mass fraction and 8 iterations for entropy). Therefore, achieving high accuracy on these variables is essential.

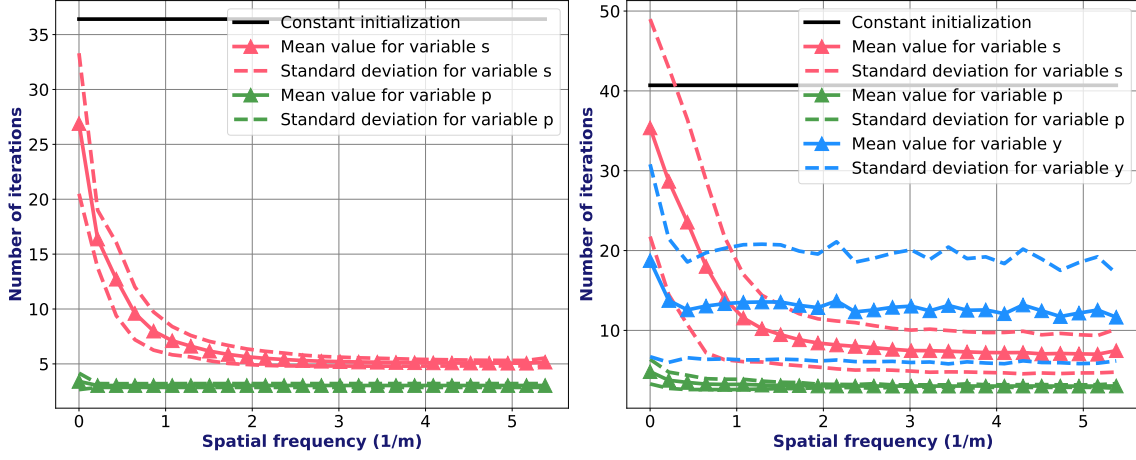


FIGURE 8.6

Number of iterations to reach convergence as a function of the spatial frequency of the perturbation - 3-equation (left) and 4-equation model (right).

The considered system thus struggles to eliminate low-frequency perturbations for entropy (and for mass fraction). The objective is to try to reduce low-frequency perturbations in the predicted solutions. Therefore, the neural network must be trained to predict accurate solutions with as few low-frequency errors as possible for entropy. For the 4-equation model, the mass fraction would also need to be addressed, but this work focuses on handling entropy for now. A new loss function is proposed with an additional term that minimizes low-frequency perturbations in entropy. A cutoff frequency f_c is considered, beyond which perturbation frequencies are no longer taken into account. The cut index is noted i_c such that $i_c = f_c n_x \Delta x$, $i_c \in [0, n_x]$. Noting $\alpha > 0$ as the proportion of the frequency term in the total loss and $\text{DFT}(f)$ as the Discrete spatial Fourier Transform of function f , the MSE+DFT Loss is written, using the same vectors $\mathbf{V}$ and $\hat{\mathbf{V}}$, as in Equation (8.7),

$$\text{Loss}(\mathbf{V}, \hat{\mathbf{V}}) = (1 - \alpha) \sum_{\mathbf{x} \in \{\widetilde{\Delta \mathbf{s}}, \widetilde{\Delta \mathbf{p}}, \widetilde{\mathbf{y}}\}} \text{MSE}(\mathbf{x}, \hat{\mathbf{x}}) + \alpha \times \left\| \text{DFT}(\widetilde{\Delta \mathbf{s}} - \widetilde{\Delta \hat{\mathbf{s}}})[0, i_c] \right\|_2^2. \quad (8.11)$$

8.3.3 Acceleration results with regularization on normal modes

Using the MSE+DFT Loss, different neural networks are studied, varying the parameters $\alpha \in [0.1, 0.9]$ and $i_c \in [1, i_m]$ with $i_m = n_x/2 (= 25)$ as the maximal value for i_c . A detailed sensitivity analysis on the two parameters i_c and α can be found in **Appendix 8.E**. This study shows that for $i_c > 5$ and $\alpha < 0.7$, the MSE+DFT Loss leads to a performance increase of approximately 10%. The performance is analyzed in detail for the parameters $i_c = 25$ and $\alpha = 0.5$. The histograms of the number of cases per relative gain are represented in Figure 8.7, on the left for the 3-equation model and on the right for the 4-equation model. For both models, the average gain increases by more than 10% with the new loss function. For the 3-equation model, almost maximum acceleration is achieved for all cases, with an average gain of $83 \pm 11\%$. Compared to the results in Figure 8.3 for the 3-equation model, most of the cases that were previously in the $[30, 60]\%$ range have now reached maximum acceleration ($> 80\%$). This new NN has achieved an acceleration for the 3-equation model that approaches the maximum reachable value. Indeed, 67.3% of the cases require

fewer than 4 iterations to converge, i.e., are maximally accelerated. Moreover, 90% of the cases converge in fewer than 10 iterations (whereas no case converges in fewer than 10 iterations with constant initialization).

For the 4-equation model, the average gain reaches $61 \pm 18\%$. While the single-phase cases are very well accelerated, the gain for the two-phase cases remains around 50%. Only 4 cases are accelerated to the maximum (fewer than 4 iterations to converge), and 40.6% of the cases converge in fewer than 10 iterations. This represents 98% of the single-phase cases and only 7% of the two-phase cases. Further work can therefore be undertaken to try to accelerate the two-phase cases. The problem may be partly due to the mass fraction. Therefore, it is important to also address the low-frequency perturbations of the mass fraction. However, the final processing that ensures $y \in [0, 1]$ tends to inject low-frequency perturbations in all cases, even if the prediction shows only high-frequency errors. As a consequence, adding a DFT term for mass fraction similar to the one for entropy is not suited for this variable, and a different method for handling mass fraction is required.

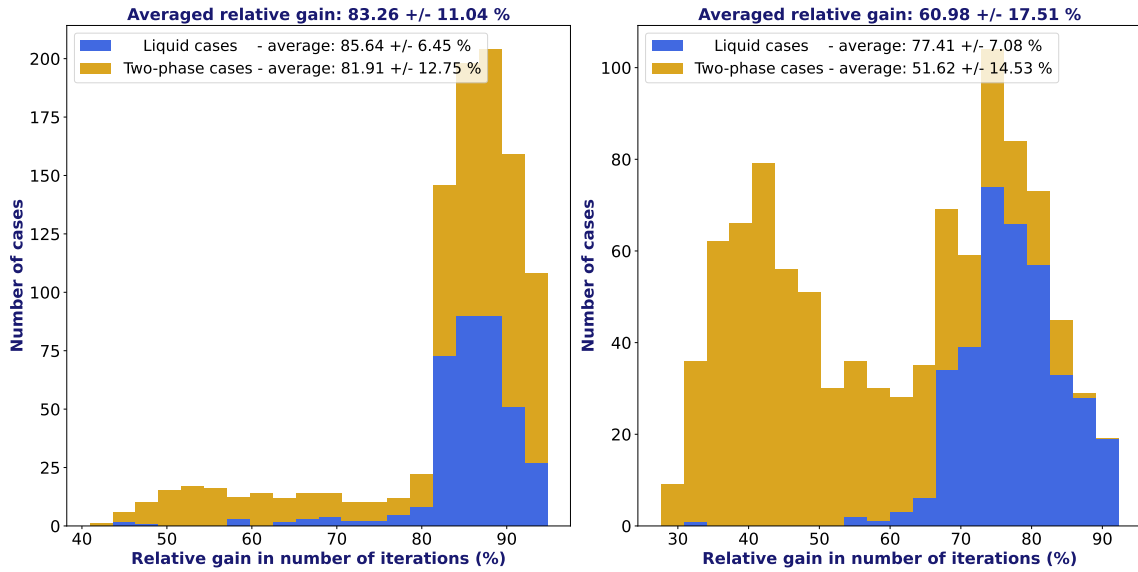


FIGURE 8.7

Histogram of relative gain on the test dataset of 1000 cases - MSE+DFT Loss - 3-equation (left) and 4-equation model (right).

8.4 Conclusion and perspectives

The proposed method with a simple MSE Loss enables an interesting acceleration of ThermoTorch 1D once the network hyperparameters are optimized (about 67% for the 3-equation model and 48% for the 4-equation model on average). Adding low-frequency processing for entropy in the MSE+DFT Loss increases the gain obtained by 10%, regardless of the model. For the 4-equation model, additional processing on the mass fraction is necessary to further increase the gain.

This work reveals the potential performance of the proposed method in accelerating the convergence to steady state of a finite volume thermal hydraulic code. It is now necessary to increase the complexity of the physics (pressure losses, diffusion, turbulence, etc.) and verify if the method remains efficient. The addition of these new terms may have an impact on the frequency behavior of the system of equations. Therefore, the normal mode analysis for each variable, especially pressure, will need to be repeated. Furthermore, in the current model without pressure losses, the pressure does not vary significantly across the domain, allowing for accurate predictions by the neural network. The introduction of pressure losses could complicate the pressure profile and thus affect the performance of the method.

The next important step in this work is the transition to three-dimensional cases. Indeed, the current model is one-dimensional and provides almost constant pressure. The transition to three-dimensional changes this, and the prediction of other fields may also become important. In particular, it will become necessary to predict the mass flux in all three directions, meaning three additional fields that will significantly impact convergence. Furthermore, the mesh size will increase (moving to thousands of cells). Therefore, the neural network will need to be adapted to maintain sufficient accuracy for each field, or multiple independent neural networks could be proposed, each trained for a specific variable. Finally, 3D geometry also complicates the analysis of spatial normal modes. The analysis must be carried out in each direction, applying sinusoidal perturbation separately in one direction at a time.

8.A Implementation

In Figure 8.8, the implementation of the Python model for training and testing neural networks is detailed. Python files are represented in blue. The data is shown in green and framed in navy blue if normalized. Finally, ThermoTorch computations and NN predictions are represented in orange. The DataLoader format from the PyTorch package is used to facilitate training, particularly batch processing. The entire code is written in Python, and the package used for the neural network is exclusively the PyTorch package¹.

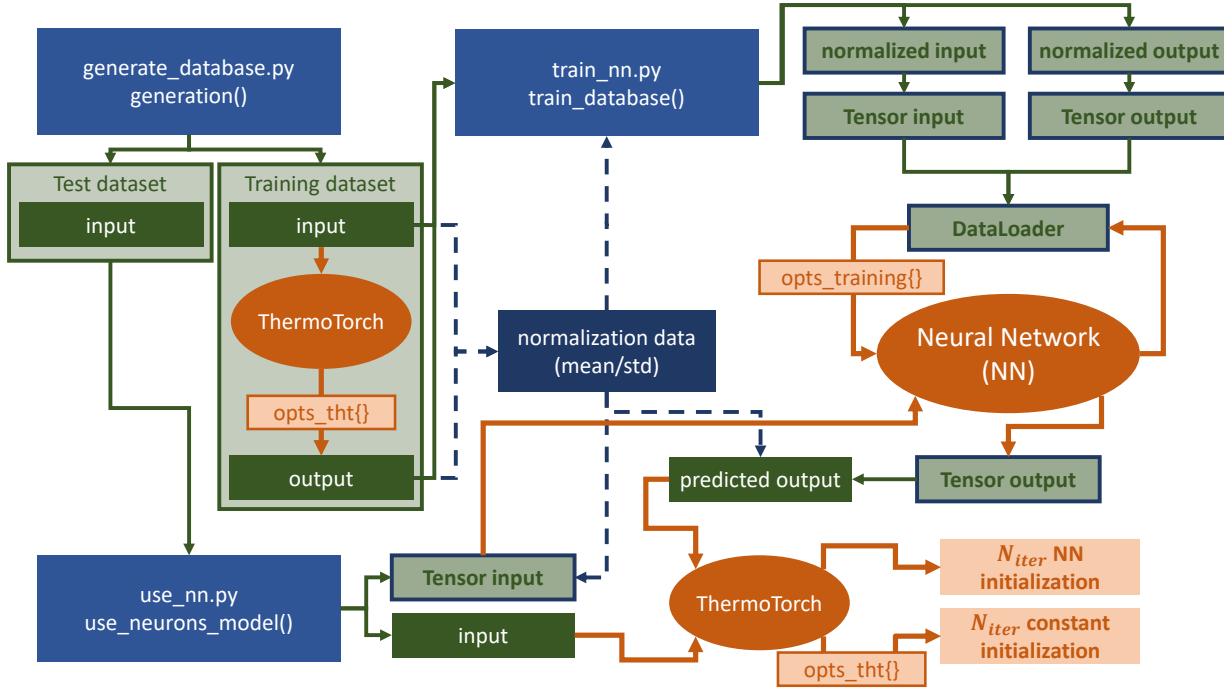


FIGURE 8.8

Implementation of the method for accelerating ThermoTorch using neural networks – Training and usage.

8.B Sensitivity analysis on the size of the datasets

8.B.1 Training dataset size

The performance of the obtained NNs is evaluated on the same dataset of 1000 test cases. Figure 8.9 shows the average gain and standard deviation on this test dataset as a function of the training dataset size, which varies from 100 to 20 000 cases. For small training dataset sizes ($< 2\,500$), the gain is relatively low, as the neural network does not have enough data to perform well. From 5000 cases onward, the gain stabilizes regardless of the equation model. To ensure a sufficiently large dataset for sensitivity studies, a default dataset of 10 000 cases is selected. The same cases (only the boundary conditions) are used for both the 3-equation and 4-equation models to allow for result comparisons.

¹<https://pytorch.org> - V1.13.1/

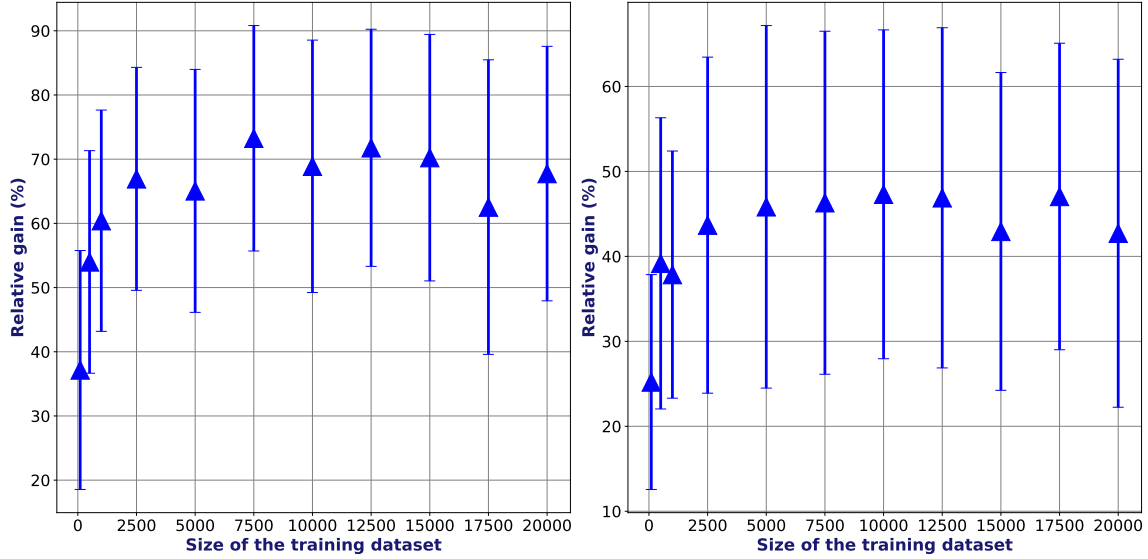


FIGURE 8.9

Relative gain as a function of the size of the training dataset - 3-equation (left) and 4-equation (right) model.

8.B.2 Test dataset size

For the test dataset, the goal is to determine the dataset size at which the average gain becomes representative of the method. A set of 20 000 different cases is randomly generated, independently of the training dataset.

To evaluate a test dataset size N_{test} , N_{test} cases are randomly drawn from this set of 20,000 cases. Ten different draws are performed for each N_{test} value. The average gain (over the entire test set) is then computed for each draw and averaged over all draws. The mean gain, averaged over all test sets and draws, is plotted as a function of the test dataset size in Figure 8.10. The standard deviation (computed over all draws) is also shown. Regardless of the model used, the gain stabilizes with a standard deviation of less than 2% from 500 cases onward. Therefore, the default test dataset size is set to 1 000.

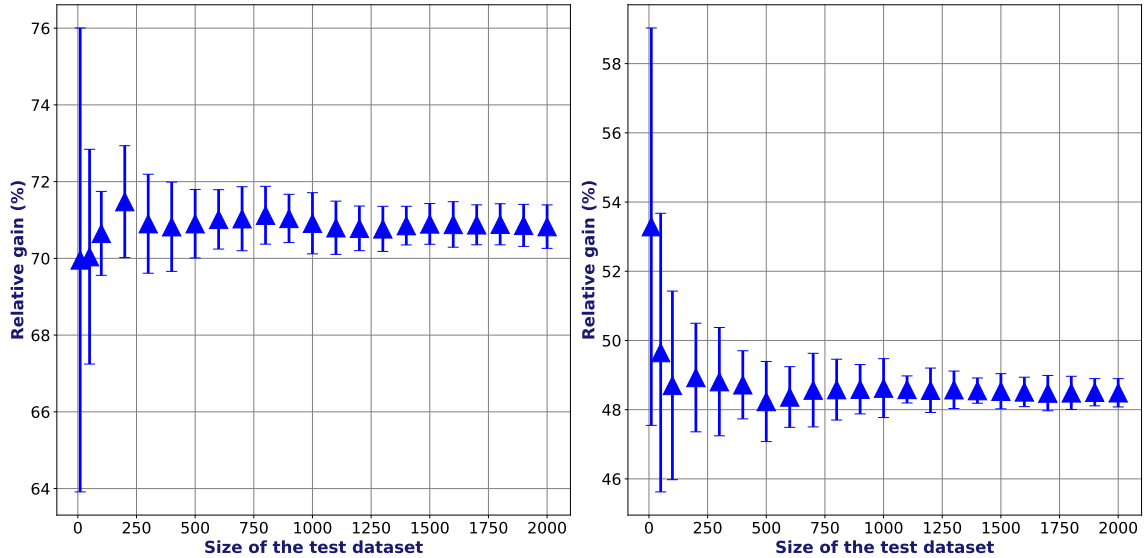


FIGURE 8.10

Relative gain as a function of the size of the test dataset - 3-equation (left) and 4-equation (right) model.

8.C Sensitivity analysis on ThermoTorch numerical parameters

To study the numerical parameters of ThermoTorch, a NN with fixed characteristics, as listed in Table 8.1, is used. Using the trained NN with a dataset of 10 000 cases, the relative gain is evaluated on a test dataset of 1 000 cases (see **Appendix 8.B** for the choice of test and training dataset sizes).

Figure 8.11 shows the average gain and its standard deviation as a function of the time step, while Figure 8.12 illustrates the average gain and its standard deviation as a function of the stopping criterion. In Figure 8.11, as the time step increases, the gain decreases slightly, with the effect being more noticeable for the 3-equation model. In Figure 8.12, the gain increases significantly as the stopping criterion becomes stricter. This is because the NN-based prediction method saves iterations that correspond to the initial iterations in a constant initialization approach. Increasing the stopping criterion greatly reduces the number of iterations before convergence, thereby significantly increasing the relative gain. Thus, the numerical parameters of ThermoTorch have a major impact on the NN's performance. These values should be fixed for all sensitivity studies.

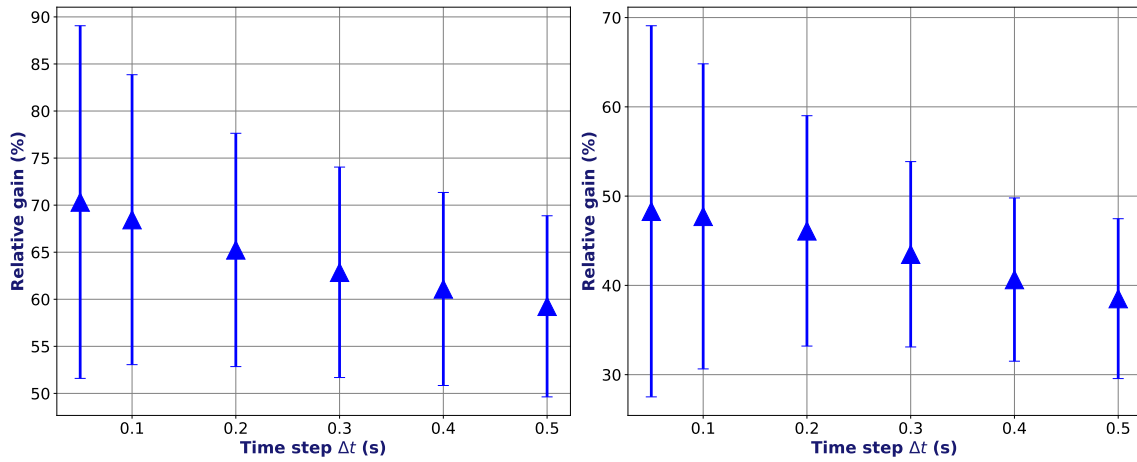


FIGURE 8.11

Relative gain as a function of the time step used in ThermoTorch - 3-equation (left) and 4-equation (right) model.

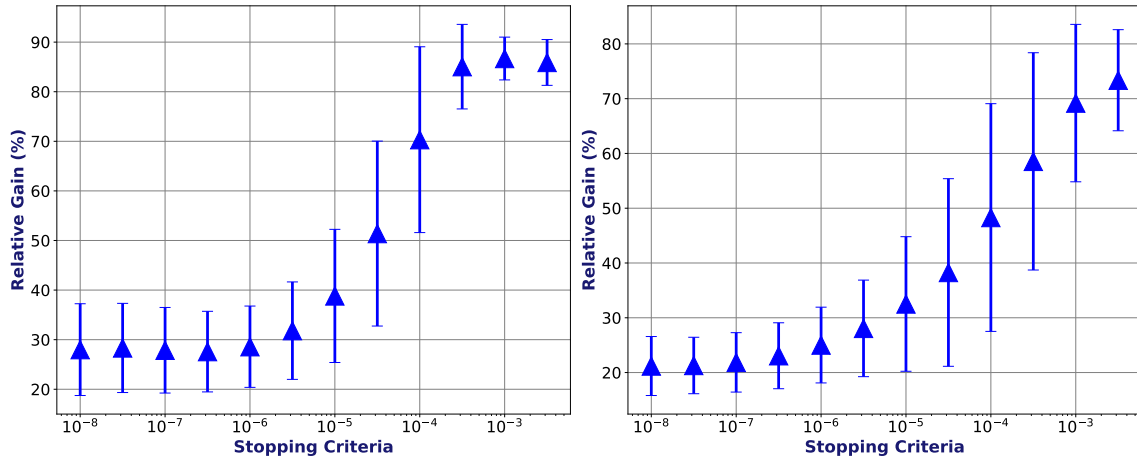


FIGURE 8.12

Relative gain as a function of the stopping criterion used in ThermoTorch - 3-equation (left) and 4-equation (right) model.

8.D Sensitivity analysis on hyperparameters of the NN

To study the individual impact of each hyperparameter on the observed gain, the other parameters are kept fixed. The default values are listed in Table 8.1.

8.D.1 Stopping number of epochs for training

The final number of epochs corresponds to the number of training iterations (gradient descent steps) performed on the entire training dataset. Figure 8.13 illustrates the loss during the training of the 3-equation model's NN as a function of the number of epochs. The training loss is shown in blue, while the validation loss is shown in orange.

During training, gradient descent optimization reduces the training loss. Simultaneously, the validation loss also decreases until it reaches an oscillating plateau at around 200 epochs. Beyond this point, the network no longer learns to generalize but instead overfits the training data. However, the obtained performance is not affected by overfitting because the provided data is not noisy, as it originates from ThermoTorch computational results.

To verify this, the relative gain was evaluated as a function of the stopping number of training epochs and is shown in Figure 8.14 for both models, ranging the number of epochs from 1 to 1000. As expected, the gain reaches a plateau at around 200 epochs and remains nearly unchanged, even at 1000 epochs where the network has overfitted. To ensure convergence across all tests performed, the stopping number of iterations is set to 500 by default, except for the sensitivity analysis for the batch size (see below).

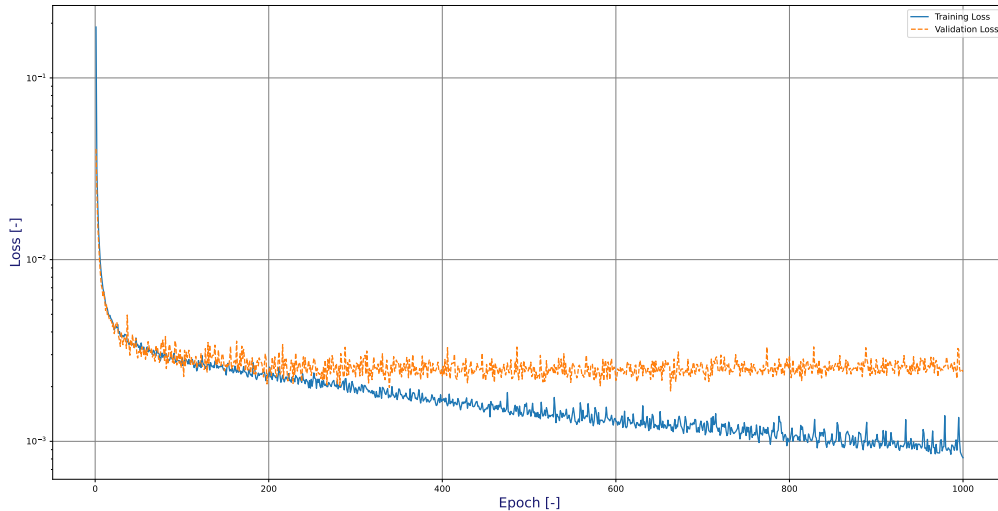


FIGURE 8.13

Training and validation loss as a function of the number of epochs - 3-equation model.

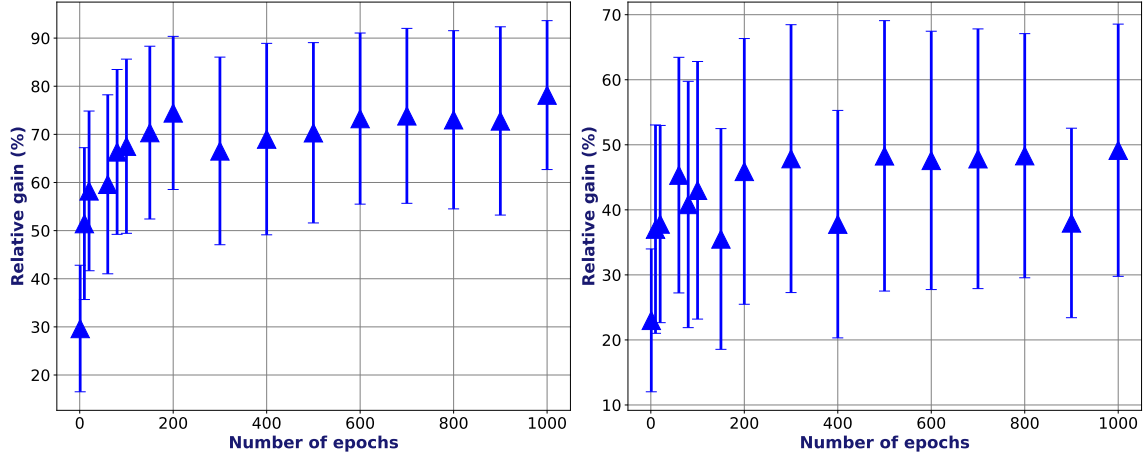


FIGURE 8.14

Relative gain as a function of the stopping number of epochs - 3-equation (left) and 4-equation (right) model.

8.D.2 Batch size

The batch size corresponds to the number of cases fed into the training process at once during gradient descent. In the literature, it is generally recommended to use a relatively small batch size to improve prediction accuracy [88]. Training and validation losses are shown in Figure 8.15 for two different batch sizes: one of 2 and one of 9000. A small batch size allows convergence to be reached more quickly in terms of the number of epochs required (although the computational time can be higher) compared to a large batch size. Convergence is achieved after 150 epochs for a batch size of 2, whereas 2000 epochs are needed for a batch size of 9000. However, with a smaller batch size, the loss exhibits much greater oscillations, which makes automatic convergence detection more difficult. This is not an issue here since no early stopping criterion is used, and a fixed number of epochs is employed.

For batch sizes ranging from 2 to 9000, the performance of the resulting neural networks is shown in Figure 8.16. Increasing the batch size requires a higher number of training epochs. For batch sizes greater than 1000, the maximum number of epochs is increased to 2000. It can be observed that for the smallest batch size of 2, the performance is degraded. Aside from this case, batch size has little impact on the final performance. To maintain a stopping number of epochs of 500, the default value of 50 is retained for batch size.

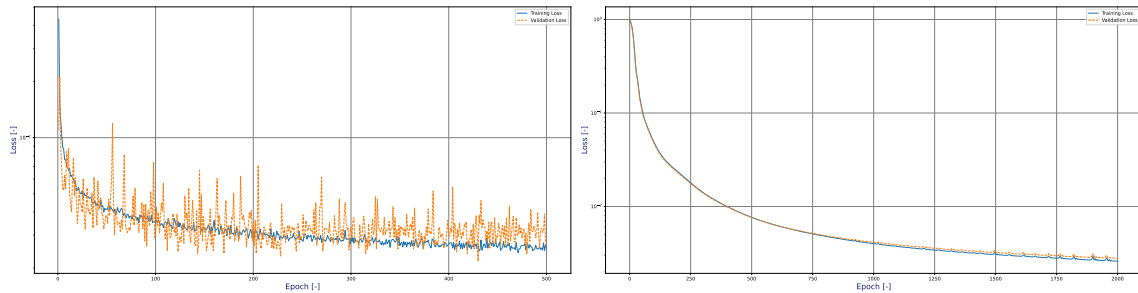


FIGURE 8.15

Training and validation loss as a function of the number of epochs for batch size of 2 (left) and 9000 (right) - 3-equation model.

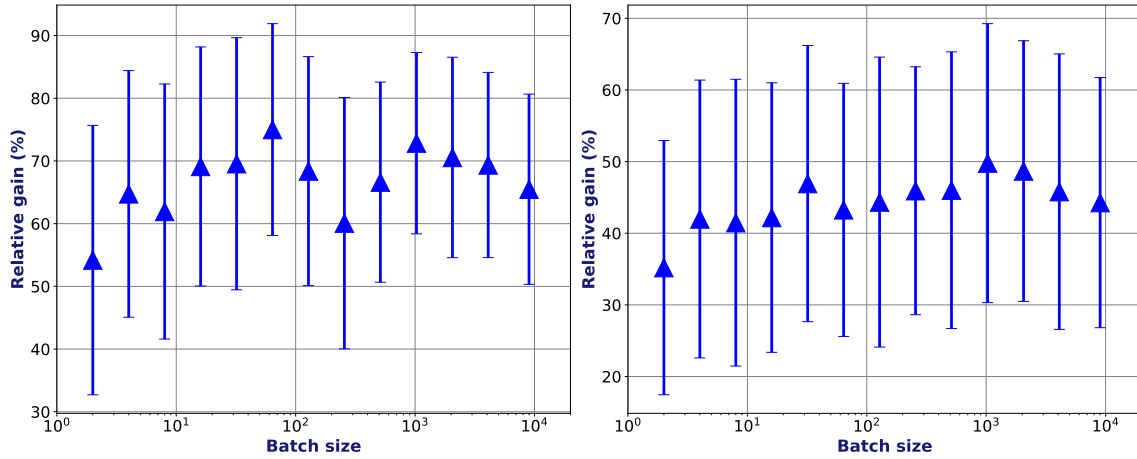


FIGURE 8.16

Relative gain as a function of the batch size - 3-equation (left) and 4-equation (right) model.

8.D.3 Number of neurons per layer and number of layers

The number of neurons and the number of layers are the key structural elements of the network, as they define the number of degrees of freedom to be trained. To limit the number of computations, only specific tests were conducted. First, a two-hidden-layer structure with equal-sized layers was chosen, and the number of neurons per layer varied between 10 and 500. Figure 8.17 shows the relative gain obtained as a function of the number of neurons per layer. From 100 neurons per layer onward, the relative gain remains stable. Increasing the number of neurons further increases training and prediction time without improving performance. For the 4-equation model, the relative gain even tends to decrease with a large number of neurons per layer. The default value selected is 200 neurons per layer. This value depends on the size of the numerical mesh in ThermoTorch, which is kept at 50 cells here.

Once the number of neurons per layer was fixed, an analysis was conducted on the number of layers, ranging from 1 to 4. Figure 8.18 illustrates the relative gain as a function of the number of layers. Performance does not appear to vary significantly with the number of layers, so a two-hidden-layer architecture was chosen as the default.

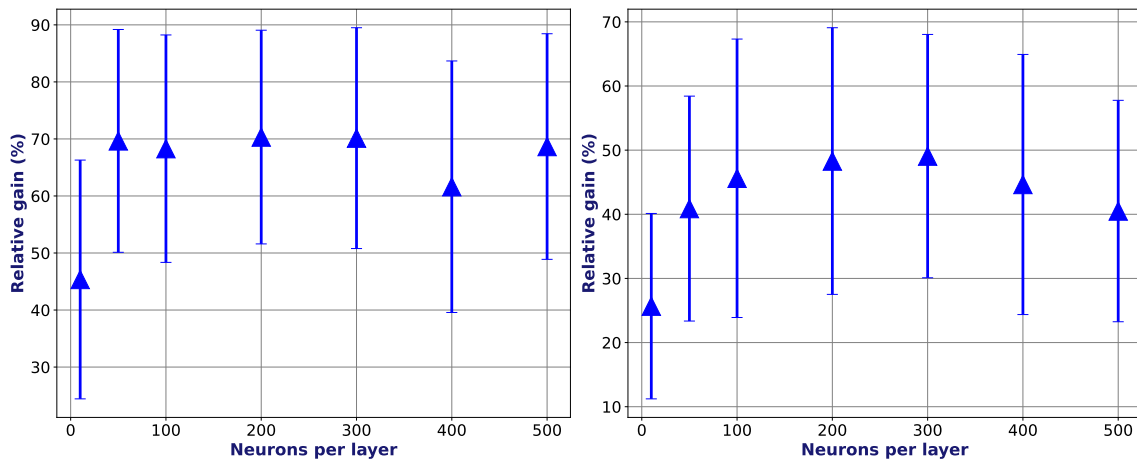


FIGURE 8.17

Relative gain as a function of the number of neurons per layer for NN with two hidden layers - 3-equation (left) and 4-equation (right) model.

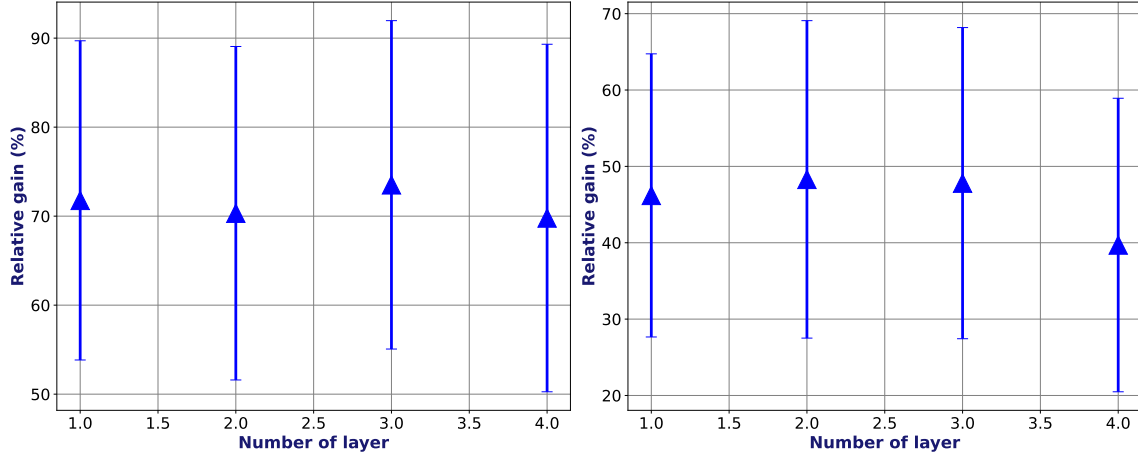


FIGURE 8.18

Relative gain as a function of the number of hidden layers of 200 neurons per layer - 3-equation (left) and 4-equation (right) model.

8.E Sensitivity analysis on the parameters of the MSE+DFT Loss

In the MSE+DFT Loss, two new parameters are introduced: the frequency term proportion $\alpha \in [0, 1]$ and the cutoff index $i_c \in [0, 25]$. A sensitivity analysis was performed on these two parameters for $\alpha \in [0.1, 0.9]$ and $i_c \in [0, 25]$. Figure 8.19 shows the relative gain as a function of the parameter α for different values of the cutoff index, and Figure 8.20 shows the relative gain as a function of the cutoff index i_c for different values of the parameter α . For each figure, the left figure is for the 3-equation model and the right figure for the 4-equation model.

When enough spatial frequencies are considered ($i_c > 5$), neural networks trained with the MSE + DFT Loss become more performant than those trained with the MSE Loss, represented by a dotted black line in Figure 8.20. The treatment of low spatial frequencies for entropy perturbation has a positive impact on the relative gain, as expected. If $i_c > 5$, the gain remains constant for $\alpha < 0.7$. When $\alpha > 0.7$, the spatial frequency term becomes too dominant in the loss. The learning process will optimize the low frequencies of entropy but will no longer predict the pressure and mass fraction accurately. The selected values are $i_c = 25$ and $\alpha = 0.5$. Regardless of the model, adding the spatial frequency term to the loss function results in a performance increase of approximately 10%.

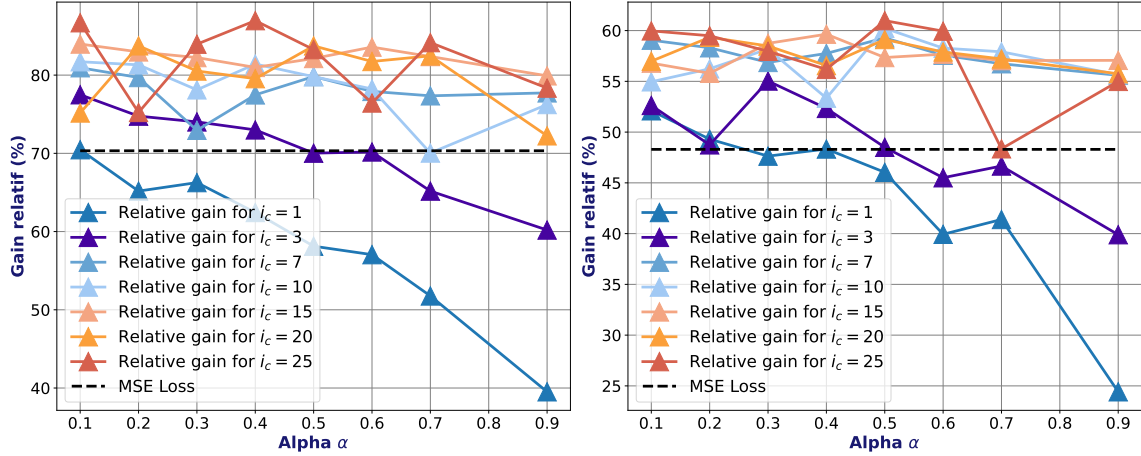


FIGURE 8.19

Relative gain as a function of the parameter α for different values of the cutoff index i_c - MSE+DFT Loss - 3-equation (left) and 4-equation (right) model.

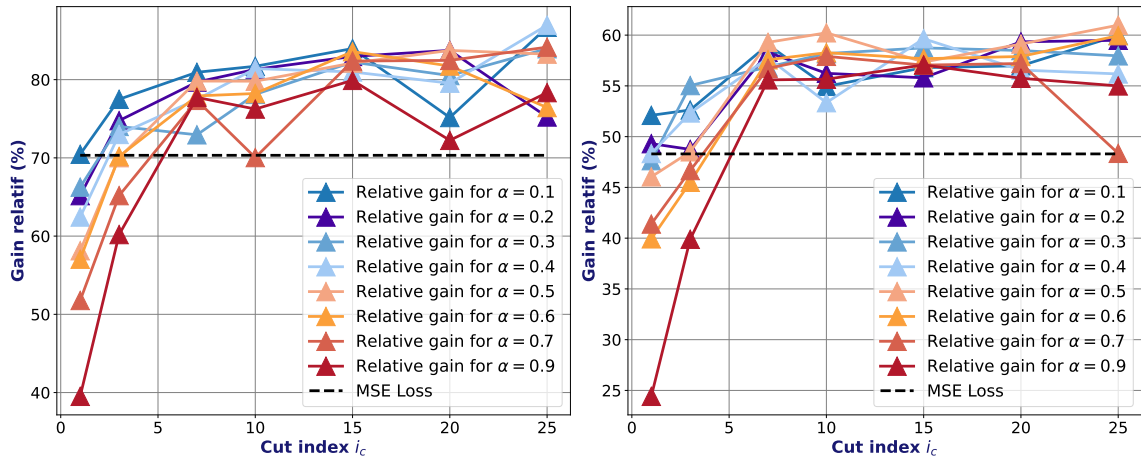


FIGURE 8.20

Relative gain as a function of the cutoff index i_c for different values of the parameter α - MSE+DFT Loss - 3-equation (left) and 4-equation (right) model.

8.F Using the 3-equation NN to predict the 4-equation model

The neural network trained for the 3-equation model is referred to as NN-3eq, and the one trained for the 4-equation model as NN-4eq. The relative gains for each loss function and each model are presented in Table 8.2. The last row shows the relative gain obtained when using NN-3eq to initialize the 4-equation model. In this case, the mass fraction is initialized using the thermodynamic equilibrium value computed from entropy and pressure. With this initialization, the gain obtained is higher than that achieved using NN-4eq, particularly for the MSE Loss. It accelerates the two-phase flow cases, in particular. For two-phase flow, the average gain is 36.7% for NN-4eq and 42.4% for NN-3eq. This result may seem surprising, as NN-3eq predicts a different solution for entropy and mass fraction. Several arguments can explain this outcome.

First, NN-3eq only needs to predict two fields (100 values), whereas NN-4eq must predict three (150 values). NN-3eq is therefore more accurate for entropy, as can be seen by comparing the losses in Figures 8.4 and 8.5. The average entropy MSE Loss is 4.9×10^{-2} for the 3-equation model and 8.4×10^{-2} for the 4-equation model (on the test dataset for the MSE Loss NN). This is even

more pronounced when only the two-phase cases are considered. In that case, the average errors in entropy are 6.3×10^{-2} for NN-3eq and 1.38×10^{-1} for NN-4eq.

However, these average errors are computed by comparing NN-3eq to the exact solution of the 3-equation model, which is not the same as that of the 4-equation model. But the difference between the 3-equation and 4-equation steady states only appears in the subcooled boiling zone, where the liquid is under-saturated and vapor begins to form. In both THYC-coeur and ThermoTorch, the models define this subcooled boiling region quite narrowly. It is rare for the mass fraction to deviate significantly from its equilibrium value. As a result, entropy and mass fraction fields are very similar in both the 3- and 4-equation models. In fact, if the 4-equation model is initialized using the steady-state solution of the 3-equation model computed with ThermoTorch, the simulation converges in an average of 7 iterations (on the test dataset), corresponding to an average gain of 81%.

Finally, the NN-4eq model predicts the mass fraction, which is a particularly challenging field to estimate. Mass fraction variations are typically small and can differ significantly between cases. All cases start with a region of variable size in which the water remains in the liquid phase, and the neural network must predict a strictly zero value there. The prediction is often worse than the thermodynamic equilibrium mass fraction, which is already very close to the non-equilibrium solution. In fact, the predicted field often shows small oscillations near zero in the single-phase region that must be damped out by the solver during convergence.

All of these reasons help explain the improved performance obtained by using the NN-3eq network. This indicates that further work is needed on mass fraction prediction for NN-4eq to become more effective. One idea, for instance, to reduce oscillations in the mass fraction within the single-phase region, would be to train the network to also predict the boiling onset location and enforce a strictly zero mass fraction up to the beginning of boiling.

Initialization	MSE (%)	MSE+DFT (%)	Gain due to DFT (%)
3-equation model			
NN-3eq	70 ± 19	83 ± 11	+13
4-equation model			
NN-4eq	48 ± 21	61 ± 18	+13
NN-3eq	53 ± 23	64 ± 21	+11

TABLE 8.2

Relative gain as a function of the used initialization - MSE and MSE+DFT Loss.

Conclusion and perspectives

Conclusion

The objective of this PhD thesis was to accelerate the THYC-coeur code [7], a component code used to simulate two-phase flows in the reactor cores of Pressurized Water Reactors. Compared to most of two-phase codes, the specificities of THYC-COEUR model are :

- The use of a fully 3D porous model approach. This means that the solid is integrated into the fluid domain through porosity, which calculates the fraction of liquid in a control volume. The exact geometry of the solid is disregarded, and macroscopic interaction terms with the solid must be modeled.
- The use of a drift-flux model for relative velocity between liquid and vapor [81, 82]. The drift-flux correlation for the relative velocity [145] accounts for buoyancy of vapor structures and for the non-uniform distribution of mixture velocity and vapor void fraction.
- The use of a fourth equation for the mass of the vapor phase, to better model subcooled boiling.

The numerical scheme of the code is based on staggered meshes and an upwind convection scheme, with a decoupling of the energy equation. To achieve acceleration of the code, the work of the PhD thesis was divided into three main parts.

Part I - Two-phase flow model

- The first chapter of the thesis has focused on studying the drift-flux models and the various correlations used to evaluate the relative velocity in the THYC-coeur code: the Bestion correlation [14] and the Chexal-Lellouche correlation [22].
- The relative velocity and the fourth disequilibrium equation both introduce significant complexity to the model. **Chapter 2** attempts to provide initial insights into the study of hyperbolicity of the 4-equation drift-flux model. Only the case of a constant relative velocity could be studied. In this simplified framework, a sufficient condition for the model to be hyperbolic was determined. This result relies on the intermediate value theorem, and the eigenvalues of the system could not be obtained analytically. Increasing the complexity of the problem by considering a variable relative velocity appears to be challenging.
- In the final chapter of this first part, some analytical solutions are provided for models derived from the 4-equation and 3-equation models. By considering simple configurations, these analytical solutions enable the verification of codes in the second and third parts of this thesis. This chapter has provided two new self-similar solutions for the unsteady 3-equation model [70]. The first one is a porous medium solution without heat exchange and the second one is a solution with heat exchange in a free medium.

Part II - THYC-coeur with relative velocity: After studying the drift-flux models to better understand their behavior, the second part of the thesis proposes a first acceleration method by analyzing the finite volume schemes used in the THYC-coeur. It focuses specifically on the numerical schemes for the fourth equation of the model, which concerns the mass fraction of vapor. This work on the schemes was divided into two phases.

- In a first phase, three new finite volume schemes for the fourth equation have been proposed [93]: the QRd scheme, the QRq scheme, and the QG scheme. After studying the maximum principle on the continuous equation to ensure $y \in [0, 1]$, the discrete maximum principle has been analyzed for the three new schemes. The QRd and QRq schemes have no constraint on the time step but the QRq scheme can only be used for co-current flows. The last scheme, called QG Scheme, presents a constraint on the time step, which is not limiting for nuclear reactor core applications. All three schemes are linear and first-order in space schemes. The convergence rate was confirmed by a convergence study on a prototype isolating the disequilibrium equation. The QG and QRq schemes appear slightly more accurate for the intended applications. The QG scheme has been also verified for unsteady simulations and a convergence rate of one in space and time was obtained. The QRd and QRq schemes are not conservative in unsteady conditions. Therefore, they cannot be used for transient computations. The three schemes are particularly interesting because they offer a discrete equation for the mass fraction that is independent of the other variables during time stepping. This allows the equation to be solved independently, thereby saving computational time. It is recommended to use the QG scheme by default.
- In the second phase of the work on the disequilibrium equation schemes, the three newly proposed schemes were implemented in the industrial THYC-coeur code. These schemes are compatible with an unstructured staggered grid mesh, and thus can be implemented in THYC-coeur, which uses a Cartesian staggered grid. The three schemes were tested, and their results were compared with those obtained using the default THYC-coeur scheme.

The Bias Curve Database (a database of 36148 representative cases of normal and accidental reactor core flows) was used to evaluate the behavior of two different models with the scheme. These two models differ by the correlation used for the relative velocity: one uses the Bestion correlation [14], and the other the Chexal-Lellouche correlation [22]. Both correlations are used industrially in THYC-coeur. It was shown that the new schemes improve the robustness of the code. For the model using Chexal-Lellouche, a computational time gain of over 50% was also observed compared to the reference scheme. For the Bestion correlation, performance was similar between the new schemes and the reference scheme.

This work on the disequilibrium equation scheme led to the proposal of a summary table in the conclusion of **Chapter 5**, which recommends the scheme to use, depending on the simulation (steady or unsteady) and the model considered (Bestion or Chexal-Lellouche).

Part III - ThermoTorch 1D without relative velocity: In this final part of the thesis work, a 1D prototype called ThermoTorch is presented. It reproduces part of the behavior and numerical schemes of THYC-coeur with a simplified model. This code, developed in collaboration with the thesis director in [68], is a tool that allows for testing new methods and precisely analyzing results before moving on to the industrial code and potentially time-consuming developments. The physical model is simplified, in particular by considering zero relative velocity. It is implemented in a one-dimensional framework on a staggered grid mesh. Both the 3-equation and 4-equation models have been developed. The code has been used in **Chapter 7** for the verification of the numerical schemes implemented in both steady and unsteady conditions; and in **Chapter 8** to evaluate the performance of a method of initialization by a neural network to accelerate the convergence toward the steady-state solution.

- **Verification of the numerical schemes:** Several solutions from **Chapter 3** were implemented and tested using ThermoTorch. This confirmed first a convergence rate of one in space for steady-state solutions. In a second phase, unsteady solutions were used to evaluate the impact of the consistency error in ThermoTorch scheme for transient regimes due to an approximation in the pseudo-entropy linearization.

These difficulties are linked to the fact that the entropy of the drift-flux 4-equation model cannot be obtained analytically for any equation of state, even with zero relative velocity. A pseudo-entropy, along with an approximation in its linearization, was therefore used in the numerical schemes to decouple the energy equation from the coupled momentum-pressure system. This element is key to achieving an efficient code for steady-state solutions. The consistency error does not exist in the steady regime.

The thesis work showed that this error increases as the flow deviates from thermodynamic equilibrium. Moreover, it demonstrated that the error introduced is negligible compared to the error caused by using an industrial mesh for the majority of conditions encountered in reactor core flows.

- **Steady-state convergence acceleration using initialization from a neural network:** The ThermoTorch code was used to evaluate an acceleration method based on machine learning. A neural network was trained to predict the steady-state solution of the ThermoTorch code, and the network's predictions were used as initializations for ThermoTorch computations. This method is particularly appealing because it combines the speed of neural network evaluation with the physical reliability of a validated code. Moreover, the method can be implemented without requiring intrusive modifications to the code.

Very promising results were obtained with a first network, showing a 50% – 70% reduction in the number of iterations to reach convergence (depending on the physical model). A study of the code's behavior with respect to the spatial frequency of perturbations in the solution fields revealed that low-frequency perturbations in the entropy prevent the code to converge quickly. It led to the design of a second network incorporating a discrete Fourier transform term on entropy in the loss function to reduce these types of perturbations, which further improved the method's performance by an additional 10%. This preliminary study validated the approach, which now needs to be tested on the industrial THYC-coeur code.

Summary: The thesis work contributed to a deeper understanding of the drift-flux model and proposed several acceleration methods. The first method, focused on numerical schemes, have been developed right through the implementation and testing within the industrial THYC-coeur code. The second method, based on AI-driven initialization, remains at the prototype stage. However, the preliminary results obtained with the ThermoTorch code are particularly promising and pave the way for further investigation using the industrial code. Finally, the verification of numerical schemes with ThermoTorch provided valuable insights into the use of pseudo-entropy and its impact on unsteady solutions. This helped to clarify the limitations and behavior of the scheme in transient regimes.

Perspectives

The work in this thesis has answered many questions, but raises many more. Some leads have been fully explored, such as the finite volume schemes developed for the fourth equation. For other directions of research, several perspectives are proposed here.

- **Work on hyperbolicity:** The work on the hyperbolicity of the drift-flux model proposed in the thesis is unfortunately only partial, stopping at the case of a constant relative velocity. It

would be interesting to improve our understanding of the sufficient condition of hyperbolicity obtained, but also to possibly push the work even further by working with a variable relative velocity, starting with a relative velocity depending only on the mass fraction (like the approximated Bestion correlation proposed in the appendix of **Chapter 1**). This might need to no longer consider a general equation of state, but to tackle the problem with increasing difficulty, starting for example with a perfect gas mixture.

- **Analytical solutions:** In **Chapter 3**, two self-similar unsteady solutions have been proposed for the 3-equation model: one with a variable porosity and the other in a free medium with a heat source term. It would be interesting to implement them for ThermoTorch in order to further verify the numerical schemes.

Moreover, all the proposed analytical solutions have zero relative velocity. Work could be carried out to find a configuration for which an analytical solution exists. This would enable verification of the numerical schemes implemented in THYC-coeur for terms containing relative velocity.

- **Verification of ThermoTorch and scheme for the energy equation:** The thesis work concluded that the use of pseudo-entropy introduces a consistency error for unsteady solutions. It would be interesting to test alternatives to this pseudo-entropy scheme that would not introduce a consistency error, as proposed in the conclusion of **Chapter 7**. This will provide a scheme that can be used under all conditions for unsteady solutions. For these new proposed methods, however, it will be necessary to evaluate the performance obtained for the search of steady-state solutions.
- **Acceleration by Machine Learning:** The initial results on ThermoTorch are very promising. The next step is transitioning to the industrial THYC-coeur code. It presents several challenges:
 - Increased physical complexity: THYC-coeur includes numerous closure laws, making the physical model more difficult to predict accurately with a neural network.
 - 3D implementation: moving from one-dimensional to three-dimensional flows could significantly impact the method's performance. The spatial mode analysis performed in one-dimensional configuration must be extended to all three directions to understand how perturbations behave in the numerical model depending on orientation.
 - Mass fraction prediction: the method could be further improved by refining how the mass fraction is handled in the 4-equation model.

Despite these challenges, the method shows strong potential. It can be generalized to any code computing steady-state solutions on a fixed mesh and could be a valuable tool for broader industrial use.

Résumé en français

A Introduction

A.1 Fonctionnement d'un réacteur à eau pressurisée

En 2023, l'énergie nucléaire a permis de produire 320,4 TWh d'électricité en France, représentant 65,3% de la production totale d'électricité. En 2025, Électricité de France (EDF) exploite 57 réacteurs nucléaires. Ces réacteurs ont des puissances différentes : 32 réacteurs ont une puissance électrique de 900 MWe, 20 réacteurs de 1300 MWe, 4 réacteurs de type N4 ont une puissance de 1450 MWe, et un réacteur EPR a une puissance de 1650 MWe. Tous ces réacteurs nucléaires sont des Réacteurs à Eau Pressurisée (REP), comme représenté dans la Figure 8.21. Un REP fonctionne avec trois circuits : le circuit primaire, le circuit secondaire et le circuit tertiaire (également appelé boucle de refroidissement).

Le circuit primaire comprend plusieurs composants majeurs : le coeur du réacteur, le pressuriseur, le générateur de vapeur et la pompe primaire. Un réacteur est constitué de 3 ou 4 boucles primaires (générateur de vapeur, pompe) alimentées par un seul coeur de réacteur. Pour extraire l'énergie libérée par la fission, un fluide caloporteur (l'eau) est pompé dans le coeur par les pompes primaires. Il se réchauffe au contact du combustible. Pendant le fonctionnement normal d'une centrale nucléaire, l'eau reste principalement à l'état liquide grâce à la haute pression maintenue par le pressuriseur. L'eau chauffée circule ensuite vers le générateur de vapeur, qui est un échangeur de chaleur qui transfère la chaleur du circuit primaire au circuit secondaire.

Dans le circuit secondaire, l'eau est vaporisée dans le générateur de vapeur puis entraîne les différentes turbines pour produire de l'électricité. Après avoir fourni son travail dans la turbine, la vapeur est recondensée dans le condenseur, puis réinjectée dans le générateur de vapeur. Le refroidissement dans le condenseur est assuré par le circuit tertiaire, qui puise de l'eau dans une rivière ou dans la mer. Selon la configuration de la centrale (en boucle ouverte ou fermée), une tour de refroidissement peut être ajoutée au circuit tertiaire.

Cette thèse se concentre sur l'écoulement au sein d'un seul composant du circuit primaire : le coeur du réacteur. Ce composant est détaillé dans la section suivante.

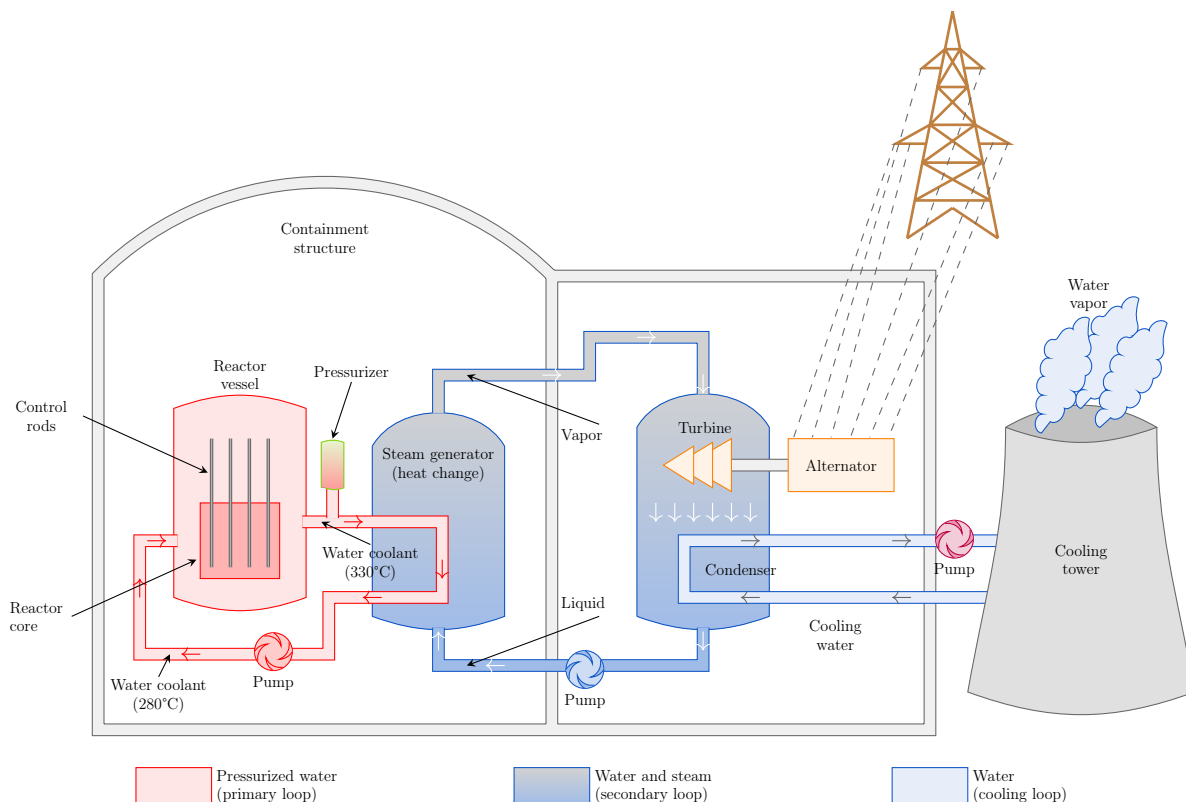


FIGURE 8.21

Schéma d'un Réacteur à Eau Préssurisée (REP). Schéma Tikz de la thèse de Gloria Facanoni[41].

A.2 Présentation du coeur de réacteur nucléaire

Dans le coeur du réacteur, l'eau circule de bas en haut. Elle entre à une température d'environ 290°C et sort du coeur à environ 320°C . Elle est maintenue sous une pression d'environ 155 bars, ce qui permet à l'eau de rester principalement à l'état liquide malgré la température très élevée. En conditions normales de fonctionnement, une petite partie de l'écoulement peut se vaporiser. En cas d'accident, on peut rencontrer un mélange diphasique eau-vapeur contenant une quantité non négligeable de vapeur. Les vitesses du mélange sont de l'ordre de 5 m.s^{-1} en régime nominal.

Le coeur du réacteur est chargé avec du combustible nucléaire composé d'oxyde d'uranium (UO_2) enrichi à environ 4% en uranium-235. L'élément fissile peut également être du plutonium dans le cas du combustible MOX. Ce combustible est fabriqué sous forme de pastilles de 1.35 cm de hauteur. Plusieurs pastilles sont insérées dans des crayons combustibles, comportant une gaine qui entoure le combustible. Cette gaine, faite d'un alliage de zirconium, assure l'étanchéité du crayon afin d'éviter le rejet de matières radioactives dans le circuit primaire.

Un assemblage est constitué de 264 crayons regroupés ensemble et mesure 4 à 5 mètres de long. La Figure 8.22 montre les pastilles, les crayons et un assemblage combustible. Un assemblage comprend également des tubes guides permettant l'insertion des barres de contrôle, qui régulent la réaction nucléaire. Ces barres de contrôle sont fixées à une grappe de contrôle situé au-dessus de l'assemblage.

À l'intérieur de l'assemblage, des grilles d'espacement et de mélange sont placées à intervalles réguliers. Les grilles d'espacement maintiennent les crayons combustibles en place dans l'assemblage, tandis que les grilles de mélange favorisent le mélange pour homogénéiser l'écoulement. Cette homogénéisation permet d'éviter les points chauds en introduisant de la turbulence.

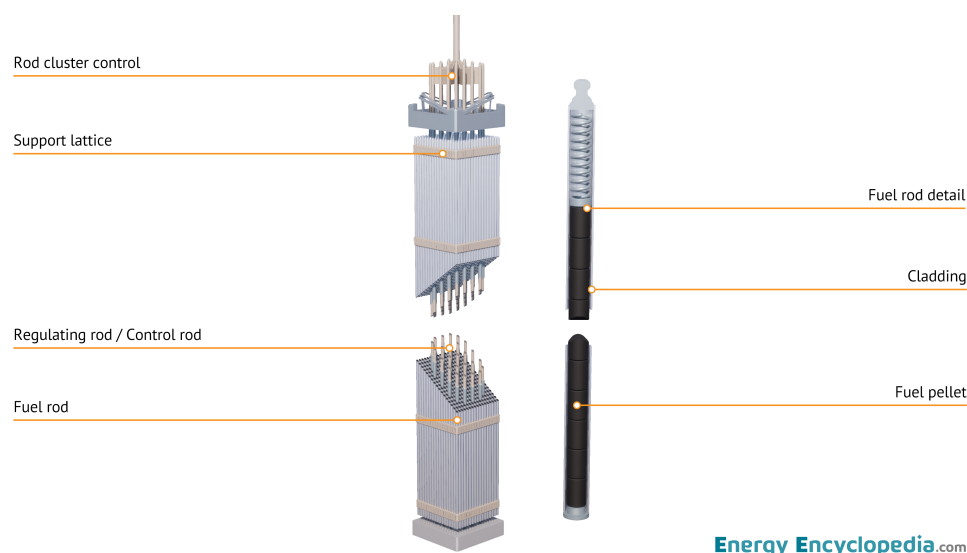


FIGURE 8.22

Schéma d'un assemblage combustible composé de crayons combustibles contenant des pastilles d'uranium. Crédit: EnergyEncyclopedia.com.

Le coeur du réacteur est constitué d'une cuve en acier qui résiste à la pression interne du coeur. Cette cuve est remplie d'un nombre variable d'assemblages selon la puissance souhaitée du coeur : 157 assemblages pour un réacteur de 900 MWe, 193 pour 1300 MWe, 205 pour 1450 MWe et 241 pour l'EPR. L'eau circule dans cet environnement fortement obstrué, présentant des géométries solides très complexes (grilles, crayons, tubes guides, etc.). Bien que l'écoulement soit principalement axial, de bas en haut, des écoulements transverses peuvent se produire entre les assemblages, car l'espace entre eux est laissé ouvert. À l'intérieur des assemblages, l'eau circule principalement entre les crayons combustibles et évacue la chaleur produite par la réaction nucléaire. L'espace entre quatre crayons combustibles est appelé un sous-canal, comme illustré dans la Figure 8.23.



FIGURE 8.23

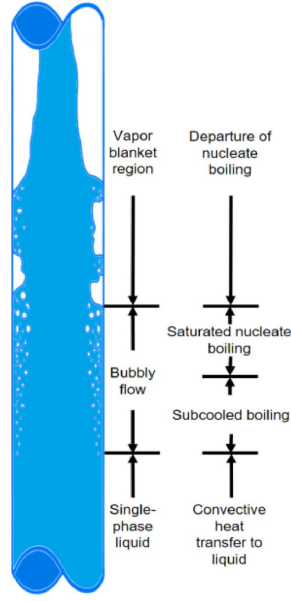
Image d'une maquette d'assemblage combustible avec un zoom sur le sous-canal. Source EDF R&D.

A.3 Régimes d'ébullition

À une pression de 155 bars dans le coeur du réacteur, la température de saturation de l'eau est d'environ 350°C . L'eau entre dans le coeur du réacteur sous forme d'un écoulement monophasique turbulent, avec un nombre de Reynolds de l'ordre de $Re \sim 500000$. Au fur et à mesure que le fluide est chauffé lors de son ascension dans le coeur, l'écoulement peut traverser différents régimes d'ébullition et mécanismes de transfert thermique. Les deux phases, liquide et vapeur, peuvent présenter une grande variété de structures complexes. La vapeur peut apparaître sous forme de petites bulles (de l'ordre de $\sim 10^{-4}$ m) jusqu'à de grandes poches de vapeur occupant tout l'espace entre les crayons combustibles (environ $\sim 10^{-2}$ m). Les différents régimes sont illustrés dans la Figure 8.24. Le transfert thermique débute par un transfert convectif en liquide pur, lorsque le liquide est chauffé vers la température de saturation. Lorsque le flux thermique augmente, la température de paroi T_w augmente. Une fois que T_w dépasse la température de saturation T_{sat} , le **début de l'ébullition nucléée** (ONB, en anglais) se produit. À partir de ce point, l'ébullition nucléée (voir Figure 8.24) a lieu. La surface de la paroi est suffisamment chaude pour vaporiser l'eau à son contact. Des bulles de vapeur apparaissent à la surface de la paroi et peuvent se détacher. Le liquide n'est pas saturé en moyenne, mais de la vapeur apparaît déjà sur les crayons : c'est l'**ébullition sous-saturée**.

Lorsque l'enthalpie du mélange fluide augmente, l'**ébullition nucléée** se poursuit. Des colonnes ou poches de vapeur plus grandes peuvent apparaître. Ces structures se détachent de la paroi et sont entraînées vers le haut. Avec une enthalpie encore plus élevée, les structures de vapeur s'agglomèrent en poches de vapeur. L'ébullition nucléée peut alors basculer brutalement vers une **ébullition en film** : lorsque ce point est atteint, le flux thermique est appelé **flux thermique critique**. Ce phénomène est appelé **départ de l'ébullition nucléée** (DNB, en anglais). La différence entre la température de la gaine et celle du fluide varie fortement, passant de quelques degrés à plusieurs centaines de degrés. C'est la **crise d'ébullition**, durant laquelle la température de paroi devient si élevée ($> 1000^{\circ}\text{C}$) que des dommages thermo-mécaniques peuvent survenir dans la gaine, menaçant l'intégrité des crayons combustibles dans le coeur du réacteur.

La crise d'ébullition est particulièrement dangereuse car elle implique un phénomène d'hystérésis. Une fois le flux thermique critique dépassé, il faut réduire le flux thermique jusqu'au point de Leidenfrost [55] pour revenir à l'ébullition nucléée. Il est donc crucial de s'assurer que le flux thermique critique ne soit atteint nulle part dans le coeur du réacteur nucléaire. En conditions normales de fonctionnement du coeur, le flux thermique reste dans la zone de l'ébullition nucléée sous-saturée.



(a) DNB

FIGURE 8.24

Régimes d'ébullition pour un écoulement ascendant et faible titre vapeur. Illustration extraite de [141].

A.4 Rapport de Flux Thermique Critique

La distance à la crise d'ébullition est mesurée avec le **Rapport de Flux Thermique Critique** (RFTC). Il est défini comme le rapport entre le flux de chaleur local ϕ et le flux de chaleur critique ϕ_{DNB} :

$$\text{RFTC} = \frac{\phi_{DNB}}{\phi}. \quad (8.12)$$

Le RFTC est une grandeur locale définie en chaque point du coeur du réacteur. Il est essentiel de s'assurer que sa valeur soit toujours supérieure à un partout dans le coeur du réacteur ($\text{RFTC} > 1$).

Le flux ϕ_{DNB} est obtenu à l'aide d'une corrélation. Cette corrélation permet de prédire la valeur locale de ϕ_{DNB} en fonction des valeurs moyennes de surface dans un sous-canal, notamment la pression P , le flux massique du mélange G , et le titre à l'équilibre X (voir l'Équation (5.1)), tel que

$$\phi_{DNB} = \phi_{DNB}(P, G, X). \quad (8.13)$$

Les paramètres de la corrélation sont déterminés à partir de résultats expérimentaux d'essais. Pour évaluer le RFTC dans le coeur d'un réacteur lors de la conception ou de la planification du rechargement du combustible, on peut utiliser un code thermo-hydraulique. Ce code doit être capable de fournir les champs locaux de pression, de flux massique et du titre dynamique dans l'ensemble du coeur, pour un écoulement diphasique avec ébullition sous-saturée. Une fois le champ de RFTC déterminé, on considère la valeur minimale sur l'ensemble du coeur, car c'est le point le plus limitant en termes de marge. Cette valeur minimale est appelée **RFTCmin** et doit satisfaire

$$\text{RFTCmin} > 1. \quad (8.14)$$

Pour garantir la sûreté des REP, EDF a développé le code thermo-hydraulique THYC-coeur [7], qui simule l'écoulement diphasique dans le coeur du réacteur. Ce code permet de calculer le champ 3D du ratio du Rapport de Flux Thermique Critique (RFTC) dans le coeur. Cette section commence par une brève revue des codes de coeur de réacteur nucléaire. Deux approches possibles pour traiter

la géométrie sont décrites dans la Section A.5. L'approche milieu poreux, utilisée dans THYC-coeur, est présentée dans la Section A.6. Les caractéristiques du code THYC-coeur sont détaillées dans la Section A.7. Enfin, la Section A.8 présente les études de sûreté réalisées avec THYC-coeur et explique la motivation pour accélérer le code.

A.5 Etude bibliographique des codes de réacteur nucléaire

Pour les simulations thermo-hydrauliques, plusieurs échelles peuvent être considérées (voir Figure 8.25).

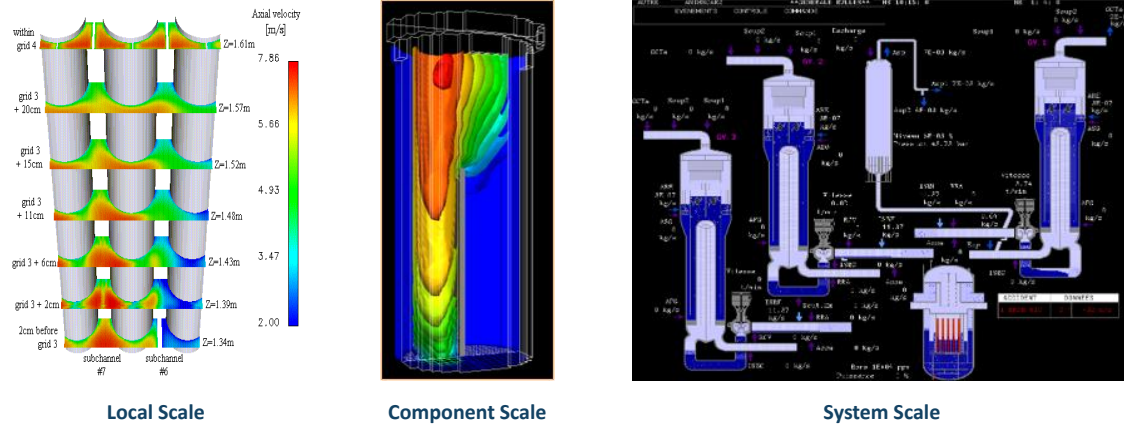


FIGURE 8.25

Illustration des différentes échelles de simulations: local, composante and système. Source: EDF R&D.

- La première et la plus macroscopique des échelles est **l'échelle système**, où plusieurs composants d'un circuit sont simulés simultanément. Ces codes système incluent CATHARE [14] en France, TRACE [8, 112], RELAP5 [46] aux États-Unis, et MARS [26] en Corée du Sud. Pour évaluer précisément le champ RFTC sur l'ensemble du cœur du réacteur, cette échelle est trop macroscopique.
- La deuxième échelle que l'on peut considérer est **l'échelle locale CFD**, utilisant des codes CFD tels que code_saturne [4] développé par EDF ou NEPTUNE_CFD [65, 103], développé conjointement par EDF, Framatome, le CEA et l'ASNR. L'échelle locale est précise mais ne permet pas de simuler un cœur de réacteur complet avec les capacités de calcul actuelles. Il faudrait des centaines de milliards de cellules pour représenter le cœur complet avec un maillage résolvant la paroi. Ces codes sont actuellement utilisés pour simuler localement l'écoulement, sur un faisceau de quelques crayons au maximum lorsque la paroi est résolue, ou sur un assemblage sans résolution de la paroi.
- La troisième échelle est **l'échelle composant**. Le cœur entier du réacteur est représenté avec un maillage de cellules de taille de l'ordre du centimètre, ce qui permet de calculer l'écoulement dans tout le cœur avec un temps CPU acceptable. Avec un code à l'échelle composant, des champs tels que P , G et X peuvent être obtenus sur l'ensemble du cœur du réacteur, permettant d'évaluer le champ RFTC et le RFTCmin. Le compromis d'un code rapide à l'échelle composant est que le modèle fonctionne à une échelle macroscopique, impliquant des termes macroscopiques qui nécessitent des lois de fermeture. Des expériences et des simulations CFD haute fidélité sont nécessaires pour déterminer les paramètres des corrélations utilisées dans ces lois de fermeture. Par conséquent, ces lois de fermeture ne sont validées que sur des plages spécifiques de grandeurs physiques et de configurations.

De nombreux codes composant existe dans le monde pour simuler des écoulements diphasiques en cœur de réacteur. Deux approches sont possibles pour prendre en compte la géométrie complexe.

- L'**approche sous-canaux couplés** simule chaque sous-canal indépendamment et les couple numériquement. C'est une approche très efficace en terme de temps de calcul.
- L'**approche poreuse** intègre les structures solides dans un maillage cartésien à travers un champ de porosité (voir Section A.6). Avec son modèle 3D réel, l'approche poreuse permet de mieux prendre en compte les écoulements transverses.

Pour chaque approche, deux modèles physiques principaux sont utilisés.

- Le **modèle drift-flux** [81, 82] considère les phases eau et vapeur comme un mélange homogène. Il peut être formulé avec 3 ou 4 équations selon les phénomènes physiques pris en compte. Le modèle drift-flux est robuste et permet des simulations rapides. La différence de vitesse entre les deux phases est modélisée par une vitesse relative, obtenue à partir d'une loi de corrélation drift-flux [145]. Une équation aux dérivées partielles pour la vitesse relative peut être ajoutée au modèle à la place de la corrélation drift-flux.
- Le **modèle bi-fluides** [82] représente chaque phase séparément par ses lois de conservation. Dans ce modèle, la difficulté réside dans la modélisation des termes de transfert entre les phases. Ce modèle permet de simuler facilement des configurations avec deux phases hors équilibre thermodynamique ou des cas où les vitesses de vapeur et de liquide diffèrent fortement. Le modèle à deux phases peut également être étendu à un cadre à trois phases avec un modèle à 9 équations. C'est le cas des codes COBRA-TF [135] et CTF [125].

A.6 Approche poreuse

L'approche poreuse utilisée dans THYC-coeur considère des volumes de contrôle contenant à la fois un volume solide V_s et un volume fluide V_f (voir Figure 8.26). La géométrie exacte du solide est ignorée. Pour chaque volume de contrôle de volume total $V_f + V_s$, la porosité est définie comme

$$\varepsilon = \frac{V_f}{V_f + V_s}. \quad (8.15)$$

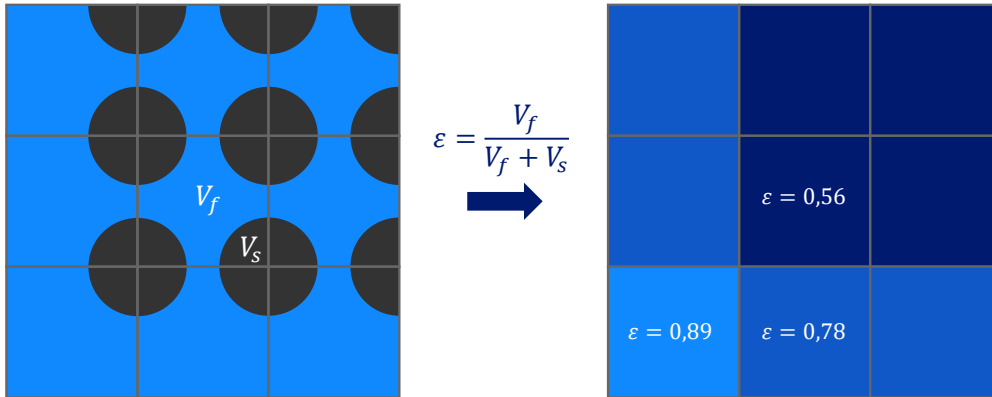


FIGURE 8.26
Schéma expliquant l'approche poreuse.

Les équations du modèle sont intégrées sur ces volumes de contrôle. La présence de matière solide n'est donc prise en compte que par la porosité ε et les termes sources apparaissant dans les équations (apport de chaleur, pertes de charge). Ces termes de fermeture sont modélisés par des lois physiques représentatives du cas étudié. Par exemple, les grilles de mélange et d'espacement sont représentées par des pertes de charge singulières dans le modèle.

Cette méthode permet tout d'abord de s'affranchir de la géométrie complexe des solides dans le cœur du réacteur. Elle permet de modéliser les écoulements dans un milieu encombré avec des

mailles de l'ordre du centimètre. Elle utilise un maillage cartésienne 3D pour représenter le cœur du réacteur, ce qui rend le code efficace en temps de calcul.

A.7 Caractéristiques du code THYC-coeur

Les premiers développements du code THYC-coeur ont commencé en 1986. Un rapport de validation du code a été rédigé en 1998, basé sur de nombreux résultats expérimentaux d'essais. En 2006, l'autorité française de sûreté nucléaire a approuvé l'utilisation de THYC-coeur pour les démonstrations de sûreté. En 2018, un co-développement entre Framatome et EDF a été lancé pour intégrer THYC-coeur dans la nouvelle chaîne de calcul ODYSSEE. Un nouveau rapport de validation a été soumis à l'autorité de sûreté en 2025.

Chez EDF, le code THYC-coeur est développé au sein du département Mécanique des Fluides, Énergies et Environnement (MFEE) de EDF R&D. Afin de répondre aux besoins décrits dans les Sections A.2 et A.3, plusieurs choix de modélisation ont été faits pour le code THYC-coeur.

- Calcul rapide et précis d'écoulement 3D à travers tout le coeur de réacteur.
→ **Code à échelle composante.**
- Considération d'écoulements vraiment 3D, particulièrement avec prise en compte les écoulements transverses entre les sous-canaux et assemblages. Simulation de l'échelle composante avec des écoulements dans une géométrie complexe.
→ **Approche poreuse 3D.**
- Simulation d'un écoulement diphasique avec changement d'état à haute température et haute pression. Considération de la vitesse relative entre la phase vapeur et la phase liquide.
→ **Modèle de dérive, appelé modèle drift-flux** [81, 82].
- Prise en compte du phénomène d'ébullition sous-saturée.
→ Ajout d'une **équation de déséquilibre** sur le titre massique vapeur, menant à un modèle drift-flux à **4 équations**.

Pour définir le modèle drift-flux dans THYC-coeur avec une approche poreuse, plusieurs lois de fermeture sont nécessaires pour les grandeurs macroscopiques apparaissant dans les équations (ces termes sont abordés dans le **Chapitre 1**). Des corrélations sont utilisées comme lois de fermeture. Ces corrélations sont développées et calibrées à partir de résultats d'essais expérimentaux ou de calculs CFD haute fidélité (DNS, LES).

En particulier, le modèle drift-flux repose sur la corrélation de drift-flux [145] pour la vitesse relative entre les phases, notée $\mathbf{u}_r$. Cette corrélation repose sur une formulation qui prend en compte certains phénomènes physiques tels que la poussée d'Archimède de la vapeur dans le liquide, mais aussi les effets de distribution non uniforme de la fraction volumique de vapeur et de la vitesse d'écoulement (vitesse plus faible près de la paroi et plus élevée au centre des sous-canaux).

Dans THYC-coeur, ces paramètres sont obtenus soit avec la corrélation de Chexal-Lellouche [22, 23, 24], soit avec la corrélation de Bestion [14]. Dans THYC-coeur, la puissance à la surface des crayons combustibles est un paramètre d'entrée. Elle est soit fournie par l'utilisateur, soit issue d'un couplage avec un code de neutronique et un code thermique de crayon combustible.

A.8 Motivations industrielles d'accélérer le code THYC-coeur

THYC-coeur est principalement utilisé pour obtenir l'état stationnaire des écoulements diphasiques dans des conditions nominales ou accidentelles, via une simulation transitoire. Dans le cadre des études de sûreté menées pour un nouveau réacteur, pour les études de conception lors des visites décennales, ou pour les analyses de sûreté liées au rechargement du combustible, un très grand nombre de scénarios accidentels doivent être analysés. À l'époque du développement du code, un

petit nombre de calculs avec des hypothèses très conservatrices étaient réalisés pour chaque étude, ce qui conduisait à des analyses très pénalisantes, parfois éloignées de la réalité. Récemment, un traitement plus réaliste des conditions initiales du cœur et des paramètres physiques a conduit à une augmentation significative du nombre de configurations étudiées.

Par ailleurs, en 2017, THYC-cœur a été sélectionné pour être intégré dans la nouvelle chaîne de calcul industrielle ODYSSEE pour les études futures, en commençant par la conception des réacteurs EPR2. En conséquence, le nombre de calculs réalisés avec THYC-cœur a fortement augmenté au fil des années, atteignant désormais plusieurs millions de simulations par an. Comparée à une approche par sous-canaux, la modélisation 3D avec une approche poreuse permet une meilleure représentation des écoulements, notamment des écoulements transverses entre sous-canaux. Cependant, ce choix a un impact sur le temps de calcul. Un calcul du cœur de réacteur avec une maille industrielle prend environ 10 secondes avec THYC-cœur (sur une station mono-cœur avec un maillage industrielle utilisant la symétrie quart de cœur), alors qu'un code sous-canaux comme FLICA ne prend que quelques secondes. Les calculs avec THYC-cœur sont déjà rapides, surtout comparés aux calculs CFD, mais le lancement simultané de millions de simulations pour les études industrielles représente un temps CPU non négligeable. Il est donc crucial de réduire le temps de calcul. Cette thèse s'inscrit dans l'effort en cours visant à accélérer le code THYC-cœur, avec pour objectif de réduire le temps de calcul sans compromettre la précision des résultats.

B Synthèse des travaux de thèse

B.1 Chapitre 1

Dans ce chapitre, les modèles physiques considérés dans le cadre de cette thèse sont présentés. Ces modèles permettent de simuler un écoulement diphasique (eau-vapeur) dans un milieu obstrué par la présence de matière solide. Ils sont dérivés des modèles proposés par Ishii dans [81, 82].

Pour chaque phase, notée $k \in \{g, l\}$, avec g désignant le gaz et l le liquide, les équations locales de conservation (masse, quantité de mouvement et énergie) sont considérées. Afin de définir les grandeurs pour chaque phase à tout instant, les équations locales sont moyennées temporellement. Ce moyennage temporel introduit la fraction volumique locale α_k , qui correspond à la probabilité de présence de la phase k à un instant et un endroit donnés. Le moyennage temporel introduit également des termes d'échange interfacial entre les phases (transfert de masse, de quantité de mouvement et d'énergie).

Les équations sont ensuite moyennées en espace sur un volume V_0 contenant un volume solide V_s et un volume fluide V_f . Cela permet d'éviter d'avoir à traiter la géométrie complexe des solides. Cette méthode fait apparaître la porosité fluide ε donné par

$$\varepsilon = \frac{V_f}{V_f + V_s}. \quad (8.16)$$

Le moyennage spatial introduit également des termes sources dus à la présence des solides. On obtient un modèle à six équations moyenné dans le temps et l'espace. En considérant que l'écoulement diphasique se comporte comme un mélange homogène, l'objectif est ici de simplifier le modèle afin d'obtenir un modèle physique robuste et peu coûteux en temps de calcul. Après avoir défini les grandeurs du mélange, les équations sont sommées pour chaque phase afin d'obtenir trois équations de mélange. Cette sommation permet d'éliminer les termes de transfert interfacial. En prenant la différence entre les équations d'énergie de chaque phase, on obtient une équation de déséquilibre sur l'enthalpie spécifique relative. Le même procédé est appliqué aux équations de quantité de mouvement pour obtenir une équation sur la vitesse relative entre les phases, notée $\mathbf{u}_r$ et définie par

$$\mathbf{u}_r = \mathbf{u}_g - \mathbf{u}_l, \quad (8.17)$$

avec $\mathbf{u}_g$ la vitesse de la phase gazeuse et $\mathbf{u}_l$ la vitesse de la phase liquide. La dernière équation de déséquilibre est celle sur la fraction massique de vapeur. Pour réduire le nombre d'équations, plusieurs hypothèses sont faites :

- La phase minoritaire est toujours considérée à saturation, ce qui permet de supprimer l'équation de déséquilibre sur l'enthalpie spécifique relative,
- La vitesse relative est obtenue via une loi de fermeture, ce qui élimine l'équation sur la vitesse relative. Cette loi de fermeture est dérivée de la méthode de drift-flux développée dans [145].

Si l'on considère l'équation sur la fraction massique de vapeur, on obtient un modèle à 4 équations déséquilibré. Cela permet de modéliser un écoulement thermodynamique hors équilibre, en particulier le phénomène d'ébullition sous-saturée, où l'ébullition peut commencer même lorsque le liquide n'est pas encore à saturation en moyenne (un opérateur de moyennage spatial est utilisé). Pour cela, le terme de transfert de masse interfacial impliqué dans l'équation de la fraction massique de vapeur doit être modélisé. Si le mélange est considéré à saturation, cette équation disparaît et on obtient le modèle à 3 équations à l'équilibre.

Dans ce chapitre, deux modèles à 4 équations sont présentés : un modèle conservatif en énergie totale, et un modèle basé sur l'enthalpie. Le modèle basé sur l'enthalpie correspond à celui implémenté dans le code industriel THYC-coeur [7]. Pour l'obtenir, plusieurs hypothèses supplémentaires ont été faites : la contribution thermique due au frottement visqueux entre les phases et au frottement solide a été négligée, et la somme des transferts d'énergie thermique entre les phases a été négligée dans le bilan d'enthalpie.

Pour fermer les systèmes d'équations présentés, plusieurs termes doivent être modélisés, tels que le frottement fluide, le frottement solide, la vitesse relative, le transfert de masse... Pour cela, des lois de fermeture modélisant les principaux phénomènes physiques sont utilisées. Dans ce chapitre, ces lois de fermeture sont décrites, en particulier pour le terme de transfert de masse et pour la vitesse relative. Pour cette dernière, deux corrélations différentes sont proposées : la corrélation de Bestion [14] et la corrélation de Chexal-Lellouche [22].

B.2 Chapitre 2

Les modèles du **Chapitre 1** sont étudiés dans le **Chapitre 2** dans des configurations simplifiées afin de déterminer si les modèles dérivés du modèle drift-flux à 4 équations sont hyperboliques, ce qui permet de garantir des solutions transitoires stables. Les modèles suivants sont analysés dans ce chapitre :

- **Modèle barotrope de drift-flux avec une vitesse relative constante** : Le modèle de drift-flux est considéré sans équation sur l'énergie et avec une vitesse relative constante. Les valeurs propres du système ne peuvent pas être obtenues analytiquement. En utilisant le théorème des valeurs intermédiaires, le modèle est **strictement hyperbolique** pour $u_{r0} \neq 0$.
- **Modèle à 4 équations sans vitesse relative** : Ce modèle est équivalent à un modèle HRM [16]. En considérant c comme la vitesse du son du mélange, les valeurs propres sont $\lambda \in \{u, u, u + c, u - c\}$. Ce modèle est **hyperbolique**. Cela implique de disposer d'une équation d'état pour le mélange garantissant une vitesse du son réelle.
- **Modèle drift-flux à 4 équations avec une vitesse relative constante u_{r0}** : Ce modèle est plus complexe que le modèle HRM. Une seule valeur propre analytique est obtenue : $u_l = u - yu_{r0}$. Les autres valeurs propres analytiques ne sont pas accessibles. Le polynôme caractéristique est évalué pour la vitesse de la phase gazeuse $u_g = u + (1 - y)u_{r0}$. En utilisant le théorème des valeurs intermédiaires, le système est hyperbolique si la dérivée du polynôme caractéristique est négative en $u_l = u - yu_{r0}$. Il s'agit d'une condition suffisante

d'hyperbolicité. Avec p la pression, ρ la densité du mélange et y la fraction massique, le système est **hyperbolique** si

$$C_1^h(p, \rho, y) < 0 \quad \text{ou} \quad |u_{r0}| < u_r^{max}(p, \rho, y) = \sqrt{\frac{C_2^h}{C_1^h}} c(p, \rho, y), \quad (8.18)$$

avec

$$\begin{cases} C_1^h(p, \rho, y) = ya^{-1}(p, \rho, y) \left(y \frac{dh_g}{dp}(p) - \frac{1}{\rho} \right), \\ C_2^h(p, \rho, y) = ((1 - \rho y v_g(p))), \end{cases} \quad (8.19)$$

où c est la vitesse du son du mélange et $a = \left(\frac{\partial e}{\partial p} \right)_{\rho, y}$. Encore une fois, ce résultat est valable lorsque l'équation d'état du mélange garantit une vitesse du son réelle. Dans l'annexe de ce chapitre, cette condition suffisante est étudiée pour un mélange eau-vapeur à 155 bars.

B.3 Chapitre 3

L'objectif de ce chapitre est de proposer des solutions analytiques pour des versions simplifiées des modèles présentés dans le **Chapitre 1**. La première partie se concentre sur la quatrième équation concernant uniquement la fraction massique de vapeur, afin de fournir des solutions analytiques pour tester les schémas volumes finis dans le **Chapitre 4**, en isolant cette équation du système complet. Un milieu libre sans porosité est considéré ici.

Trois nouvelles solutions exactes sont présentées. Tout d'abord, deux solutions à l'état stationnaire, avec la plupart des paramètres de l'équation considérés comme constants ($\rho_0, \tau_0, u_0, \bar{y}_0$), sont proposées. La première utilise une vitesse relative constante u_{r0} . La seconde utilise la corrélation approchée de Bestion présentée dans le **Chapitre 1**. Cette corrélation propose la vitesse relative comme une fonction de la fraction massique :

$$u_r(y) = \frac{1 + (\delta - 1)y}{(1 - y)u_b}, \quad (8.20)$$

avec δ et u_b des paramètres constants. Ces solutions analytiques sont utilisées dans le **Chapitre 4** pour vérifier les schémas numériques stationnaires pour la quatrième équation sur la fraction massique. La troisième solution est une solution instationnaire de la quatrième équation, utilisant à nouveau la corrélation approchée de Bestion $u_r(y)$. Cette solution auto-similaire donne la fraction massique en fonction de la variable ξ :

$$\xi(x, t) = \frac{x}{u_0 t}. \quad (8.21)$$

Cette solution est également utilisée dans le **Chapitre 4** pour vérifier le schéma pour les solutions instationnaires.

La seconde section du chapitre est extraite d'un article publié pendant la thèse [70]. Le modèle à trois équations est considéré sans vitesse relative $u_r = 0$, correspondant à un modèle de type HEM [16] dans un milieu poreux de porosité $\varepsilon(x)$ et avec un apport d'énergie $\phi(x, t)$. Deux nouvelles solutions analytiques auto-similaires issues de [70] sont présentées pour deux configurations différentes de ce modèle. Ces solutions seront utilisées pour la vérification du code industriel THYC-coeur dans des travaux futurs. Les solutions dépendent de la variable ξ :

$$\xi(x, t) = \frac{x}{t + t_0}, \quad t_0 > 0. \quad (8.22)$$

La première solution est obtenue sans apport de chaleur et avec une porosité non uniforme $\varepsilon(x)$ vérifiant :

$$\varepsilon = \left(\frac{x}{x_0} \right)^\alpha, \quad (8.23)$$

avec α et x_0 des constantes. La solution analytique est proposée pour une équation d'état générale, et le cas particulier du gaz parfait est détaillé à la fin. Pour la seconde solution, un écoulement avec une équation d'état arbitraire dans un milieu libre est considéré. Un apport de chaleur de la forme suivante est appliqué :

$$\phi(x, t) = \frac{1}{t + t_0} \psi(\xi). \quad (8.24)$$

La dérivation de cette solution conduit à un système d'EDO en ξ qui doit être résolu numériquement.

Le **Chapitre 3** inclut également trois annexes pour détailler des solutions déjà connues utilisées dans ce travail. La première annexe présente une solution stationnaire du modèle à trois équations issue de [78], adaptée au cas étudié. Cette solution analytique est obtenue via une méthode de Newton et pour une équation d'état générale. Elle est utilisée dans le **Chapitre 7** pour vérifier le code ThermoTorch dans la configuration stationnaire du modèle à trois équations.

Dans la seconde annexe, des solutions de problèmes de Riemann pour le modèle à trois équations avec une équation d'état de gaz parfait sont présentées. En particulier, les trois problèmes de Riemann considérés sont : une double onde de détente symétrique, une double onde de choc symétrique, et le tube à choc de Sod. Ils sont utilisés dans le **Chapitre 7** pour étudier le comportement des schémas numériques de ThermoTorch, aussi bien pour les solutions régulières que pour les solutions avec chocs, en utilisant le modèle à trois équations.

Enfin, la dernière annexe présente une équation d'état de mélange dérivée d'un mélange de gaz parfait ayant le même indice polytropique γ . L'équation d'état est écrite comme :

$$e(p, \rho, y) = ye_g + (1 - y)e_l = \frac{p}{\rho(\gamma - 1)} + h_0(y), \quad (8.25)$$

avec

$$h_0(y) = yh_{0g} + (1 - y)h_{0l}, \quad (8.26)$$

où γ , h_{0g} et h_{0l} sont des paramètres constants à fixer. Cette équation d'état particulière est utilisée pour trouver une solution à un problème de Riemann pour le modèle à 4 équations. Cette solution est utilisée dans le **Chapitre 7** pour étudier le schéma numérique instationnaire de ThermoTorch avec le modèle à quatre équations, en particulier le schéma utilisé pour l'équation d'énergie.

B.4 Chapitre 4

Ce chapitre est une version étendue d'un article publié durant la thèse [93]. Il est centré sur l'équation de déséquilibre pour la fraction massique de vapeur y :

$$\frac{\partial(\rho y)}{\partial t} + \nabla \cdot (y \mathbf{q}) + \nabla \cdot (y(1 - y) \mathbf{q}_r) = \rho \frac{\bar{y} - y}{\tau} + \Gamma_p, \quad (8.27)$$

où ρ est la masse volumique du mélange, $\mathbf{q}$ est le flux massique du mélange, $\mathbf{q}_r = \rho \mathbf{u}_r$ le flux massique relatif, $\bar{y}$ le titre massique à l'équilibre, τ le temps de relaxation à l'équilibre et $\Gamma_p > 0$ la production de vapeur. Cette équation est similaire à celle étudiée dans [53]. Elle modélise un écoulement hors équilibre, avec une fraction massique y qui tend à s'écarter de la fraction massique d'équilibre $\bar{y}$ sous l'effet d'un terme de production direct Γ_p , et qui est ramenée vers l'équilibre après un temps caractéristique τ . L'originalité de cette équation de transport réside dans le terme de drift-flux, qui est un terme de convection de la quantité non linéaire $y(1 - y)$ par le flux massique relatif $\mathbf{q}_r$.

Dans un premier temps, le principe du maximum continu est étudié sur l'Équation (8.27) afin de garantir que la fraction massique reste comprise entre zéro et un. Ensuite, trois schémas volumes finis - appelés QRd, QRq et QG - sont proposés pour cette équation. Ces schémas sont présentés

dans un article publié durant la thèse [93]. Il s'agit de schémas implicites linéaires, présentés dans un cadre multidimensionnel non structuré. L'étude est limitée à des schémas numériques linéaires, afin qu'ils puissent être utilisés dans le code THYC-coeur. Ils respectent le principe du maximum pour la fraction massique de vapeur, qui doit rester entre 0 et 1. Pour cela, aucune condition sur le pas de temps n'est requise pour les schémas QRd et QRq. Le schéma QG impose une contrainte sur le pas de temps, qui est non limitante dans les applications concernées. Dans ce contexte, le terme de production, qui ne respecte pas nécessairement le principe du maximum continu, n'est pas pris en compte dans ces schémas.

Les schémas sont implémentés dans un prototype 1D qui simule uniquement la quatrième équation. En utilisant les solutions analytiques de la quatrième équation issues du **Chapitre 3**, les schémas sont d'abord testés sur des solutions stationnaires unidimensionnelles. Cette solution stationnaire est obtenue pour des paramètres constants ($\rho, q, q_r, \tau, \bar{y}$). Deux cas tests représentatifs des conditions d'écoulement en réacteur sont étudiés. Cette vérification démontre la cohérence des trois schémas, qui présentent un ordre de convergence spatial égal à un. Les schémas QG et QRq apparaissent plus précis que le schéma QRd pour une maille fixée, notamment lorsque l'on considère une vitesse relative négative significative.

Les schémas QRd et QRq proposés ne sont pas conservatifs en régime transitoire. Ils ne sont donc pas adaptés à la simulation de solutions instationnaires, comme montré dans l'annexe de ce chapitre. Le schéma QG peut être vérifié sur une solution instationnaire. En considérant une valeur de CFL constante, une courbe de convergence est produite pour la solution instationnaire de la quatrième équation proposée dans le **Chapitre 3**. Un ordre de convergence proche de un est obtenu pour le schéma QG, comme attendu.

Dans le **Chapitre 5**, ces schémas sont ensuite implémentés dans le code industriel THYC-coeur afin d'évaluer leur impact sur les performances par rapport au schéma de référence de THYC-coeur. Dans cette étude sur le code THYC-coeur, l'équation de déséquilibre n'est plus traitée indépendamment comme dans le **Chapitre 4**, mais comme partie intégrante du système complet THYC-coeur (voir **Chapitre 1** pour le modèle).

Dans l'annexe de ce chapitre, un schéma numérique est proposé pour le terme de production Γ_p afin de rétablir le principe du maximum au niveau discret, c'est-à-dire garantir que la fraction massique reste toujours entre 0 et 1. En effet, sans contrôle, un modèle arbitraire pour le terme $\Gamma_p > 0$ peut entraîner une évolution de la fraction massique vers des valeurs supérieures à un. Au niveau discret, le schéma correctif GAMc proposé permet de modifier localement l'équation pour garantir que la solution reste inférieure à un. Ce schéma correctif ne peut être utilisé que pour la recherche de solution stationnaire, car le comportement transitoire est altéré en raison du contrôle.

B.5 Chapitre 5

Les trois schémas proposés dans le **Chapitre 4** ont été implémentés dans le code industriel THYC-coeur [7]. Pour vérifier leur implémentation et leur cohérence, une base de données de 36148 cas industriels stationnaires est utilisée. Ces cas correspondent à une configuration de réacteur de type N4, sous différentes conditions physiques (température d'entrée et flux massique, pression de sortie, distribution de puissance). Cette base de données est appelée Bias Curves Database. Elle a été initialement créée pour configurer un système de protection appelé SPIN (Système de Protection Intégré Numérique), utilisé pour garantir la sûreté d'une installation en fonctionnement.

Dans un premier temps, la Bias Curves Database est présentée, ainsi que le système de protection SPIN. Ensuite, les trois schémas introduits dans le **Chapitre 4** [93] sont comparés au schéma de référence de THYC-coeur sur cette base de données. Deux modèles différents sont considérés, selon la corrélation utilisée pour la vitesse relative. Le premier modèle repose sur la corrélation de Bestion [14], tandis que le second repose sur la corrélation de Chexal-Lellouche [22]. Pour chaque schéma,

les performances en termes de précision (par rapport au schéma de référence) et en termes de temps CPU sont évaluées. Les comparaisons sont réalisées en utilisant des paramètres numériques industriels (taille de maillage, pas de temps, etc.) et visent à évaluer la capacité des schémas à prédire la solution du modèle d'équation continue lorsque les paramètres numériques industriels sont fixés.

Les résultats obtenus valident l'implémentation et la cohérence des trois nouveaux schémas pour les deux modèles. Ces nouveaux schémas apportent une robustesse accrue au code. Pour le modèle utilisant la corrélation de Chexal-Lellouche, les schémas permettent également un gain de 50% en temps CPU par rapport au schéma de référence, en considérant les mêmes paramètres numériques (pas de temps, critères d'arrêt, etc.). Les conclusions de cette comparaison sont résumées dans le Tableau 8.3, qui liste les avantages et les limitations de l'utilisation de chaque schéma dans le contexte de THYC-coeur. Il en ressort que le schéma QG est le plus adapté à une utilisation dans THYC-coeur. En particulier, il est très robuste, offre la meilleure précision pour un maillage donnée avec la corrélation de Chexal-Lellouche, et peut être utilisé pour des simulations transitoires. Le schéma QG présente une faible contrainte sur le pas de temps, contrairement aux schémas QRd et QRq, mais cette contrainte n'est pas limitante en pratique. En particulier, elle n'est jamais atteinte pour les applications testées ici.

Schéma	Contraintes pas de temps	Autres contraintes	Stationnaire Bestion	Stationnaire Chexal-Lellouche	Simulations instationnaires
REF	Complexe	-	Valide	Moins recommandé	-
QRd	Aucune	CL de sortie Non limitante	Valide	Moins recommandé	Non adapté
QRq	Aucune	Ecoulements co-courant	Valide	Recommandé	Non adapté
QG	Faible Non limitante	Aucune	Valide	Recommandé	Recommandé

TABLE 8.3

Tableau de comparaison des caractéristiques des schémas pour les applications THYC-coeur.

B.6 Chapitre 6

Le code ThermoTorch 1D (voir l'article publié durant la thèse [68]) est un code unidimensionnel en volumes finis simulant un écoulement diphasique. Il s'agit d'un prototype partageant les mêmes caractéristiques générales que le code industriel THYC-coeur [7], mais avec une physique simplifiée. Ainsi, les modèles à 4 équations et à 3 équations de THYC-coeur présentés dans le **Chapitre 1** sont considérés dans ThermoTorch sans termes du second ordre (diffusion, frottement...) et avec une équation d'état simplifiée pour le mélange. De plus, une vitesse relative nulle est considérée ici. Cela permet d'obtenir un modèle simple qui sera utilisé dans les deux chapitres suivants pour vérifier les schémas numériques de ThermoTorch 1D pour des solutions stationnaires et instationnaires (voir **Chapitre 7**), ou pour évaluer les performances d'une méthode d'accélération de la recherche de l'état stationnaire par initialisation via un réseau de neurones (voir **Chapitre 8**).

Le **Chapitre 6** commence par la présentation des modèles implémentés dans ThermoTorch : le modèle à 3 équations et le modèle à 4 équations. Le modèle est manipulé pour exprimer le système avec les inconnues suivantes : pression p , flux massique $q = \rho u$, enthalpie spécifique $h = e + p/\rho$, et fraction massique y (uniquement pour le modèle à 4 équations). Ensuite, les discrétisations spatiale et temporelle sont détaillées. En particulier, un maillage à grilles décalées est utilisée avec une taille de maille uniforme Δx . Dans ThermoTorch, les champs solutions sont initialisés, puis un instationnaire est calculée avec une discrétisation temporelle basée sur un pas de temps Δt . Ce pas de temps peut être constant ou calculé à partir d'un nombre de CFL dérivé de la vitesse d'une

des ondes du système. Cette méthode permet de simuler des écoulements transitoires ainsi que d'atteindre l'état stationnaire en avançant dans le temps jusqu'à ce que la solution ne varie plus.

Les schémas numériques en volumes finis utilisés pour discrétiser le système d'équations continues sont présentés dans la troisième partie du chapitre. Des schémas semi-implicites sont proposés, permettant d'écrire chaque équation linéairement en fonction de l'incrément temporel des inconnues. Une fonction appelée pseudo-entropie $\mathfrak{s}$ est introduite pour découpler l'équation d'énergie du reste du système. Elle correspond à une grandeur du mélange dérivée des entropies de chaque phase :

$$\mathfrak{s} = y s_g + (1 - y) s_l, \quad (8.28)$$

avec s_g l'entropie de la phase gazeuse et s_l celle de la phase liquide. Pour le modèle à 4 équations, cette fonction n'est pas l'entropie du système. Elle satisfait

$$T_l d\mathfrak{s} = dh - (v + y(T_l - T_s(p)) \frac{ds_g}{dp}) dp + \mathfrak{s}_y dy, \quad (8.29)$$

avec

$$\mathfrak{s}_y = (h_l - \bar{h}_g) - T_l(s_l - \bar{s}_g) = \mu_l(p, T_l) - \bar{\mu}_g(p, T_s) + (T_s - T_l)\bar{s}_g. \quad (8.30)$$

Les termes en bleu sont négligés pour découpler complètement l'équation d'énergie. La fonction $\mathfrak{s}$ est uniquement utilisée dans la partie instationnaire du schéma. Ainsi, cette approximation n'affecte que les solutions transitoires du modèle à 4 équations. L'impact de cette approximation est évalué dans le **Chapitre 7**. Le système matriciel du modèle à 4 équations, obtenu en considérant les incréments temporels des inconnues $(p, \mathfrak{s}, q, y)$, peut s'écrire sous la forme :

$$\begin{pmatrix} A_{\mathfrak{s},\mathfrak{s}} & 0 & 0 & 0 \\ 0 & A_{y,y} & 0 & 0 \\ A_{p,\mathfrak{s}} & A_{p,y} & A_{p,p} & A_{p,q} \\ 0 & 0 & A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta\mathcal{S} \\ \delta Y \\ \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_{\mathfrak{s}} \\ B_y \\ B_p \\ B_q \end{pmatrix}, \quad (8.31)$$

où $\delta\mathcal{S}$, δY , δP et δQ (avec $\delta\phi = \phi^{n+1} - \phi^n$, $\phi \in \{S, Y, P, Q\}$) sont les vecteurs solutions correspondant aux incréments temporels de la fonction $\mathfrak{s}$, de la fraction massique y , de la pression p et du flux massique q sur leurs maillages respectifs. Par exemple, $\delta\mathcal{S} = (\mathfrak{s}^{n+1} - \mathfrak{s}^n)_{i \in [0, n_x]}$. Ce schéma en volumes finis permet de résoudre un pas de temps en plusieurs étapes indépendantes, en découplant la fraction massique et la fonction $\mathfrak{s}$ des autres variables. Le vecteur $\delta\mathcal{S}$ est obtenu en utilisant

$$\delta\mathcal{S} = A_{\mathfrak{s},\mathfrak{s}}^{-1} B_{\mathfrak{s}}, \quad (8.32)$$

et le vecteur δY en utilisant

$$\delta Y = A_{y,y}^{-1} B_y. \quad (8.33)$$

Alors, le système couplé vitesse-pression peut s'écrire :

$$\begin{pmatrix} A_{p,p} & A_{p,q} \\ A_{q,p} & A_{q,q} \end{pmatrix} \begin{pmatrix} \delta P \\ \delta Q \end{pmatrix} = \begin{pmatrix} B_p - A_{p,\mathfrak{s}}\delta\mathcal{S} - A_{p,y}\delta Y \\ B_q \end{pmatrix}. \quad (8.34)$$

La matrice de ce système est principalement creuse, donc la représentation Python `scipy.sparse.csc_matrix` est utilisée. Les vecteurs δP et δQ sont obtenus en inversant directement la matrice creuse, en utilisant la méthode Python `scipy.sparse.linalg.spsolve`. Le modèle à trois équations peut être résolu de manière similaire, sans prendre en compte l'équation pour la fraction massique y . La dernière partie du chapitre vise à présenter les configurations du code utilisées dans les chapitres suivants. Les équations d'état disponibles dans le code sont également présentées.

B.7 Chapitre 7

Dans ce chapitre, les schémas numériques de ThermoTorch présentés dans le **Chapitre 6** sont testés. L'objectif est de vérifier l'implémentation des schémas numériques dans ThermoTorch et d'évaluer leur ordre de convergence en espace et en temps. Les configurations du **Chapitre 3** sont mises en œuvre, et les solutions numériques approchées obtenues avec ThermoTorch sont comparées aux solutions analytiques décrites dans le **Chapitre 3**. Tout au long de ce chapitre, aucune vitesse relative n'est considérée, car à notre connaissance, aucune solution analytique n'est disponible pour le modèle avec vitesse relative. Plusieurs types de solutions sont étudiés dans ce chapitre afin de vérifier différentes propriétés des schémas.

- **Solutions stationnaires du modèle à trois équations** : Ces solutions permettent d'étudier l'ordre de convergence spatiale du schéma ThermoTorch pour des solutions régulières 1D du modèle à trois équations. La configuration étudiée est celle d'un canal chauffé avec des conditions proches de celles d'un écoulement dans un cœur de réacteur. Un ordre de convergence spatial égal à un est obtenu.
- **Solutions instationnaires de problèmes de Riemann 1D pour le modèle à trois équations** : Ces solutions sont utilisées pour étudier le modèle à trois équations de ThermoTorch dans des configurations instationnaires. Les différents problèmes de Riemann 1D étudiés sont : un cas symétrique avec deux ondes de détente (solution régulière), un cas symétrique avec deux ondes de choc (solutions avec chocs), et le cas du tube à choc de Sod [131]. Les simulations sont réalisées avec un nombre CFL constant (en considérant les ondes rapides) :

$$\text{CFL}_{u+c} = 0.5. \quad (8.35)$$

Cette étude montre que les schémas ThermoTorch pour le modèle à trois équations présentent un ordre de convergence supérieur à $1/2$ (et semblent tendre vers un) en espace et en temps pour la solution régulière instationnaire considérée (double onde de détente symétrique) [52]. Pour les solutions avec chocs, une erreur de consistance est introduite par les schémas non conservatifs utilisés ici, comme attendu [51, 76]. En conséquence, les plateaux de pression et de densité ne sont pas parfaitement prédits dans le cas des deux ondes de choc symétriques. Il est montré que l'amplitude de cette erreur de consistance est négligeable comparée à l'erreur introduite par l'utilisation d'un maillage industriel dans des conditions proches de celles d'un écoulement en cœur de réacteur.

- **Solutions instationnaires de problèmes de Riemann réguliers 1D pour le modèle à quatre équations** : Ces problèmes de Riemann sont utilisés pour évaluer le comportement du schéma ThermoTorch avec le modèle à quatre équations dans des conditions instationnaires. En particulier, cette étude évalue l'impact de l'utilisation de la pseudo-entropie et de l'approximation faite lors de sa linéarisation (voir l'Équation (8.29)) par rapport à l'utilisation de l'entropie réelle du système s . Cela est rendu possible grâce à l'équation d'état proposée dans le **Chapitre 3**, pour laquelle l'entropie du mélange est connue.

L'approximation de la pseudo-entropie introduit une erreur de consistance dans l'équation de l'énergie pour les solutions instationnaires. Cette erreur est mise en évidence par une étude de convergence. Lorsque la pseudo-entropie est utilisée, l'erreur entre la solution analytique et la solution numérique atteint un plateau à mesure que le maillage est raffiné. En revanche, avec l'entropie, le schéma ThermoTorch montre un ordre de convergence de $1/2$ (attendu en raison de la présence d'une onde de contact dans la solution). L'amplitude de l'erreur de consistance due à l'approximation sur la pseudo-entropie est évaluée et comparée à l'erreur introduite par l'utilisation d'un maillage industriel. Les erreurs sont estimées pour diverses conditions d'écoulement rencontrées dans les cœurs de réacteurs nucléaires. Il est observé que dans la plupart des cas, l'erreur de consistance est négligeable.

Dans ThermoTorch, l'utilisation de la pseudo-entropie $\mathfrak{s}$ permet de découpler complètement l'équation de l'énergie des autres équations. Cela donne un modèle numériquement robuste pour le calcul stationnaire avec d'excellentes performances en temps de calcul. En contrepartie, une erreur de consistance est introduite en régime transitoire, qui devient plus significative lorsque la configuration s'éloigne de l'équilibre thermodynamique.

B.8 Chapitre 8

Ce chapitre est une version étendue d'un article de conférence [92]. Il propose une méthode basée sur l'apprentissage automatique pour accélérer la convergence d'un code volumes finis lors de la recherche de solutions stationnaires. Un réseau de neurones profond est développé afin de prédire les solutions stationnaires. Ces solutions prédites sont utilisées pour initialiser le calcul, dans le but de réduire le nombre d'itérations externes (l'effet recherché est illustré dans la Figure 8.27) par rapport à une autre initialisation, comme une initialisation par champ constant. Le développement de cette méthode est apparu pour la première fois dans l'article [68] publié durant la thèse.

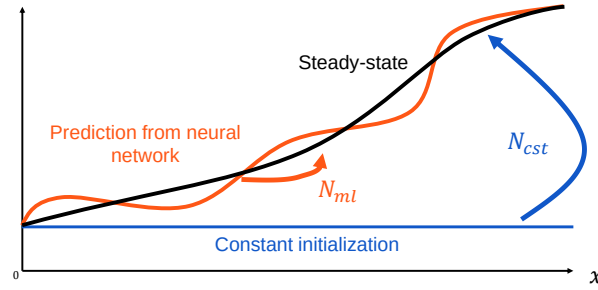


FIGURE 8.27

Illustration de l'initialisation par réseau de neurones pour accélérer la recherche du stationnaire.

Les performances de cette méthode, c'est-à-dire le gain en nombre d'itérations avant convergence vers l'état stationnaire, sont évaluées avec le prototype 1D ThermoTorch, présenté dans le **Chapitre 6**. La configuration du canal chauffé (issue du **Chapitre 6**) est utilisée, avec quatre conditions aux limites (flux massique q_{in} et température T_{in} entrants, pression de sortie p_{out} et flux de chaleur uniforme ϕ_0). La méthodologie détaillée pour l'entraînement et l'utilisation du réseau de neurones est représentée dans la Figure 8.28. Les modèles à trois et quatre équations de ThermoTorch sont tous deux testés.

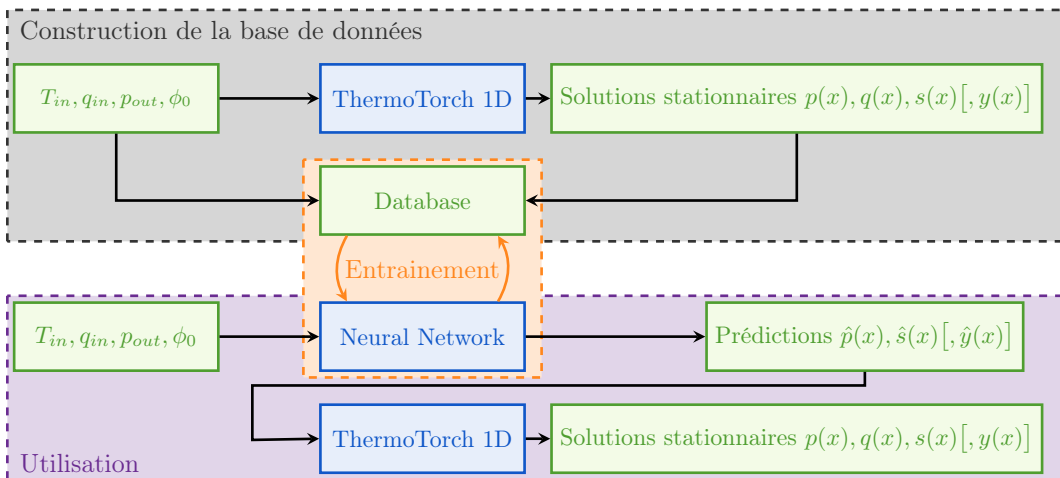


FIGURE 8.28

Construction de la base de données utilisée pour l'entraînement et utilisation du réseau de neurones pour accélérer ThermoTorch 1D.

Un premier réseau de neurones a été développé avec l'erreur quadratique moyenne entre la prédiction et la solution stationnaire (pour chaque champ) comme fonction de coût. Après optimisation des hyperparamètres du réseau, de bons résultats ont été observés pour un réseau de neurones comportant deux couches cachées de 200 neurones chacune. Un ensemble de données d'apprentissage de 10000 cas et un ensemble de test de 1000 cas sont utilisés. La méthode permet d'accélérer tous les cas, quel que soit le modèle. Le gain moyen obtenu est de $70 \pm 19\%$ pour le modèle à trois équations et de $48 \pm 21\%$ pour le modèle à quatre équations.

La fonction de coût n'est pas entièrement corrélée au gain observé. Une étude sur les fréquences de l'erreur entre la prédiction et la solution a été menée. Il a été observé que le gain n'est pas le même selon la fréquence de l'erreur de la prédiction (à amplitude constante). En particulier, les perturbations de basses fréquences sur l'entropie ont un impact majeur sur les performances pour atteindre l'état stationnaire.

Dans la seconde partie du chapitre, une nouvelle fonction de coût est proposée. Elle prend en compte les premières fréquences de la transformée de Fourier discrète de l'erreur sur l'entropie. Avec cette fonction de coût, le gain est augmenté de plus de 10%. Il atteint $83 \pm 11\%$ pour le modèle à trois équations et $61 \pm 18\%$ pour le modèle à quatre équations.

Bibliography

- [1] Ajuria-Illaramendi, E., Alguacil, A., Bauerheim, M., Misdariis, A., Cuenot, B. and Benazera, E. “Towards a hybrid computational strategy based on Deep Learning for incompressible flows”. In: *AIAA AVIATION FORUM*. 2020, pp. 1–17. URL: <https://hal.science/hal-02923501>.
- [2] Ambroso, A., Chalons, C., Coquel, F., Galié, T., Godlewski, E., Raviart, P. A. and Seguin, N. “The drift-flux asymptotic limit of barotropic two-phase two-pressure models”. In: *Communications in Mathematical Sciences* 6.2 (2008), pp. 521–529.
- [3] Archambeau, F., Hérard, J.-M. and Laviéville, J. “Comparative study of pressure-correction and Godunov-type schemes on unsteady compressible cases”. In: *Computers & Fluids* 38.8 (2009), pp. 1495–1509.
- [4] Archambeau, F., Méchitoua, N. and Sakiz, M. “Code Saturne: A finite volume code for the computation of turbulent incompressible flows-Industrial applications”. In: *International Journal on Finite Volumes* 1.1 (2004). URL: <https://hal.science/hal-01115371>.
- [5] Areva NP. *A Core Thermal-Hydraulic Analysis Code*. Topical Report ANP-10311NP. 2010.
- [6] Armand, A. *The Resistance During the Movement of a Two-phase System in Horizontal Pipes*. AERE-Trans. 1946, pp. 16–23.
- [7] Aubry, S., Caremoli, C., Olive, J. and Rasche, P. “The THYC Three-Dimensional Thermal-Hydraulic Code for Rod Bundles: Recent Developments and Validation Tests”. In: *Nuclear Technology* 112.3 (1995), pp. 331–345. DOI: 10.13182/NT95-A35159.
- [8] Avramova, M., Velazquez-Lozada, A. and Rubin, A. “Comparative Analysis of CTF and Trace Thermal-Hydraulic Codes Using OECD/NRC PSBT Benchmark Void Distribution Database”. In: *Science and Technology of Nuclear Installations* 2013 (Feb. 2013), p. 725687. ISSN: 1687-6075. DOI: 10.1155/2013/725687.
- [9] Baer, M. R. and Nunziato, J. W. “A two-phase mixture theory for the deflagration-to-detonation transition (DDT) in reactive granular materials”. In: *International journal of multiphase flow* 12.6 (1986), pp. 861–889.
- [10] Bai, N., Zhu, Y.-B., Ren, Z.-H., Chen, J., Zhou, Y.-X., Li, J.-G. and He, H.-B. “Development and verification of subchannel analysis code LINDEN”. In: *Yuanzineng Kexue Jishu/Atomic Energy Science and Technology* 47 (June 2013), pp. 299–301. DOI: 10.7538/yzk.2013.47.S0.0299.
- [11] Bankoff, S. G. “A Variable Density Single-Fluid Model for Two-Phase Flow With Particular Reference to Steam-Water Flow”. In: *Journal of Heat Transfer* 82.4 (Nov. 1960), pp. 265–272. ISSN: 0022-1481. URL: <https://doi.org/10.1115/1.3679930>.
- [12] Barret, M., Faucher, E. and Hérard, J.-M. “Schemes to Compute Unsteady Flashing Flows”. In: *AIAA Journal* 40.5 (2002), pp. 905–913. DOI: 10.2514/2.1727.
- [13] Behringer, H. “The flow of liquid-gas mixtures in vertical tubes”. In: (Feb. 1952). URL: <https://www.osti.gov/biblio/12405591>.
- [14] Bestion, D. “The physical closure laws in the CATHARE code”. In: *Nuclear Engineering and Design* 124.3 (1990), pp. 229–245. ISSN: 0029-5493. DOI: [https://doi.org/10.1016/0029-5493\(90\)90294-8](https://doi.org/10.1016/0029-5493(90)90294-8).

- [15] Bézenac, E., Pajot, A. and Gallinari, P. “Deep learning for physical processes: incorporating prior scientific knowledge”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.12 (2019), p. 124009. DOI: 10.1088/1742-5468/ab3195. URL: <https://dx.doi.org/10.1088/1742-5468/ab3195>.
- [16] Bilicki, Z. and Kestin, J. “Physical aspects of the relaxation model in two-phase flow”. In: *Proc. R. Soc. Lond.* 428 (1990), pp. 379–397. DOI: 10.1098/rspa.1990.0040.
- [17] Bilicki, Z., Kestin, J. and Pratt, M. M. “A Reinterpretation of the Results of the Moby Dick Experiments in Terms of the Nonequilibrium Model”. In: *Journal of Fluids Engineering* 112.2 (1990), pp. 212–217. ISSN: 0098-2202. DOI: 10.1115/1.2909390.
- [18] Brunel, A., Herbin, R. and Latché, J.-C. “A centered limited finite volume approximation of the momentum convection operator for low-order nonconforming face-centered discretizations”. In: *International Journal for Numerical Methods in Fluids* 96.6 (2024), pp. 1104–1135. DOI: <https://doi.org/10.1002/fld.5276>.
- [19] Brunel, A., Herbin, R. and Latché, J.-C. “A Staggered Scheme for the Compressible Euler Equations on General 3D Meshes”. In: *Journal of Scientific Computing* 100.2 (June 2024), p. 30. ISSN: 1573-7691. DOI: 10.1007/s10915-024-02560-y. URL: <https://doi.org/10.1007/s10915-024-02560-y>.
- [20] Brunton, S. L., Noack, B. R. and Koumoutsakos, P. “Machine Learning for Fluid Mechanics”. In: *Annual Review of Fluid Mechanics* 52 (2020), pp. 477–508. ISSN: 1545-4479. DOI: <https://doi.org/10.1146/annurev-fluid-010719-060214>. URL: <https://www.annualreviews.org/content/journals/10.1146/annurev-fluid-010719-060214>.
- [21] Chelemer, H., Hochreiter, L., Boman, L. and Chu, P. “An improved thermal-hydraulic analysis method for rod bundle cores”. In: *Nuclear Engineering and Design* 41.2 (1977), pp. 219–229. ISSN: 0029-5493. DOI: [https://doi.org/10.1016/0029-5493\(77\)90111-X](https://doi.org/10.1016/0029-5493(77)90111-X).
- [22] Chexal, B. and Lellouche, G. *Full-range drift-flux correlation for vertical flows Revision 1*. Tech. rep. EPRI-NP-3989-SR-Rev1. United States, 1986, p. 93. URL: http://inis.iaea.org/search/search.aspx?orig_q=RN:18019604.
- [23] Chexal, B., Lellouche, G., Horowitz, J. and Healzer, J. “A void fraction correlation for generalized applications”. In: *Progress in Nuclear Energy* 27.4 (1992), pp. 255–295. ISSN: 0149-1970. DOI: [https://doi.org/10.1016/0149-1970\(92\)90007-P](https://doi.org/10.1016/0149-1970(92)90007-P).
- [24] Chexal, B. *Void fraction technology for design and analysis*. EPRI, 1997.
- [25] Chu, P. T., Hochreiter, L. E., Chelemer, H., Boman, L. H. and Tong, L. S. “THINC-IV: a new thermal-hydraulic code for PWR thermal design.” In: *Trans. Amer. Nucl. Soc.* 15: No. 2, 876-8(Nov 1972). (Jan. 1972). URL: <https://www.osti.gov/biblio/4583820>.
- [26] Chung, B. D., Kim, K. D., Bae, S. W., Jeong, J. J., Lee, S. W., Hwang, M. K. and Yoon, C. “MARS code manual volume I: code structure, system models, and solution methods”. In: (2010).
- [27] Ciarlet, P. G. “Discrete Maximum Principle for Finite-Difference Operators.” In: *Aequationes mathematicae* 4 (1970), pp. 338–352. URL: <http://eudml.org/doc/136095>.
- [28] Coddington, P. and Macian, R. “A study of the performance of void fraction correlations used in the context of drift-flux two-phase flow models”. In: *Nuclear Engineering and Design* 215.3 (2002), pp. 199–216. ISSN: 0029-5493.
- [29] Courant, R. and Friedrichs, K. O. *Supersonic Flow and Shock Waves*. Applied Mathematical Sciences. Springer New York, 1999. ISBN: 9780387902326. URL: <https://books.google.fr/books?id=Qsxec0QfYw8C>.
- [30] Cuta, J. M., Koontz, A. S., Stewart, C. W., Montgomery, S. D. and Nomura, K. K. *VIPRE-01: A thermal-hydraulic code for reactor cores*. Tech. rep. Electric Power Research Inst., Palo Alto, CA (USA); Pacific Northwest Lab., Richland, WA (USA), Aug. 1989. URL: <https://www.osti.gov/biblio/5740653>.

- [31] Cybenko, G. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of Control, Signals and Systems* 2.4 (1989), pp. 303–314. ISSN: 1435-568X. DOI: 10.1007/BF02551274.
- [32] Daw, A., Thomas, R. Q., Carey, C. C., Read, J. S., Appling, A. P. and Karpatne, A. “Physics-Guided Architecture (PGA) of Neural Networks for Quantifying Uncertainty in Lake Temperature Modeling”. In: *Proceedings of the 2020 SIAM International Conference on Data Mining (SDM)*. 2020, pp. 532–540. DOI: 10.1137/1.9781611976236.60. URL: <https://epubs.siam.org/doi/abs/10.1137/1.9781611976236.60>.
- [33] Delhaye, J., Giot, M. and Riethmuller, M. *Thermohydraulics of Two-phase Systems for Industrial Design and Nuclear Engineering*. McGraw-Hill Series on Computer Communications. Hemisphere Publishing Corporation, 1981. ISBN: 9780070162686.
- [34] Deng, J., Lu, Q., Liu, Y., Wang, X., Yang, X., Zhang, Y. and Zhang, J. “Review of sub-channel code development for pressurized water reactor and introduction of CORTH-V2.0 sub-channel code”. In: *Progress in Nuclear Energy* 125 (2020), p. 103373. ISSN: 0149-1970. DOI: <https://doi.org/10.1016/j.pnucene.2020.103373>.
- [35] Domanus, H., Shah, V. and Sha, W. “Applications of the COMMIX code using the porous medium formulation”. In: *Nuclear Engineering and Design* 62.1 (1980), pp. 81–100. ISSN: 0029-5493. DOI: [https://doi.org/10.1016/0029-5493\(80\)90022-9](https://doi.org/10.1016/0029-5493(80)90022-9).
- [36] Dubois, F. *Boundary conditions and the Osher scheme for the Euler equations of gas dynamics*. Ecole polytechnique, Centre de mathématiques appliquées, 1987.
- [37] Dubois, F. and Le Floch, P. “Boundary conditions for nonlinear hyperbolic systems of conservation laws”. In: *Journal of Differential Equations* 71.1 (1988), pp. 93–122. ISSN: 0022-0396. DOI: [https://doi.org/10.1016/0022-0396\(88\)90040-X](https://doi.org/10.1016/0022-0396(88)90040-X).
- [38] Eymard, R., Gallouët, T., Herbin, R. and Latché, J.-C. “Finite volume schemes and Lax-Wendriff Consistency”. In: *CRAS Compte Rendu Académie des Sciences*. More than a half century of Computational Fluid Dynamics online first (2022), pp. 1–13. DOI: 10.5802/crmeca.132.. URL: <https://hal.science/hal-03257774>.
- [39] Eymard, R., Gallouët, T. and Herbin, R. *Finite volume methods*. Vol. 7. Handbook of Numerical Analysis. Elsevier, 2000, pp. 713–1018.
- [40] Eymard, R., Henry, G., Herbin, R., Hubert, F., Klöfkom, R. and Manzini, G. “3D Benchmark on Discretization Schemes for Anisotropic Diffusion Problems on General Grids”. In: *Finite Volumes for Complex Applications VI Problems & Perspectives*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 895–930. ISBN: 978-3-642-20671-9.
- [41] Faccanoni, G. “Étude d’un modèle fin de changement de phase liquide-vapeur: contribution à l’étude de la crise d’ébullition.” Thèse en cotutelle entre l’École Polytechnique et l’Université di Trento (Italie). Theses. Ecole Polytechnique X, Nov. 2008. URL: <https://pastel.hal.science/tel-00363460>.
- [42] Faccanoni, G. and Mathis, H. “Admissible equations of state for immiscible and miscible mixtures”. In: *ESAIM: ProcS* 66 (2019), pp. 1–21. DOI: 10.1051/proc/201966001. URL: <https://doi.org/10.1051/proc/201966001>.
- [43] Facchini, A., Lee, J. and Joo, H. G. “Development and Validation of the Two-Phase Flow Analysis Capability of a Drift-flux Model Based Core Thermal-Hydraulics Code ESCOT”. In: *ICAPP 2019* (2019).
- [44] Fathalli, M., Petitpas, F. and Le Martelot, S. “Liquid-vapor flows in heated ducts: Reference solutions”. In: *International Journal of Heat and Mass Transfer* 125 (2018), pp. 1–16. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2018.04.031>.
- [45] Faucher, E. “Simulation numérique des écoulements unidimensionnels instationnaires avec autovaporisation”. PhD thesis. Université Paris XII Val de Marne, Jan. 2000. URL: <https://theses.hal.science/tel-01026372>.

- [46] Fletcher, C. and Schultz, R. *RELAP5/MOD3 Code Manual*. Tech. rep. Nuclear Regulatory Commission, Washington, DC (United States), 1992. URL: <https://www.osti.gov/biblio/5559769>.
- [47] Frolkovič, P. “Maximum principle and local mass balance for numerical solutions of transport equation coupled with variable density flow”. In: *Acta Mathematica Universitatis Comenianae. New Series* 67 (Aug. 1998).
- [48] Fukushima, K. “Cognitron: A self-organizing multilayered neural network”. In: *Biological Cybernetics* 20 (1975), pp. 121–136. URL: <https://doi.org/10.1007/BF00342633>.
- [49] Fukushima, K. “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position”. In: *Biological Cybernetics* 36.4 (1980), pp. 193–202. ISSN: 1432-0770. DOI: 10.1007/BF00344251.
- [50] Gallouët, T., Helluy, P., Hérard, J.-M. and Nussbaum, J. “Hyperbolic relaxation models for granular flows”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 44.2 (Feb. 2011), pp. 371–400. DOI: 10.1051/m2an/2010006. URL: <https://hal.science/hal-00288620>.
- [51] Gallouët, T., Hérard, J.-M. and Seguin, N. “A hybrid scheme to compute contact discontinuities in one-dimensional Euler systems”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 36.6 (2002), pp. 1133–1159.
- [52] Gallouët, T., Hérard, J.-M. and Seguin, N. “Some recent finite volume schemes to compute Euler equations using real gas EOS”. In: *International Journal for Numerical Methods in Fluids* 39.12 (2002), pp. 1073–1138. DOI: <https://doi.org/10.1002/flid.346>.
- [53] Gastaldo, L., Herbin, R. and Latché, J.-C. “A discretization of phase mass balance in fractional step algorithms for the drift-flux model”. In: *IMA Journal of Applied Mathematics* 31.1 (2011), pp. 116–146. URL: <https://hal.science/hal-00308838>.
- [54] Gastaldo, L., Herbin, R. and Latché, J.-C. “An unconditionally stable finite element-finite volume pressure correction scheme for the drift-flux model”. en. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 44.2 (2010), pp. 251–287. URL: <http://www.numdam.org/articles/10.1051/m2an/2010002/>.
- [55] Gavriluk, S. and Gouin, H. “Theoretical model of the Leidenfrost temperature”. In: *Phys. Rev. E* 106 (5 Nov. 2022), p. 055102. DOI: 10.1103/PhysRevE.106.055102. URL: <https://link.aps.org/doi/10.1103/PhysRevE.106.055102>.
- [56] Glorot, X., Bordes, A. and Bengio, Y. “Deep Sparse Rectifier Neural Networks”. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Geoffrey Gordon, David Dunson, and Miroslav Dudík. Vol. 15. Proceedings of Machine Learning Research. PMLR, 2011, pp. 315–323. URL: <https://proceedings.mlr.press/v15/glorot11a.html>.
- [57] Glück, M. “Sub-channel analysis with F-COBRA-TF—Code validation and approaches to CHF prediction”. In: *Nuclear Engineering and Design* 237.6 (2007), pp. 655–667. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2006.08.011>.
- [58] Godlewski, E. and Raviart, P.-A. *Numerical Approximation of Hyperbolic Systems of Conservation Laws*. Vol. 118. Applied Mathematical Sciences. Springer New York, 2013. ISBN: 9781461207139.
- [59] Goodfellow, I., Bengio, Y. and Courville, A. *Deep Learning*. MIT Press, 2016. URL: <http://www.deeplearningbook.org>.
- [60] Grapsas, D., Herbin, R., Latché, J.-C. and Nasser, Y. “Modelling of a spherical deflagration at constant speed”. In: *Computational and Applied Mathematics* 40.5 (2021), p. 163. ISSN: 1807-0302. DOI: 10.1007/s40314-021-01543-1.
- [61] Grégoire, O. and Martin, M. “Derivation of a well-posed and multidimensional drift-flux model for boiling flows”. In: *Comptes Rendus Mécanique* 333.6 (2005), pp. 459–466. ISSN: 1631-0721. DOI: <https://doi.org/10.1016/j.crme.2005.05.002>.

- [62] Greydanus, S., Dzamba, M. and Yosinski, J. “Hamiltonian Neural Networks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 32. Curran Associates, Inc., 2019, pp. 15353–15363. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/26cd8ecadce0d4efd6cc8a8725cbd1f8-Paper.pdf.
- [63] Griffith, P. and Wallis, G. B. “Two-Phase Slug Flow”. In: *Journal of Heat Transfer* 83.3 (Aug. 1961), pp. 307–318. ISSN: 0022-1481. URL: <https://doi.org/10.1115/1.3682268>.
- [64] Gronwall, T. H. “Note on the Derivatives with Respect to a Parameter of the Solutions of a System of Differential Equations”. In: *Annals of Mathematics* 20.4 (1919), pp. 292–296. ISSN: 0003486X. URL: <http://www.jstor.org/stable/1967124>.
- [65] Guelfi, A., Bestion, D., Boucker, M., Boudier, P., Fillion, P., Grandotto, M., Hérard, J.-M., Hervieu, E. and Péturaud, P. “NEPTUNE: A New Software Platform for Advanced Nuclear Thermal Hydraulics”. In: *Nuclear Science and Engineering* 156.3 (2007), pp. 281–324. DOI: 10.13182/NSE05-98. URL: <https://doi.org/10.13182/NSE05-98>.
- [66] Gui, M., Liu, Z., Wang, T., Sui, Z. and Bi, Q. “Void fractions in a rod bundle geometry at high pressure—Part II : Drift-flux model assessment and development”. In: *International Journal of Multiphase Flow* 125 (2020). ISSN: 0301-9322. DOI: <https://doi.org/10.1016/j.ijmultiphaseflow.2020.103231>.
- [67] Guillemaud, V. “Modélisation et simulation numérique des écoulements diphasiques par une approche bifluide à deux pressions”. Theses. Université de Provence - Aix-Marseille I, Mar. 2007. URL: <https://theses.hal.science/tel-00169178>.
- [68] Helluy, Philippe and Lazare, Gauthier. “Analysis and optimization of a liquid-vapor thermohydraulic model”. In: *ESAIM: Proceedings and Surveys* 81 (2025), pp. 77–103. DOI: 10.1051/proc/202581077. URL: <https://doi.org/10.1051/proc/202581077>.
- [69] Helluy, P., Hérard, J.-M. and Mathis, H. “A well-balanced approximate Riemann solver for compressible flows in variable cross-section ducts”. In: *Journal of Computational and Applied Mathematics* 236.7 (2012), pp. 1976–1992. ISSN: 0377-0427. DOI: <https://doi.org/10.1016/j.cam.2011.11.008>.
- [70] Hérard, J.-M. and Lazare, G. “Two unsteady solutions of a homogeneous equilibrium model in a porous medium”. In: *ESAIM: Proceedings and Surveys, in Press* (2024). URL: <https://hal.science/hal-04733944>.
- [71] Herbin, R., Latché, J.-C. and Saleh, K. “Low Mach number limit of some staggered schemes for compressible barotropic flows”. In: *Mathematics of Computation* 90.329 (Feb. 2021), pp. 1039–1087. DOI: 10.1090/mcom/3604. URL: <https://hal.science/hal-01672909>.
- [72] Hibiki, T. and Ishii, M. “One-dimensional drift-flux model and constitutive equations for relative motion between phases in various two-phase flow regimes”. In: *International Journal of Heat and Mass Transfer* 46.25 (2003), pp. 4935–4948. ISSN: 0017-9310. DOI: [https://doi.org/10.1016/S0017-9310\(03\)00322-3](https://doi.org/10.1016/S0017-9310(03)00322-3).
- [73] Hibiki, T. and Ishii, M. “One-dimensional drift-flux model for two-phase flow in a large diameter pipe”. In: *International Journal of Heat and Mass Transfer* 46.10 (2003), pp. 1773–1790. ISSN: 0017-9310. DOI: [https://doi.org/10.1016/S0017-9310\(02\)00473-8](https://doi.org/10.1016/S0017-9310(02)00473-8).
- [74] Hinton, G., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. “Improving neural networks by preventing co-adaptation of feature detectors”. In: *arXiv* (2012).
- [75] Hornik, K., Stinchcombe, M. and White, H. “Multilayer feedforward networks are universal approximators”. In: *Neural networks* 2.5 (1989), pp. 359–366.
- [76] Hou, T. Y. and LeFloch, P. G. “Why nonconservative schemes converge to wrong solutions: error analysis”. In: *Mathematics of computation* 62.206 (1994), pp. 497–530.
- [77] Huang, J., Gou, J., Ding, W., Zhang, B. and Shan, J. “Development of a sub-channel analysis code based on the two-fluid model and its preliminary assessment”. In: *Nuclear Engineering and Design* 340 (2018), pp. 17–30. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2018.09.023>.

- [78] Hurisse, O. “Numerical simulations of steady and unsteady two-phase flows using a homogeneous model”. In: *Computers & Fluids* 152 (2017), pp. 88–103. ISSN: 0045-7930. DOI: <https://doi.org/10.1016/j.compfluid.2017.04.007>.
- [79] Imke, U. and Sanchez, V. H. “Validation of the Subchannel Code SUBCHANFLOW Using the NUPEC PWR Tests (PSBT)”. In: *Science and Technology of Nuclear Installations* 2012.1 (2012), p. 465059. DOI: <https://doi.org/10.1155/2012/465059>.
- [80] Inoue, T. “An excess motion of the ascending node of mercury in the observations used by Le Verrier”. In: *Celestial Mechanics and Dynamical Astronomy* 56.1-2 (1993), p. 69.
- [81] Ishii, M. *One-dimensional drift-flux model and constitutive equations for relative motion between phases in various two-phase flow regimes*. Tech. rep. Argonne National Lab., Ill. (USA), Oct. 1977. DOI: 10.2172/6871478.
- [82] Ishii, M. *Thermo-fluid Dynamics Theory of Two-phase Flow*. Eyrolles, 1975.
- [83] Ishii, M. and Hibiki, T. *Thermo-fluid dynamics of two-phase flow*. Springer Science & Business Media, 2006.
- [84] Jeon, J., Lee, J. and Kim, S. J. “Finite volume method network for the acceleration of unsteady computational fluid dynamics: Non-reacting and reacting flows”. In: *International Journal of Energy Research* 46.8 (2022), pp. 10770–10795. DOI: <https://doi.org/10.1002/er.7879>.
- [85] Juliá, J., Hibiki, T., Ishii, M., Yun, B.-J. and Park, G.-C. “Drift-flux model in a sub-channel of rod bundle geometry”. In: *International Journal of Heat and Mass Transfer* 52 (June 2009), pp. 3032–3041. DOI: 10.1016/j.ijheatmasstransfer.2009.02.012.
- [86] Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S. and Yang, L. “Physics-informed machine learning”. In: *Nature Reviews Physics* 3.6 (2021), pp. 422–440. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00314-5.
- [87] Kelly, J. M., Stewart, C. W. and Cuta, J. M. “VIPRE-02- a two-fluid thermal-hydraulics code for reactor core and vessel analysis; Mathematical modeling and solution methods”. In: *Nuclear Technology; (United States)* 100:2 (Nov. 1992). ISSN: ISSN 0029-5450. URL: <https://www.osti.gov/biblio/7163755>.
- [88] Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M. and Tang, P. T. P. *On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima*. 2017. arXiv: 1609.04836 [cs.LG]. URL: <https://arxiv.org/abs/1609.04836>.
- [89] Kingma, D. and Ba, J. “Adam: A Method for Stochastic Optimization”. In: *International Conference on Learning Representations* (2014).
- [90] Lagaris, I., Likas, A. and Fotiadis, D. “Artificial neural networks for solving ordinary and partial differential equations”. In: *IEEE Transactions on Neural Networks* 9.5 (1998), pp. 987–1000. DOI: 10.1109/72.712178.
- [91] Larrouturou, B. *How to preserve the mass fractions positivity when computing compressible multi-component flows*. Research Report RR-1080. 1989. URL: <https://inria.hal.science/inria-00075479>.
- [92] Lazare, G., Coudroy, C., Helluy, P., Le Coupanec, E., Feng, Q., Hulsemann, F., Pujet, S. and Rotach, P. “Acceleration of the Convergence of a Core Thermal Hydraulic Code Using Initialization from a Neural Network”. In: *21st International Topical Meeting on Nuclear Reactor Thermal Hydraulics*. Aug. 2025. URL: <https://hal.science/hal-05321965>.
- [93] Lazare, G., Feng, Q., Helluy, P., Hérard, J.-M., Hulsemann, F. and Pujet, S. “Maximum principle for the mass fraction in a system with two mass balance equations”. In: *Comptes Rendus. Mécanique* 352 (2024), pp. 81–98. DOI: 10.5802/crmeca.244.
- [94] Le Corre, J.-M. “Validation of VIPRE-W/MEFISTO-T Subchannel Analysis Code for Void Fraction Predictions in Modern BWR Fuel Assemblies”. In: Jan. 2023, pp. 5526–5540. DOI: 10.13182/NURETH20-40152.
- [95] Le Métayer, O., Massoni, J. and Saurel, R. “Élaboration des lois d’état d’un liquide et de sa vapeur pour les modèles d’écoulements diphasiques”. In: *International Journal of Thermal*

- Sciences* 43.3 (2004), pp. 265–276. ISSN: 1290-0729. DOI: <https://doi.org/10.1016/j.ijthermalsci.2003.09.002>.
- [96] Le Métayer, O. and Saurel, R. “The Noble-Abel Stiffened-Gas equation of state”. In: *Physics of Fluids* 28.4 (Apr. 2016), p. 046102. ISSN: 1070-6631. DOI: 10.1063/1.4945981. URL: <https://doi.org/10.1063/1.4945981>.
- [97] Lee, J., Facchini, A. and Joo, H. G. “Development of a drift-flux model based core thermal-hydraulics code for efficient high-fidelity multiphysics calculation”. In: *Nuclear Engineering and Technology* 51.6 (2019), pp. 1487–1503. ISSN: 1738-5733. DOI: <https://doi.org/10.1016/j.net.2019.04.002>.
- [98] Lemaire, C. “Mechanical reliability of structures subjected to time-variant physical phenomena”. PhD thesis. France: Institut National Polytechnique de Grenoble, France (in French), Nov. 1999, p. 281. URL: http://inis.iaea.org/search/search.aspx?orig_q=RN:48055203.
- [99] Lewandowski, R. and Mohammadi, B. “Existence and positivity results for the φ - θ and a modified $k - \varepsilon$ two-equation turbulence models”. In: *Mathematical Models and Methods in Applied Sciences* 03.02 (1993), pp. 195–215. URL: <https://doi.org/10.1142/S0218202593000114>.
- [100] Lipnikov, K., Svyatskiy, D. and Vassilevski, Y. “Minimal stencil finite volume scheme with the discrete maximum principle”. In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 27.4 (2012), pp. 369–386. URL: <https://doi.org/10.1515/rnam-2012-0020>.
- [101] Maier, D. and Coddington, P. “Validation of RETRAN-03 against a wide range of rod bundle void fraction data”. In: *Transactions of the American Nuclear Society* (1996), pp. 372–374. ISSN: 0003-018X. URL: http://inis.iaea.org/search/search.aspx?orig_q=RN:28028314.
- [102] Mekkas, A., Anne, C. and Sami, K. “A new numerical algorithm for two-phase flows drift-flux model with staggered grid in porous media”. In: *ESAIM: Proceedings and Surveys* 58 (2017), pp. 58–77. DOI: 10.1051/proc/201758058. URL: <https://cea.hal.science/cea-02389194>.
- [103] Mérigoux, N., Laviéville, J., Mimouni, S., Guingo, M., Baudry, C. and Bellet, S. “Verification, validation and application of NEPTUNE CFD to two-phase Pressurized Thermal Shocks”. In: *Nuclear Engineering and Design* 312 (2017). 16th International Topical Meeting on Nuclear Reactor Thermal Hydraulics, pp. 74–85. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2016.06.041>.
- [104] Miao, C., Baumann, W., Domanus, H., Shah, V. and Sha, W. “Two-phase thermal-hydraulic simulations with COMMIX-2”. In: *Nuclear Engineering and Design* 82.2 (1984), pp. 205–214. ISSN: 0029-5493. DOI: [https://doi.org/10.1016/0029-5493\(84\)90212-7](https://doi.org/10.1016/0029-5493(84)90212-7).
- [105] Miwa, S., Hibiki, T. and Katono, K. “Performance of drift-flux correlations for predicting void fraction of two-phase flow in tight-lattice rod bundles”. In: *International Journal of Heat and Mass Transfer* 191 (2022), p. 122664. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2022.122664>.
- [106] Mohan, A. T., Lubbers, N., Chertkov, M. and Livescu, D. “Embedding hard physical constraints in neural network coarse-graining of three-dimensional turbulence”. In: *Phys. Rev. Fluids* 8 (1 2023), p. 014604. DOI: 10.1103/PhysRevFluids.8.014604.
- [107] Nakoryakov, V. E., Pokusaev, B. G. and Shreiber, I. R. *Wave propagation in gas-liquid media*. CRC Press, 1993.
- [108] Neal, L. G. “An analysis of slip in gas-liquid flow applicable to the bubble and slug flow regimes”. In: (Dec. 1963). URL: <https://www.osti.gov/biblio/4090193>.
- [109] Nicklin, D. J., Wilkes, J. O. and Davidson, J. F. “Two-Phase Flow in Vertical Tube”. In: *Trans. Instn. Chem. Engrs.* 40 (1962), pp. 61–68.

- [110] Novak, A. J., Zou, L., Peterson, J. W., Andrs, D., Kelly, J., Slaybaugh, R. N., Martineau, R. C. and Gougar, H. D. *Pronghorn theory manual*. Tech. rep. Idaho National Lab.(INL), Idaho Falls, ID (United States), 2018.
- [111] Nowlan, S. J. and Hinton, G. E. “Simplifying Neural Networks by Soft Weight-Sharing”. In: *Neural Computation* 4.4 (1992), pp. 473–493. DOI: 10.1162/neco.1992.4.4.473.
- [112] NRC US. “TRACE V5.0 User’s manual”. In: *Division of Safety Analysis Office of Nuclear Regulatory Research US Nuclear Regulatory Commission Washington, DC* (2010), pp. 20555–0001.
- [113] Ozaki, T. and Hibiki, T. “Drift-flux model for rod bundle geometry”. In: *Progress in Nuclear Energy* 83 (2015), pp. 229–247. ISSN: 0149-1970. DOI: <https://doi.org/10.1016/j.pnucene.2015.03.015>.
- [114] Panyotov, D., Ilieva, B., Avramova, M., Toshev, V., Argirov, J. and Bylgarska Akademiya na Naukite, S. B. *Some research activities of the Institute for Nuclear Research and Nuclear Energy in the development, validation and application of the reactor safety and thermal-hydraulics computer codes*. Tech. rep. Bylgarska Akademiya na Naukite, Sofia (Bulgaria). Inst. za Yadrena, 1996.
- [115] Pinkus, A. “Approximation theory of the MLP model in neural networks”. In: *Acta Numerica* 8 (1999), pp. 143–195. DOI: 10.1017/S0962492900002919.
- [116] Quintard, M. and Whitaker, S. “Transport in ordered and disordered porous media II: Generalized volume averaging”. In: *Transport in Porous Media* 14.2 (1994), pp. 179–206. ISSN: 1573-1634. DOI: 10.1007/BF00615200.
- [117] Raissi, M., Perdikaris, P. and Karniadakis, G. “Physics Informed Deep Learning (Part II): Data-driven Discovery of Nonlinear Partial Differential Equations”. In: (2017). DOI: 10.48550/arXiv.1711.10566.
- [118] Rao, Y., Cheng, Z., Waddington, G. and Nava-Dominguez, A. “ASSERT-PV 3.2: Advanced subchannel thermalhydraulics code for CANDU fuel bundles”. In: *Nuclear Engineering and Design* 275 (2014), pp. 69–79. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2014.04.016>.
- [119] Rowe, D. S. *COBRA IIIC: digital computer program for steady state and transient thermal-hydraulic analysis of rod bundle nuclear fuel elements*. Tech. rep. Battelle Pacific Northwest Labs., Richland, WA (United States), Mar. 1973. DOI: 10.2172/4480166. URL: <https://www.osti.gov/biblio/4480166>.
- [120] Rowe, D. *Cross-Flow Mixing Between Parallel Flow Channels during Boiling. Part I. COBRA: Computer Programm for Coolant Boiling in Rod Arrays*. Tech. rep. Battelle Pacific Northwest Labs., Richland, WA (United States), 1967.
- [121] Rubim, A., A. S., Avramora, M., Ustuno, H., Bajorek, S. and Velazquez-Lozada. *OECD/NRC benchmark based on NUPEC PWR subchannel and bundle tests (PSBT). Volume I: experimental database and final problem specifications*. Tech. rep. 2010.
- [122] Ruckenstein, E. “On heat transfer between vapour bubbles in motion and the boiling liquid from which they are generated”. In: *Chemical Engineering Science* 10.1 (1959), pp. 22–30. ISSN: 0009-2509. DOI: [https://doi.org/10.1016/0009-2509\(59\)80021-X](https://doi.org/10.1016/0009-2509(59)80021-X).
- [123] Rumelhart, D. E., Hinton, G. E. and Williams, R. J. “Learning representations by back-propagating errors”. In: *Nature* 323.6088 (1986), pp. 533–536. ISSN: 1476-4687. DOI: 10.1038/323533a0.
- [124] Saha, P. and Zuber, N. “Point of net vapor generation and vapor void fraction in subcooled boiling”. In: *International Heat Transfer Conference 5*. Vol. IV. Jan. 1974. DOI: 10.1615/IHTC5.430.
- [125] Salko, R., Wysocki, A., Blyth, T., Toptan, A., Hu, J., Kumar, V., Dances, C., Dawn, W., Sung, Y., Kucukboyaci, V., Gurecky, W., Lange, T., Zhao, X., Rader, J., Jernigan, C., Collins, B., Avramova, M., Magedanz, J., Palmtag, S., Clarno, K., Kropaczek, D., Hizoum, B., Godfrey, A., Pointer, D., Turner, J., Sankaran, R., Schmidt, R., Hooper, R.,

- Bartlett, R., Baird, M. and Pilch, M. “CTF: A modernized, production-level, thermal hydraulic solver for the solution of industry-relevant challenge problems in pressurized water reactors”. In: *Nuclear Engineering and Design* 397 (2022), p. 111927. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2022.111927>.
- [126] Sanchez, V., Imke, U., Ivanov, A. and Gomez, R. “SUBCHANFLOW: a thermal hydraulic sub-channel program to analyse fuel rod bundles and reactor cores”. In: (2010).
- [127] Schropff, S., Petitpas, F. and Daniel, E. “Reference solutions for compressible single-phase flows in heated and cooled ducts”. In: *Physics of Fluids* 36.5 (2024), p. 056126. ISSN: 1070-6631. DOI: 10.1063/5.0209500.
- [128] Shan, J., Zhang, B., Li, C. and Leung, L. K. “SCWR subchannel code ATHAS development and CANDU-SCWR analysis”. In: *Nuclear Engineering and Design* 239.10 (2009), pp. 1979–1987. ISSN: 0029-5493. DOI: <https://doi.org/10.1016/j.nucengdes.2009.05.008>.
- [129] Sirignano, J. and Spiliopoulos, K. “DGM: A deep learning algorithm for solving partial differential equations”. In: *Journal of Computational Physics* 375 (2018), pp. 1339–1364. ISSN: 0021-9991. DOI: <https://doi.org/10.1016/j.jcp.2018.08.029>.
- [130] Smoller, J. *Shock waves and reaction—diffusion equations*. Vol. 258. Springer Science & Business Media, 2012.
- [131] Sod, G. A. “A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws”. In: *Journal of computational physics* 27.1 (1978), pp. 1–31.
- [132] Srikantiah, G. S. “VIPRE - A Reactor Core Thermal-Hydraulics Analysis Code for Utility Applications”. In: *Nuclear Technology* 100.2 (1992), pp. 216–227. DOI: 10.13182/NT92-A34744. URL: <https://doi.org/10.13182/NT92-A34744>.
- [133] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. “Dropout: a simple way to prevent neural networks from overfitting”. In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 1929–1958. ISSN: 1532-4435.
- [134] Stewart, C. W., Wheeler, C. L., Cena, R. J., McMonagle, C. A., Cuta, J. M. and Trent, D. S. *COBRA-IV: the model and the method*. Tech. rep. Pacific Northwest National Lab. (PNNL), Richland, WA (United States), July 1977. DOI: 10.2172/5358588. URL: <https://www.osti.gov/biblio/5358588>.
- [135] Thurgood, M., Kelly, J., Basehore, K. and George, T. *COBRA-TF: a three-field two-fluid model for reactor safety analysis*. American Society of Mechanical Engineers., 1980, pp. 31–39.
- [136] Toumi, I., Bergeron, A., Gallo, D., Royer, E. and Caruge, D. “FLICA-4: a three-dimensional two-phase flow computer code with advanced numerical methods for nuclear applications”. In: *Nuclear Engineering and Design* 200.1 (2000), pp. 139–155. ISSN: 0029-5493. DOI: [https://doi.org/10.1016/S0029-5493\(99\)00332-5](https://doi.org/10.1016/S0029-5493(99)00332-5).
- [137] Varga, R. S. *Matrix Iterative Analysis*. Springer Berlin, Heidelberg, 2000.
- [138] Vinuesa, R. and Brunton, S. L. “Enhancing computational fluid dynamics with machine learning”. In: *Nature Computational Science* 2.6 (2022), pp. 358–366. ISSN: 2662-8457. DOI: 10.1038/s43588-022-00264-7.
- [139] Wagner, W., Cooper, J. R., Dittmann, A., Kijima, J., Kretzschmar, H.-J., Kruse, A., Mares, R., Oguchi, K., Sato, H., Stöcker, I., Sifner, O., Takaishi, Y., Tanishita, I., Trübenbach, J. and Willkommen, T. “The IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam”. In: *Journal of Engineering for Gas Turbines and Power* 122.1 (Jan. 2000), pp. 150–184. ISSN: 0742-4795. DOI: 10.1115/1.483186. URL: <https://doi.org/10.1115/1.483186>.
- [140] Whitaker, S. “Flow in porous media I: A theoretical derivation of Darcy’s law”. In: *Transport in Porous Media* 1.1 (1986), pp. 3–25. ISSN: 1573-1634. DOI: 10.1007/BF01036523.
- [141] Wu, W., Ding, W., Hampel, U. and Sun, B. “Analysis of the influence of swirling flow on the boiling heat transfer characteristics of two-phase flow”. In: *International Journal of*

- Heat and Mass Transfer* 221 (2024), p. 125075. ISSN: 0017-9310. DOI: <https://doi.org/10.1016/j.ijheatmasstransfer.2023.125075>.
- [142] Yoo, Y.-J., Hwang, D.-H. and Sohn, D.-S. “Development of a subchannel analysis code MATRA applicable to PWRs and ALWRs”. In: *Nuclear Engineering and Technology* 31.3 (1999), pp. 314–327.
 - [143] Yoon, H. Y., Lee, J. R., Kim, H., Park, I. K., Song, C.-H., Cho, H. K. and Jeong, J. J. “Recent improvements in the CUPID code for a multi-dimensional two-phase flow analysis of nuclear reactor components”. In: *Nuclear Engineering and Technology* 46.5 (2014), pp. 655–666. ISSN: 1738-5733. DOI: <https://doi.org/10.5516/NET.02.2014.023>.
 - [144] Yoon, S. J., Kim, S. B., Park, G. C., Yoon, H. Y. and Cho, H. K. “Application of CUPID for subchannel-scale thermal–hydraulic analysis of pressurized water reactor core under single-phase conditions”. In: *Nuclear Engineering and Technology* 50.1 (2018), pp. 54–67. ISSN: 1738-5733. DOI: <https://doi.org/10.1016/j.net.2017.09.008>.
 - [145] Zuber, N. and Findlay, J. “Average Volumetric Concentration in Two-Phase Flow Systems”. In: *J. Heat Transfer* 87.4 (1965), pp. 453–468. DOI: 10.1115/1.3689137.

Pour assurer la sûreté des REP, EDF a développé le code de thermo-hydraulique THYC-coeur qui simule l'écoulement de l'eau dans le coeur. Ce code permet de calculer en tout point du coeur le Rapport de Flux Thermique Critique (RFTC), mesurant l'écart entre le flux de chaleur réel et le flux de chaleur critique à ne pas dépasser pour ne pas causer la crise d'ébullition qui peut menacer l'intégrité du coeur du réacteur.

THYC-coeur utilise des schémas volumes finis et est principalement utilisé pour résoudre des problèmes stationnaires à des instants critiques d'accidents. Malgré un temps de calcul réduit (environ 10 secondes pour une configuration industrielle de coeur de réacteur), un très grand nombre de calculs (plusieurs millions) est nécessaire. L'objectif de cette thèse est donc d'accélérer le temps de calcul du code THYC-coeur.

Pour cela, les modèles de drift-flux sont présentés (Chapitre 1) et analysés (Chapitre 2). Plusieurs solutions analytiques de ces modèles sont présentées (Chapitre 3). Par la suite, plusieurs méthodes d'accélération sont proposées. Tout d'abord, des nouveaux schémas volumes finis pour l'équation de déséquilibre du modèle sont proposés et vérifiés (Chapitre 4). Une fois implémentées dans le code industriel, la robustesse et l'accélération obtenue sont évaluées (Chapitre 5). Afin de pouvoir étudier plus facilement certains phénomènes numériques et tester une méthode d'accélération par IA, un prototype simplifié de THYC-coeur, appelé ThermoTorch est développé (Chapitre 6). Une vérification en stationnaire et en instationnaire du schéma numérique global est réalisé (Chapitre 7). Enfin, une méthode d'accélération par IA utilisant une initialisation issue de réseaux de neurones a été testée (Chapitre 8).

INSTITUT DE RECHERCHE MATHÉMATIQUE AVANCÉE

UMR 7501

Université de Strasbourg et CNRS

7 Rue René Descartes

67 084 STRASBOURG CEDEX

cnrs

Tél. 03 68 85 01 29

Fax 03 68 85 03 28

<https://irma.math.unistra.fr>

web-irma@math.unistra.fr

Université

de Strasbourg

IRMA

Institut de Recherche
Mathématique Avancée

ISSN 0755-3390

IRMA 2025/008

<https://tel.archives-ouvertes.fr/tel-00000000>

Gauthier LAZARE

Développement d'une méthode numérique performante pour la résolution d'un modèle diphasique homogène partiellement déséquilibré en milieu poreux hétérogène.

Résumé

EDF a développé le code thermo-hydraulique THYC-cœur pour simuler l'écoulement d'eau dans le cœur des Réacteurs à Eau Pressurisée (REP) et évaluer le Rapport de Flux Thermique Critique (RFTC), indicateur de la marge à la crise d'ébullition pouvant menacer l'intégrité du cœur. THYC-cœur résout principalement des problèmes stationnaires via des schémas volumes finis. Bien que le calcul soit rapide (10 s pour une configuration industrielle), le grand nombre de simulations requises (plusieurs millions) rend nécessaire une accélération du code, objectif de cette thèse. Les modèles drift-flux sont d'abord présentés et analysés. Plusieurs solutions analytiques sont ensuite construites. Afin d'étudier plus facilement certains phénomènes numériques, un prototype simplifié, appelé ThermoTorch est développé. Après une vérification en stationnaire et en instationnaire du schéma numérique global, une méthode d'accélération utilisant une initialisation issue de réseaux de neurones a été testée.

Mots clés : Thermo-hydraulique, schémas volumes finis, sûreté nucléaire, écoulement diphasique, équations aux dérivées partielles.

Abstract

EDF has developed the thermo-hydraulic code THYC-coeur to simulate water flow in the cores of PWRs (Pressurized Water Reactors) and to evaluate the Departure from Nucleate Boiling Ratio (DNBR), an indicator of the margin to the boiling crisis that could threaten core integrity. THYC-coeur mainly solves steady-state problems using finite volume schemes. Although each computation is fast (about 10 seconds for an industrial configuration), the very large number of required simulations (several million) makes code acceleration necessary, which is the main goal of this thesis. The drift-flux models are first presented and analyzed. Several analytical solutions are then constructed. To more easily study certain numerical phenomena, a simplified prototype called ThermoTorch has been developed. After verifying the global numerical scheme in both steady-state and transient regimes, an acceleration method using initialization from neural networks was tested.

Keywords: Thermo-hydraulics, finite volume schemes, nuclear safety, two-phase flows, partial differential equations

