Dissertation submitted for the degree of

Doctor of Philosophy

ALE and SPH formulations for Fluid Structure Interaction: Shock waves impact

Ramzi MESSAHEL

Thesis defended the 30 March 2016 in front of jury:

H. Naji	Professor	LGCgE, Université d'Artois	President
E. Longatte	Research Engineer-HDR	EDF, Electricité De France	Reviewer
E. Lefrançois	Professor	ROBERVAL, UTC Compiègne	Reviewer
G. Mompean	Professor	LML, Université de Lille 1	Examiner
E. Liberge	Associate Professor	LASIE, Université de La Rochelle	Examiner
M. Moatamedi	Professor	NUC Narvik University College	Guest
H. A. Khawaja	Associate Professor	UiT The Arctic University of Norway	Guest
A. Boudlal	Associate Professor	LML, Université de Lille 1	Guest
M. Souli	Professor	LML, Université de Lille 1	Supervisor

Université de Lille 1 – Sciences et Technologies   Order number : 42008

Ecole Doctorale Sciences pour l'Ingénieur (EDSPI 72)

Laboratoire De Mécanique de Lille (LML – FRE3723)

Dissertation submitted for the degree of

Doctor of Philosophy

ALE and SPH formulations for Fluid Structure Interaction: Shock waves impact

Ramzi MESSAHEL

Thesis defended the 30 March 2016 in front of jury:

H. Naji	Professor	LGCgE, Université d'Artois	President
E. Longatte	Research Engineer-HDR	EDF, Electricité De France	Reviewer
E. Lefrançois	Professor	ROBERVAL, UTC Compiègne	Reviewer
G. Mompean	Professor	LML, Université de Lille 1	Examiner
E. Liberge	Associate Professor	LASIE, Université de La Rochelle	Examiner
M. Moatamedi	Professor	NUC Narvik University College	Guest
H. A. Khawaja	Associate Professor	UiT The Arctic University of Norway	Guest
A. Boudlal	Associate Professor	LML, Université de Lille 1	Guest
M. Souli	Professor	LML, Université de Lille 1	Supervisor

Thèse présentée devant l'Université de Lille 1 pour obtenir le grade de Docteur

Spécialité : Mécanique

par

Ramzi MESSAHEL

Formulations ALE et SPH en Interaction Fluide-Structure :

Impact d'ondes de choc

Thèse soutenue le 30 Mars 2016, devant le jury composé de:

H. Naji	Professeur	LGCgE, Université d'Artois	Président du Jury
E. Longatte	Ingénieur de Recherche-HDR	EDF, Electricité De France	Rapporteur
E. Lefrançois	Professeur	ROBERVAL, UTC Compiègne	Rapporteur
G. Mompean	Professeur	LML, Université de Lille 1	Examinateur
E. Liberge	Maître de Conférence	LASIE, Université de La Rochelle	Examinateur
M. Moatamedi	Professeur	NUC Narvik University College	Invité
H. A. Khawaja	Maître de Conférence	UiT The Arctic University of Norway	Invité
A. Boudlal	Maître de Conférence	LML, Université de Lille 1	Invité
M. Souli	Professeur	LML, Université de Lille 1	Directeur de thèse

This thesis has been prepared at

I know that I know nothing.

Socrates

Truth is ever to be found in
simplicity, and not in the
multiplicity and confusion of
things.

Isaac Newton

I've never lost a game I just ran out
of time.

Michael Jordan

It always seems impossible until
it's done.

Nelson Mandela

ALE and SPH formulations for Fluid Structure Interaction: Shock waves impact

Abstract

This thesis focuses on the numerical study of the propagation of shock waves in compressible multiphase flows and fluid structure interaction. Two approaches are being studied for the numerical solution of the fluid part: the ALE approach (Arbitrary Lagrangian Eulerian) present in the commercial code LS-DYNA and the SPH (Smoothed Particle Hydrodynamics) Lagrangian approach; while the structure part is solved by a conventional FE (Finite Element). The numerical investigation of the ALE and SPH methods are the two main areas of research.

The combination of the ALE formalism to fluid structure coupling techniques are used to solve complex industrial problems. Water Hammers phenomena occuring in nuclear industries are investigated in this thesis. During a Water Hammer, the shock waves reflections in nuclear piping may drop locally the water pressure below its saturation pressure and generate cavitation. The HEM (Homogeneous Equilibrium Model) three equations phase change model proposed by Saurel et al. (1999) (temperatures, pressures and velocities equilibrium between the vapor and liquid phases) is studied and applied to solve water hammers. The obtained results are compared with experimental data.

The meshless Lagrangian nature of the SPH method made it very competitive with the grid methods for solving multiphase flows. The traditional derivation of the WC-SPH (Weakly Compressible SPH) method, which is present in most commercial code, assumes that the smoothing length is almost constant, and therefore implies that the spatial derivative of the kernel function is calculated only with respect to position vector. Despite the use of renormalization techniques, instabilities (numerical oscillations) are developed at the interface between particles of different materials. These instabilities restrict the use of traditional WC-SPH schemes to problems with low density ratio. This limitation was remedied in the weakly compressible regime by Hu and Adams (2006). In order to solve the shock problems in the compressible regime, the schemes originally proposed by Hu and Adams (2006) is adapted to fully compressible regime (FC-SPH) by considering the coupling between the density and the smoothing length as proposed by Price (2004). The different SPH schemes are compared for 1-D and 2-D multiphase shock problems. Validation is performed in comparison with exact solutions for 1-D problems and ALE solution for 2-D problems.

Keywords: arbitrary lagrangian-eulerian (ale), smoothed particle hydrodynamics (sph), fluid structure interaction (fsi), shock waves, multiphase flow

FORMULATIONS ALE ET SPH EN INTÉRACTION FLUIDE-STRUCTURE : IMPACT D'ONDES DE CHOC

Résumé

Ce travail de thèse porte sur l'étude numérique de la propagation d'ondes de choc dans les écoulements compressibles multiphasiques et en interaction (fluide-structure). Deux approches sont étudiées pour la résolution numérique de la partie fluide : L'approche ALE (Arbitrary Lagrangian Eulerian) présente dans le code de calcul LS-DYNA et l'approche lagrangienne SPH (Smoothed Particle Hydrodynamics) ; la partie structure, quant à elle, est résolue par une approche classique EF (Éléments finis). L'étude des méthodes ALE et SPH constituent les deux principaux axes de recherche.

La combinaison du formalisme ALE aux techniques de couplage fluide-structure est utilisée pour résoudre des problèmes industriels complexes. La problématique des coups de bélier dans l'ingénierie nucléaire est abordée dans cette thèse. Lors d'un coup de bélier, les nombreuses réflexions d'ondes de choc dans les tuyauteries nucléaires peuvent faire baisser la pression de l'eau en dessous de sa pression de saturation et générer localement de la cavitation. Le modèle HEM (Homogeneous Equilibrium Model) de changement de phase proposé par Saurel et al. (1999) à trois équations (équilibre des températures, pressions et vitesses entre les phases vapeur et liquide) est étudié et appliqué aux coups de bélier. Les résultats obtenus sont comparés aux données expérimentales.

La nature lagrangienne de la méthode sans maillage SPH l'a rendue très compétitive par rapport aux méthodes à grille pour la résolution d'écoulements multiphasiques. La dérivation des méthodes classiques WC-SPH (Weakly Compressible SPH), présentes dans la plupart des codes commerciaux, suppose que la longueur de lissage est quasi-constante, et donc implique que la dérivée spatiale du noyau d'interpolation ne repose que sur le vecteur position. Malgré l'utilisation des techniques de renormalisation, des instabilités (oscillations numériques) se développent à l'interface entre des particules de matériaux différents. Ces instabilités restreignent l'utilisation des schémas classiques WC-SPH pour des problèmes à faible ratio de densité. Cette limite fut palliée dans le régime faiblement compressible par Hu et Adams (2006). Afin de résoudre les problèmes de choc, le schéma proposé par Hu et Adams (2006) est adapté au régime fortement compressible en considérant le couplage entre la densité et la longueur de lissage tel que proposé par Price (2004). Les différents schémas SPH sont comparés entre eux pour les problèmes de chocs multiphasiques en 1-D et 2-D. Les résultats SPH sont validés avec la solution exacte pour les problèmes 1-D et la solution ALE pour les problèmes 2-D.

Mots clés : arbitrary lagrangian-eulerian (ale), smoothed particle hydrodynamics (sph), interaction fluide-structure (ifs), onde de choc, ecoulements multiphasiques

Remerciements

Je tiens tout d'abord à remercier mon directeur de thèse, monsieur M'Hamed Souli, pour avoir accepté d'encadrer et de diriger ma thèse. Je le remercie aussi pour sa disponibilité, ses nombreux conseils ainsi que sa patience tout au long de ma thèse.

Je remercie monsieur Hassane Naji pour m'avoir fait l'honneur de présider mon jury de soutenance.

Je remercie aussi madame Elisabeth Longatte et monsieur Emmanuel Lefrançois pour avoir accepté de rapporter mes travaux de thèse.

Je remercie également messieurs Mojtaba Moatamedi, Gilmar Mompean, Abdelaziz Boudlal, Hassan Abbas Khawaja et Erwan Liberge pour avoir accepté d'examiner mes travaux de thèse et de faire partie du jury de ma soutenance.

Je tiens à remercier aussi l'ensemble des membres du Laboratoire de Mécanique de Lille, pour avoir fait de ces années passées au sein du laboratoire une expérience scientifique et humaine riche. Je remercie tout particulièrement mesdames Marie-Christine Nodot, Nathalie Labaeye et Anne-Marie Moudart pour leurs aides et disponibilités tout au long de ma thèse.

Enfin, je remercie mes amis et collègues Mehdi Amoura, Nassim Delenda, Mohammed Amdi, Malik Atik, Abdoulaye Gueye, Madani Hamlaoui, Ahmed El Moumen et Iason Zisis pour leurs amitiés, leurs aides et leurs soutiens, ainsi que tous ceux qui ont contribué à l'aboutissement de ces travaux de thèses: les ingénieurs d'EDF-UTO Bernard Cohen, Sylvain Durel et Romain Ceyrolles; les chercheurs EDF-R&D-CEA et développeurs du code Europlexus Frédéric Daude, Fabien Crouzet et Pascal Galon; les membres du projet FP7 SAFUEL Pierre Sinou, Bruno Reynard, Carlo Rens, Yana Lazarova, Susan Barreault et Tina Tefy; l'équipe enseignante du master Calcul Scientifique de l'université de Lille; et mesdames Tram Nguyen et Malika Debuysschere de l'EDSPI.

Cette thèse est dédiée à mes parents, mon épouse, mes frères et ma belle sœur.

Contents

List of Tables

List of Figures

Introduction

Context

Over the last few years, Fluid Structure Interaction (FSI) has become more and more the focus of computational engineering in many industries (petroleum, nuclear, defense, spatial, sports, ...). These problems are computer time consuming and require new stable and accurate coupling algorithms in order to be solved. Over the last few decades, both the new development of coupling algorithms and the advancement of computer performance have enabled some of the problems to be solved and have led to more physical applications that were inaccessible in the past; in the future this trend is likely to continue and to take more realistic problems into account.

This thesis follows my training period and its theme is the Water Hammer (WH) at the technical engineering unit EDF-UTO. Through the use of FSI capabilities and the simple "cut-off" cavitation model of the commercial code LS-DYNA, we have successfully solved coupled single-phase WHs and coupled two-phase WHs for specific cases in which the amount of vapor created was small enough so that the cases were not considered more sophisticated phase change models. This internship helped us to identify some limitations of LS-DYNA features in order to model complex industrial problems. To overcome these limitations, this research investigates the Homogeneous Equilibrium Models (HEM).

In order to test the actual Arbitrary Lagrangian Eulerian (ALE) and Smoothed Particle Hydrodynamics (SPH) methods that are implemented in LS-DYNA

to model real-life shock wave problems, the following collaboration projects supported this thesis:

- **University of Narvik (Norway) [65, 64] :** Investigation of composite shells structure subjected to water shock waves.

- **EDF-UTO [89, 93, 92, 91] :** Simulation of two phase WHs with phase change.

- **Technical University of Eindhoven (The Netherlands) [90, 171, 172, 170] :** Comparison of ALE and SPH methods.

Parallel to this thesis, I have also been involved in the two following projects:

- **FP7 European project SAFUEL** [39, 137]. Tasks : Numerical simulation of fuel flow in an aircraft fuel tank (during the refueling and the flight phases) and development of a software package (fortran) that computes the fuel mass within the fuel tank based on pressure gauging.

- **Investigation of Statistical Energy Analysis (SEA) methods for high frequency acoustic** [138]. Tasks : Development of a Fortran SEA code (ongoing development in LS-DYNA) and its validation against the free SEA software **FreeSEA** developed by professor Sarradj [123].

State of the Art

Homogeneous Equilibrium Model (HEM)

Under certain configurations, fluids (water particularly) may be put into tension and the local pressure may fall below the saturated pressure that gives rise to cavitation, small liquid-free zones ("bubbles" or "voids"). Such phenomena occur mostly near industrial apparatuses such as pumps, propellers, impellers and control valves. The rapid collapse of cavitation produces strong shock waves that may harm the interacting structure. In order to simulate fast transient phenomena such as Water Hammers (WHs) or underwater explosions (UNDEX), one must consult realistic compressible fluid models that take into consideration phase change, shock wave generation and its propagation. From the existing approaches, we are able to distinguish two major categories:

- The "Two-Fluid Models" [5, 128, 147], where each material (liquid and vapor) has its own set of governing equations that include additional closing relations and coupling process at the interface between the two materials.

- The "One-fluid Models", where only the average flow is considered by solving a unique set of governing equations and it can be based on a pure phase model (Vacuum Model [142], Cut-off Model [3]) or a mixture model (Schmidt model [126], Isentropic model [84], modified Schmidt model [157] or Saurel et al. model [124]).

In this thesis we investigate the one-fluid models due to their simplicity, easy implementation within existing codes (ALE and Lagrangian SPH), and finally, their ability to model water phase changes in many industrial applications. From the point of view of a numerical simulation, the set of governing equations (in the continuum domain) solved are the same for both applications whether or not a phase change takes place. Except that for the phase change model, the solved density and energy variables are mixture quantities expressed in function of the saturated vapor fraction. An adequate equation of state is based on the mixture quantities, and furthermore, is able to describe the kinematic and the

thermodynamic behaviors of the phase change phenomenon (creation of the cavitation and its collapse).

Schmidt [126] proposed the barotropic phase change model, which was originally designed for high speed nozzle cavitating flows in diesel injectors. When the Schmidt model is applied to water, it was shown by Xie et al. [157] that when the vapor fraction is larger than a small quantity $O\left(10^{-2}\right)$, the pressure cannot stay positive which is nonphysical. In order to overcome this limitation and to apply the phase change model to water, [157] proposed a modification to the Schmidt model adding a small positive cut-off pressure that can prevent the cavitation pressure from reaching negative values. This last model is referred as the modified Schmidt model.

The Isentropic model proposed by Liu et al. [84], is another barotropic model that overcomes the limitations of Schmidt's model. The main difficulty in the implementation of this model is the computation of a parameter that is based on the physical properties of the flow and the cavitation number which makes the model problem dependent. A numerical procedure to obtaining a value of the Isentropic model parameter is given in detail in [84].

In this thesis work, we decided to study the model proposed by Saurel et al. [124], which was designed to model high-velocity compressible flows including phase change caused by low pressure. The main advantage of the method proposed by Saurel et al. is that it is based on the physical properties of water and does not require non-physical parameters.

Arbitrary Lagrangian Eulerian (ALE) formulation

In the last few years, the simulation of FSI problems has become more and more the focus of computational engineering. The Lagrangian formulation, in which the mesh moves with material is mainly used to solve problems in solid mechanics, and can be used for FSI problems. For small deformations, Lagrangian formulation can accurately solve fluid structure interface and material boundary; its main limitation is mesh distortion for large deformation and moving structure, since the fluid is solved on a moving domain due to the structure motion. Various approaches have been investigated to solve FSI prob-

lems. The ALE formulation represents one of the commonly used approaches, which has been used with success in the simulation of fluid with large motion such as sloshing tanks [10, 108], bird impact in aeronautic industry [135], and explosions or shock wave impact [6, 107, 163, 164]. Fluid problems that include interfaces between different materials are more easily modeled through the use of Lagrangian mesh. If an analysis for complex geometry is required, however, the distortion of the Lagrangian mesh makes such a method difficult to use, and as a result, many re-meshing steps are necessary to continue the calculation. The Eulerian formulation present another method. This change from a Lagrangian to an Eulerian formulation, however, introduces two problems; the first is interface tracking [162] and the second is the advection of fluid material across element boundaries.

Various approaches have been investigated to solve FSI problems, including ALE muti-material formulation as in [10] and Lagrangian formulations [17]. An ALE formulation contains both pure Lagrangian and pure Eulerian formulations. The pure Lagrangian description is the approach in which the mesh moves with the material, which makes it easier to track interfaces and to apply boundary conditions. In an Eulerian description, the mesh remains fixed while the material passes through it. It is difficult to track interfaces and boundary conditions through the use of this approach. However, mesh distortion is not problematic because the mesh never changes. In solid mechanics, a pure Eulerian formulation is not useful because it can handle only a single material in an element, while in an ALE formulation, it is assumed to be capable of handling more than one material in an element.

Smoothed Particle Hydrodynamics (SPH) Method

The SPH method provides many advantages to model severe deformations in comparison to classical FEM formulations that suffers from high mesh distortion. Lucy [86] and Monaghan and Gingold [97] first introduced the method for gas dynamic problems and for problems in which the main concern is a set of discrete physical particles than the continuum media. The method was extended

to solve high velocity impact in solid mechanics, CFD applications governed by Navier-Stokes equations, and lastly, FSI problems. From previous studies (see [148]), it is well-established that the standard SPH method suffers from its lack of consistency, which can lead to poor accuracy in motion approximation.

Multiphase problems in SPH have received substantial attention, due to SPH's straightforward way of introducing more than one fluid in the computational domain. For the weakly compressible regime especially, multiphase algorithms have been extensively studied [28, 45, 54, 98, 95, 99]and remedies have been pinpointed and often fixed. Validated results have been reported that involve density ratios of up to $1,000$ and sound-speed ratios of almost 10 through the simple use of standard SPH algorithms by simply using standard SPH algorithms [99].

One of the currents in the SPH community is improve the quality of operators to move to a higher order of accuracy. The classical WC-SPH method has been improved and gave arise to a new class of hybrid methods such as the SPH-ALE method [149, 151] where Godunov schemes, increasing the stability of the WC-SPH scheme. For more details on these methods, the book from Magoules [117, 151] contains two detailed chapters on the hybrid WC-SPH, SPH-ALE and FVPM (Finite Volume Particle Method) methods.

Another current in the SPH community is the study and the derivation of consistent fully-compressible schemes and to trust the Lagrangian [113, 22, 115]. In this thesis, we present the mass-based scheme from Price [113] and the extended number-density scheme from [54] to fully-compressible regime for handling shock problems with high density ratio of the order of $100 - 200$. The two schemes were selected among a selection of schemes previously investigated by Zisis et al. [168]. It was shown by Zisis et al. [167] that the energy as well as the angular momentum are conserved for various impact tests.

Thesis objectives and development

The objectives of this thesis are as follows:

- Investigation and implementation of a HEM phase change model to solve WHs problems.

- Validation of SPH, ALE and FSI methods in LS-DYNA for shock wave impact on industrial structures.

- Identification of LS-DYNA SPH solver limits to solve shock problems in compressible multiphase flows.

- Development of an in-house fully compressible SPH code to solve shock problems in multiphase flows.

Chapter1 presents the physical models that are solved numerically in chapters 2 and 3. In consideration of the appropriate material model and constitutive equations, both solids and fluids can be described through the same set of conservation equations (mass, linear momentum and energy). However, the governing equations must be written in the proper formulation that describes the physics that is most compatible with the applied numerical methods. The Lagrangian and ALE governing equations are derived in the first subsection of the chapter. The last two subsections are dedicated to material constitutive equations for solids and fluids and the implementation of one-fluid HEM phase change model in LS-DYNA using the "User defined interface".

Chapter2 describes the ALE and FSI numerical methods implemented in LS-DYNA. Furthermore, it presents two industrial applications from collaborations with EDF and the University of Narvik. In the case of WHs, chapter2 investigates and validates HEM model coupled with LS-DYNA capabilities against experimental data and existing numerical solutions. In the case of the *"numerical investigation of shell structures subjected to water shock waves"*, the LS-DYNA results are compared to the experimental data provided by the University of Narvik (Norway).

In chapter 3, we present the weakly-compressible SPH methods implemented in LS-DYNA and its application to *"underwater explosion problems"*. We then validate the SPH solution the ALE solution in the absence of experimental data. The limitation of the implemented SPH method is highlighted through 1-D shock tube tests. To overcome the limitation of the weakly-compressible SPH schemes, the chapter presents fully compressible SPH schemes with multiphase shock validation test cases. The end of the chapter presents a description of the developed in-house code.

Physical Models

In this chapter, we describe the basis of continuum mechanics and the associated governing equations that the next chapters solve numerically. The first section presents Lagrangian description that is commonly used in solid mechanics. The second section consists of the more general ALE description that contains as both limit cases the Lagrangian and the Eulerian descriptions. The third section presents the different material models and constitutive equations used in this thesis. Finally, the last section presents the one-fluid HEM phase change model that is implemented in LS-DYNA and the Saurel et al. [124] HEM model.

There are many books and papers in literature that provide detailed derivations of the continuum equations in both Lagrangian and ALE formulations, as well as the different constitutive equations employed in continuum mechanics. For further information about the theory and mathematical derivation of the equations, see [17, 10, 134, 15].

1.1 Lagrangian Equations

1.1.1 Kinematics

Kinematic variables   For a given finite continuum body, let us consider the motion of all the particles within the material. At a given time t, the body occupies in space the domain Ω_t with boundary Γ_t. The reference domain is the domain occupied by the continuum body at time $t = 0$. The initial domain and its boundary are respectively denoted by Ω_0 and Γ_0. The motion from the initial domain Ω_0 to the current domain Ω_t can be described by the injective function $\vec{\phi}$ that associates to each material point of coordinate $\vec{X}$ at time t, the position in space $\vec{x}$ defined by:

$$\vec{x} = \vec{\phi}(\vec{X},t), \tag{1.1}$$

where $\vec{X}$ are the Lagrangian coordinates and $\vec{x}$ are the Eulerian coordinates. Lagrangian coordinates are link to the material and are invariant in time in the local configuration attached to the current domain Ω_t, whereas the Eulerian coordinates are invariant in space and time.
The displacement vector $\vec{u}$ of a material point is defined by:

$$\vec{u}(\vec{X},t) = \vec{\phi}(\vec{X},t) - \vec{X} = \vec{x} - \vec{X},$$
$$u_i(\vec{X},t) = \phi_i(\vec{X},t) - X_i = x_i - X_i \tag{1.2}$$

The velocity vector of a material point is obtained by taking the time derivative of the displacement vector given by Eq.1.2:

$$\dot{\vec{u}}(\vec{X},t) = \vec{v}(\vec{X},t) = \frac{\partial \vec{u}}{\partial t} = \frac{\partial \vec{\phi}}{\partial t}(\vec{X},t),$$
$$\dot{u}_i(\vec{X},t) = v_i(\vec{X},t) = \frac{\partial u_i}{\partial t} = \frac{\partial \phi_i}{\partial t}(\vec{X},t) \tag{1.3}$$

In a similar way, we obtain the acceleration vector by taking the derivative of Eq.1.3 with respect to time:

$$\vec{v}(\vec{X},t) = \vec{a}(\vec{X},t) = \frac{\partial \vec{v}}{\partial t} = \frac{\partial^2 \vec{\phi}}{\partial t^2}(\vec{X},t),$$

$$\dot{v}_i(\vec{X},t) = a_i(\vec{X},t) = \frac{\partial v_i}{\partial t} = \frac{\partial^2 \phi_i}{\partial t^2}(\vec{X},t) \tag{1.4}$$

The partial derivative $\frac{\partial}{\partial t}$ represent the derivative in the Eulerian framework. It is related to the material derivative $\frac{d}{dt}$ by:

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \vec{v}.\nabla_{\vec{x}} \tag{1.5}$$

Fig.1.1 provides a description of the transformation of a material continuum body from the reference domain Ω_0 to the current domain Ω_t.
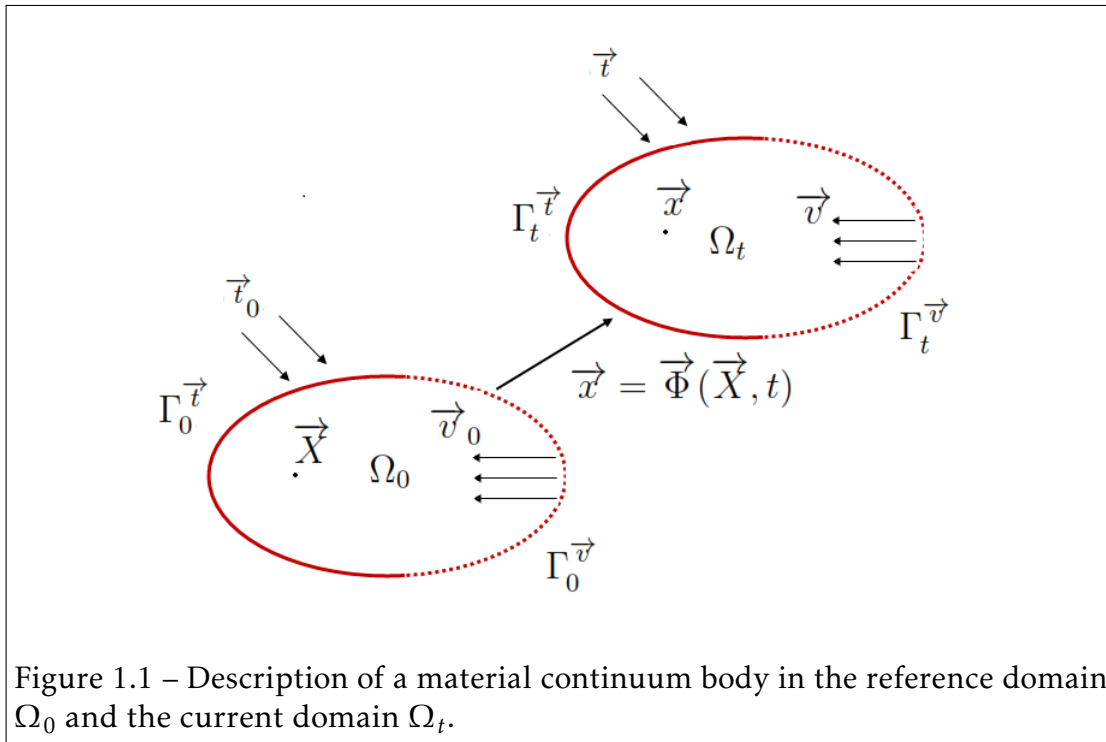
Figure 1.1 – Description of a material continuum body in the reference domain Ω_0 and the current domain Ω_t.

Deformation and Strain measure   Let $d\vec{X}$ be an infinitesimal line in the reference configuration and $d\vec{x}$ be its transformation in the current configuration. Since $\vec{x}$ is function of $\vec{X}$, from $x_1 = (X_1, X_2, X_3)$, $x_2 = (X_1, X_2, X_3)$ and $x_3 = (X_1, X_2, X_3)$ we obtain an expression of the differential $d\vec{x}$:

$$dx_1 = \frac{\partial x_1}{\partial X_1}dX_1 + \frac{\partial x_1}{\partial X_2}dX_2 + \frac{\partial x_1}{\partial X_3}dX_3,$$

$$dx_2 = \frac{\partial x_2}{\partial X_1}dX_1 + \frac{\partial x_2}{\partial X_2}dX_2 + \frac{\partial x_2}{\partial X_3}dX_3, \qquad (1.6)$$

$$dx_3 = \frac{\partial x_3}{\partial X_1}dX_1 + \frac{\partial x_3}{\partial X_2}dX_2 + \frac{\partial x_3}{\partial X_3}dX_3$$

Or in matrix form:

$$\underbrace{\begin{bmatrix} dx_1 \\ dx_2 \\ dx_3 \end{bmatrix}}_{d\vec{x}} = \underbrace{\begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix}}_{\underline{\underline{F}}} \underbrace{\begin{bmatrix} dX_1 \\ dX_2 \\ dX_3 \end{bmatrix}}_{d\vec{X}}, \qquad (1.7)$$

where $\underline{\underline{F}}$ is the second order deformation gradient tensor, which is also called the Jacobian matrix of the transformation ϕ. It is defined by:

$$\underline{\underline{F}} = \frac{\partial \vec{\phi}}{\partial \vec{X}} = \frac{\partial \vec{x}}{\partial \vec{X}},$$

$$F_{ij} = \frac{\partial \phi_i}{\partial X_j} = \frac{\partial x_i}{\partial X_j} \qquad (1.8)$$

For any function f defined in the current and the reference states, we have the equality:

$$\int_{\Omega_t} f(\vec{x}, t)d\Omega_t = \int_{\Omega_0} f(\vec{\phi}(\vec{X}, t))J d\Omega_0, \qquad (1.9)$$

where J is the determinant of the Jacobian matrix $\underline{\underline{F}}$ defined by:

$$J = det(\underline{\underline{F}}) = \frac{d\Omega_t}{d\Omega_0} \qquad (1.10)$$

The time derivative of the Jacobian is given by:

$$\frac{dJ}{dt} = J\nabla_{\vec{x}}.\vec{v},$$

$$\frac{dJ}{dt} = J\frac{\partial v_i}{\partial x_i} \qquad (1.11)$$

The second order velocity gradient tensor is given by:

$$\underline{\underline{L}} = \frac{\partial \vec{v}}{\partial \vec{x}},$$
$$L_{ij} = \frac{\partial v_i}{\partial x_j} \tag{1.12}$$

The second order velocity gradient tensor can be rewritten in term of its symmetric part $\underline{\underline{D}}$ and its anti-symmetric part $\underline{\underline{W}}$:

$$\underline{\underline{L}} = \underbrace{\frac{1}{2}(\underline{\underline{L}} + \underline{\underline{L}}^T)}_{\underline{\underline{D}}} + \underbrace{\frac{1}{2}(\underline{\underline{L}} - \underline{\underline{L}}^T)}_{\underline{\underline{W}}}, \tag{1.13}$$

where $\underline{\underline{D}}$ denotes the strain rate tensor. It can be expressed in term of the velocity vector:

$$\underline{\underline{D}} = \frac{1}{2}[(\nabla_{\vec{x}} \otimes \vec{v}) + (\nabla_{\vec{x}} \otimes \vec{v})^T],$$
$$D_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) \tag{1.14}$$

Stress measure   Stress measure is defined by the Cauchy law that gives a relation between the second order Cauchy stress tensor $\underline{\underline{\sigma}}$ to the traction vector $\vec{t}$, which is applied to the boundary Γ_t, such that:

$$\vec{n}.\underline{\underline{\sigma}}d\Gamma_t = \vec{t}\,d\Gamma_t,$$
$$n_j\sigma_{ji}d\Gamma_t = t_i d\Gamma_t, \tag{1.15}$$

where $\vec{n}$ is the exterior normal vector to the boundary Γ_t.
The Cauchy stress tensor can be expressed in its matrix form:

$$\underline{\underline{\sigma}} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \tag{1.16}$$

Or, using the Voigt notations, in a 6-dimensional (respectively 3-dimensional) for 3-dimensional problems (2-dimensional problems) due to the symmetry

$\sigma_{ij} = \sigma_{ji}$ (see Fig.1.2):

$$\overrightarrow{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{23}, \sigma_{13}, \sigma_{12}]^T \qquad (1.17)$$

The stress tensor $\underline{\underline{\sigma}}$ is often called the physical or the true stress. Unlike the first Piola-Kirchhoff stress and the second Piola-Kirchhoff stress, which refer to the undeformed state, the Cauchy stress tensor is related to the boundary Γ_t in the current configuration. The trace of the Cauchy stress tensor provides the true pressure, which is defined by:

$$\begin{aligned} p &= -\frac{1}{3} tr(\underline{\underline{\sigma}}), \\ p &= -\frac{1}{3} \sigma_{ii} \end{aligned} \qquad (1.18)$$

The Cauchy stress tensor can also be written into the following form:

$$\underline{\underline{\sigma}} = \underline{\underline{\sigma}}^d + p.\underline{\underline{I_d}}, \qquad (1.19)$$

where $\underline{\underline{\sigma}}^d$, $\underline{\underline{I_d}}$ and p denote the deviatoric stress tensor, the second order identity tensor and the pressure defined in Eq.1.18, respectively.

Figure 1.2 – Left: The six components of the Cauchy stress vector (using Voigt's notations) in 3D. Right: The three components of the Cauchy stress vector (using Voigt's notations) in 2D.
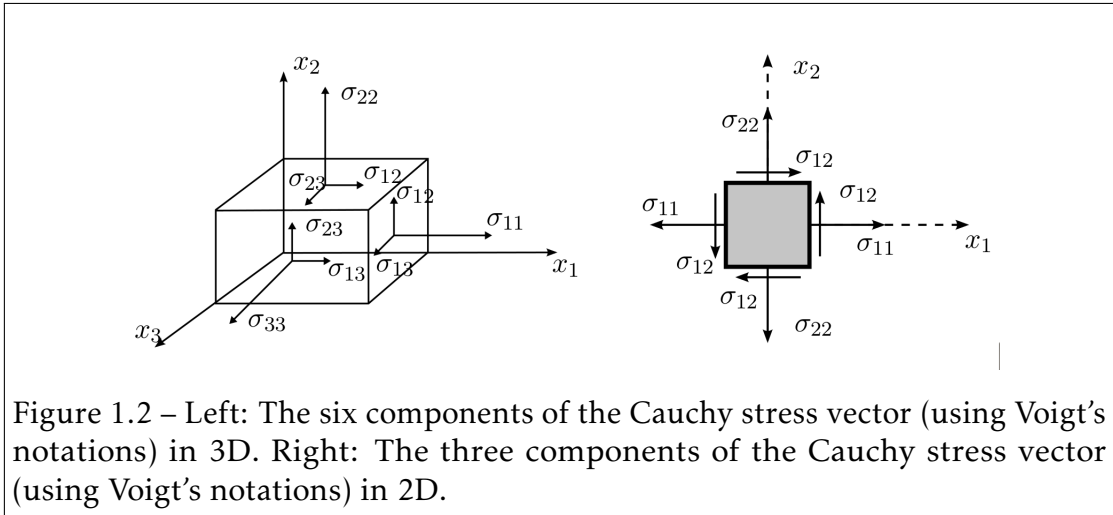
1.1.2   Conservation Equations

Reynold's transport theorem   Let $f(\vec{x}, t)$ be a function of space and time. The material time derivative of the integral of the function f over an arbitrary material domain Ω is defined by:

$$\frac{d}{dt} \int_\Omega f d\Omega = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left(\int_{\Omega_{t+\Delta t}} f(\vec{x}, t + \Delta t) d\Omega - \int_{\Omega_t} f(\vec{x}, t) d\Omega \right), \qquad (1.20)$$

where Ω_t and $\Omega_{t+\Delta t}$ are the spatial material domain Ω occupied by the same material points at times t and $t + \Delta t$, respectively. Applying Eq.1.9 to Eq.1.20, we have:

$$\frac{d}{dt} \int_\Omega f d\Omega = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \int_{\Omega_0} \left(f(\vec{X}, t + \Delta t) J(\vec{X}, t + \Delta t) - f(\vec{X}, t) J(\vec{X}, t) \right) d\Omega_0,$$

$$(1.21)$$

where Ω_0 is the material domain in the initial configuration at time t_0.
Now that the material domain is independent to the time variable, we can rewrite Eq.1.21 such that:

$$\frac{d}{dt} \int_\Omega f d\Omega = \int_{\Omega_0} \frac{\partial}{\partial t} \left(f(\vec{X}, t) J(\vec{X}, t) \right) d\Omega_0, \qquad (1.22)$$

where the partial time derivative $\frac{\partial}{\partial t}$ is a material time derivative $\frac{d}{dt}$ since the space variables are the material coordinates. Using the product rule for the derivative, we have:

$$\frac{d}{dt} \int_\Omega f d\Omega = \int_{\Omega_0} \left(\frac{\partial f}{\partial t} J + f \frac{\partial J}{\partial t} \right) d\Omega_0 \qquad (1.23)$$

Reminding that the partial time derivative is a material time derivative and using Eq.1.11, we have:

$$\frac{d}{dt} \int_\Omega f d\Omega = \int_{\Omega_0} \left(\frac{\partial f}{\partial t} J + J \nabla_{\vec{x}} \cdot \vec{v} \right) d\Omega_0 = \int_{\Omega_0} \left(\frac{\partial f}{\partial t} + \nabla_{\vec{x}} \cdot \vec{v} \right) J d\Omega_0 \qquad (1.24)$$

The material time derivative of the integral is given by:

$$\frac{d}{dt}\int_{\Omega} f\, d\Omega = \int_{\Omega}\left(\frac{df(\overrightarrow{x},t)}{dt} + f\nabla_{\overrightarrow{x}}.\overrightarrow{v}\right)d\Omega,$$

$$\frac{d}{dt}\int_{\Omega} f\, d\Omega = \int_{\Omega}\left(\frac{df(\overrightarrow{x},t)}{dt} + f\frac{\partial v_i}{\partial x_i}\right)d\Omega \tag{1.25}$$

Finally, applying the relation in Eq.1.5 to Eq.1.25 we have:

$$\frac{d}{dt}\int_{\Omega} f\, d\Omega = \int_{\Omega}\left(\frac{\partial f(\overrightarrow{x},t)}{\partial t} + \overrightarrow{v}\nabla_{\overrightarrow{x}}.f + f\nabla_{\overrightarrow{x}}.\overrightarrow{v}\right)d\Omega,$$

$$\frac{d}{dt}\int_{\Omega} f\, d\Omega = \int_{\Omega}\left(\frac{\partial f(\overrightarrow{x},t)}{\partial t} + \nabla_{\overrightarrow{x}}.\left(f\overrightarrow{v}\right)\right)d\Omega \tag{1.26}$$

Conservation of mass   Let $m(\Omega)$ be the mass of an arbitrary material domain Ω given by:

$$m(\Omega) = \int_{\Omega} \rho(\overrightarrow{X},t)d\Omega, \tag{1.27}$$

where $\rho(\overrightarrow{X},t)$ is the density. Through the omission of the conversion of mass to energy, the mass conservation ensures us that the mass within a material domain remains constant through time. Taking the material time derivative of $m(\Omega_t)$, we have:

$$\frac{dm}{dt} = \frac{d}{dt}\int_{\Omega}\rho\, d\Omega = 0 \tag{1.28}$$

Applying the Reynold's theorem to Eq.1.28, we have:

$$\frac{dm}{dt} = \int_{\Omega}\left(\frac{d\rho}{dt} + \rho\nabla_{\overrightarrow{x}}.\overrightarrow{v}\right)d\Omega = 0 \tag{1.29}$$

Since Eq.1.29 holds true for any arbitrary sub-domain Ω, we finally obtain the conservation of mass equation:

$$\frac{d\rho}{dt} + \rho\nabla_{\overrightarrow{x}}.\overrightarrow{v} = 0,$$

$$\frac{d\rho}{dt} + \rho\frac{\partial v_i}{\partial x_i} = 0 \tag{1.30}$$

Conservation of linear momentum   Let Ω be an arbitrary material domain with boundary Γ. The total force $\overrightarrow{f}(t)$ applied to Ω is given by:

$$\overrightarrow{f}(t) = \underbrace{\int_\Omega \rho\, \overrightarrow{b}(\overrightarrow{x},t)d\Omega}_{\text{body force}} + \underbrace{\int_\Gamma \overrightarrow{t}(\overrightarrow{x},t)d\Gamma}_{\text{Area traction}}, \tag{1.31}$$

where $\overrightarrow{b}$ is a force per unit mass and $\overrightarrow{t}$ is a force per unit area. The linear momentum $\overrightarrow{p}(t)$ is given by:

$$\overrightarrow{p}(t) = \int_\Omega \rho\, \overrightarrow{v}(\overrightarrow{x},t)d\Omega \tag{1.32}$$

Newton's second law of motion for a continuum domain states that the time derivative of the linear momentum is equal to the total force. Through the use of Eq.1.31 and Eq.1.32, we have:

$$\int_\Omega \rho\, \overrightarrow{v}(\overrightarrow{x},t)d\Omega = \int_\Omega \rho\, \overrightarrow{b}(\overrightarrow{x},t)d\Omega + \int_\Gamma \overrightarrow{t}(\overrightarrow{x},t)d\Gamma \tag{1.33}$$

Applying Reynold's theorem to the left hand side of Eq.1.33, we have:

$$\int_\Omega \rho\, \overrightarrow{v}\, d\Omega = \int_\Omega \left(\frac{d}{dt}\left(\rho\, \overrightarrow{v}\right) + \left(\rho\, \overrightarrow{v}\right)\nabla_{\overrightarrow{x}}.\overrightarrow{v} \right)d\Omega,$$

$$\int_\Omega \rho\, \overrightarrow{v}\, d\Omega = \int_\Omega \left[\rho\frac{d\overrightarrow{v}}{dt} + \overrightarrow{v}\underbrace{\left(\frac{d\rho}{dt} + \rho\nabla_{\overrightarrow{x}}.\overrightarrow{v} \right)}_{= \, 0 \text{ due to Eq.1.30}} \right]d\Omega \tag{1.34}$$

Applying the mass conservation equation (Eq.1.30) to the last equation, we have:

$$\int_\Omega \rho\, \overrightarrow{v}\, d\Omega = \int_\Omega \rho\frac{d\overrightarrow{v}}{dt}d\Omega \tag{1.35}$$

Through the successful use of Cauchy's relation (Eq.1.15) and the Gauss theorem to the second term of the right hand side of Eq.1.33, we have:

$$\int_\Gamma \vec{t}\, d\Gamma = \int_\Gamma \vec{n}.\underline{\underline{\sigma}}\, d\Gamma = \int_\Omega \nabla_{\vec{x}}.\underline{\underline{\sigma}}\, d\Omega,$$

$$\int_\Gamma t_i\, d\Gamma = \int_\Gamma n_j \sigma_{ji}\, d\Gamma = \int_\Omega \frac{\partial \sigma_{ji}}{\partial x_j}\, d\Omega \tag{1.36}$$

Combining Eq.1.33, Eq.1.35 and Eq.1.36, we have:

$$\int_\Omega \left(\rho \frac{d\vec{v}}{dt} - \rho \vec{b} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} \right) d\Omega = 0 \tag{1.37}$$

Since Eq.1.37 holds true for any arbitrary sub-domain Ω, we finally obtain the conservation of linear momentum equation:

$$\rho \frac{d\vec{v}}{dt} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} - \rho \vec{b} = 0,$$

$$\rho \frac{dv_i}{dt} - \frac{\partial \sigma_{ji}}{\partial x_j} - \rho b_i = 0 \tag{1.38}$$

Conservation of energy   Let Ω be an arbitrary material domain with boundary Γ. Since the first law of thermodynamics states that the rate of change of the total energy $\mathcal{P}^{tot}$ (kinetic $\mathcal{P}^{kin}$ plus internal $\mathcal{P}^{int}$) is equal to the rate of work by the external forces $\mathcal{P}^{ext}$ and the rate of work provided by heat flux and energy sources $\mathcal{P}^{heat}$:

$$\mathcal{P}^{tot} = \mathcal{P}^{kin} + \mathcal{P}^{int}, \tag{1.39}$$

$$\mathcal{P}^{tot} = \mathcal{P}^{ext} + \mathcal{P}^{heat}, \tag{1.40}$$

where:

$$\mathcal{P}^{kin} = \frac{d}{dt} \int_\Omega \frac{1}{2} \rho \, \vec{v}.\vec{v}\, d\Omega, \tag{1.41}$$

$$\mathcal{P}^{int} = \frac{d}{dt} \int_\Omega \rho e^{int}\, d\Omega, \tag{1.42}$$

$$\mathcal{P}^{ext} = \int_\Omega \vec{v}.\rho \vec{b}\, d\Omega + \int_\Gamma \vec{v}.\vec{t}\, d\Gamma, \tag{1.43}$$

$$\mathcal{P}^{heat} = \int_{\Omega} \rho s \, d\Omega - \int_{\Gamma} \vec{n}.\vec{q} \, d\Gamma = \int_{\Omega} \rho s - \nabla_{\vec{x}}.\vec{q} \, d\Omega, \tag{1.44}$$

where e^{int}, s and $\vec{q}$ denote the internal energy, the heat source and the heat flux vector applied to the boundary Γ, respectively.

Substituting Eq.1.41 and Eq.1.42 into Eq.1.39 and applying Reynold's theorem , we have:

$$\mathcal{P}^{tot} = \frac{d}{dt} \int_{\Omega} \left(\rho e^{int} + \frac{1}{2} \rho \vec{v}.\vec{v} \right) d\Omega = \int_{\Omega} \left(\rho \frac{d e^{int}}{dt} + \frac{1}{2} \rho \frac{d\left(\vec{v}.\vec{v}\right)}{dt} \right) d\Omega,$$

$$= \int_{\Omega} \left(\rho \frac{d e^{int}}{dt} + \rho \vec{v}.\frac{d\vec{v}}{dt} \right) d\Omega \tag{1.45}$$

Converting the traction boundary integral in Eq.1.43 to a domain integral by the use of Gauss's theorem and Cauchy's law, we have:

$$
\begin{aligned}
\int_{\Gamma} \vec{v}.\vec{t} \, d\Gamma &= \int_{\Gamma} \vec{n}.\underline{\underline{\sigma}}.\vec{v} \, d\Gamma = \int_{\Gamma} n_j \sigma_{ji} v_i d\Gamma \\
&= \int_{\Omega} \frac{\partial \sigma_{ji} v_i}{\partial x_j} d\Omega = \int_{\Omega} \left(\frac{\partial v_i}{\partial x_j} \sigma_{ji} + v_i \frac{\partial \sigma_{ji}}{\partial x_j} \right) d\Omega \\
&= \int_{\Omega} \left((D_{ji} + W_{ji}) \sigma_{ji} + v_i \frac{\partial \sigma_{ji}}{\partial x_j} \right) d\Omega \\
&= \int_{\Omega} \left(D_{ji}\sigma_{ji} + \underbrace{W_{ji}\sigma_{ji}}_{= \, 0} + v_i \frac{\partial \sigma_{ji}}{\partial x_j} \right) d\Omega \\
&= \int_{\Omega} \left(D_{ji}\sigma_{ji} + v_i \frac{\partial \sigma_{ji}}{\partial x_j} \right) d\Omega \\
&= \int_{\Omega} \left(\underline{\underline{D}} : \underline{\underline{\sigma}} + \left(\nabla_{\vec{x}}.\underline{\underline{\sigma}} \right).\vec{v} \right) d\Omega
\end{aligned}
\tag{1.46}
$$

Substituting Eq.1.46 into Eq.1.43 and combining Eq.1.40, Eq.1.43, Eq.1.44 and Eq.1.45, we have:

$$\int_{\Omega} \left(\rho \frac{d e^{int}}{dt} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\vec{x}}.\vec{q} - \rho s + \vec{v}.\underbrace{\left(\rho \frac{d\vec{v}}{dt} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} - \rho \vec{b} \right)}_{= \, 0 \text{ due to Eq. 1.38}} \right) d\Omega = 0 \tag{1.47}$$

Applying the linear momentum conservation equation (Eq.1.38) to the latest equation we obtain:

$$\int_{\Omega}\left(\rho\frac{de^{int}}{dt} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\overrightarrow{x}}.\overrightarrow{q} - \rho s\right)d\Omega = 0 \tag{1.48}$$

Since Eq.1.48 holds true for any arbitrary sub-domain Ω, we finally obtain the conservation of energy equation:

$$\rho\frac{de^{int}}{dt} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\overrightarrow{x}}.\overrightarrow{q} - \rho s = 0,$$
$$\rho\frac{de^{int}}{dt} - \frac{\partial v_i}{\partial x_j}\sigma_{ji} + \frac{\partial q_j}{\partial x_j} - \rho s = 0 \tag{1.49}$$

Initial and Boundary conditions   In order to define a well-posed mathematical problem (existence of the solution, uniqueness and continuity at the boundaries), we need to define the appropriate initial and boundary conditions.

- **Boundary conditions**
  As shown in Fig. 1.1, let the boundary $\Gamma = \Gamma^{\overrightarrow{v}} \cup \Gamma^{\overrightarrow{t}}$ be the union of the boundary $\Gamma^{\overrightarrow{v}}$, where the velocity is imposed and the boundary Γ^{σ} where the stresses are imposed such that $\Gamma^{\overrightarrow{v}} \cap \Gamma^{\overrightarrow{t}} = \emptyset$. The boundary conditions applied on boundary Γ are defined by:

$$n_j\sigma_{ji} = t_i \text{ on boundary } \Gamma^{\overrightarrow{t}} \tag{1.50}$$

  and
$$v_i = v_i^{bdy} \text{ on boundary } \Gamma^{\overrightarrow{v}}. \tag{1.51}$$

- **Initial conditions**
  We define at time $t = t_0$ the initial conditions on the reference material domain Ω_0 by:

$$\overrightarrow{v}(\overrightarrow{X}, t_0) = \overrightarrow{v}_0(\overrightarrow{X}), \tag{1.52}$$

$$\underline{\underline{\sigma}}(\overrightarrow{X}, t_0) = \underline{\underline{\sigma}}_0(\overrightarrow{X}) \tag{1.53}$$

The following box summarizes the Lagrangian conservation equations in both tensor and suffix notations.

In tensor notations

- Mass conservation

$$\frac{d\rho}{dt} + \rho \nabla_{\vec{x}} . \vec{v} = 0$$

- Linear momentum conservation

$$\rho \frac{d\vec{v}}{dt} - \nabla_{\vec{x}} . \underline{\underline{\sigma}} - \rho \vec{b} = 0$$

- Energy conservation

$$\rho \frac{de^{int}}{dt} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\vec{x}} . \vec{q} - \rho s = 0$$

- Boundary and initial conditions

$$\vec{n} . \underline{\underline{\sigma}} = \vec{t} \text{ on boundary } \Gamma^{\vec{t}}$$

$$\vec{v} = \vec{v}^{bdy} \text{ on boundary } \Gamma^{\vec{v}}$$

$$\vec{v}(\vec{X}, t_0) = \vec{v}_0(\vec{X}) \text{ and } \underline{\underline{\sigma}}(\vec{X}, t_0) = \underline{\underline{\sigma_0}}(\vec{X})$$

In suffix notations

- Mass conservation

$$\frac{d\rho}{dt} + \rho \frac{\partial v_i}{\partial x_i} = 0$$

- Linear momentum conservation

$$\rho \frac{dv_i}{dt} - \frac{\partial \sigma_{ji}}{\partial x_j} - \rho b_i = 0$$

- Energy conservation

$$\rho \frac{de^{int}}{dt} - \frac{\partial v_i}{\partial x_j} \sigma_{ji} + \frac{\partial q_j}{\partial x_j} - \rho s = 0$$

- Boundary and initial conditions

$$n_j \sigma_{ji} = t_i \text{ on boundary } \Gamma^{\vec{t}}$$

$$v_i = v_i^{bdy} \text{ on boundary } \Gamma^{\vec{v}}$$

$$v_i(\vec{X}, t_0) = v_{i0}(\vec{X}) \text{ and } \sigma_{ji}(\vec{X}, t_0) = \sigma_{ji_0}(\vec{X})$$

1.2   Arbitrary Lagrangian Eulerian (ALE) Equations

1.2.1   Kinematics

Kinematic variables   In the previous section, we demonstrated that the Eulerian coordinates $\overrightarrow{x}$ can be expressed in term of the Lagrangian coordinates $\overrightarrow{X}$ by the function $\overrightarrow{x} = \overrightarrow{\phi}(\overrightarrow{X}, t)$. The injective function $\overrightarrow{\phi}(\overrightarrow{X}, t)$ associates to each material point of coordinates $\overrightarrow{X}$ in Ω_0 at time t, the position $\overrightarrow{x}$ in Ω_t.

Let Ω_t^{ALE} be an arbitrary domain of boundary Γ_t^{ALE}, we define the injective function $\overrightarrow{\psi}(\overrightarrow{\chi}, t)$ that associates the ALE coordinates $\overrightarrow{\chi}$ in Ω_t^{ALE} at time t to the Eulerian coordinates $\overrightarrow{x}$ in Ω_t by:

$$\overrightarrow{x} = \overrightarrow{\psi}(\overrightarrow{\chi}, t) \tag{1.54}$$

A description of the transformation of a material continuum body from the reference domain Ω_0 to the current domain Ω_t and ALE domain Ω_{ALE} is shown in Fig.1.3.

Figure 1.3 – Description of a material continuum body in the reference domain Ω_0, the current domain Ω_t and the arbitrary domain Ω_{ALE}
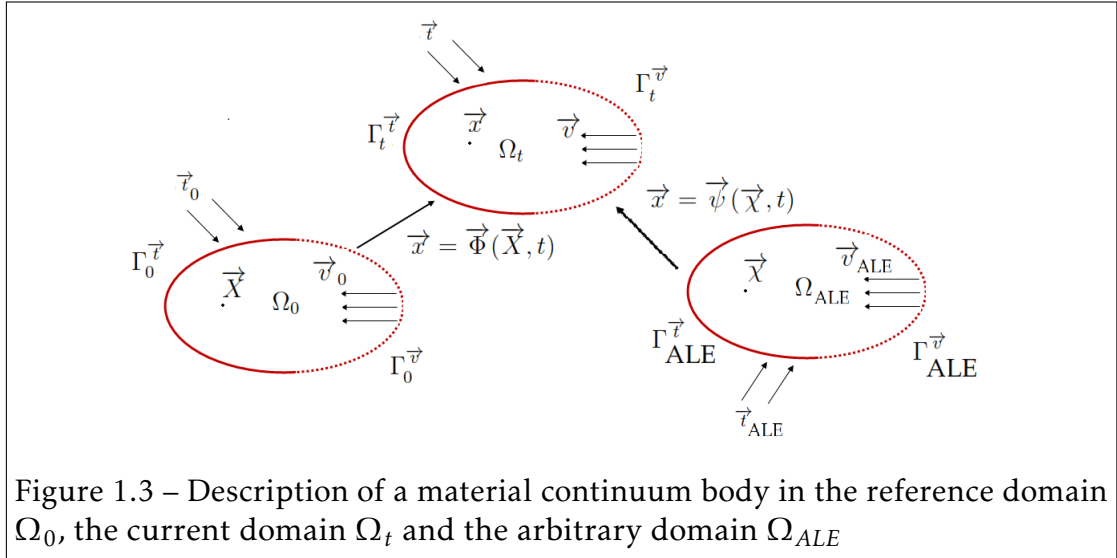
Let $f(\overrightarrow{x}, t)$ be a function of the Eulerian coordinates $\overrightarrow{x}$ and the time t. The expression of $f(\overrightarrow{x}, t)$ in the ALE referential is given by:

$$f(\vec{x},t) = f\left(\vec{\psi}(\vec{\chi},t),t\right) = f^{ALE}\left(\vec{\chi},t\right) \tag{1.55}$$

The partial time derivative of the function $f^{ALE}\left(\vec{\chi},t\right)$ in the ALE referential is given by:

$$\frac{\partial f^{ALE}\left(\vec{\chi},t\right)}{\partial t} = \frac{\partial f\left(\vec{x},t\right)}{\partial t} + \frac{\partial \vec{\psi}(\vec{\chi},t)}{\partial t}.\nabla_{\vec{x}} f\left(\vec{x},t\right), \tag{1.56}$$

where $\frac{\partial \vec{\psi}(\vec{\chi},t)}{\partial t} = \vec{v}_{ALE}$ is the ALE velocity. After applying Eq.1.5 to the function $f(\vec{x},t)$, we have:

$$\frac{df(\vec{x},t)}{dt} = \frac{\partial f\left(\vec{x},t\right)}{\partial t} + \vec{v}.\nabla_{\vec{x}} f\left(\vec{x},t\right) \tag{1.57}$$

Now subtracting Eq.1.56 to Eq.1.57, we finally obtain the relation between the material time derivative and the partial time derivative in the ALE referential:

$$\frac{df}{dt} = \frac{\partial f^{ALE}}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}} f \tag{1.58}$$

We denote by $\vec{w} = \left(\vec{v} - \vec{v}_{ALE}\right)$ the convective velocity (difference between the fluid's velocity and the ALE domain velocity). The ALE description is a more general description that has the limit cases:

- **Lagrangian description** If $\Omega_t^{ALE} = \Omega_0$ then $\vec{\chi} = \vec{X}$ and $\vec{\psi}(\vec{\chi},t) = \vec{\phi}(\vec{X},t)$ since the functions are injective. Thus $\vec{w} = 0 \Rightarrow \vec{v} = \vec{v}_{ALE}$ and we recover the expression of the Lagrangian time derivative:

$$\frac{df}{dt} = \frac{\partial f^{ALE}}{\partial t}$$

- **Eulerian description** If $\Omega_t^{ALE} = \Omega_t$ then $\vec{\chi} = \vec{x}$ and:

$$\vec{v}_{ALE} = \frac{\partial \vec{\psi}(\vec{\chi},t)}{\partial t} = \frac{\partial \vec{\psi}(\vec{x},t)}{\partial t} = \frac{\partial \vec{x}}{\partial t} = \vec{0}$$

Thus $\vec{w} = \vec{v}$ and we recover the expression of the Eulerian time derivative:

$$\frac{df}{dt} = \frac{\partial f^{ALE}}{\partial t} + \vec{v}.\nabla_{\vec{x}} f$$

1.2.2   Conservation Equations

The ALE formulation for the conservation equations are obtained by substituting the function $f(\vec{x}, t)$ in Eq.1.58 by the density, velocity vector and the internal energy in the equations of the conservation of mass, conservation of linear momentum and conservation of the energy, respectively. Thus, the ALE conservation equations are given by:

- **Conservation of mass**

$$\frac{\partial \rho}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}\rho + \rho\nabla_{\vec{x}}.\vec{v} = 0 \tag{1.59}$$

- **Conservation of linear momentum**

$$\rho\frac{\partial \vec{v}}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}\vec{v} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} - \rho\,\vec{b} = 0 \tag{1.60}$$

- **Conservation of energy**

$$\rho\frac{\partial e^{int}}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}e^{int} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\vec{x}}.\vec{q} - \rho s = 0 \tag{1.61}$$

The following box summarizes the ALE conservation equations in tensor notation.

Lagrangian equations

- Mass conservation

$$\frac{d\rho}{dt} + \rho\nabla_{\vec{x}}.\vec{v} = 0$$

- Linear momentum conservation

$$\rho\frac{d\vec{v}}{dt} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} - \rho\vec{b} = 0$$

- Energy conservation

$$\rho\frac{de^{int}}{dt} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\vec{x}}.\vec{q} - \rho s = 0$$

ALE equations

- Mass conservation

$$\frac{\partial\rho}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}\rho + \rho\nabla_{\vec{x}}.\vec{v} = 0$$

- Linear momentum conservation

$$\rho\frac{\partial\vec{v}}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}\vec{v} - \nabla_{\vec{x}}.\underline{\underline{\sigma}} - \rho\vec{b} = 0$$

- Energy conservation

$$\rho\frac{\partial e^{int}}{\partial t} + \left(\vec{v} - \vec{v}_{ALE}\right).\nabla_{\vec{x}}e^{int} - \underline{\underline{D}} : \underline{\underline{\sigma}} + \nabla_{\vec{x}}.\vec{q} - \rho s = 0$$

1.3   Material Models and Constitutive Equations

In the previous subsection, the continuum equations were derived that are to be solved numerically in chapters 2 and 3. In total there are five scalar equations (one for mass conservation, three for linear momentum conservation and one for energy conservation) for eleven unknown scalars (the density, the three components of the velocity vector, the internal energy and the six components of the stress tensor).

The material models in continuum mechanics, are designed to model the behavior of continuum material bodies by empirical laws throughout their deformation. Material models define the link between the kinematic variables (displacement vectors, strain tensors, ...) and dynamic variables (stress tensor). Using an appropriate material model, the stress tensor can be deduced from the kinematic variables and the system of conservation equations is now closed. The constitutive equations do provide a solution to the stress tensor (six scalars unknowns), which leaves five unknowns that are solved in the five equations.

In this subsection, we present the different material models for fluids and structures that are used in the numerical modeling in chapters 2 and 3.

1.3.1   Material Models for Structures

LS-DYNA MAT_001: Isotropic Linear Elasticity Materials   For most solid materials, the measured strains are functions of the applied forces. For linear materials, when the load does not exceed the elastic limit of the material the strains are proportional to the applied force. In one dimension, considering that the stress can be defined as the ratio between the applied force to Area($\sigma = \frac{F}{A}$), it follows that the stress is then proportional to the applied force and thus to the strain ε (see Fig.1.4):

$$\sigma = E.\varepsilon \tag{1.62}$$

Figure 1.4 – Elastic-plastic behavior with isotropic and kinematic hardening where l_0 and l are the undeformed and deformed length of uniaxial tension specimen, respectively [48]
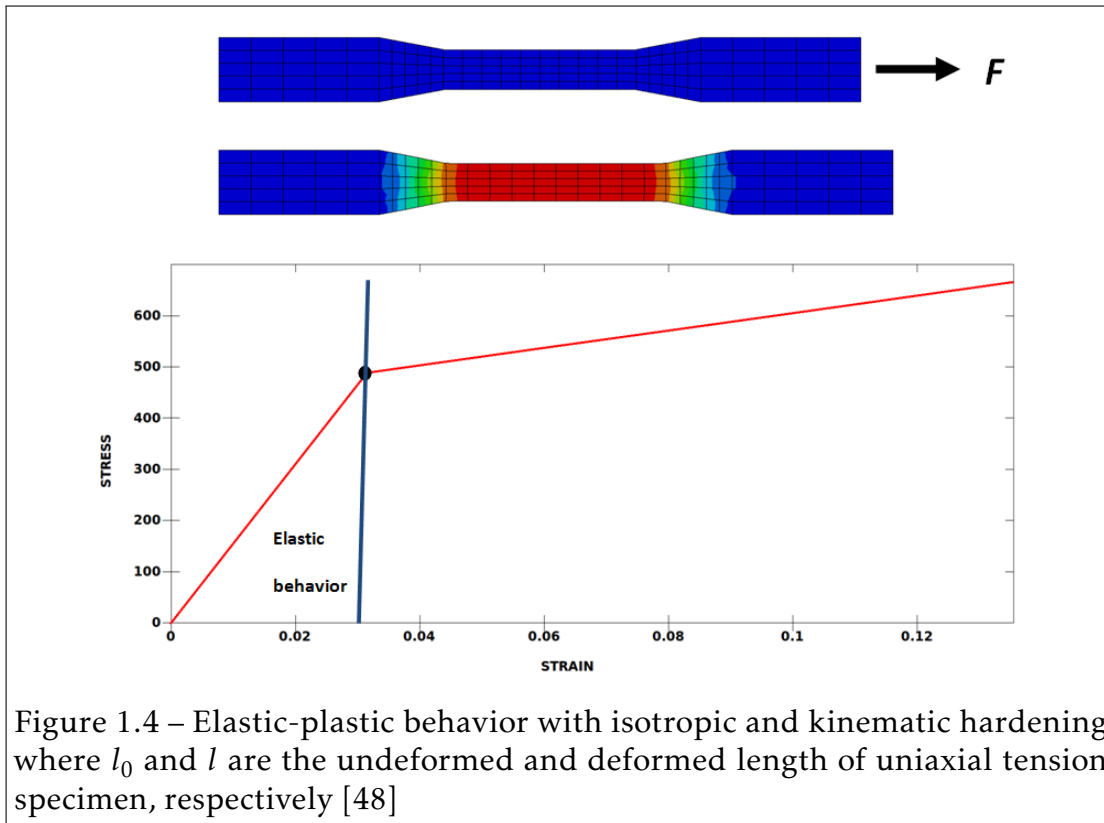
The last statements are known as Hook's law and do not apply to viscoelastic, plastic, or viscoplastic materials. It can be generalized to multidimensional problems through the use of the tensor notations:

$$\underline{\underline{\sigma}} = \underline{\underline{E}} : \underline{\underline{\varepsilon}}, \tag{1.63}$$

or the use of vectors and matrix notations:

$$
\begin{bmatrix}
\sigma_{11} \\
\sigma_{22} \\
\sigma_{33} \\
\sigma_{23} \\
\sigma_{13} \\
\sigma_{12}
\end{bmatrix}
=
\begin{bmatrix}
E_{11} & E_{12} & E_{13} & E_{14} & E_{15} & E_{16} \\
E_{21} & E_{22} & E_{23} & E_{24} & E_{25} & E_{26} \\
E_{31} & E_{32} & E_{33} & E_{34} & E_{35} & E_{36} \\
E_{41} & E_{42} & E_{43} & E_{44} & E_{45} & E_{46} \\
E_{51} & E_{52} & E_{53} & E_{54} & E_{55} & E_{56} \\
E_{61} & E_{62} & E_{63} & E_{64} & E_{65} & E_{66}
\end{bmatrix}
\begin{bmatrix}
\varepsilon_{11} \\
\varepsilon_{22} \\
\varepsilon_{33} \\
\varepsilon_{23} \\
\varepsilon_{13} \\
\varepsilon_{12}
\end{bmatrix}, \tag{1.64}
$$

where $\underline{\underline{\varepsilon}}$ is the second order Green-Lagrange strain tensor defined by:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \sum_{k=1}^{3}\frac{\partial u_k}{\partial x_i}\frac{\partial u_k}{\partial x_j}\right), \; i, j = 1, 2, 3 \qquad (1.65)$$

and $\underline{\underline{E}}$ is the fourth order tensor of elastic coefficients defined by:

$$\underline{\underline{E}} = \begin{bmatrix} \lambda + 2\mu & & & & & \\ \lambda & \lambda + 2\mu & & & \text{symmetric} & \\ \lambda & \lambda & \lambda + 2\mu & & & \\ 0 & 0 & 0 & 2\mu & & \\ 0 & 0 & 0 & 0 & 2\mu & \\ 0 & 0 & 0 & 0 & 0 & 2\mu \end{bmatrix}, \qquad (1.66)$$

where the Lamé material coefficients (λ, μ) are defined by:

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \qquad (1.67)$$

and

$$\mu = \frac{E}{2(1+\nu)}, \qquad (1.68)$$

where E is the Young modulus and ν the Poisson ratio.

Finally, when substituting Eq.1.65 and Eq.1.66 into Eq.1.64, we obtain the expression of the Cauchy stress tensor in term of the strain tensor components:

$$\begin{aligned} \sigma_{ii} &= \lambda(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{ii} \; i = 1, 2, 3, \\ \sigma_{ij} &= 2\mu\varepsilon_{ij} \; i \neq j \end{aligned} \qquad (1.69)$$

LS-DYNA MAT_003: Elastic-Plastic Materials with Kinematic Hardening   When a solid continuum body is subjected to stresses characterized by multidimensional stress field $\underline{\underline{\sigma}}$ and reaches a certain yield stress σ_y, the material becomes permanently stretched in such a way that the material does not return to its original length when the force is removed. $\underline{\underline{\alpha}}$ is an internal variable used in the measure of the plastic deformation.

In the elastic domain the stresses depend only on the state of the strain. Above the elastic limit the yield stress changes with increasing plastic deformations.

For elastic-plastic materials, the stresses are integrated in time:

$$\sigma_{ij}(t+dt) = \sigma_{ij}(t) + dt.\dot{\sigma}_{ij}(t), \tag{1.70}$$

where $\underline{\underline{\dot{\sigma}}}$ is the time derivative of Cauchy stress tensor defined by:

$$\dot{\sigma}_{ij} = \sigma_{ij}^{\nabla} + \sigma_{ik}\omega_{kj} + \sigma_{jk}\omega_{ki} \tag{1.71}$$

in which $\underline{\underline{\omega}}$ is the spin tensor given by:

$$\omega_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i}\right) \tag{1.72}$$

and $\underline{\underline{\sigma}}^{\nabla}$ is the Jaumann (co-rotational) stress rate given by:

$$\underline{\underline{\sigma}}^{\nabla} = \underline{\underline{E}} : \underline{\underline{\dot{\varepsilon}}} \tag{1.73}$$

The LS-DYNA material model **MAT_003** can model isotropic hardening, kinematic hardening or a combination of both varying a parameter $\beta \in [0,1]$(see Fig.1.5). The works published by Krieg and Key [70] form the basis for the model implemented in LS-DYNA.

The yield condition $\phi(\underline{\underline{\sigma}}, \underline{\underline{\alpha}})$ is a function of the stress tensor $\underline{\underline{\sigma}}$ and the internal variable $\underline{\underline{\alpha}}$. It is given by:

$$\phi(\underline{\underline{\sigma}}, \underline{\underline{\alpha}}) = \frac{3}{2}\left(\sigma_{ij}^d - \alpha_{ij}\right)\left(\sigma_{ij}^d - \alpha_{ij}\right) - \sigma_y^2 = 0 \tag{1.74}$$

The yield stress σ_y is defined by:

$$\sigma_y = \sigma_0 + \beta E_p \varepsilon_{ij}^p, \tag{1.75}$$

where

- σ_0 is the initial yield stress.

- E_p is the plastic hardening modulus defined by:

Figure 1.5 – Strain vs Stress curve for uniaxial stretching test using LS-DYNA finite element solver [48]
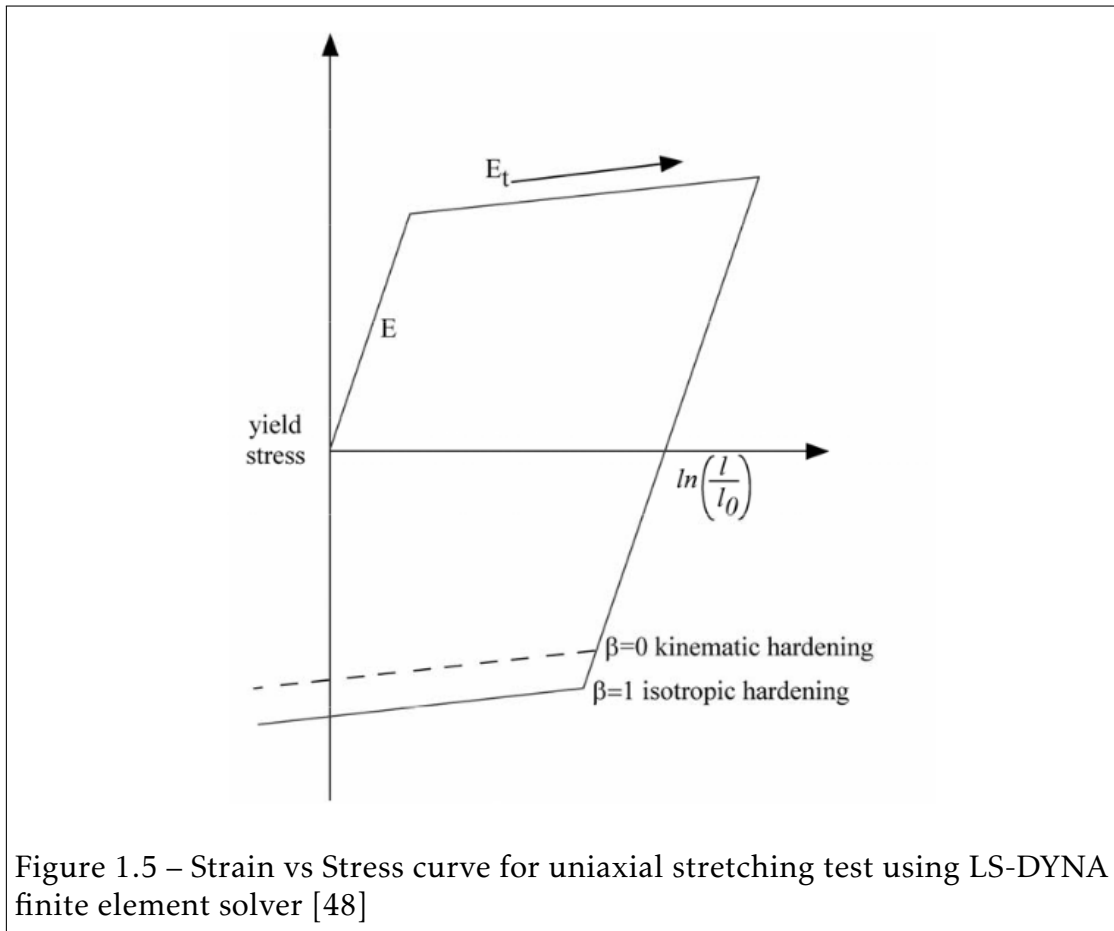
$$E_p = \frac{E_t E}{E - E_t} \tag{1.76}$$

- ε_{ij}^p is the effective plastic strain defined by:

$$\varepsilon_{ij}^p = \int_0^t \left(\frac{2}{3} \dot{\varepsilon}_{ij}^p \dot{\varepsilon}_{ij}^p \right)^{\frac{1}{2}} dt, \tag{1.77}$$

where the plastic strain rate $\dot{\varepsilon}_{ij}^p$ is the difference between the total strain rate $\dot{\varepsilon}_{ij}$ and the elastic strain rate $\dot{\varepsilon}_{ij}^e$

$$\dot{\varepsilon}_{ij}^p = \dot{\varepsilon}_{ij} - \dot{\varepsilon}_{ij}^e \tag{1.78}$$

Through the use of the Cowper-Symonds model [58], the yield stress is scaled by a strain rate dependent factor fac given by:

$$\sigma_y = fac\left(\sigma_0 + \beta E_p \varepsilon_i j^p\right),$$

$$fac = 1 + \left(\frac{\sqrt{\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij}}}{C}\right)^{\frac{1}{p}}, \tag{1.79}$$

where the constants C and p are input user-defined parameters.

The internal variable $\underline{\underline{\alpha}}(t)$ is a time dependent parameter defined by:

$$\alpha_{ij}(t+dt) = \alpha_{ij}(t)\left[\alpha_{ij}^{\nabla}\left(t + \frac{dt}{2}\right) + \alpha_{ik}(t)\Omega_{kj}\left(t + \frac{dt}{2}\right) + \alpha_{jk}(t)\Omega_{ki}\left(t + \frac{dt}{2}\right)\right]dt, \tag{1.80}$$

in which $\underline{\underline{\alpha}}^{\nabla}$ is the co-rotational rate of $\underline{\underline{\alpha}}$

$$\alpha_{ij}^{\nabla} = (1 - \beta)\frac{2}{3}E_p\dot{\varepsilon}_{ij}^{p} \tag{1.81}$$

and

$$\Omega_{ij} = \dot{R}_{ik}R_{jk} \tag{1.82}$$

R is found through the application of the polar decomposition theorem (see [143, 48]).

The implementation of the method is described in details in [70, 59, 15, 48]

1.3.2   Material Models for Fluids

Unlike solids, any shear stress force applied to a fluid produces motion. The stress tensor $\underline{\underline{\sigma}}$ is expressed as a function of the strain rate tensor $\underline{\underline{\dot{\varepsilon}}}$.

In this thesis, only compressible Newtonian fluids which shear stresses are proportional to strain rates are considered (**LS-DYNA MAT_009: MAT_NULL**). The constitutive equations for such fluids are given by:

$$\underline{\underline{\sigma}} = -p\left(\rho, e^{int}\right)\underline{\underline{I_d}} + \underline{\underline{\sigma^d}},$$

$$\sigma_{ij} = -p\left(\rho, e^{int}\right)\delta_{ij} + \sigma_{ij}^d, \tag{1.83}$$

where the pressure $p\left(\rho, e^{int}\right)$ is related to density and internal energy by an equation of state (see next subsection), δ_{ij} is the Kronecker symbol and the deviatoric stress tensor $\underline{\underline{\sigma^d}}$ is a linear function of the viscosity.

Newtonian Inviscid Fluids   A fluid is said to be inviscid when the viscosity coefficient is small enough and the effects of shear are negligible. For such fluids, the components of the deviatoric stress tensor are identically equal to zero $\sigma_{ij} = 0$. The constitutive equation is reduced to the normal part of the stress tensor:

$$\underline{\underline{\sigma}} = -p\left(\rho, e^{int}\right)\underline{\underline{I_d}},$$

$$\sigma_{ij} = -p\left(\rho, e^{int}\right)\delta_{ij} \tag{1.84}$$

If both heat source and heat flux terms are neglected, and if Eq.1.84 is substituted into the linear momentum and energy conservation equations, we can obtain the fluid dynamics Euler equations(the mass conservation equation does not change).

- **Conservation of mass**

$$\frac{d\rho}{dt} = -\rho\nabla_{\overrightarrow{x}}.\overrightarrow{v} \tag{1.85}$$

- **Conservation of linear momentum**

$$\rho\frac{d\overrightarrow{v}}{dt} = -\nabla_{\overrightarrow{x}}p + \rho\overrightarrow{b} \tag{1.86}$$

- **Conservation of energy**

$$\rho\frac{de^{int}}{dt} = -p\nabla_{\overrightarrow{x}}.\overrightarrow{v} \tag{1.87}$$

Newtonian Viscous Fluids   For Newtonian viscous fluids, the shear stresses are proportional to the rate of strains. The constant of proportionality is the

viscosity coefficient. The deviatoric stress tensor is given by:

$$\underline{\underline{\sigma}}^d = \left[\lambda tr(\underline{\underline{D}})\right]\underline{\underline{I_d}} + 2\mu\underline{\underline{D}},$$

$$\sigma_{ij}^d = \lambda D_{kk}\delta_{ij} + 2\mu D_{ij}, \tag{1.88}$$

where λ and μ are viscosity coefficients. Considering Stoke's relation $\lambda + (2\mu/3) = 0$ Eq.1.88 becomes:

$$\underline{\underline{\sigma}}^d = \mu\underline{\underline{\varepsilon}} = \mu\left[-\frac{2tr(\underline{\underline{D}})}{3}\underline{\underline{I_d}} + 2\underline{\underline{D}}\right],$$

$$\sigma_{ij}^d = \mu\varepsilon_{ij} = \mu\left[-\frac{2D_{kk}}{3}\delta_{ij} + 2D_{ij}\right] \tag{1.89}$$

If both heat source and heat flux terms are neglected, and if Eq.1.83 and Eq.1.89 are substituted into the linear momentum and energy conservation equations, we obtain the fluid dynamics compressible Navier-Stokes equations (the mass conservation equation remains unchanged):

- **Conservation of mass**

$$\frac{d\rho}{dt} = -\rho\frac{\partial v_i}{\partial x_i} \tag{1.90}$$

- **Conservation of linear momentum**

$$\rho\frac{d\vec{v}}{dt} = -\nabla_{\vec{x}}p + \mu\left(\nabla^2_{\vec{x}}\vec{v}\right) + \frac{1}{3}\mu\left(\nabla_{\vec{x}}\left(\nabla_{\vec{x}}.\vec{v}\right)\right) + \rho\vec{b} \tag{1.91}$$

- **Conservation of energy**

$$\rho\frac{de^{int}}{dt} = -p\frac{\partial v_i}{\partial x_i} + \frac{1}{2}\mu\varepsilon_{ij}\varepsilon_{ij} \tag{1.92}$$

1.3.3   Equations of State (EOS) for Fluids

The pressure was introduced as a new unknown in the set of governing equations. There are then five scalar equations for six unknowns (density, the three components of the velocity vector, the internal energy and the pressure). In order

to close both compressible Euler and Navier-Stokes equations, an equation of state (EOS) is needed in which the pressure $p\left(\rho, e^{int}\right)$ is expressed as a function of the density and the internal energy.

The EOS for compressible media must be completed with the speed of sound equation:

$$c = \sqrt{\left.\frac{dp}{d\rho}\right|_s},$$

(1.93)

where s denotes the entropy.

According to the first law of thermodynamics we have:

$$Tds = de^{int} + pd\left(\frac{1}{\rho}\right),$$

(1.94)

where T denotes the temperature. T is related to the internal energy by $e^{int} = c_v T$ where c_v denotes the specific heat at constant volume.

In consideration that:

$$de = \frac{\partial e^{int}}{\partial p}dp + \frac{\partial e^{int}}{\partial \rho}d\rho,$$

(1.95)

$$d\left(\frac{1}{\rho}\right) = -\frac{1}{\rho^2}d\rho,$$

(1.96)

and that at constant entropy $Tds = 0$, Eq.1.93 becomes:

$$c^2 = \frac{dp}{d\rho} = \frac{\frac{p}{\rho^2} - \frac{\partial e^{int}}{\partial \rho}}{\frac{\partial e^{int}}{\partial p}}$$

(1.97)

The EOS presented in this thesis are linear in volume and can be written in the following form:

$$p\left(\rho, e^{int}\right) = A(\rho) + B(\rho)e^{int}$$

(1.98)

We will briefly present the EOS used in this thesis for gases and liquids.

Ideal gas (IG) EOS   The ideal gas EOS is used to model gas compression and expansion such as air, helium and water vapor. It is defined by:

$$p\left(\rho, e^{int}\right) = (\gamma - 1)\rho e^{int} \tag{1.99}$$

where $\gamma = c_p/c_v$ is the ration between the specific heat at constant pressure c_p and the specific heat at constant volume c_v.

Jones-Wilkins-Lee (JWL) EOS   The Jones-Wilkins-Lee EOS is commonly used to model the thermodynamic behavior of explosives gas produced by detonation. Jones [57] originally proposed the EOS in which the pressure was a function of the relative volume and the temperature. Lately, Wilkins [155] provided another expression of the pressure as a function of the relative volume and the internal energy. Finally, Lee [78] extended both equations to provide the JWL EOS in its final form:

$$p^{jwl}\left(\rho, e^{int}\right) = A\left(1 - \frac{\omega\rho}{R_1\rho_0}\right)exp\left(-R_1\frac{\rho_0}{\rho}\right) + B\left(1 - \frac{\omega\rho}{R_2\rho_0}\right)exp\left(-R_2\frac{\rho_0}{\rho}\right) + \omega\rho e^{int},$$
$$\tag{1.100}$$

where A, B, R_1, R_2 and ω are material dependent parameters (see [34]) and ρ_0 is the reference density of the explosive. The name Jones-Wilkins-Lee of the EOS stands for the successive modifications of the EOS.

During the detonation process, the use of Eq.1.100 does not control the chemical energy released. To consider those effects the computed pressure p^{jwl} is multiplied a burn fraction factor $\eta \in [0, 1]$. In LS-DYNA, two types of burn fractions (η_1 and η_2) that are proposed based on the work of Wilkins [155] and Giroux [42]:

- η_1 computation:

$$\eta_1 = \frac{2(t - t_l)D_{CJ}}{3V_e/A_{emax}} \text{ if } t > t_l\,,$$
$$\eta_1 = 0 \text{ if } t \leq t_l\,, \tag{1.101}$$

  where the ratio V_e/A_{emax}, between the element volume and its maximum area, is a characteristic length. D_{CJ} and t_l are the detonation velocity and

the lightning time, respectively.

- η_2 computation:

$$\eta_2 = \frac{\rho D_{CJ}^2}{P_{CJ}} \left(1 - \frac{\rho_0}{\rho} \right), \tag{1.102}$$

where P_{CJ} is the Chapman-Jouguet pressure.

The user can choose between $\eta = 1$ or $\eta = \eta_1$ or $\eta = \eta_2$ or $\eta = max(\eta_1, \eta_2)$. When η is equal to 1, η remains constant and is equal to 1 which corresponds to an element that is burnt totally (fully transformed into gas). Finally the JWL EOS is given by:

$$p\left(\rho, e^{int}\right) = \eta p^{jwl}\left(\rho, e^{int}\right) \tag{1.103}$$

One may notice that for $A = B = 0$ and $\omega = \gamma - 1$ we recover the ideal gas EOS given in Eq.1.99.

Tait EOS   The barotropic Tait's EOS is often use to model both weakly-compressible and compressible water flow and has different form in the literature. Thompson [144] provides the most commonly used expression:

$$p(\rho) = B\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right] + p_0, \tag{1.104}$$

where $B = \rho_0 c^2$ is the bulk modulus, γ is a material dependent parameter. ρ_0 and p_0 are the reference density and pressure, respectively.

For barotropic EOS, the internal energy is not included in the computation of the pressure. Thus, it is not required to solve the conservation of energy equation to model the behavior of such fluids.

Stiffened Gas (SG) EOS   In consideration of water under very high pressures such as in UNDEX or WHs applications, another possible choice is the Stiffened Gas EOS defined by:

$$p\left(\rho, e^{int}\right) = (\gamma - 1)\rho e^{int} - \gamma \pi, \tag{1.105}$$

where γ and π are empirical material dependent parameters.

For $\pi = 0$ we recover the IG EOS given in Eq.1.99.

Mie-Gruneïsen (MG) EOS   For applications where the material acts differently when its under compression and expansion, the EOS must depend on the state of the material. For such materials, the MG EOS can be used. It is defined by:

- In compression: $\mu = \frac{\rho_0}{\rho} - 1 > 0$

$$p\left(\rho, e^{int}\right) = \frac{\rho_0 c^2 \left[1 + \left(1 - \frac{\gamma_0}{2}\right)\mu - \frac{a}{2}\mu^2\right]}{\left[1 - (S_1 - 1)\mu - S_2 \frac{\mu^2}{\mu+1} - S_3 \frac{\mu^3}{(\mu+1)^2}\right]^2} + (\gamma_0 + a\mu)\,e^{int}, \qquad (1.106)$$

- In expansion: $\mu = \frac{\rho_0}{\rho} - 1 < 0$

$$p\left(\rho, e^{int}\right) = \rho_0 c^2 \mu + (\gamma_0 + a\mu)\,e^{int}, \qquad (1.107)$$

where ρ_0 is the initial density, c is the speed of sound, γ_0 is the Gruneïsen coefficient, a is a volume correction coefficient. S_1, S_2 and S_3 are fitting coefficients provided by the Shock Hugoniot experimental data [88].

1.4   Phase change Modeling: One-Fluid HEM Model

1.4.1   Cut-Off Model

The Cut-Off model is a pure phase model. The fluid is assumed to be cavitating when it is below a pre-set cut-off pressure, which is usually the saturated water pressure p_{sat}. When the pressure is lower than the cut-off pressure $p \leq p_{sat}$ it is simply assumed to be equal to the cut-off pressure $p = p_{sat}$. To compute the liquid water pressure ($p > p_{sat}$), any classical EOS for modeling water can be used. Among the EOS defined in the previous subsection, one can choose either the Tait EOS (Eq.1.104), SG EOS (Eq.1.105) or the MG EOS (Eq.1.106 and Eq.1.107). Finally the one-fluid Cut-Off Model EOS becomes:

$$p = p_{eos} \text{ if } p > p_{sat}$$
$$p = p_{sat} \text{ if } p \leq p_{sat}$$

$$(1.108)$$

Although the pressure maintains constant when pressure drops below the saturated pressure, the density is still updated by solving the conservation of mass equation (Eq.1.30 and Eq.1.59). Otherwise, if the density was kept constant the sound would become zero. To artificially keep the system of conservation equations hyperbolic, the speed of sound must be kept positive. In order to do so, the following two methods are commonly used:

- I) Set the speed of sound to be constant. The saturated pressure p_{sat} and the saturated density ρ_{sat} are used to compute the speed of sound.

- II) Compute the speed of sound based on the updated density and constant cut-off pressure p_{sat}.

Fig.1.6 illustrates the pressure behavior versus density for water using the Tait EOS (Eq.1.104) combined with the cut-off at temperature $T = 293.15K$. The parameters for water are given in Tab. 1.1.

B (Pa)	ρ_0 (kg/m^3)	p_0 (Pa)	γ	p_{sat} (Pa)
3.3e+8	1000	1e+5	7.15	239.2

Table 1.1 – Tait EOS parameters for water at temperature $T = 293.15K$
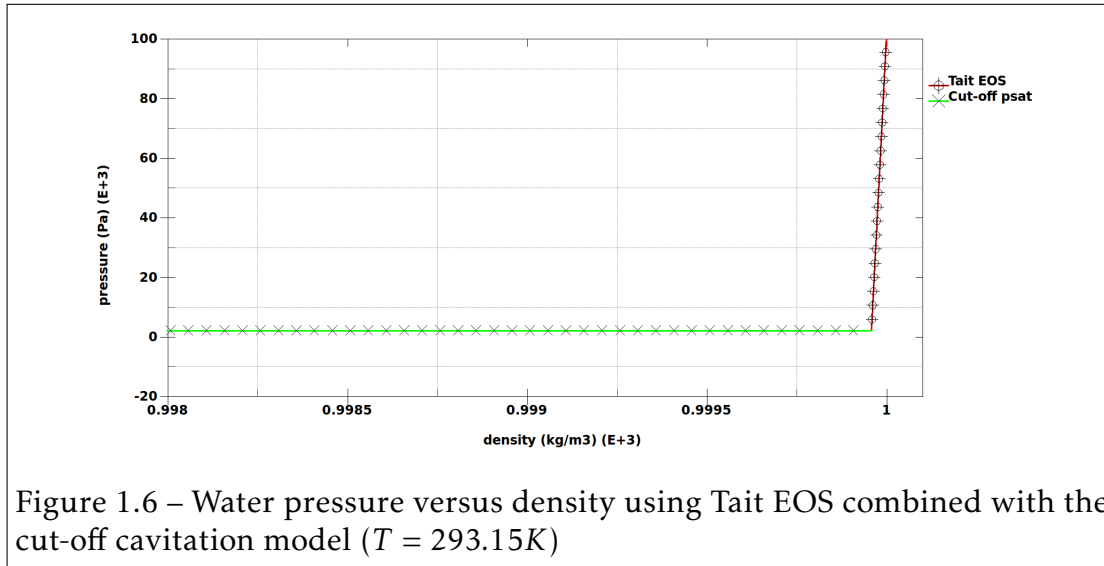
Figure 1.6 – Water pressure versus density using Tait EOS combined with the cut-off cavitation model ($T = 293.15K$)

1.4.2 Saurel et al. model

In this subsection we present the two models that we decided to study and implement to model phase change in compressible flows: the pure phase cut-off model that is already implemented in LS-DYNA and the model proposed by Saurel et al. [124]. Both models have been extensively used and validated for different applications [1, 2, 130, 24, 21, 68, 61, 67, 145].

Physical Assumptions and HEM equations

The method proposed by Saurel et al. [124] is used to model phase change in inviscid non-barotropic compressible fluids. The method is based on the following three main assumptions in the mixture region:

- The state variables ρ and e^{int} are mean quantities of both liquid and vapor phases. The mixture density ρ_m is a convex function of the vapor fraction α_v and both saturated liquid and vapor densities. The mixture internal

energy e_m^{int} is a convex function of the vapor mass fraction $Y_v = \frac{\rho_{v,sat}\alpha_v}{\rho}$ and both saturated liquid and vapor internal energies:

$$\rho_m = \alpha_v \rho_{v,sat}(T) + (1 - \alpha_v)\rho_{l,sat}(T), \quad (1.109)$$

$$e_m^{int} = Y_v e_{v,sat}^{int}(T) + (1 - Y_v)e_{l,sat}^{int}(T), \quad (1.110)$$

where the subscripts l,v,m and sat denote the liquid phase, the vapor phase, the mixture phase and the saturated state of the fluid, respectively.

The vapor fraction α_v is defined by:

$$\alpha_v = \frac{V_v}{V_m} = \begin{cases} 0 & if & \rho_m \geq \rho_{l,sat}(T) \\ \frac{\rho_m - \rho_{l,sat}(T)}{\rho_{v,sat}(T) - \rho_{l,sat}(T)} & if & \rho_{v,sat}(T) \leq \rho_m \leq \rho_{l,sat}(T) \\ 1 & if & \rho_m \leq \rho_{v,sat}(T) \end{cases} \quad (1.111)$$

where V_v is the vapor volume and V_m is the total mixture volume (see Fig.1.7).

The vapor mass fraction Y_v is defined by:

$$Y_v = \frac{\rho_{v,sat}(T)\alpha_v}{\rho_m} \quad (1.112)$$

In a similar way we define the liquid fraction α_l and the liquid mass fraction Y_l such that $\alpha_v + \alpha_l = 1$ and that $Y_v + Y_l = 1$.

- The liquid and vapor phases are in kinematic equilibrium:

$$\overrightarrow{v}_{l,sat} = \overrightarrow{v}_{v,sat} \quad (1.113)$$

- The liquid and vapor phases are in thermodynamic equilibrium:

$$p_{l,sat}(T) = p_{v,sat}(T) = p_{sat}(T) \quad (1.114)$$

$$T_{l,sat} = T_{v,sat} = T_{sat} \quad (1.115)$$

In this thesis, the academic and the industrial applications involving phase change are solved using Euler's equations (Eq.1.85, Eq.1.86 and Eq.1.87) under the assumptions for the mixture quantities (Eq.1.109 to Eq.1.115) as proposed

Figure 1.7 – Left: Discrete vapor bubbles. Right: Fringe levels of the vapor fraction α_v using the homogeneous model representation (Eq.1.111).
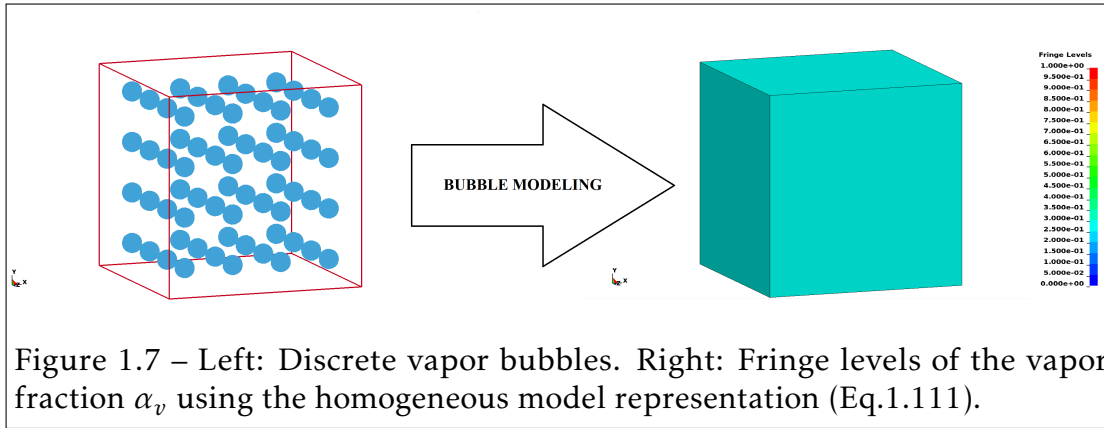
in the original paper of Saurel et al. [124].

conservation of mass

$$\frac{d\rho_m}{dt} = -\rho_m \nabla_{\overrightarrow{x}} . \overrightarrow{v} \tag{1.116}$$

conservation of linear momentum

$$\rho_m \frac{d\overrightarrow{v}}{dt} = \nabla_{\overrightarrow{x}} p \tag{1.117}$$

conservation of energy

$$\rho_m \frac{de_m^{int}}{dt} = -p \nabla_{\overrightarrow{x}} \overrightarrow{v} \tag{1.118}$$

More details on the derivation and the investigation of the HEM equations are provided in [27, 46, 71].

To close the HEM set of equations (Eq.1.116, Eq.1.117 and Eq.1.118), we need to provide an EOS that preserves the hyperbolic character of the HEM equations and that handles continuous phase transition (between liquid to vapor and between vapor to liquid), shock wave generation and its propagation.
In the two next subsections, we present the model used to define the physical properties of the saturated water, the EOS and sound speed equation which are based on the physical properties of water. The speed of sound is computed

using the simplified Wallis formula [154] in the mixture phase to preserve the hyperbolic character of the HEM equations.

In summary, the HEM model is a three equations model in which the density and the energy are mixture state variables. The model assumes that the pressure, the velocity and the temperature of the two phases are in equilibrium. The HEM model is conservative, hyperbolic (using Wallis formula [154]) and guarantees that the jump conditions are unique. A further advantage of the HEM model is that it does not require any empirical parameters to model phase change.

Physical parameters of saturated water

In the previous subsection, we defined mixture state variables in function of the fluid's parameters at the saturation state. The pressure, liquid density and vapor density at saturation state are given by the Oldenbourg polynomials [127]:

$$ln\left(\frac{p_{sat}(T)}{p_{cr}}\right) = \frac{T_{cr}}{T}\sum_{i=1}^{6} a_i\theta(T)^{\hat{a}_i}, \tag{1.119}$$

$$\frac{\rho_{l,sat}(T)}{\rho_{cr}} = 1 + \sum_{i=1}^{6} b_i\theta(T)^{\hat{b}_i}, \tag{1.120}$$

$$ln\left(\frac{\rho_{v,sat}(T)}{\rho_{cr}}\right) = \sum_{i=1}^{6} c_i\theta(T)^{\hat{c}_i}, \tag{1.121}$$

where $\theta(T)$ is the temperature function defined by:

$$\theta(T) = 1 - \frac{T}{T_{cr}} \tag{1.122}$$

$p_{cr} = 22.064.10^6 Pa, T_{cr} = 647.096K$ and $\rho_{cr} = 322.0 kg/m^3$ are the critical pressure, temperature and density for water, respectively. The coefficients of the Oldenbourg polynomials for water are given in Tab.1.2. The saturation equations (Eq.1.119, Eq.1.120 and Eq.1.121) are validated against NIST data provided online from [82]. The saturation pressure equation (Eq.1.119) is plotted in Fig.1.8 and saturation density equations for both liquid and vapor phases (Eq.1.120 and

Eq.1.121) are plotted in Fig.1.9.

Index	a	b	c	$\hat{a}$	$\hat{b}$	$\hat{c}$
1	-7.85823	1.99206	-2.02957	1	1/3	2/6
2	1.83991	1.10123	-2.68781	1.5	2/3	4/6
3	-11.7811	-0.512506	-5.38107	3	5/3	8/6
4	22.6705	-1.75263	-17.3151	3.5	16/3	18/6
5	-15.9393	-45.4485	-44.6384	4.0	43/3	37/6
6	1.77516	6.75615×10^5	-64.3486	7.5	110/3	71/6

Table 1.2 – Parameters of the saturation equation for water taken from [127]

Figure 1.8 – Water saturation pressure vs temperature. The red line represents the pressure from Schmidt (Eq.1.119) and the crossed circle symbol represents NIST pressure data from [82].
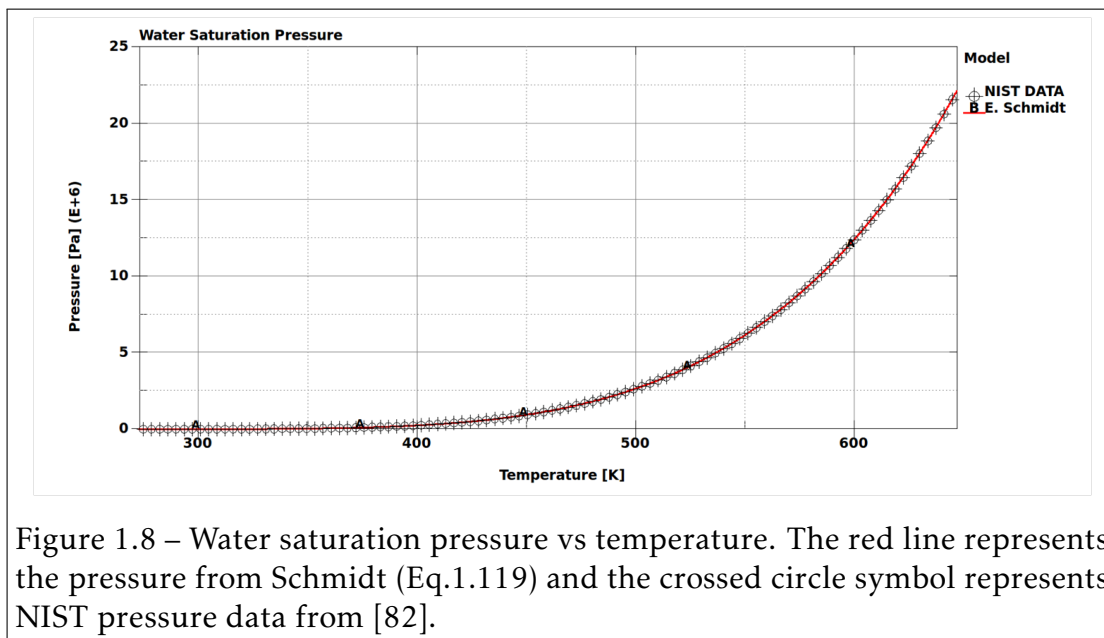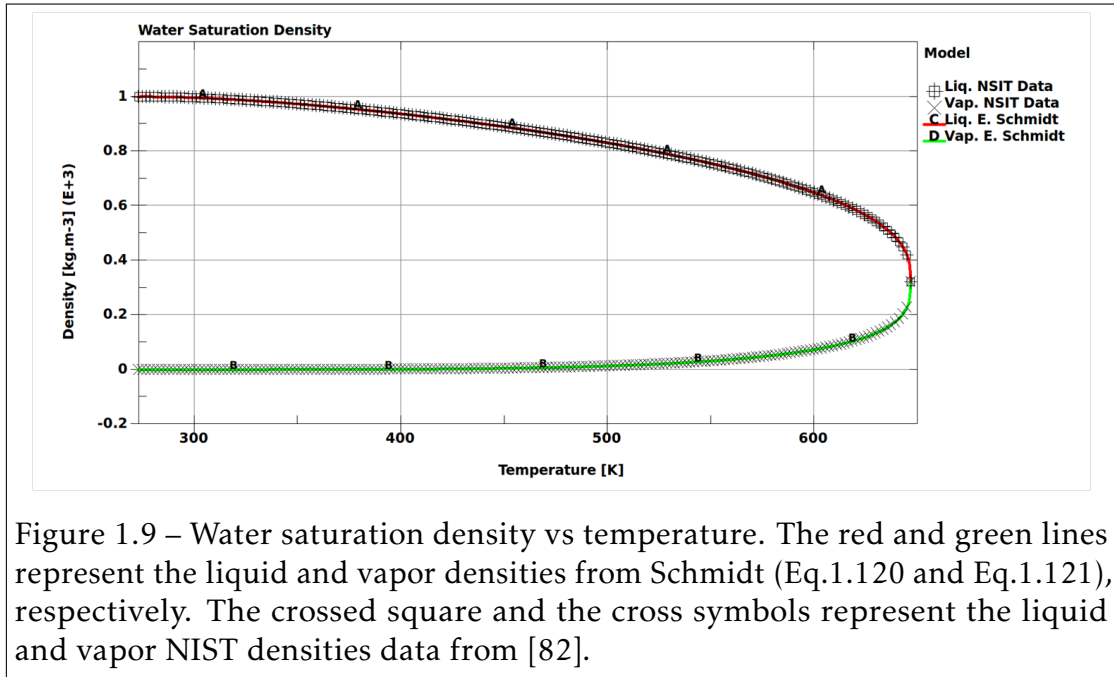
Figure 1.9 – Water saturation density vs temperature. The red and green lines represent the liquid and vapor densities from Schmidt (Eq.1.120 and Eq.1.121), respectively. The crossed square and the cross symbols represent the liquid and vapor NIST densities data from [82].

Equation of State and Speed of Sound

Liquid Phase ($\alpha_v = 0$)   To model compressible pure liquid water at high pressure (such as in UNDEX applications) three EOS are commonly used:

- The Tait EOS: Eq.1.104.

- The stiffened gas EOS: Eq.1.105.

- The Mie-Gruneïsen EOS: Eq.1.106 and Eq.1.107.

Among those three equations, only the Tait EOS is independent of temperature. Saurel et al. [124] suggests to include a temperature dependency that subsitutes the reference density ρ_0 by the saturated liquid water density $\rho_{l,sat}(T)$ and the reference pressure p_0 by the saturated water pressure $p_{sat}(T)$ in the original Tait EOS. The pressure in the liquid water region is computed using the modified Tait EOS defined by:

$$p = p\left(\rho(T), T\right) = B\left[\left(\frac{\rho(T)}{\rho_{l,sat}(T)}\right)^{\gamma_l} - 1\right] + p_{sat}(T), \tag{1.123}$$

where $B = 3.3 \times 10^8 Pa$ and $\gamma_l = 7.15$ (see [109, 130]). The pressure $p_{sat}(T)$ and the density $\rho_{l,sat}(T)$ are defined in Eq.1.119 and Eq.1.120, respectively. The associated internal energy equation for pure liquid water phase e_l^{int} is given by:

$$e_l^{int}(T) = C_{v,l}(T - T_0) + e_{l0}, \tag{1.124}$$

where $C_{v,l} = 4180 J.kg^{-1}.K^{-1}$, $T_0 = 273.15K$ and $e_{l0} = 617 J/kg$ are the specific heat of liquid water at constant volume, the reference temperature and the reference energy at the reference temperature T_0, respectively. The speed of sound of the liquid water phase $c_l(T)$ is given by:

$$c_l(T) = \sqrt{\frac{\gamma_l}{\rho(T)} (p + B)} \tag{1.125}$$

As discussed in Saurel et al. [130] and Sezal's thesis report [130], the modified Tait EOS (Eq.1.123) gives the best approximation for IAPWS data [153]. Thus, it was finally selected to model liquid water. It is noticed that for $\rho(T) = \rho_{l,sat}(T)$ the saturation pressure $p(\rho_{l,sat}(T), T) = p_{sat}$ is recovered.

Vapor phase ($\alpha_v = 1$)   The pressure in the water vapor phase region is computed using the IG EOS (Eq.1.99):

$$p = p(\rho(T), T) = \rho(T)RT, \tag{1.126}$$

where $R = 461.5 J.kg^{-1}.K^{-1}$ is the specific gas constant for vapor water. The associated internal energy equation for pure water vapor phase e_v^{int} is given by:

$$e_v^{int}(T) = C_{v,v}(T - T_0) + L_{v,0} + e_{l0}, \tag{1.127}$$

where $C_{v,v} = 1410.8 J.kg^{-1}.K^{-1}$, $T_0 = 273.15K$, $L_{v,0} = 2753.3 \times 10^3 J/kg$ and $e_{l0} = 617 J/kg$ are the specific heat of water vapor at constant volume, the reference temperature, the latent heat of vaporization of water at reference temperature T_0 and the reference energy at the reference temperature T_0, respectively. The speed of sound of the water vapor phase $c_v(T)$ is given by:

$$c_v(T) = \sqrt{\gamma_v R T}, \tag{1.128}$$

where $\gamma_v = 1.327$ is the adiabatic coefficient of the water vapor.

Mixture phase $(0 < \alpha_v < 1)$ In the mixture region the pressure is assumed to be in equilibrium between both liquid and vapor phases ($p = p_l(T) = p_v(T) = p_{sat}(T)$). This equilibrium is also assumed in the temperature in the mixture region ($T = T_l = T_v = T_{sat}$).

The mixture internal energy e_m^{int} is obtained by combining Eq.1.109, Eq.1.110, Eq.1.124 and Eq.1.127:

$$\rho_m(T)e_m^{int}(T) = \left[\alpha_v \rho_{v,sat}(T)e_v^{int}(T) + (1 - \alpha_v)\rho_{l,sat}(T)e_l^{int}(T)\right] \tag{1.129}$$

For the special case $\rho_m(T) = \rho_{l,sat}$ we have $\alpha_v = 0$ and we recover Eq.1.124 ($e_m^{int}(T) = e_l^{int}(T)$). In a similar way, for the special case $\rho_m(T) = \rho_{v,sat}$ we have $\alpha_v = 1$ and we recover Eq.1.127 ($e_m^{int}(T) = e_v^{int}(T)$).

The speed of sound in the mixture region c_m is computed using the simplified Wallis formula [154] ($c_m = c_{wallis}$):

$$\frac{1}{\rho_m(T)c_{wallis}^2} = \frac{\alpha_v}{\rho_{v,sat}(T)c_v^2} + \frac{1 - \alpha_v}{\rho_{l,sat}(T)c_l^2} \tag{1.130}$$

The approximation of the mixture speed of sound using the Wallis formula is shown in Fig.1.10 for $T = 293.15K$. The Wallis speed of sound formula increases the stability of the numerical simulation (see [130, 24, 61]) in addition to preserving the hyperbolicity of the system.

Numerical Procedure: To Solve numerically the HEM conservation equations (Eq.1.116, Eq.1.117 and Eq.1.118), the mixture variables ρ_m, $\overrightarrow{v}$ and e_m^{int} are obtained at a given time t_n. In order to compute the solution at the next time step ($t_{n+1} = t_n + \Delta t$) a leap frog time integration scheme (see section 2.1) is used in which the mixture variables are updated on half a time step $t_{n+1/2} = t_n + \Delta t/2$. We denote by $(.)^{n+1/2}$ a variable at time $t_{n+1/2}$, where $(.)$ can be replaced by any time dependent variable that has been previously defined. The HEM pressure

and the speed of sound at half a time step are computed as follows:
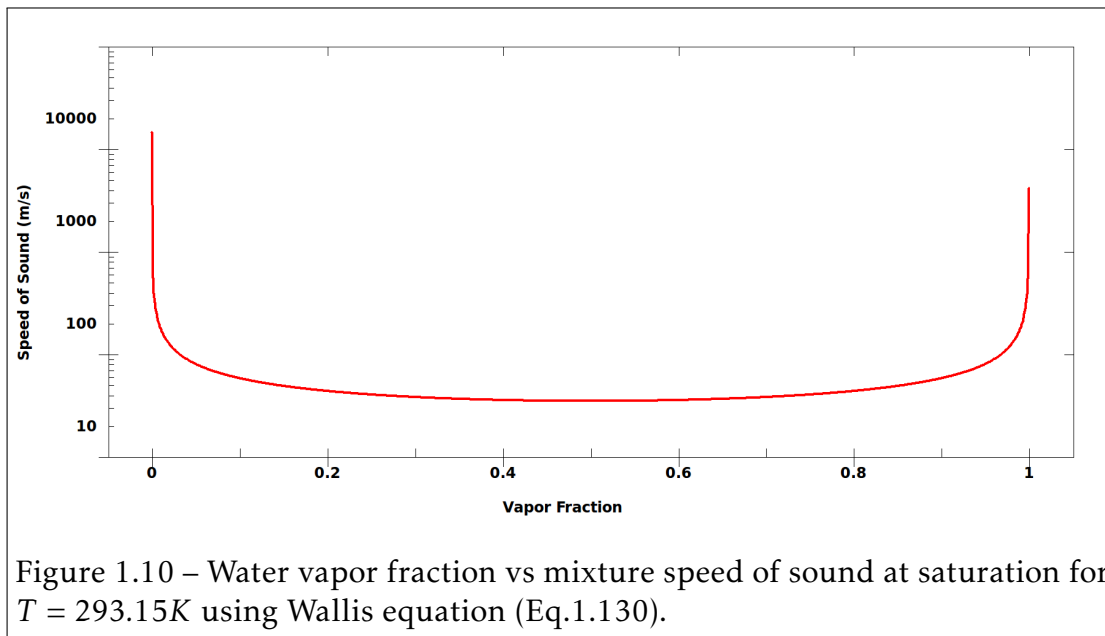
Figure 1.10 – Water vapor fraction vs mixture speed of sound at saturation for $T = 293.15K$ using Wallis equation (Eq.1.130).

Inputs: $\rho_m^{n+1/2}$, $e_m^{n+1/2}$ (obtained from Eq.1.116, Eq.1.117 and Eq.1.118)

- **Temperature estimation:** $T^{n+1/2}$

    1. Define initial guess $T_i = T_0^{n+1/2}$, $i = 0$. ($T_0^{n+1/2} = T^{n-1/2}$ was found to be a good starting value using fixed-point method).

    2. Considering the inputs $\rho_m^{n+1/2}$ and $e_m^{n+1/2}$ to be numerical values (solved by at previous time step and integrated in time), solve the non-linear equation Eq.1.129 iteratively (Newton or fixed-point method) to obtain the solution T_{i+1}.

    3. Compute the error and check the convergence:

        – IF $|T_{i+1} - T_i| < tol$ THEN $T^{n+1/2} = T_{i+1}$.

        – ELSE $T_i = T_{i+1}$ and repeat step 2 until the solution converges.

- **Parameters of saturated water computation:**
  We compute:

    – α_v using Eq.1.111.

    – $p_{sat}(T^{n+1/2})$ using Eq.1.119.

    – $\rho_{v,sat}(T^{n+1/2})$ using Eq.1.120.

    – $\rho_{l,sat}(T^{n+1/2})$ using Eq.1.121.

- **Pressure and speed of sound computation:**

    – IF $\alpha_v = 0$ THEN
      Compute $p^{n+1/2}$, $c_m^{n+1/2}$ using Eq.1.123 and Eq.1.125, respectively.

    – IF $0 < \alpha_v < 1$ THEN Compute $c_m^{n+1/2}$ using Eq.1.130 and $p^{n+1/2} = p_{sat}(T^{n+1/2})$.

    – IF $\alpha_v = 1$ THEN Compute $p^{n+1/2}$, $c_m^{n+1/2}$ using Eq.1.126 and Eq.1.128, respectively.

Outputs: $p^{n+1/2}$, $c_m^{n+1/2}$

Chapter 2

Arbitrary Lagrangian Eulerian and Fluid Structure Interaction

There are two ways to implement ALE equations, which correspond to the two approaches taken to implement the Eulerian viewpoint in fluid mechanics. The first way solves the fully coupled equations for computational fluid mechanics; this approach used by different authors can handle only a single material in an element as described for example in [108]. The alternative approach is referred to as an ***operator splitting method***, where the ALE governing equations (Eq.1.59, Eq.1.60 and Eq.1.61) are solved in two sequential steps, separating the time derivation operator (see [26, 17, 134]). The analysis performed by Chorin et al. [26] shows that, at its best, the method is second order accurate.

Let $\phi\left(\vec{x}, t\right)$ be a function of the space variable $\vec{x}$ and time variable t. We consider the hyperbolic equation defined by:

$$\frac{\partial \phi\left(\vec{x}, t\right)}{\partial t} + \vec{w}.\nabla_{\vec{x}}\phi\left(\vec{x}, t\right) = S\left(\vec{x}, t\right), \tag{2.1}$$

where $\phi\left(\vec{x}, t\right)$ is the solution, $\vec{w}$ is the convective velocity defined in subsection 1.3.1 and $S\left(\vec{x}, t\right)$ is the source term. Instead of solving Eq.2.1 directly, using the ***operator splitting method*** we solve the two following equations sequentially:

$$\frac{\partial \phi\left(\overrightarrow{x},t\right)}{\partial t} = S\left(\overrightarrow{x},t\right), \tag{2.2}$$

$$\frac{\partial \phi\left(\overrightarrow{x},t\right)}{\partial t} + \overrightarrow{w}.\nabla_{\overrightarrow{x}}\phi\left(\overrightarrow{x},t\right) = 0 \tag{2.3}$$

The first step is called the ***Lagrangian step*** (indeed Eq.2.2 is equivalent to Eq.2.1 if $\overrightarrow{w} = 0$). The second step is called the ***Eulerian step*** and corresponds to the transport equation of the quantity $\phi\left(\overrightarrow{x},t\right)$ at the convective velocity $\overrightarrow{w}$. Neglecting the heat source term s and the heat flux vector $\overrightarrow{q}$, we obtain the two sets of equations to be solved numerically by applying the ***operator splitting method*** to the ALE equations (Eq.1.59, Eq.1.60 and Eq.1.61):

- **Lagrangian step equation:**

$$\frac{\partial \rho}{\partial t} + \rho\nabla_{\overrightarrow{x}}.\overrightarrow{v} = 0$$

$$\rho\frac{\partial \overrightarrow{v}}{\partial t} - \nabla_{\overrightarrow{x}}.\underline{\underline{\sigma}} - \rho\,\overrightarrow{b} = 0 \tag{2.4}$$

$$\rho\frac{\partial e^{int}}{\partial t} - \underline{\underline{D}} : \underline{\underline{\sigma}} = 0$$

- **Eulerian step equation:**

$$\frac{\partial \rho}{\partial t} + \overrightarrow{w}.\nabla_{\overrightarrow{x}}\rho = 0$$

$$\rho\frac{\partial \overrightarrow{v}}{\partial t} + \overrightarrow{w}.\nabla_{\overrightarrow{x}}\overrightarrow{v} = 0 \tag{2.5}$$

$$\rho\frac{\partial e^{int}}{\partial t} + \overrightarrow{w}.\nabla_{\overrightarrow{x}}e^{int} = 0$$

The Lagrangian step (Eq.2.4) is solved using a finite element method (FEM), whereas the Eulerian step equations (Eq.2.5), which is also called the transport or advection step, is solved using a finite volume method (FVM).

Eulerian step equations (Eq.2.5) are rewritten in its conservative form in order to compute the FVM solution. Considering the Jacobian J_ψ of the function $\psi(\overrightarrow{\chi},t)$,

the conservative form of Eq.2.5 is given by:

$$\frac{\partial(\rho J_\psi)}{\partial t} + J_\psi \nabla_{\overrightarrow{x}}(\rho \overrightarrow{w}) = 0$$

$$\frac{\partial(\rho \overrightarrow{v} J_\psi)}{\partial t} + J_\psi \nabla_{\overrightarrow{x}}(\rho \overrightarrow{w} \otimes \overrightarrow{v}) = 0 \qquad (2.6)$$

$$\frac{\partial(\rho e^{int} J_\psi)}{\partial t} + J_\psi \nabla_{\overrightarrow{x}}(\rho e^{int} \overrightarrow{w}) = 0$$

From a numerical point of view, the material velocity $\overrightarrow{v}$ is the velocity of the nodes of the Lagrangian mesh. The convective velocity $\overrightarrow{w} = \overrightarrow{v} - \overrightarrow{v}_{ALE}$ is the velocity of the nodes in the more general ALE mesh. In particular, if the ALE velocity $\overrightarrow{v}_{ALE}$ is set to $\overrightarrow{v}$ then $\overrightarrow{w} = 0$ and the grid is fixed and referred to the Eulerian mesh. To sequentially solve the Lagrangian equations by FEM and the conservative Eulerian equations by FVM, we first allow the mesh to deform according to the material velocity in the Lagrangian phase. Secondly the finite volume fluxes are computed and the convection variables are projected to the ALE mesh. The two steps are illustrated in Fig.2.1 for both Eulerian and ALE formulation [133].

In this chapter, we present the ALE and FSI numerical solutions implemented in LS-DYNA [48] that were used to solve the continuum equations described in the two previous chapters. We first describe the FEM approximation of the Lagrangian step equations(Eq.2.4), then the FVM approximation of the conservative form of the Eulerian step equations (Eq.2.6). Thirdly, we describe the ALE multi-material formulation that is needed to model multiphase flows, and finally, the FSI coupling algorithms implemented in LS-DYNA.
At the end of the chapter, we present numerical examples and discuss the validation, the limitations, the advantages and the drawbacks of the different methods.

Figure 2.1 – One-dimensional advection for the Eulerian formulation (left) and the ALE formulation (right)[133].
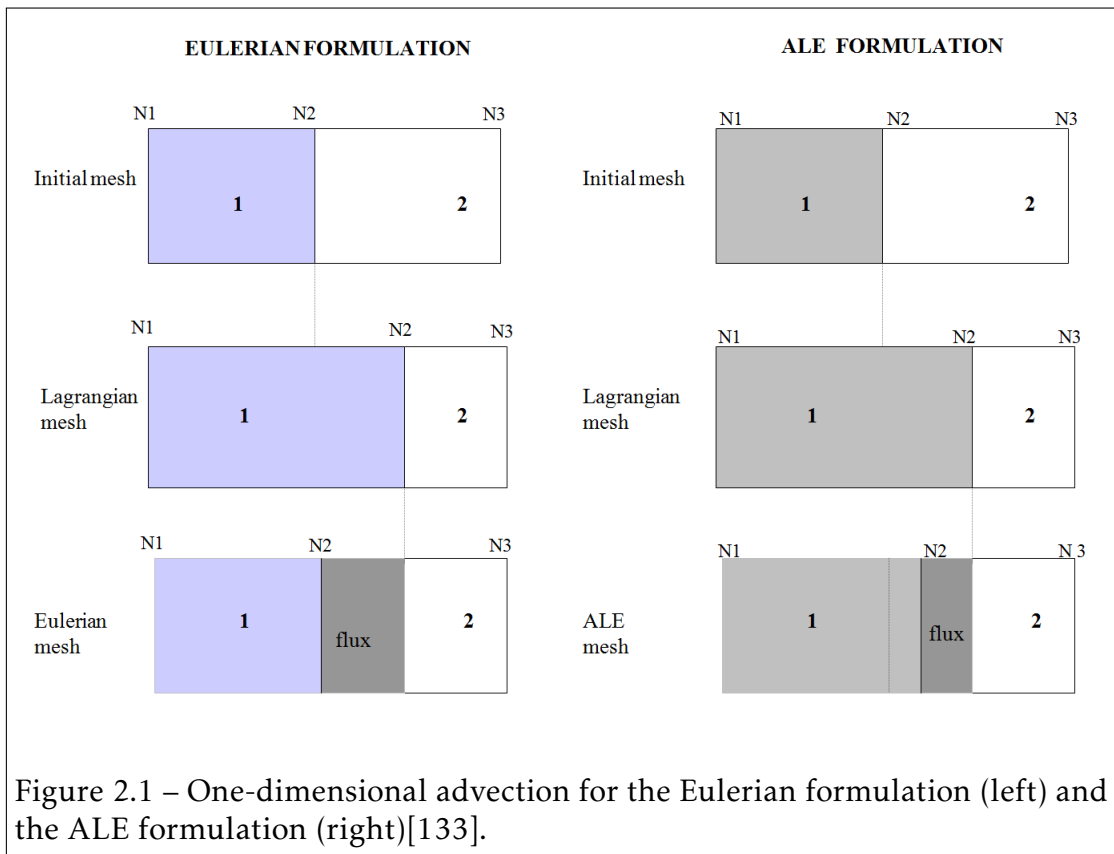
2.1   Time integration: Leapfrog (Verlet) method

The numerical solution is computed using the explicit Leapfrog time integration scheme. The method is based on a central finite difference approximation of the time derivative.

Let us consider the time dependent function $f(t)$ and a time-step Δt. The Taylor series for $f(t + \Delta t/2)$ and $f(t - \Delta t/2)$ are given by:

$$f(t + \Delta t/2) = f(t) + \left(\frac{\Delta t}{2}\right) \cdot \frac{df(t)}{dt} + \frac{1}{2!}\left(\frac{\Delta t}{2}\right)^2 \cdot \frac{d^2 f(t)}{dt^2} + \frac{1}{3!}\left(\frac{\Delta t}{2}\right)^3 \cdot \frac{d^3 f(t)}{dt^3} + O\left((\Delta t)^4\right)$$
$$(2.7)$$

$$f(t - \Delta t/2) = f(t) - \left(\frac{\Delta t}{2}\right) \cdot \frac{df(t)}{dt} + \frac{1}{2!}\left(\frac{\Delta t}{2}\right)^2 \cdot \frac{d^2 f(t)}{dt^2} - \frac{1}{3!}\left(\frac{\Delta t}{2}\right)^3 \cdot \frac{d^3 f(t)}{dt^3} + O\left((\Delta t)^4\right)$$
$$(2.8)$$

When Eq.2.8 is subtracted from Eq.2.7 and the result is divided by Δt, we obtain an approximation for the time derivative of the function $f(t)$:

$$\dot{f}(t) = \frac{df(t)}{dt} \simeq \frac{f(t + \frac{1}{2}\Delta t) - f(t - \frac{1}{2}\Delta t)}{\Delta t} \qquad (2.9)$$

And finally the expression of $f(t + \frac{1}{2}\Delta t)$:

$$f(t + \Delta t/2) \simeq f(t - \Delta t/2) + \Delta t.\dot{f}(t) \qquad (2.10)$$

The approximation in Eq.2.9 is second order accurate and conditionally stable. The computational time-step Δt_{cpu} is limited by the Courant-Friedrichs-Lewy (CFL) condition (see [30, 29, 17]) to ensure the stability of the explicit time integration scheme, and thus the convergence. The CFL condition for an element e_i is defined by:

$$\Delta t_{cpu} \leq \Delta t_{crit}^{e_i} = \frac{l_c^{e_i}}{c^{e_i}}, \qquad (2.11)$$

where $\Delta t_{crit}^{e_i}$ is the critical time-step of the element e_i. It is defined as the ratio between the characteristic length $l_c^{e_i}$ of the element and the speed of sound c^{e_i} through the material in the element.

In practice, the critical time-step $\Delta t_{crit}^{e_i}$ is computed for all the elements

$(i = 1, Nelts)$ and the computational time-step is taken as:

$$\Delta t_{cpu} = fac. \min_{e_i}(\Delta t^{e_i}_{crit}),\tag{2.12}$$

where fac is a scaling factor between 0 and 1.

Replacing the function $f(t)$ in Eq.2.10 by the density $\rho(t)$, the internal energy e^{int}, the velocity vector $\overrightarrow{v}(t)$ and the position $\overrightarrow{v}(t)$ we have:

$$
\begin{cases}
\rho(t + \Delta t/2) &= \rho(t - \Delta t/2) + \Delta t.\frac{d\rho(t)}{dt} \\[2ex]
e^{int}(t + \Delta t/2) &= e^{int}(t - \Delta t/2) + \Delta t.\frac{de^{int}(t)}{dt} \\[2ex]
\overrightarrow{v}(t + \Delta t/2) &= \overrightarrow{v}(t - \Delta t/2) + \Delta t.\frac{d\overrightarrow{v}(t)}{dt} \\[2ex]
\overrightarrow{x}(t + \Delta t) &= \overrightarrow{x}(t) + \Delta t.\overrightarrow{v}(t + \Delta t/2)
\end{cases}
\tag{2.13}
$$

Numerical Implementation   At the beginning of the computation (time t_0) the quantities $\rho(t_0)$, $e^{int}(t_0)$, $\overrightarrow{v}(t_0)$, $\overrightarrow{x}(t_0)$ and the time derivatives $\dot{\rho}(t_0)$, $\dot{e}^{int}(t_0)$, $\dot{\overrightarrow{v}}(t_0)$ are known.

At the first time-step, the variables $\rho(t_0)$, $e^{int}(t_0)$ and $\overrightarrow{v}(t_0)$ are updated on half time-step $t = t_0 + \Delta t/2$ and the variable $\overrightarrow{x}(t_0)$ on a full time-step $t = t_0 + \Delta t$ by:

$$
\begin{cases}
\rho(t_0 + \Delta t/2) &= \rho(t_0) + \frac{1}{2}\Delta t.\dot{\rho}(t_0) \\[2ex]
e^{int}(t_0 + \Delta t/2) &= e^{int}(t_0) + \frac{1}{2}\Delta t.\dot{e}^{int}(t_0) \\[2ex]
\overrightarrow{v}(t_0 + \Delta t/2) &= \overrightarrow{v}(t_0) + \frac{1}{2}\Delta t.\dot{\overrightarrow{v}}(t_0) \\[2ex]
\overrightarrow{x}(t_0 + \Delta t) &= \overrightarrow{x}(t_0) + \Delta t.\overrightarrow{v}(t_0 + \Delta t/2)
\end{cases}
\tag{2.14}
$$

At each subsequent time-step, the density, internal energy and velocity are

updated to half-time-step:

$$
\begin{cases}
\rho(t) &=& \rho(t - \Delta t/2) + \frac{1}{2}\Delta t . \dot{\rho}(t - \Delta t) \\[2em]
e^{int}(t) &=& e^{int}(t - \Delta t/2) + \frac{1}{2}\Delta t . \dot{e}^{int}(t - \Delta t) \\[2em]
\overrightarrow{v}(t) &=& \overrightarrow{v}(t - \Delta t/2) + \frac{1}{2}\Delta t . \overrightarrow{\dot{v}}(t - \Delta t)
\end{cases}
\tag{2.15}
$$

At the end of the subsequent time-step the density, internal energy, velocity and position vectors are updated according to the Leapfrog scheme (Eq.2.10):

$$
\begin{cases}
t &=& t + \Delta t \\[2em]
\rho(t + \Delta t/2) &=& \rho(t - \Delta t/2) + \Delta t . \dot{\rho}(t) \\[2em]
e^{int}(t + \Delta t/2) &=& e^{int}(t - \Delta t/2) + \Delta t . \dot{e}^{int}(t) \\[2em]
\overrightarrow{v}(t + \Delta t/2) &=& \overrightarrow{v}(t - \Delta t/2) + \Delta t . \overrightarrow{\dot{v}}(t) \\[2em]
\overrightarrow{x}(t + \Delta t) &=& \overrightarrow{x}(t) + \Delta t . \overrightarrow{v}(t + \Delta t/2)
\end{cases}
\tag{2.16}
$$

2.2   Lagrangian FE formulation: Lagrangian phase

2.2.1   Weak formulation

In the Lagrangian formulation, the resolution of the mass conservation equation is trivial. Indeed, in consideration of the definition of the Jacobian J_ϕ of the transformation function $\vec{\phi}\left(\vec{X},t\right)$ between the Eulerian coordinates $\vec{x}$ and the Lagrangian coordinates $\vec{X}$, we have:

$$\rho\left(\vec{X},t\right)J_\phi\left(\vec{X},t\right) = \rho\left(\vec{X},0\right)J_\phi\left(\vec{X},0\right) = \rho\left(\vec{X},0\right) \tag{2.17}$$

Thus the mass conservation equation is solved algebraically using Eq.2.17.

In FEM, we seek a weak solution of the continuum equations based on an integral formulation. For the sake of clarity, the equations in this subsection are expressed in suffix notations, except when it is mentioned.

Let U_0 and U be the functions spaces defined by:

$$\delta v_i\left(\vec{X},t\right) \in U_0 \quad , \quad U_0 = \{\delta v_i | \delta v_i \in H^1 \, , \, \delta v_i = 0 \text{ on } \Gamma^{\vec{v}}\}, \tag{2.18}$$

$$v_i\left(\vec{X},t\right) \in U \quad , \quad U = \{v_i | v_i \in H^1 \, , \, v_i = v_i^{bdy} \text{ on } \Gamma^{\vec{v}}\}, \tag{2.19}$$

where H^1 is the first order Sobolev space and δv_i is the virtual velocity (test function) defined to be zero on the boundary $\Gamma^{\vec{v}}$.

To obtain the weak formulation, we first multiply the linear momentum conservation equation by the test function δv_i and we integrate the result over the spatial domain Ω:

$$\int_\Omega \delta v_i\left(\rho\frac{\partial v_i}{\partial t} - \frac{\partial \sigma_{ji}}{\partial x_j} - \rho b_i\right)d\Omega = 0 \tag{2.20}$$

Considering that:

$$\int_\Omega \delta v_i \frac{\partial \sigma_{ji}}{\partial x_j} d\Omega = \int_\Omega \left[\frac{\partial(\delta v_i \sigma_{ji})}{\partial x_j} - \sigma_{ji} \frac{\partial \delta v_i}{\partial x_j} \right] d\Omega \qquad (2.21)$$

And applying the Gauss theorem to the right hand side first term in Eq.2.21 integral, we have:

$$\int_\Omega \frac{\partial(\delta v_i \sigma_{ji})}{\partial x_j} d\Omega = \underbrace{\int_{\Gamma^{\vec{v}}} \delta v_i \sigma_{ji}.n_j d\Gamma}_{= 0 \text{ due to definition of } U_0 \text{ in Eq.2.18}} + \int_{\Gamma^{\vec{t}}} \delta v_i \sigma_{ji}.n_j d\Gamma$$

$$ \qquad (2.22)$$

$$\int_\Omega \frac{\partial(\delta v_i \sigma_{ji})}{\partial x_j} d\Omega = \int_{\Gamma^{\vec{t}}} \delta v_i t_i d\Gamma$$

Substituting the right hand side of Eq.2.22 into Eq.2.21 and replacing the result in Eq.2.20, we finally obtain the weak formulation for the linear momentum equation:

$$\int_\Omega \left[\frac{\partial \delta v_i}{\partial x_j} \sigma_{ji} - \delta v_i \rho b_i + \delta v_i \rho \frac{\partial v_i}{\partial t} \right] d\Omega - \int_{\Gamma^{\vec{t}}} \delta v_i t_i d\Gamma = 0 \qquad (2.23)$$

Eq.2.23 can be expressed in term of the virtual powers:

$$\delta P = \delta P^{int} - \delta P^{ext} + \delta P^{kin} = 0, \qquad (2.24)$$

where

$$\delta \mathcal{P}^{int} = \int_\Omega \frac{\partial \delta v_i}{\partial x_j} \sigma_{ji} d\Omega,$$

$$\delta \mathcal{P}^{ext} = \int_\Omega \delta v_i \rho b_i d\Omega + \int_{\Gamma^{\vec{t}}} \delta v_i t_i d\Gamma, \qquad (2.25)$$

$$\delta \mathcal{P}^{kin} = \int_\Omega \delta v_i \rho \frac{\partial v_i}{\partial t} d\Omega$$

There are several element technologies to discretize the continuum model into a FE mesh. Most of them are present in FE commercial codes [48, 119, 9, 4]. In the next subsection, we present the two finite element technologies implemented in LS-DYNA that are used in this thesis:

- The eight-node solid hexahedron element to model Langrangian structures

and ALE fluids.

- The four nodes Belytschko-Lin-Tsay (BLT) shell element to model thin plate structures.

Let us denote by δe^{int} the virtual internal energy. The weak formulation for the energy conservation equation can be developed in a similar way. It is given by:

$$\int_{\Omega} \delta e^{int} \frac{\partial e^{int}}{\partial t} d\Omega - \int_{\Omega} \delta e^{int} \frac{\partial v_i}{\partial x_j} \sigma_{ij} d\Omega - \int_{\Omega} \delta e^{int} v_i \rho b_i d\Omega = 0 \qquad (2.26)$$

2.2.2   Internal and Hourglass forces computation

Let us consider the subdivision of the problem domain Ω into a set of k finite element Ω_{e_k} such that $\Omega = \cup_{e_k} \Omega_{e_k}$, the FEM approximates the solution of the field variables $x_{iI}(\overrightarrow{X}, t)$ and $v_{iI}(\overrightarrow{X}, t)$ at each node I of the FE mesh.
Let us consider the basis of the linear shape functions $N_I(\overrightarrow{X})$ that satisfy:

$$N_I(\overrightarrow{X_J}) = \begin{cases} 1 & if \quad I = J \\ 0 & if \quad I \neq J \end{cases} \qquad (2.27)$$

and

$$\sum_I N_I(\overrightarrow{X}) = 1 \qquad (2.28)$$

The expressions of the field variables $x_i(\overrightarrow{X}, t)$, $v_i(\overrightarrow{X}, t)$ and $\delta v_i(\overrightarrow{X}, t)$ in the basis of the shape functions $N_I(\overrightarrow{X})$ are given by:

$$x_i(\overrightarrow{X}, t) = x_{iI}(t) N_I(\overrightarrow{X}) \qquad (2.29)$$

$$v_i(\overrightarrow{X}, t) = v_{iI}(t) N_I(\overrightarrow{X}) \qquad (2.30)$$

$$\delta v_i(\overrightarrow{X}, t) = \delta v_{iI}(t) N_I(\overrightarrow{X}) \qquad (2.31)$$

Substituting Eq.2.29 and Eq.2.30 into Eq.2.25, we have:

$$
\begin{aligned}
\delta P^{int} &= \delta v_{iI} \int_\Omega \frac{\partial N_I}{\partial x_j} \sigma_{ji} d\Omega \\[2mm]
\delta P^{ext} &= \delta v_{iI} \left[\int_\Omega N_I \rho b_i d\Omega + \int_{\Gamma_{\vec{t}}} N_I t_i d\Gamma \right] \\[2mm]
\delta P^{kin} &= \delta v_{iI} \frac{\partial v_{iJ}}{\partial t} \int_\Omega N_I N_J \rho d\Omega
\end{aligned}
\tag{2.32}
$$

Now, substituting Eq.2.32 into Eq.2.24, we have:

$$
\delta v_{iI} \left[\int_\Omega \frac{\partial N_I}{\partial x_j} \sigma_{ji} d\Omega - \left(\int_\Omega N_I \rho b_i d\Omega + \int_{\Gamma_{\vec{t}}} N_I t_i d\Gamma \right) + \frac{\partial v_{iJ}}{\partial t} \int_\Omega N_I N_J \rho d\Omega \right] = 0
\tag{2.33}
$$

Since the test functions δv_i are arbitrary, Eq.2.33 can only be satisfied if the term between the square brackets [.] is zero. This result leads us finally to the FE discrete equation in matrix form:

$$
M^c . A = F^{ext} - F^{int},
\tag{2.34}
$$

where:

- M^c is the consistent mass matrix defined by:

$$
M_{IJ}^c = \int_\Omega N_I N_J \rho d\Omega
\tag{2.35}
$$

- A_I is the I-node unknown acceleration vector defined by:

$$
A_{iI} = \frac{\partial v_{iI}}{\partial t}
\tag{2.36}
$$

- F_I^{int} is the I-node internal force vector defined by:

$$
F_{iI}^{int} = \int_\Omega \frac{\partial N_I}{\partial x_j} \sigma_{ji} d\Omega
\tag{2.37}
$$

- F_I^{ext} is the I-node external force vector defined by:

$$F_{iI}^{ext} = \int_{\Omega} N_I \rho b_i d\Omega + \int_{\Gamma^{\vec{t}}} N_I t_i d\Gamma \tag{2.38}$$

The solution of Eq.2.34 is obtained by solving a system of linear equations that is CPU time consuming. Therefore, the consistent mass matrix M^c is replaced by a ***diagonal lumped mass matrix*** M^{lump} that is defined by:

$$M_{IJ}^{lump} = \begin{cases} \int_{\Omega} N_I \rho d\Omega & if \quad I = J \\ 0 & if \quad I \neq J \end{cases} \tag{2.39}$$

Using the lumped mass matrix instead of the consistent mass matrix, the accelerations at each node I of the FE mesh are simply given by:

$$A_I = \left(F_I^{ext} - F_I^{int}\right)/M_{II}^{lump} \tag{2.40}$$

The results for the integral $\int_{\Omega}(.)d\Omega$ is obtained by summing up all the elementary results $\int_{\Omega}(.)d\Omega = \sum_k \int_{\Omega_{e_k}}(.)d\Omega_{e_k}$.

In the previous equations, the shape functions are expressed in terms of the spatial coordinates in the current configuration. For computational purposes, it is common to express the shape functions in terms of the so-called parent domain, in which a reference element is defined in a local parametric coordinate system $(\xi, \eta, \zeta) \in [-1, 1]^3$ for three-dimensional problems $((\xi, \eta) \in [-1, 1]^2$ for two-dimensional problems, respectively).

At this stage, we have defined three different domains for a three-dimensional finite element Ω_{e_k}:

- The parent domain is expressed in terms of the local parametric coordinates (ξ, η, ζ).

- The current element domain is expressed in terms of the Eulerian coordinates $\vec{x}(t)$.

- The initial reference domain is expressed in terms of the Lagrangian coordinates $\vec{X}$.

And the different mapping functions:

- From the parent domain to the initial reference domain: $\overrightarrow{X}(\xi, \eta, \zeta)$.

- From the parent domain to the current element domain: $\overrightarrow{x}(\xi, \eta, \zeta, t)$.

- From the initial reference domain to the current element domain:
  $\overrightarrow{x} = \overrightarrow{\phi}(\overrightarrow{X}, t)$.

Let us consider the integral of the function $f(x)$ over the current finite element Ω_{e_k}. Using the mapping $\overrightarrow{x}(\xi, \eta, \zeta)$, the integrals over the finite element Ω_{e_k} are calculated by:

$$\int_{\Omega_{e_k}} f(\overrightarrow{x}) d\Omega_{e_k} = \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} f(\xi, \eta, \zeta) det[Jac(\xi, \eta, \zeta)] d\xi d\eta d\zeta, \qquad (2.41)$$

where $det[Jac(\xi, \eta, \zeta)]$ is the determinant of the Jacobian matrix $Jac(\xi, \eta, \zeta)$ defined by:

$$Jac(\xi, \eta, \zeta) = \begin{bmatrix} \frac{\partial x_1}{\partial \xi} & \frac{\partial x_1}{\partial \eta} & \frac{\partial x_1}{\partial \zeta} \\[2mm] \frac{\partial x_2}{\partial \xi} & \frac{\partial x_2}{\partial \eta} & \frac{\partial x_2}{\partial \zeta} \\[2mm] \frac{\partial x_3}{\partial \xi} & \frac{\partial x_3}{\partial \eta} & \frac{\partial x_3}{\partial \zeta} \end{bmatrix} \qquad (2.42)$$

The integrals can be evaluated numerically using a quadrature rule. To evaluate the right hand side integral in Eq.2.41, we select a reduced one-point (at the central point $(0,0,0)$) Gauss quadrature defined by:

$$\int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} f(\xi, \eta, \zeta) det[Jac(\xi, \eta, \zeta)] d\xi d\eta d\zeta = 8 f(0,0,0) det[Jac(0,0,0)] \quad (2.43)$$

The two main advantages of the reduced one-point Gauss quadrature rule are the CPU time saved (compared to fully integrated) and its ability to overcome the numerical locking phenomena (see [15]).

The counterparts of the one-point integration is the introduction of **Hourglass modes** (also called zero energy modes) [134, 15] that threaten the accuracy of the FE solution. Additional stabilization hourglass forces, which are denoted by F^{hrg}, need to be added to the total force vector to correct the parasite modes.

Figure 2.2 – Eight-node solid hexahedron element [48].
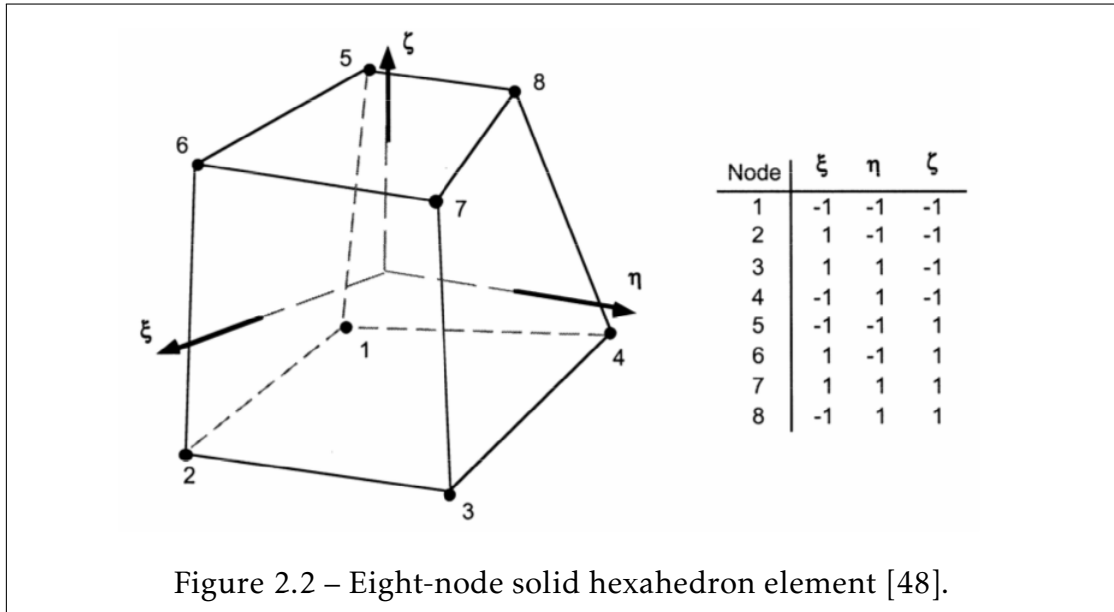
Finally, adding hourglass stabilization, Eq.2.40 becomes:

$$A_I = \left(F_I^{ext} - F_I^{int} + F_I^{hrg}\right)/M_{II}^{lump} \tag{2.44}$$

We will now describe the expression of the shape functions and the computation of the internal and hourglass forces for both eight-node solid hexahedron and four nodes BLT shell element technologies.

Eight-node solid hexahedron elements   The expression of the reference eight-node solid hexahedron element shape functions are defined in the parent domain by:

$$N_I(\xi, \eta, \zeta) = (1 + \xi\xi_I)(1 + \eta\eta_I)(1 + \zeta\zeta_I), \tag{2.45}$$

where I is the node index, from 1 to 8. The nodal parameters are given in Fig.2.2.

Internal nodal forces evaluation:

The internal nodal force vector F_I^{int} can be rewritten in matrix form:

$$F_I^{int} = \int_{\Omega_{e_k}} B_I^T . \sigma \, d\Omega_{e_k}, \tag{2.46}$$

where

$$\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T \tag{2.47}$$

and

$$B_I = \begin{bmatrix} \frac{\partial N_I}{\partial x_1} & 0 & 0 \\[2ex] 0 & \frac{\partial N_I}{\partial x_2} & 0 \\[2ex] 0 & 0 & \frac{\partial N_I}{\partial x_3} \\[2ex] \frac{\partial N_I}{\partial x_2} & \frac{\partial N_I}{\partial x_1} & 0 \\[2ex] 0 & \frac{\partial N_I}{\partial x_3} & \frac{\partial N_I}{\partial x_2} \\[2ex] \frac{\partial N_I}{\partial x_3} & 0 & \frac{\partial N_I}{\partial x_1} \end{bmatrix} \tag{2.48}$$

The terms in the matrix B_I (Eq.2.48) are given by the mapping function from the parent domain to the current element domain:

$$\begin{bmatrix} \frac{\partial N_I}{\partial x_1} \\[2ex] \frac{\partial N_I}{\partial x_2} \\[2ex] \frac{\partial N_I}{\partial x_3} \end{bmatrix} = Jac^{-1}(\xi, \eta, \zeta) . \begin{bmatrix} \frac{\partial N_I}{\partial \xi} \\[2ex] \frac{\partial N_I}{\partial \eta} \\[2ex] \frac{\partial N_I}{\partial \zeta} \end{bmatrix}, \tag{2.49}$$

where $Jac(\xi, \eta, \zeta)$ is the Jacobian matrix defined in (Eq.2.42). Finally, the integral in (Eq.2.46) is evaluated using the one-point rule given in Eq.2.43.

Hourglass nodal forces evaluation:

The hourglass modes (see Fig.2.3) are parasite modes and a threat to the accuracy of the FE solution and more generally to the stability of explicit FE

Figure 2.3 – The hourglass modes of an eight-node solid hexahedron element [38, 48].
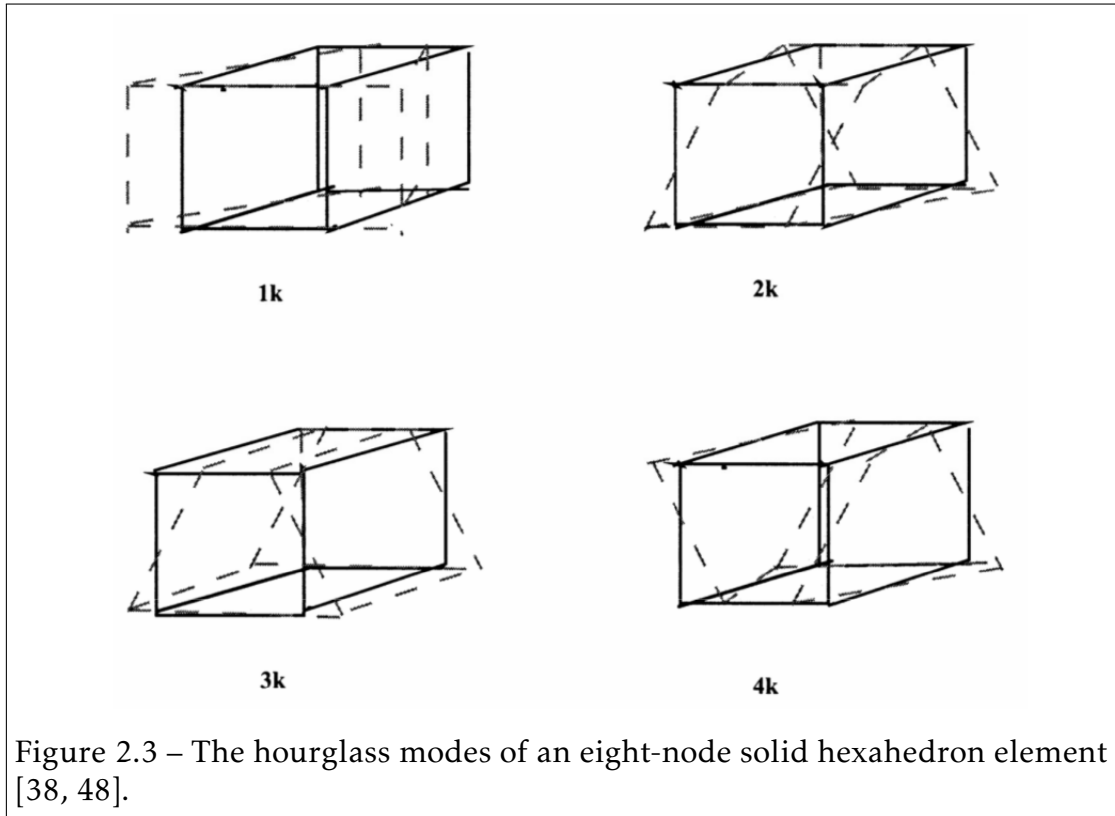
codes. The hourglass deformation modes are orthogonal to the strain calculation. Kosloff and Frazier [69] proposed to add hourglass forces in the form of a viscous damping or a small elastic stiffness that limits the formation of those undesirable modes. Let us consider the four base vectors Γ^k ($k = 1, 2, 3, 4$) that are related to the hourglass modes, which are defined by:

$$
\begin{aligned}
\Gamma^1 &= [\ \ +1 \ \ +1 \ \ -1 \ \ -1 \ \ -1 \ \ -1 \ \ +1 \ \ +1 \ \ ]^T \\[6pt]
\Gamma^2 &= [\ \ +1 \ \ -1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ +1 \ \ -1 \ \ ]^T \\[6pt]
\Gamma^3 &= [\ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ ]^T \\[6pt]
\Gamma^4 &= [\ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ ]^T
\end{aligned}
\tag{2.50}
$$

The four hourglass velocity rates $\dot{q}^k$ ($k = 1, 2, 3, 4$) are defined by:

$$
\begin{aligned}
\dot{q}^1 &= \begin{bmatrix} \sum_{I=1}^{8} \Gamma_I^1 . v_{1I} & \sum_{I=1}^{8} \Gamma_I^1 . v_{2I} & \sum_{I=1}^{8} \Gamma_I^1 . v_{3I} \end{bmatrix}^T \\[2mm]
\dot{q}^2 &= \begin{bmatrix} \sum_{I=1}^{8} \Gamma_I^2 . v_{1I} & \sum_{I=1}^{8} \Gamma_I^2 . v_{2I} & \sum_{I=1}^{8} \Gamma_I^2 . v_{3I} \end{bmatrix}^T \\[2mm]
\dot{q}^3 &= \begin{bmatrix} \sum_{I=1}^{8} \Gamma_I^3 . v_{1I} & \sum_{I=1}^{8} \Gamma_I^3 . v_{2I} & \sum_{I=1}^{8} \Gamma_I^3 . v_{3I} \end{bmatrix}^T \\[2mm]
\dot{q}^4 &= \begin{bmatrix} \sum_{I=1}^{8} \Gamma_I^4 . v_{1I} & \sum_{I=1}^{8} \Gamma_I^4 . v_{2I} & \sum_{I=1}^{8} \Gamma_I^4 . v_{3I} \end{bmatrix}^T
\end{aligned}
\tag{2.51}
$$

Finally, the node I hourglass force vector is given by:

$$
F_I^{hrg} = \frac{1}{4} \rho c Q_{hrg} (V_e)^{2/3}
\begin{bmatrix}
\sum_{k=1}^{4} \dot{q}_1^k . \Gamma_I^k \\[3mm]
\sum_{k=1}^{4} \dot{q}_2^k . \Gamma_I^k \\[3mm]
\sum_{k=1}^{4} \dot{q}_3^k . \Gamma_I^k
\end{bmatrix},
\tag{2.52}
$$

where ρ, c, Q_{hrg} and V_e are the density, the speed of sound, an input dimensional scaling coefficient and the volume of the element, respectively. Q_{hrg} is a user-defined constant value that is usually set between 0.05 and 0.15.

Four-nodes BLT shell elements   Shell elements are degenerated solid elements in which all the quantities are computed with respect to the mid-surface (see Fig.2.4). There are two main theories that describe the motion of shells:

- *Kirchoff-Love (KL):* The normal vector to the mid-surface remains straight and normal. Shear deformations are not possible.

- *Mindlin-Reissner (MR):* The normal vector to the mid-surface remains straight. Shear deformations are possible.

The BLT shells belong to the *Continuum-Based shells (CB)* family. The main assumptions for the BLT shells are the following:

- The normal stress to the mid-surface σ_{33} is zero (Plane stress theory).

- The fiber remains straight (Modified MR kinematic assumption).

Figure 2.4 – Shell element representation: a degenerated solid element [51].
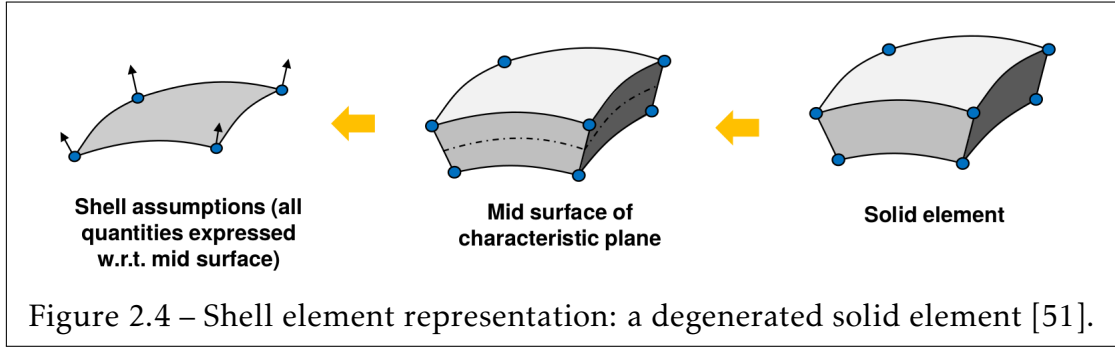
The BLT shell element was originally proposed by Belytschko et al. [13]. The BLT shells are continuum based elements that combine a co-rotational coordinate system and a velocity-strain formulation. The BLT shells are present in most of the explicit FE commercial codes [4, 9, 48, 119] due its robustness and its fast computation.

Most of the shell elements are low degree interpolation elements. The area of the elements, is interpolated from the nodes by bilinear functions.

The co-rotational assumption avoids the complexity of non-linear mechanics by embedding a coordinate system in the element. The decision to use velocity-strain in the formulation facilitates the constitutive evaluation, because the conjugate stress is the more familiar Cauchy stress.

Co-rotational coordinates:

Let $n_I(I = 1, 2, 3, 4)$ be the element's four corner nodes, and $(\overrightarrow{r_{21}} = \overrightarrow{n_2 n_1}, \overrightarrow{r_{31}} = \overrightarrow{n_3 n_1}, \overrightarrow{r_{42}} = \overrightarrow{n_4 n_2})$ be an embedded coordinate system that deforms with the element.

The different steps to compute the co-rotational coordinate system (shown in Fig.2.5) are as follows:

- Calculate a unit vector normal to the main diagonal of the element as:

$$s_{3i} = (\overrightarrow{r_{31}} \times \overrightarrow{r_{42}})_i, \tag{2.53}$$

$$\| \overrightarrow{s_3} \| = \sqrt{s_{31}^2 + s_{32}^2 + s_{33}^2}, \tag{2.54}$$

Figure 2.5 – Local coordinates on a shell element [48].
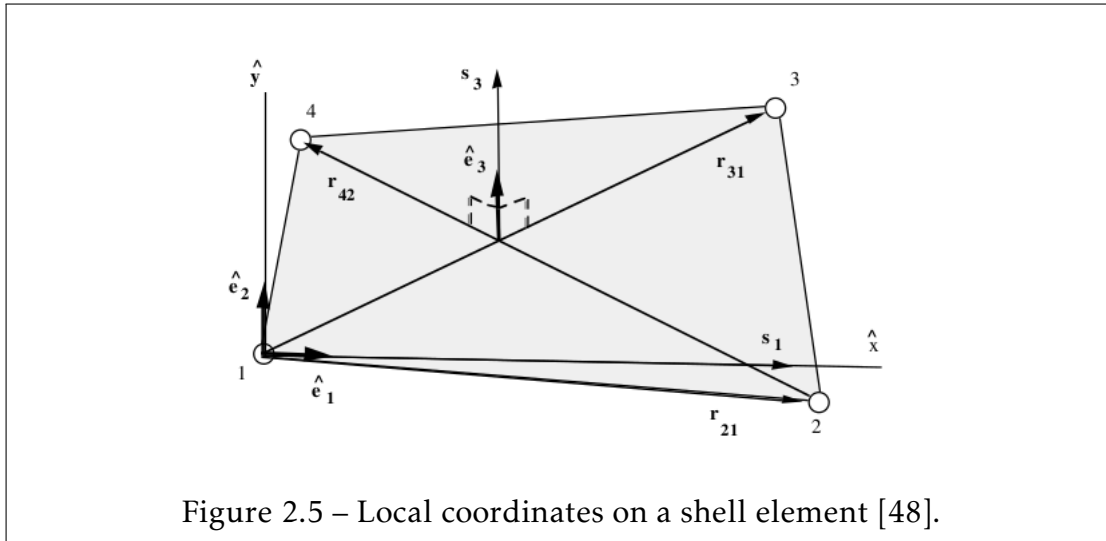
$$\hat{e}_{3i} = \frac{s_{3i}}{\| \overrightarrow{s_3} \|}, \tag{2.55}$$

where the superscript $(\hat{\cdot})$ denotes the variables in the local coordinate system.

- Establish a local x axis $\hat{x}$ approximately along the edge between n_1 and n_2. We calculate a unit vector $\hat{e}_1$ parallel to the vector r_{21} as

$$s_{1i} = r_{21i} - (r_{21i}.\hat{e}_{3i})\,\hat{e}_{3i} \tag{2.56}$$

$$\hat{e}_{1i} = \frac{s_{1i}}{\| s_{1i} \|} \tag{2.57}$$

- The remaining vector is deduced from the two previous ones taking the cross product such that:

$$\hat{e}_{2i} = (\overrightarrow{\hat{e}_3} \times \overrightarrow{\hat{e}_1})_i \tag{2.58}$$

The orthogonal transformation matrix R between the global and local coordinate systems is defined by:

$$R = \begin{bmatrix} \hat{e}_{11} & \hat{e}_{12} & \hat{e}_{13} \\ \hat{e}_{21} & \hat{e}_{22} & \hat{e}_{23} \\ \hat{e}_{31} & \hat{e}_{32} & \hat{e}_{33} \end{bmatrix} \tag{2.59}$$

Let $A = (A_1, A_2, A_3)^T$ be a vector defined in the global coordinate system and $\hat{A} = (\hat{A}_1, \hat{A}_2, \hat{A}_3)^T$ its definition in the local co-rotational coordinate system. The transformation from A to $\hat{A}$ is given by:

$$\underbrace{\begin{bmatrix} \hat{A}_1 \\ \hat{A}_2 \\ \hat{A}_3 \end{bmatrix}}_{\hat{A}} = \underbrace{\begin{bmatrix} \hat{e}_{11} & \hat{e}_{12} & \hat{e}_{13} \\ \hat{e}_{21} & \hat{e}_{22} & \hat{e}_{23} \\ \hat{e}_{31} & \hat{e}_{32} & \hat{e}_{33} \end{bmatrix}}_{R} \cdot \underbrace{\begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}}_{A} \tag{2.60}$$

The inverse transformation from $\hat{A}$ to A is given by:

$$A = R^T . \hat{A} \tag{2.61}$$

Velocity-strain and displacement:

The velocity of any point in the BLT shell is defined by:

$$v_i = v_i^m - (\overrightarrow{\hat{z}} . \overrightarrow{\hat{e}_3} \times \overrightarrow{\theta})_i, \tag{2.62}$$

where v_i^m, $\hat{z}_i$ and θ_i are the velocity of the mid surface, the distance along the fiber direction (thickness) of the element and the angular velocity vector, respectively. The corresponding co-rotational velocity-strain components are given by:

$$\hat{\dot{\varepsilon}}_{ij} = \frac{1}{2}\left(\frac{\partial \hat{v}_j}{\partial \hat{x}_i} + \frac{\partial \hat{v}_i}{\partial \hat{x}_j}\right) \tag{2.63}$$

Substituting Eq.2.63 into Eq.2.62, we have:

$$\hat{\dot{\varepsilon}}_{11} = \frac{\partial \hat{v}_1^m}{\partial \hat{x}_1} + \hat{z}\frac{\partial \hat{\theta}_2}{\partial \hat{x}_1}, \tag{2.64}$$

$$\hat{\dot{\varepsilon}}_{22} = \frac{\partial \hat{v}_2^m}{\partial \hat{x}_2} - \hat{z}\frac{\partial \hat{\theta}_1}{\partial \hat{x}_2}, \tag{2.65}$$

$$2\hat{\dot{\varepsilon}}_{12} = \frac{\partial \hat{v}_1^m}{\partial \hat{x}_2} + \frac{\partial \hat{v}_2^m}{\partial \hat{x}_1} + \hat{z}\left(\frac{\partial \hat{\theta}_2}{\partial \hat{x}_2} - \frac{\partial \hat{\theta}_1}{\partial \hat{x}_1}\right), \tag{2.66}$$

$$2\hat{\dot{\varepsilon}}_{23} = \frac{\partial \hat{v}_3^m}{\partial \hat{x}_2} - \hat{\theta}_1, \tag{2.67}$$

$$2\hat{\dot{\varepsilon}}_{31} = \frac{\partial \hat{v}_3^m}{\partial \hat{x}_1} + \hat{\theta}_2, \tag{2.68}$$

where $\hat{v}^m$, $\hat{\theta}^m$ and $\hat{x}^m$ are evaluated in term of the shape functions $N_I(\xi, \eta)$ defined in the parent space:

$$\hat{v}^m = N_I(\xi, \eta)v_I \tag{2.69}$$

$$\hat{\theta}^m = N_I(\xi, \eta)\theta_I \tag{2.70}$$

$$\hat{x}^m = N_I(\xi, \eta)x_I \tag{2.71}$$

The shape function $N_I(\xi, \eta)$ ($I = 1, 2, 3, 4$) are defined by:

$$N_1 = \frac{1}{4}(1 - \xi)(1 - \eta) \tag{2.72}$$

$$N_2 = \frac{1}{4}(1 + \xi)(1 - \eta) \tag{2.73}$$

$$N_3 = \frac{1}{4}(1 + \xi)(1 + \eta) \tag{2.74}$$

$$N_4 = \frac{1}{4}(1 - \xi)(1 + \eta) \tag{2.75}$$

Using a reduced one-point integration, the co-rotational velocity-strain components and the shape functions are evaluated at the center of the element ($\xi = 0, \eta = 0$). Eq.2.64, Eq.2.65, Eq.2.66, Eq.2.67 and Eq.2.68 become:

$$\hat{\dot{\varepsilon}}_{11} = B_{1I}\hat{v}^m_{1I} + \hat{z}B_{1I}\hat{\theta}_{2I} \tag{2.76}$$

$$\hat{\dot{\varepsilon}}_{22} = B_{2I}\hat{v}^m_{2I} - \hat{z}B_{2I}\hat{\theta}_{1I} \tag{2.77}$$

$$2\hat{\dot{\varepsilon}}_{12} = B_{1I}\hat{v}^m_{2I} + B_{2I}\hat{v}^m_{1I} + \hat{z}\left(B_{2I}\hat{\theta}^m_{2I} - B_{1I}\hat{\theta}^m_{1I}\right) \tag{2.78}$$

$$2\hat{\dot{\varepsilon}}_{23} = B_{1I}\hat{v}^m_{3I} + N_I\hat{\theta}_{2I} \tag{2.79}$$

$$2\hat{\dot{\varepsilon}}_{31} = B_{2I}\hat{v}^m_{3I} - N_I\hat{\theta}_{1I} \tag{2.80}$$

Where $B_{kj} = \frac{\partial N_I}{\partial \hat{x}_k}, k = 1,2$ are the shape functions derivative evaluated at the center of the element ($\xi = 0, \eta = 0$).

The co-rotational stress rates $\hat{\dot{\sigma}}^{n+1/2}_{ij}$ are computed using the appropriate constitutive equation and the computed velocity-strains $\hat{\dot{\varepsilon}}^{n+1/2}_{ij}$ at time $t^{n+1/2}$. The co-rotational stresses $\hat{\sigma}^{n+1}_{ij}$ at time t^{n+1} is obtained through a time integration:

$$\hat{\sigma}^{n+1}_{ij} = \hat{\sigma}^n_{ij} + \Delta t.\hat{\dot{\sigma}}^{n+1/2}_{ij} \tag{2.81}$$

Stress resultant and Internal nodal forces evaluation:

The resultant force and moment vectors are defined in the co-rotational local coordinate system by:

$$\hat{f}^R_{ij} = \int \hat{\sigma}_{ij}d\hat{z}, \tag{2.82}$$

$$\hat{m}^R_{ij} = -\int \hat{z}\hat{\sigma}_{ij}d\hat{z}, \tag{2.83}$$

where $\hat{\sigma}_{ij}$ ($i,j = 1,2,3$) are the stress tensor components evaluated in the co-rotational local coordinates system at each integration point through the thickness. The integrals are computed using a numerical quadrature rule (Gauss or Trapezoidal).

Through the use of the definition of the velocity-strain, and by recalling that

there are five integration points through the thickness, we can deduce the resultant forces and moments integrating the stress through the thickness of the shell.

Invoking the principle of virtual power and performing a one-point Gauss quadrature, we have:

$$\hat{F}_I^{int} = Area \begin{bmatrix} \left(B_{1I}\hat{f}_{11}^R + B_{2I}\hat{f}_{12}^R\right) \\ \\ \left(B_{2I}\hat{f}_{22}^R + B_{1I}\hat{f}_{12}^R\right) \\ \kappa\left(B_{1I}\hat{f}_{13}^R + B_{2I}\hat{f}_{23}^R\right) \end{bmatrix}, \tag{2.84}$$

$$\hat{m}_I^{int} = Area \begin{bmatrix} \left(B_{2I}\hat{m}_{22}^R + B_{1I}\hat{m}_{12}^R - \frac{\kappa}{4}\hat{f}_{23}^R\right) \\ \\ -\left(B_{1I}\hat{m}_{11}^R + B_{2I}\hat{m}_{12}^R - \frac{\kappa}{4}\hat{f}_{13}^R\right) \\ 0 \end{bmatrix}, \tag{2.85}$$

where $Area$ and κ are the area of the element and the shear factor correction coefficient, respectively.

Finally, the vector forces and moments are obtained in the global coordinate system using Eq.2.61:

$$F_I^{int} = R^T.\hat{F}_I^{int} \tag{2.86}$$

$$m_I^{int} = R^T.\hat{m}_I^{int} \tag{2.87}$$

Hourglass nodal forces evaluation:

The hourglass nodal forces implemented in LS-DYNA follows the formulation developed by Belytschko et al. [13] where the hourglass modes were generalized. In the original papers, the different steps in the computation of the nodal hourglass vector F_I^{hrg} are described as follows:

- Definition of the only hourglass vector Γ:

$$\Gamma = \begin{bmatrix} +1 & -1 & +1 & -1 \end{bmatrix}^T \tag{2.88}$$

- Computation of the element matrix B, given explicitly in the co-rotational coordinate system:

$$B = \frac{1}{2.Area} \begin{bmatrix} \hat{y}_2 - \hat{y}_4 & \hat{x}_4 - \hat{x}_2 \\[2ex] \hat{y}_3 - \hat{y}_1 & \hat{x}_1 - \hat{x}_3 \\[2ex] \hat{y}_4 - \hat{y}_2 & \hat{x}_2 - \hat{x}_4 \\[2ex] \hat{y}_1 - \hat{y}_3 & \hat{x}_3 - \hat{x}_1 \end{bmatrix} \tag{2.89}$$

- Computation of the element vector γ:

$$\gamma = \begin{bmatrix} \Gamma_1 - (\sum_{k=1}^{4} \Gamma_k.\hat{x}_k)B_{11} - (\sum_{k=1}^{4} \Gamma_k.\hat{y}_k)B_{12} \\[2ex] \Gamma_2 - (\sum_{k=1}^{4} \Gamma_k.\hat{x}_k)B_{21} - (\sum_{k=1}^{4} \Gamma_k.\hat{y}_k)B_{22} \\[2ex] \Gamma_3 - (\sum_{k=1}^{4} \Gamma_k.\hat{x}_k)B_{31} - (\sum_{k=1}^{4} \Gamma_k.\hat{y}_k)B_{32} \\[2ex] \Gamma_4 - (\sum_{k=1}^{4} \Gamma_k.\hat{x}_k)B_{41} - (\sum_{k=1}^{4} \Gamma_k.\hat{y}_k)B_{42} \end{bmatrix} \tag{2.90}$$

- Computation of the hourglass strain rates $\dot{q}$:

$$
\begin{bmatrix}
\dot{q}_1^m & = & \sum_{k=1}^{4} \gamma_k . \hat{\theta}_{1k} \\[2ex]
\dot{q}_2^m & = & \sum_{k=1}^{4} \gamma_k . \hat{\theta}_{2k} \\[2ex]
\dot{q}_1^b & = & \sum_{k=1}^{4} \gamma_k . \hat{v}_{1k} \\[2ex]
\dot{q}_2^b & = & \sum_{k=1}^{4} \gamma_k . \hat{v}_{2k} \\[2ex]
\dot{q}_3^b & = & \sum_{k=1}^{4} \gamma_k . \hat{v}_{3k}
\end{bmatrix}, \tag{2.91}
$$

where the superscripts b and m denote the hourglass modes associated with bending and membrane forces, respectively.

- Computation of the general hourglass stress rates $\dot{Q}$:

$$
\begin{bmatrix}
\dot{Q}_1^m & = & C_1 \dot{q}_1^m \\[2ex]
\dot{Q}_2^m & = & C_1 \dot{q}_2^m \\[2ex]
\dot{Q}_1^b & = & C_2 \dot{q}_1^b \\[2ex]
\dot{Q}_2^b & = & C_3 \dot{q}_2^b \\[2ex]
\dot{Q}_3^b & = & C_3 \dot{q}_3^b
\end{bmatrix}, \tag{2.92}
$$

where:

$$
\begin{cases}
C_1 & = & \frac{1}{8} r_m \left(E.h.Area \right) \sum_{i=1}^{2} \sum_{k=1}^{4} B_{ik} \\[2ex]
C_2 & = & \frac{1}{192} r_\theta \left(E.h^3.Area \right) \sum_{i=1}^{2} \sum_{k=1}^{4} B_{ik} \\[2ex]
C_3 & = & \frac{1}{12} r_w \left(\kappa.G.h^3 \right) \sum_{i=1}^{2} \sum_{k=1}^{4} B_{ik}
\end{cases}, \tag{2.93}
$$

where r_m, r_θ and r_w are user-defined constants with values between 0.01

and 0.05, respectively. h, E and G represent the element thickness, Young modulus and shear modulus, respectively.

- Computation of the general hourglass stresses rates Q integrating in time the hourglass stress rates $\dot{Q}$.

- Computation of the nodal hourglass forces $\hat{F}_I^{hrg}$ and moments $\hat{m}_I^{hrg}$ vectors ($I = 1, 2, 3, 4$) in the co-rotational coordinate system:

$$
\hat{F}_I^{hrg} = \begin{bmatrix} \gamma_I Q_1^m \\ \\ \gamma_I Q_2^m \\ \\ \gamma_I Q_3^b \end{bmatrix} \tag{2.94}
$$

$$
\hat{m}_I^{hrg} = \begin{bmatrix} \gamma_I Q_1^b \\ \\ \gamma_I Q_2^b \\ \\ 0 \end{bmatrix} \tag{2.95}
$$

- Computation of the nodal hourglass forces F_I^{hrg} and moments m_I^{hrg} vectors ($I = 1, 2, 3, 4$) in the global coordinate system:

$$
F_I^{hrg} = R^T . \hat{F}_I^{hrg} \tag{2.96}
$$

$$
m_I^{hrg} = R^T . \hat{m}_I^{hrg} \tag{2.97}
$$

2.2.3   Shock Viscosity

The non-linear hyperbolic characteristic of the conservation laws may naturally introduce a ***discontinuous strong solution***, even for smooth initial conditions (see [146, 11]). This difficulty can be overcome by treating the problem from a different angle by seeking a ***weak solution*** (based the weak formulations Eq.2.23 and Eq.2.26) instead of a strong solution, which would lead to the ***jump***

Figure 2.6 – Riemann Problem: Propagation of the discontinuous variables at shock speed U.
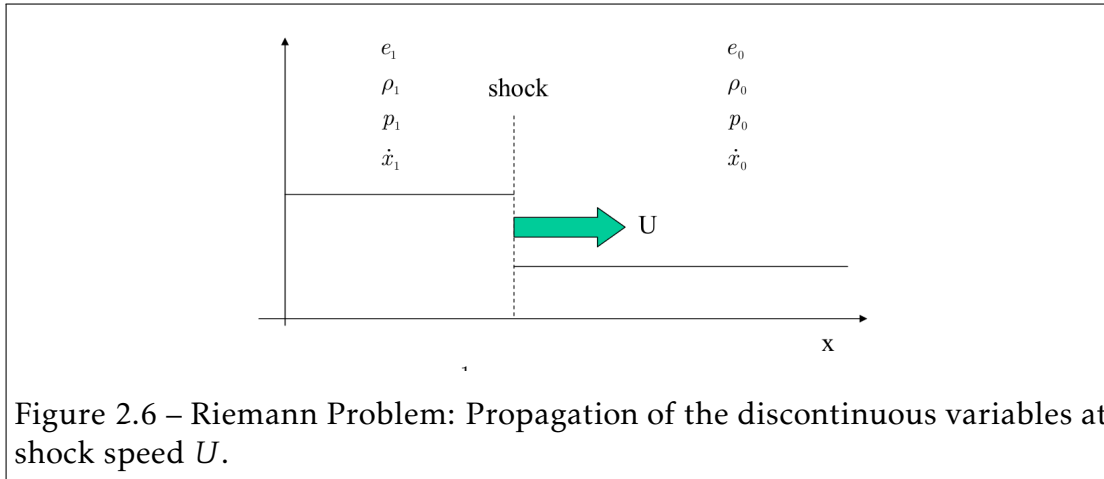
Rankine-Hugoniot relations at the discontinuous interface (see Fig.2.6). The Rankine-Hugoniot jump relations are in one-dimension given by:

$$(\rho_1 - \rho_0)\, U = \rho_1(v_1 - v_0) \qquad \text{Conservation of mass}$$

$$p_1 - p_0 = \rho_0\, U\,(v_1 - v_0) \qquad \text{Conservation of momentum} \qquad (2.98)$$

$$e_1^{int} - e_0^{int} = \frac{p_1 - p_0}{2} \cdot \frac{\rho_1 - \rho_0}{\rho_1 \rho_0} \qquad \text{Conservation of energy}$$

Where U is the shock speed. Combining the three equations, we have:

$$e_0^{int} - e_1^{int} = \frac{1}{2}(p_0 + p_1)(v_0 - v_1) \qquad (2.99)$$

The strong solution must be continuous and derivable, whereas the weak solution requires must only be continuous. As mentioned in [134], the drawback of this approach is the loss of the unity of the solution [76]. An additional entropy condition must be satisfied by adding viscous terms to the system of equations to guarantee that the computed solution is the physical one [50].

The method proposed by Von Neumann and Richtmeyer [102] allows for the automatic processing of shock waves without solving the non-linear Rankine-Hugoniot relations through the use of a costly iterative root finding method.

At the shock front, the thermodynamic irreversibility results in an in-supply of entropy. The idea of the Von Neumann and Richtmeyer bulk viscosity is to introduce into the equations, a dissipative viscous term that represents this irreversibility. An additional pressure term q is added to the physical pressure p in the shock layer. The original "bulk" pressure proposed by Von Neumann and Richtmeyer [102] is defined by the following quadratic term:

$$q = \begin{cases} \rho l_c^2 C_1 tr(\dot{\varepsilon})^2 & \text{if} \quad tr(\dot{\varepsilon}) < 0 \quad \text{(compression)} \\ 0 & \text{if} \quad tr(\dot{\varepsilon}) \geq 0 \qquad \text{(tension)} \end{cases} \qquad (2.100)$$

Where C_1 and l_c are a user-defined value (taken to be 1.5 for most of the problems) and a characteristic length ($l_c = \sqrt{area}$ in 2D and $l_c = \sqrt[3]{volume}$ in 3D), respectively. The quadratic pressure term q is added to the linear momentum and energy conservation equations.

The added quadratic pressure term smooths the shock among a few elements (typically three to six elements for $C_1 = 1.5$). The smoothed solution presents no discontinuity and prevents numerical oscillations (referred to as ***Gibb's phenomena*** in an attempt to approximate a discontinuous function through a series of continuous functions, see Fig.2.7) at the discontinuous interface.

Even if the discontinuity has been smoothed, Von Neumann and Richtmeyer [102] showed that the Rankine-Hugoniot relations (Eq.2.98 and Eq.2.99) remain valid at a large scale. Additionally, in the removal of the numerical oscillations, the physical correct solution is obtained by adding the necessary entropy as discussed in [50].

A linear term is added to the bulk pressure in Eq.2.100 by [72] to remove the spurious oscillations behind the shock. The final form of the bulk pressure is given by:

$$q = \begin{cases} \rho l_c \left[C_1 l_c tr(\dot{\varepsilon})^2 - C_2 c tr(\dot{\varepsilon}) \right] & \text{if} \quad tr(\dot{\varepsilon}) < 0 \quad \text{(compression)} \\ 0 & \text{if} \quad tr(\dot{\varepsilon}) \geq 0 \qquad \text{(tension)} \end{cases}, \qquad (2.101)$$

where C_2 and c are a use-defined value (taken to be 0.06 for most of the prob-
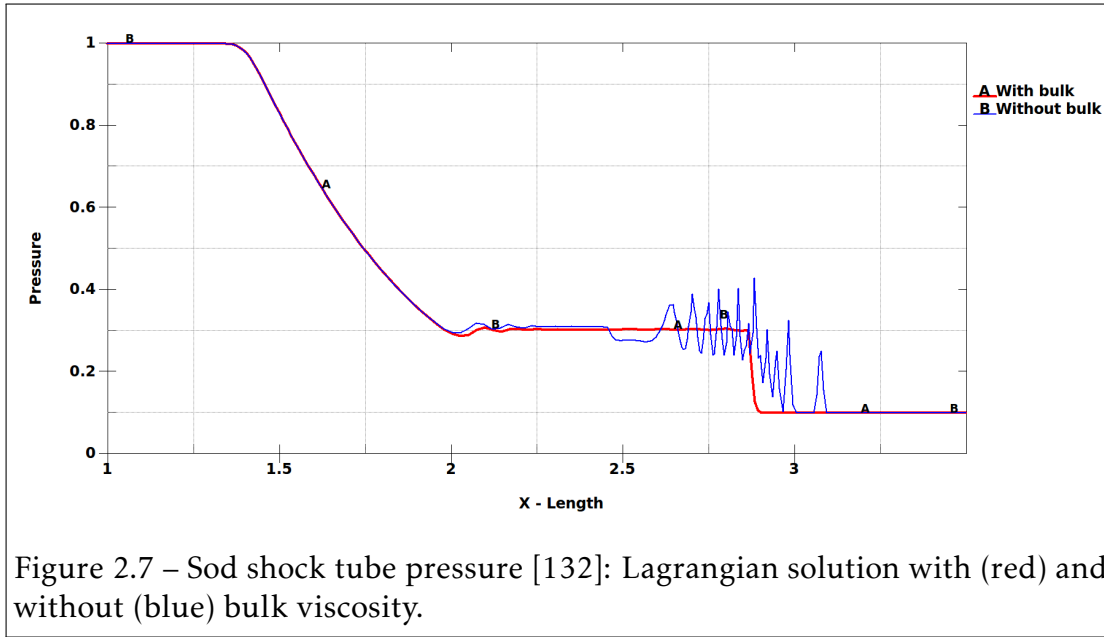
Figure 2.7 – Sod shock tube pressure [132]: Lagrangian solution with (red) and without (blue) bulk viscosity.

lems) and the element sound speed, respectively.

A generalized bulk viscosity tensor Q_{ij} can be developed in term of the strain rates tensor $\dot{\varepsilon}_{ij}$:

$$Q_{ij} = \begin{cases} \rho l_c \left[C_1 l_c \dot{\varepsilon}_{ii}^2 - C_2 c \dot{\varepsilon}_{ii} \right] \delta_{ij} + c \rho l_c \dot{\varepsilon}_{ij} & \text{if} \quad tr(\dot{\varepsilon}) < 0 \quad \text{(compression)} \\ 0 & \text{if} \quad tr(\dot{\varepsilon}) \geq 0 \qquad \text{(tension)} \end{cases} \quad (2.102)$$

2.2.4   Energy computation

Pressure is related to the internal energy through the EOS (Eq.1.99). The internal energy is evaluated using one-point integration such as stresses evaluation. Thus, the energy conservation equation is simplified to the following form:

$$\rho \dot{e}^{int} = D_{ij}(\sigma_{ij} + Q_{ij}) \quad (2.103)$$

By splitting the normal and deviatoric parts, we have:

$$\rho \dot{e}^{int} = \left(D_{ij}^d + \frac{1}{3} D_{kk} \delta_{ij} \right) \left(\sigma_{ij}^d - p \delta_{ij} + Q_{ij}^d + \frac{1}{3} Q_{kk} \delta_{ij} \right) \quad (2.104)$$

Considering the symmetry of the bulk viscosity tensor Q_{ij}, we finally obtain:

$$\rho \dot{e}^{int} = \left(\sigma_{ij}^d + Q_{ij}^d\right) D_{ij}^d + \left(\frac{1}{3} Q_{kk} - p\right) V, \qquad (2.105)$$

where:

- V is the elementary volume defined by: $V = \int_{\Omega_e} d\Omega_e$.

In applying a central finite difference scheme to Eq.2.105 and considering that the volumic strain rate $tr(\dot{\varepsilon}) = \Delta V/(V\Delta t)$, we obtain the discretized equation for the total specific energy $E = \rho e^{int} V$:

$$
\begin{aligned}
E^{n+1} \;\; &= \;\; E^n + \Delta t\left[\left(\sigma_{ij}^{d\,n+1/2} + Q_{ij}^{d\,n+1/2}\right) D_{ij}^{d\,n+1/2} V^{n+1/2}\right] \\
&+ \;\; \left(\frac{1}{3} Q_{kk}{}^{n+1/2} - p^{n+1/2}\right)\Delta V^{n+1/2}
\end{aligned}
\qquad (2.106)
$$

where the quantities at time $t = t^{n+1/2}$ are given by:

$$
\begin{cases}
D_{ij}^{d\,n+1/2} \;\; = \;\; \frac{1}{2}\left(D_{ij}^{d\,n} + D_{ij}^{d\,n+1}\right) \\[2ex]
\sigma_{ij}^{d\,n+1/2} \;\; = \;\; \frac{1}{2}\left(\sigma_{ij}^{d\,n} + \sigma_{ij}^{d\,n+1}\right) \\[2ex]
Q_{ij}^{d\,n+1/2} \;\; = \;\; \frac{1}{2}\left(Q_{ij}^{d\,n} + Q_{ij}^{d\,n+1}\right) \\[2ex]
p^{n+1/2} \;\; = \;\; \frac{1}{2}\left(p^n + p^{n+1}\right)
\end{cases}
\qquad (2.107)
$$

The equation Eq.2.106 is implicit, since the pressure is a function of the energy and the density (Eq.1.99). It can be rewritten:

$$
\begin{aligned}
E^{n+1} \;\; &= \;\; \overline{E} - \frac{1}{2} p^{n+1}(E^{n+1}, \rho^{n+1})\Delta V^{n+1/2} \\[2ex]
\overline{E} \;\; &= \;\; E^n + \Delta t\left[\left(\sigma_{ij}^{d\,n+1/2} + Q_{ij}^{d\,n+1/2}\right) D_{ij}^{d\,n+1/2} V^{n+1/2}\right] \\
&+ \;\; \left(\frac{1}{3} Q_{kk}{}^{n+1/2} - p^n\right)\Delta V^{n+1/2}
\end{aligned}
\qquad (2.108)
$$

where all the known variables are regrouped in the variable $\overline{E}$. Through the use of Eq.1.99 to evaluate the pressure, we finally obtain an explicit expression of

the pressure at time $t = t^{n+1}$:

$$p^{n+1} = \frac{A(\rho^{n+1}) + B(\rho^{n+1})\overline{E}}{1 + \frac{1}{2}B(\rho^{n+1})\Delta V^{n+1/2}}$$

(2.109)

$$E^{n+1} = \overline{E} - \frac{1}{2}p^{n+1}\Delta V^{n+1/2}$$

If the pressure is a non-linear function of the internal energy in the EOS (Eq.1.99) then the energy is computed by using an iterative root finder method.

2.2.5 Summary of the Lagrangian phase

At time $t = t^n$, the stresses, the pressure, the internal forces, the external forces, the hourglass forces, the bulk viscosity are know.

1. Computation of the nodal accelerations at time $t = t^n$.

2. Computation of the nodal velocities at time $t = t^{n+1/2}$.

3. Computation of the nodal displacements at time $t = t^{n+1}$ and update of the lagrangian mesh.

4. Computation of the bulk viscosity and hourglass forces at time $t = t^{n+1}$.

5. Computation of the internal energy at time $t = t^{n+1}$.

6. Computation of the pressure at time $t = t^{n+1}$.

7. Computation of the new time-step with respect to the CFL criteria.

2.3   Eulerian FV Formulation: Advection Phase

For the sake of clarity, we consider the pure Eulerian phase in which the ALE velocity is zero. The Jacobian in Eq.2.6 becomes $J_\psi = 1$. In this particular case, the variables computed in the Lagrangian phase are projected onto the fixed Eulerian grid. For further details on the ALE smoothing and more general arbitrary cases, see [48].

In the advection phase, the physically conserved variables ρ, ρv_i and ρe^{int} are transported from the position obtained at the end of the Lagrangian phase to the ALE grid by solving Eq.2.6 using a finite volume **Godunov method**. Among the different schemes present in LS-DYNA, this thesis uses the following two methods:

- The first order **Donor Cell** scheme [43]: fast, monotone, conservative and diffusive.

- The second order **Van Leer MUSCL** (Monotonic Upwind Schemes for Conservation Laws) scheme [79]: slow (compared to the Donor Cell scheme), monotone, conservative and dispersive.

As discussed in [134], the Lagragian phase, at its best, is second order accurate. Thus, schemes higher than the second order cannot improved the overall solution. Fig.2.8 shows the two schemes effects on the transport of a constant function.

Once again, through the use of the operator splitting technique, the multidimensional advection can be solved by splitting the spatial derivatives into a set of one dimension transport equations that are solved sequentially (see [16]). However, a particular attention should be paid to the mesh structure[134].

Figure 2.8 – Comparaison of the first and second order Godunov schemes for one dimensional advection [105].
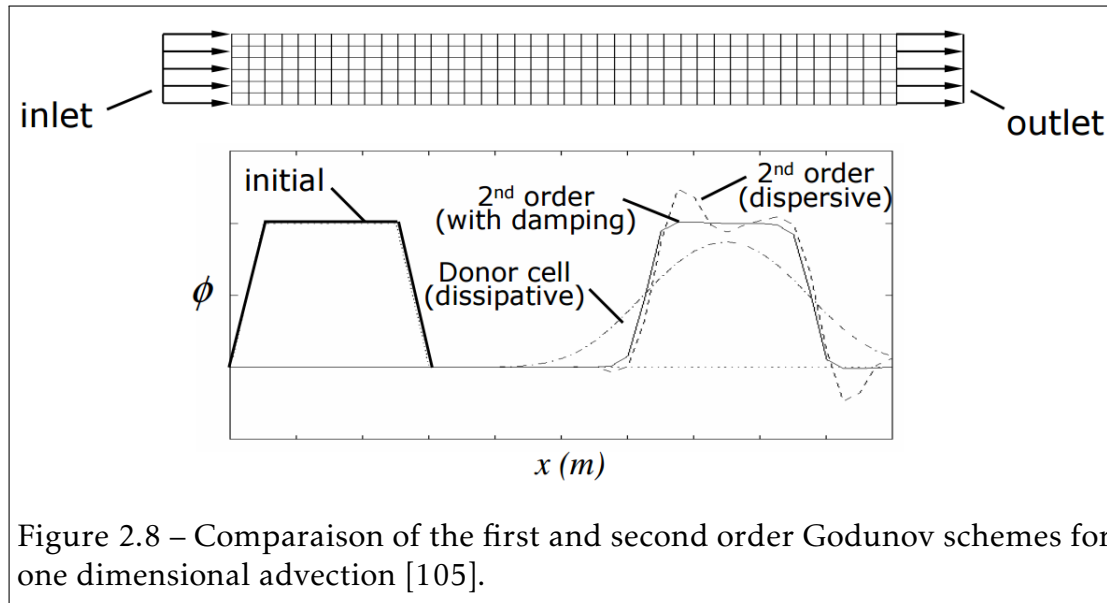
2.3.1 Godunov methods for one dimensional transport equation

In order to describe the two Godunov methods, let us consider the nonlinear hyperbolic transport equation:

$$\frac{\partial \phi}{\partial t} + a(x)\frac{\partial \phi}{\partial x} = 0, \tag{2.110}$$

where ϕ is the transported function and $a(x)$ is the transport velocity.

Let $x_j, (j = 1,...,N)$ and N be the coordinates of the node j and the total number of nodes, respectively. We denote by $\phi^n_{j-1/2}$ and $\phi^n_{j+1/2}$ the values of the function ϕ to the left and to the right of the node j, respectively.

$$\phi^n(x) = \phi^n_{j+1/2} \quad , \quad x \in]x_j, x_{j+1}[\tag{2.111}$$

The evaluation of the function ϕ in Eq.2.111 is suitable for the transport of centered values such as ρ and ρe^{int} (ρ and e^{int} are evaluated at the center of the

element during the lagrangian step using Eq.2.43) contrary to the nodal values such as the linear momentum ρv_i (v_i are evaluated at the nodal positions during the lagrangian step). The treatment of linear momentum advection presented in the next subsection is based on the following Godunov advection schemes.

"Donor Cell" (Upwind) scheme

By applying the backward finite difference scheme we obtain the donor cell method in its non-conservative form:

$$\phi_{j-1/2}^{n+1} = \phi_{j-1/2}^n + a_j \frac{\Delta t}{2\Delta x} \left\{ \begin{array}{ll} \phi_{j-1/2}^n - \phi_{j+1/2}^n & \text{if } a_j > 0 \\ \phi_{j+1/2}^n - \phi_{j-1/2}^n & \text{if } a_j < 0 \end{array} \right. , \qquad (2.112)$$

Eq.2.112 is expressed in its into conservative form by:

$$\phi_{j-1/2}^{n+1} = \phi_{j-1/2}^n + \frac{\Delta t}{\Delta x} \left(f_j^\phi - f_{j+1}^\phi \right), \qquad (2.113)$$

where the finite volume fluxes f_j^ϕ are defined by:

$$f_j^\phi = \frac{a_j}{2} \left(\phi_{j-1/2}^n + \phi_{j+1/2}^n \right) + \frac{|a_j|}{2} \left(\phi_{j-1/2}^n - \phi_{j+1/2}^n \right) \qquad (2.114)$$

Van Leer (MUSCL) scheme

In the donor cell scheme, we assumed that the distribution of the function ϕ is constant in each cell. The Van Leer scheme is obtained by replacing the constant distribution with a higher order interpolation function (see Fig.2.10).

Instead of defining a piecewise function, let us define a piecewise linear function. Thus we have:

Figure 2.9 – Description of the **Donor Cell** advection method [105].
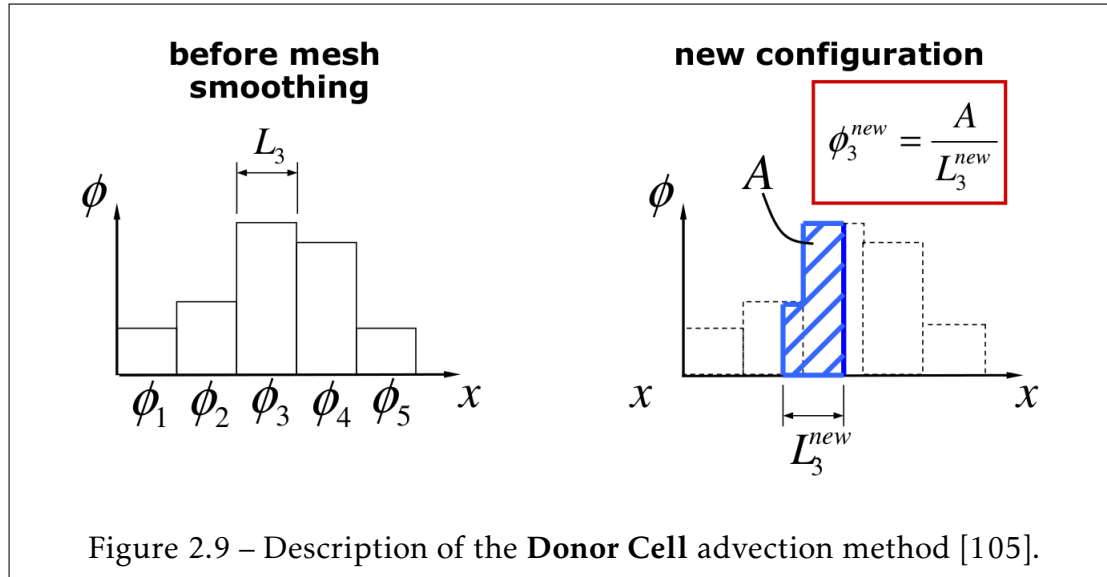
$$\phi_{j+1/2}^n(x) = S_{j+1/2}^n\left(x - x_{j+1/2}^n\right) + \phi_{j+1/2}^n, \tag{2.115}$$

where:

$$S_{j+1/2}^n = \frac{1}{2}\left(sign(s^L) + sign(s^R)\right) \times min\left(|s^L|, |s_{j+1/2}^n|, |s^R|\right) \tag{2.116}$$

$$s^L = \frac{\phi_{j+1/2}^n - \phi_{j-1/2}^n}{2\Delta x_{j+1/2}} \tag{2.117}$$

$$s^R = \frac{\phi_{j+3/2}^n - \phi_{j+1/2}^n}{2\Delta x_{j+1/2}} \tag{2.118}$$

And $s_{j+1/2}^n$ is a second order approximation of the slope.

To avoid having new extreme values at the next incrementation, and thus oscillation, we constraint $s_{j+1/2}^n$ such as:

$s^{min} < s_{j+1/2}^n < s^{max}$, where:

$s^{min} = min(\phi_{j-1}, \phi_j, \phi_{j+1})$, and

$$s^{max} = max(\phi_{j-1}, \phi_j, \phi_{j+1}).$$

That leads us to the definition of the fluxes:

$$f_j^\phi = \frac{a_j}{2}\left(\phi_j^- + \phi_j^+\right) + \frac{|a_j|}{2}\left(\phi_j^- - \phi_j^+\right), \tag{2.119}$$

where:

$$\phi_j^+ = S_{j+1/2}\left(x^c - x_{j+1/2}^n\right) + \phi_{j+1/2}^n \tag{2.120}$$

$$\phi_j^- = S_{j-1/2}\left(x^c - x_{j-1/2}^n\right) + \phi_{j-1/2}^n \tag{2.121}$$

and:

$$x^c = x_j^n + \frac{1}{2}\Delta t a_j \tag{2.122}$$

Figure 2.10 – Description of the **Van Leer advection method** [105].
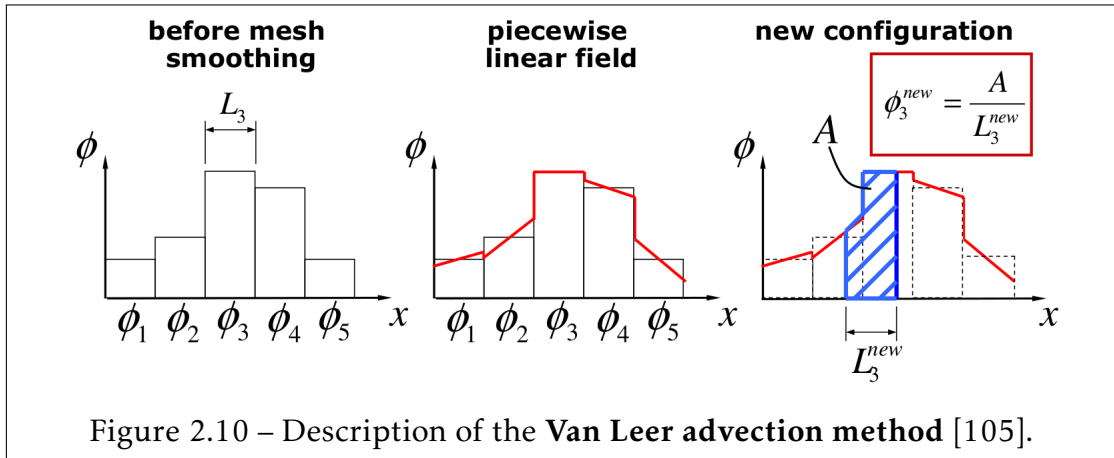
2.3.2 Advection for the linear momentum equation

In order to solve the linear momentum advection at the nodal positions, Amsdem and Hirt [8] proposed to create a dual finite volume mesh by shifting the original mesh by half of the cell size such that:

$$\phi_j^{dual} = \phi_{j+1/2}^{orig} \tag{2.123}$$

Applying to the linear momentum equation variables the evaluation of the transported quantity by the second order Van Leer method (Eq.2.115), one obtain the evaluation of the linear momentum at the nodal value j:

$$M_j^{dual} v_j^{dual} = M_j^{orig} v_j^{orig} + \frac{1}{2}\left((\rho v)_{j-1}^{orig} V_{j-1}^{orig} - (\rho v)_{j+1}^{orig} V_{j+1}^{orig}\right) \tag{2.124}$$

The different steps that lead to Eq.2.124 are given in detail in [134].

Summary of the Eulerian phase

> The conservative variables ρ and ρe^{int} has been computed at the center of the elements. The conservative variable ρv_i is computed at the nodes.
>
> 1. A new grid is generated, either by the Lagrangian mesh relaxation (ALE computation) or by going back to the fixed Eulerian grid.
>
> 2. The centered conservative variables are projected to the new grid using Eq.2.113 or Eq.2.115.
>
> 3. The nodal velocities are projected to the new grid using Eq.2.124.

2.3.3   ALE Multi-Material formulations

To model multiphase flows in fixed Eulerian or ALE grids, we consider multi-material elements such that the N_{j_k} different materials within a finite element Ω_k are mixed. The subscript j_k denotes the index j of the material contained in the mixed element k. In a multi-material we distinguish:

- The N_{j_k} individual materials within the mixed element Ω_k that are governed by their own constitutive equations $(p_{j_k}(e_{j_k}^{int}, \rho_{j_k}), \dot{\varepsilon}_{j_k}, \sigma_{j_k}, ...)$. To each individual material $j_k = 1, .., N_{j_k}$ is associated a volume fraction α_{j_k} such that:

$$0 \leq \alpha_{j_k} \leq 1$$

$$\sum_{j_k=1}^{N_{j_k}} \alpha_{j_k} = 1 \tag{2.125}$$

- The mixed element Ω_k is governed by a mixture theory [134] in which the pressure equilibrium between the N_{j_k} materials and the mean strain rates $\dot{\varepsilon}_{j_k}^m$ satisfy:

$$\dot{\varepsilon}_k^m = \sum_{j_k=1}^{N_{j_k}} \alpha_{j_k} \dot{\varepsilon}_{j_k} \tag{2.126}$$

Contrary to the phase change model described in chapter1, in which the density and pressure of the element are mixture quantities each material has its own density. The pressure of the different materials within the element are assumed to be in equilibrium. The equilibrium pressure is enforced by:

$$
\begin{aligned}
p_{i_k}(e_{i_k}^{int}, \rho_{i_k}) &= p_{j_k}(e_{j_k}^{int}, \rho_{j_k}) \quad for \quad i \neq j \\
p^{eq} &= \alpha_{j_k} p_{j_k}(e_{j_k}^{int}, \rho_{j_k})
\end{aligned} \tag{2.127}
$$

where p^{eq} denotes the equilibrium pressure.

The N_{j_k} volume fraction are obtained by solving the non-linear equations (Eq.2.127) with the additional constraint $\sum_{j_k} \mid \alpha_{j_k} \mid = 1$.

Benson [17] provided an expression of the change in the volume fraction $\Delta \alpha_{j_k}$

by using the non-linear Newton-Raphson algorithm:

$$\Delta\alpha_{j_k} = \frac{p^{eq} - p_{j_k}}{\frac{\partial p_{j_k}}{\partial\alpha_{j_k}}} \tag{2.128}$$

Considering that the sum $\sum_j \alpha_{j_k} + \Delta\alpha_{j_k} = 1$, the pressure equilibrium is obtained by combining equations Eq.2.127 and Eq.2.128.

2.3.4 Volume of Fluid (VOF) method for interface tracking

The simplest way to track the interface between the different materials is to consider the Lagrangian framework where the material is attached to mesh. The Lagrangian formulation, in which the mesh moves with the material, is mainly used to solve problems in solid mechanics. It can be used for fluids and FSI problems in which the deformation is small. The main limitation of the Lagrangian formulation is mesh distortion for large deformation and moving structures, since the fluid is solved on a moving domain as a result of the structure motion.

The VOF method consists of constructing the discontinuous material interface within the element using its volume and the volume of the surrounding neighbors. The technique implemented in LS-DYNA is the **Piecewise Linear Interface Construction** (PLIC) proposed by Youngs (see [162] and [161] for 2D and 3D problems, respectively). The PLIC method follows the early **Simple Line Interface Calculation** (SLIC) method proposed by Noh and Woodward [103].

The PLIC method approximate the interface by a linear plane(segement in 2D and planes in 3D). The parametric equation of the plane at the material interface in an element Ω_k is defined by:

$$\overrightarrow{x}.\overrightarrow{n} + C = 0, \tag{2.129}$$

where:

1. The normal $\overrightarrow{n}$ to the plane is computed by:

$$\overrightarrow{n} = \frac{\nabla_{\overrightarrow{x}}(\alpha_k)}{\|\nabla_{\overrightarrow{x}}(\alpha_k)\|} \tag{2.130}$$

2. $d = \overrightarrow{x}.\overrightarrow{n}$ is the distant function of the element computed based on the volume fractions. López and Hernández [85] explained the procedure in depth to compute the distance function. An open source package of FORTRAN routines is also provided for generalized convex polygons.

3. C is a constant.

The last term to define is the gradient of the volume fractions $\nabla_{\overrightarrow{x}}(\alpha_k)$. For the sake of clarity, the computation will be presented for the 2D case, as the computation can then easily extended to the 3D case. Let us consider the centered element (see Fig.2.11) defined by:

$$\Omega_{(i+1/2,j+1/2)} =]x(i,j),x(i,j+1)[\times]x(i,j+1),x(i+1,j+1)[\tag{2.131}$$

We denote by $\alpha_{(i+1/2,j+1/2)}$ the volume fraction of element $\Omega_{(i+1/2,j+1/2)}$ and by α_E, α_N, α_W, α_S, α_{NE}, α_{NW}, α_{SW} and α_{SE} the volume fractions of the eastern, northern, western, southern, northeastern, northwestern, southwestern and southeastern (E,N,W,S,NE,NW,SW and SE in Fig.2.11) neighbors of the centered element $\Omega_{(i+1/2,j+1/2)}$, respectively. According to Youngs [162], the gradient of the volume fraction is given by:

$$\frac{\partial \alpha_{(i+1/2,j+1/2)}}{\partial x_1} = \frac{\alpha_E^* - \alpha_W^*}{x_{1_E} - x_{1_W}}, \tag{2.132}$$

$$\frac{\partial \alpha_{(i+1/2,j+1/2)}}{\partial x_2} = \frac{\alpha_N^* - \alpha_S^*}{x_{2_N} - x_{2_S}}, \tag{2.133}$$

where $x_1(.)$ are the centroid x_1-coordinates of the eastern and western neighbors and $x_2(.)$ are the centroid x_2-coordinates of the northern and southern neighbors. The quantities $\alpha_{(.)}^*$ are defined by:

$$
\begin{cases}
\alpha_E^* & = & \frac{1}{4}[\alpha_{SE} + 2\alpha_E + \alpha_{NE}] \\[2ex]
\alpha_W^* & = & \frac{1}{4}[\alpha_{SW} + 2\alpha_W + \alpha_{NW}] \\[2ex]
\alpha_N^* & = & \frac{1}{4}[\alpha_{NW} + 2\alpha_N + \alpha_{NE}] \\[2ex]
\alpha_S^* & = & \frac{1}{4}[\alpha_{SW} + 2\alpha_S + \alpha_{SE}]
\end{cases}
\tag{2.134}
$$

Figure 2.11 – Representation of the finite volume $\Omega_{(i+1/2,j+1/2)}$ and its 8 neighbors.
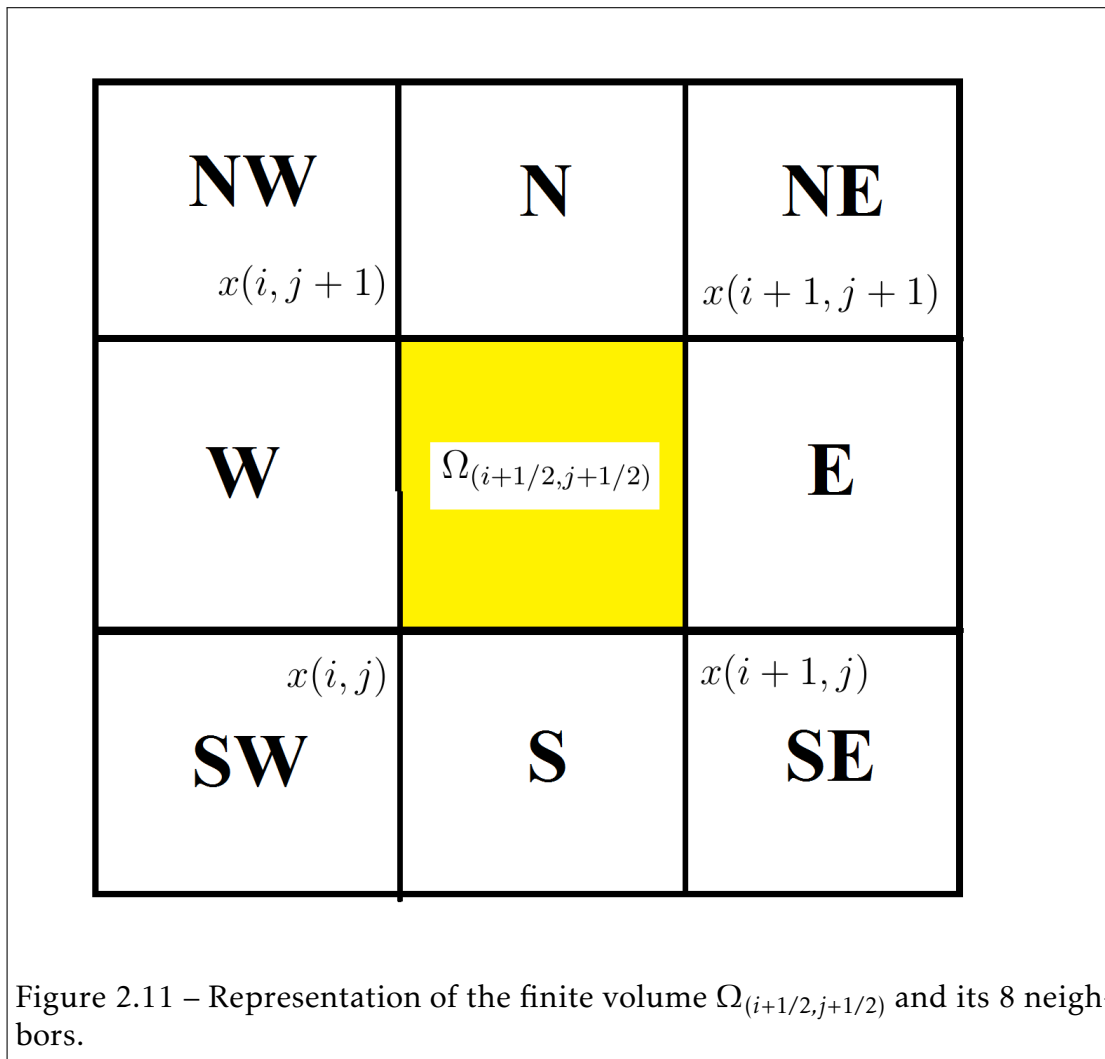
2.4   One-step ALE time integration summary

For the sake of clarity, we present in the schema bellow the different steps followed to solve the single-material Navier-Stokes equations expressed in the ALE formalism during one time-integration step.

The common Lagrangian phase steps for both ALE and Lagrangian formulations are displayed in blue boxes. Whereas, the specific advection phase steps are displayed in yellow boxes.
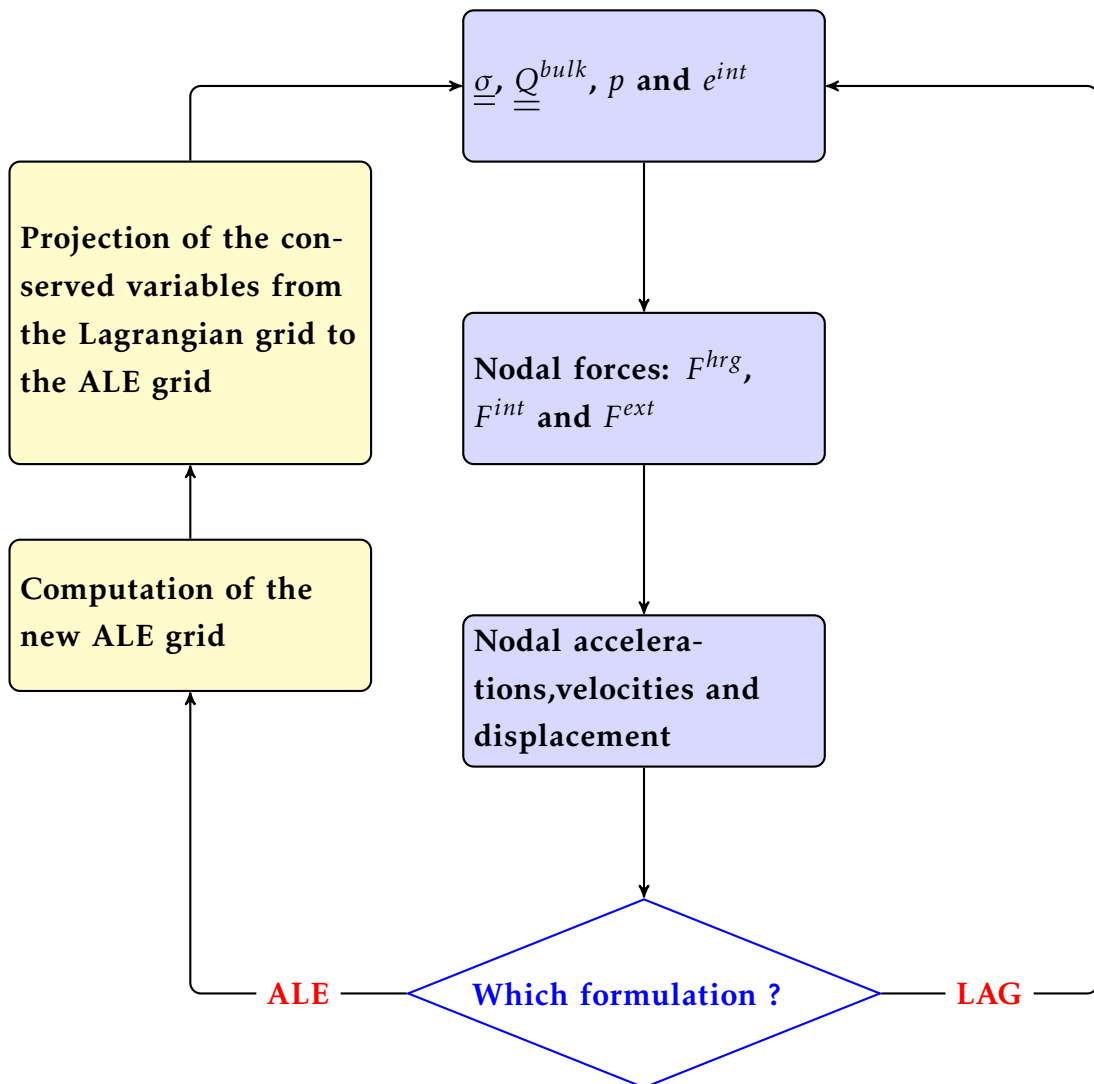
2.5 Fluid Structure Coupling methods

The coupling between the fluid and the structure acts at the interface between the fluid part and the solid part. Let Γ_s and Γ_f be the solid and fluid domain boundaries, respectively. The fluid structure interface Γ_{fsi} is defined by $\Gamma_{fsi} = \Gamma_s \cap \Gamma_f$. The coupling between the fluid and the structure acts at the interface Γ_{fsi} where fields are transferred between the fluid and the structure: the fluid pressure is transferred to the structure while the nodal velocities of the fluid are constrained to be equal to the nodal velocities of the structure. At both sides of the boundary Γ_{fsi}, we have:

- **Continuity of the velocity**: $\overrightarrow{v}_s = \overrightarrow{v}_f$.

- **Continuity of the stresses normal components**: $\sigma_s . \overrightarrow{n}_s = \sigma_f . \overrightarrow{n}_f$.

The easiest way to numerically simulate the fluid structure interaction is to consider fluid and structure meshes to be connected that share common nodes. The common nodes at the fluid structure interface are set to be Lagrangian ($v_{i_{ALE}} = 0$), which enforces the continuity of the velocities. The strategy to compute the stress components at the center of the element makes the numerical resolution of the continuity of the stress normal components easier. If the stress normal components were computed at the nodes of the mesh, projections schemes (similar to the advection of the velocity) would be required. This method is referred to as the **Lagrangian coupling**. The method is simple and robust, but limited to high mesh distortion problems. However, an ALE smoothing (re-meshing) of the grid can be used to overcome these problems (see [48]. There are two main strategies to solve FSI in LS-DYNA:

- **Constraint-based method**: Conserves momentum, stable but does not conserve energy.

- **Penalty-based method**: Conserves momentum and energy, but less stable than the constraint-based method. History variables are transported in the direction of the flow which makes it delicate to prevent leakage through the FSI interface, especially for high pressure and tangential flows to the structure [52, 105].

Brief descriptions of the two methods are presented in the next subsections. More details are provided in [10, 120, 134, 48].

2.5.1   Constraint-based coupling method

In constraint-based method, a coupling method based on contact algorithm is used. The coupling method was initially developed for fluid structure coupling, in which the fluid was modeled using an ALE or Eulerian formulation and the structure was modeled using a classical Lagrangian formulation. This coupling has been generalized to general material models with ALE formulation. Constrained coupling is based on tied contact code, where nodes on the slave interface are constrained to move with those on the master interface as described in [14]. From previous experience, tied contact has been successfully used in automotive and aerospace industries; it is stable and consistent, conserve momentum and satisfies the action reaction principle at the interface contact. Unlike contact algorithm in which a contact interface is defined though the interface mesh, in the coupling method, the structure node is embedded inside ALE of fluid element as described in Fig.2.12.
The coupling algorithm can be described though the following phases:

1. For each structure node, search for the fluid element that contains it. To ensure the search is efficient, the bucking sorting algorithm that is described in [48] can be used.

2. Distribute the nodal mass of the structure node m_i^s to the fluid element nodes m_i^f. The distribution is based on the parametric coordinates of the structure node inside the ALE element:

$$m_i^f = m_i^f + N_i.m_i^s,\qquad(2.135)$$

where N_i is linear shape function evaluation at node i.

3. Distribute structure nodal force F_i^s and momentum M_i^s from structure

nodes to fluid element nodes:

$$F_i^f = F_i^f + N_i.F_i^s$$

$$M_i^f = M_i^f + N_i.M_i^s$$

$$(2.136)$$

4. Based on new nodal forces and new momentum on the ALE element nodes, acceleration and velocity can be updated. After the finite elements are processed and assembled, the new nodal acceleration and velocity of the ALE nodes are given by:

$$a_i^f = F_i^f / m_i^f$$

$$v_i^f = M_i^f / m_i^f$$

$$(2.137)$$

5. Finally, the structure nodes are constrained to have the same acceleration and velocity as the ALE element. The structural accelerations and velocities are given by:

$$a_s = \sum_{i=1,s} N_i.a_i^f$$

$$v_s = \sum_{i=1,s} N_i.v_i^f$$

$$(2.138)$$
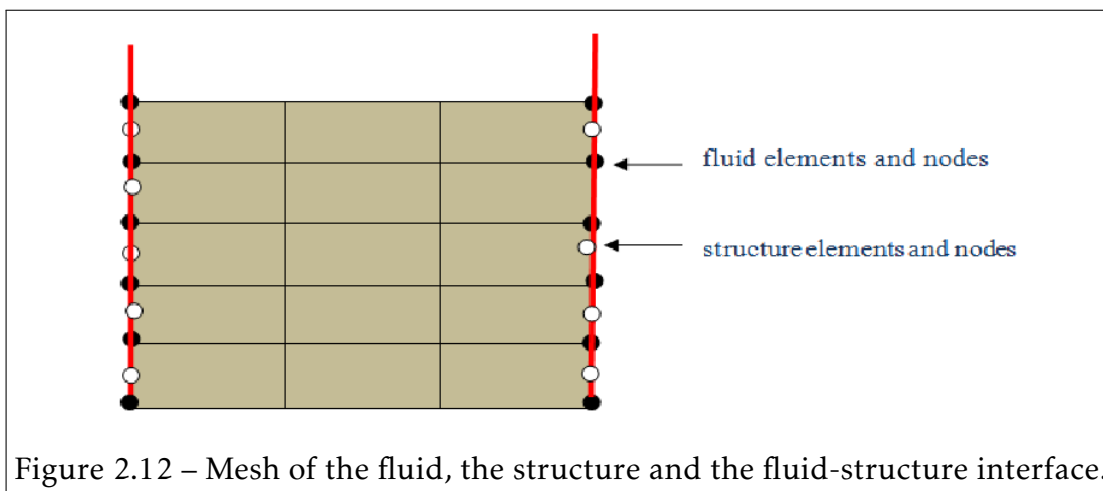
Figure 2.12 – Mesh of the fluid, the structure and the fluid-structure interface.

2.5.2   Euler-Lagrange penalty coupling method

The Euler-Lagrange coupling is a method that combines the ALE multi-material formulation described in the previous subsection and the classical master-slave penalty contact method (see [140, 110]) that is commonly used to solve contact problems in solid mechanics. The Lagrangian structure is always the slave part, and the Eulerian fluid is always the master part (see [134]).

In the Euler-Lagrange coupling the structure is embedded in an Eulerian fixed mesh. This Eulerian fixed mesh will contain the Lagrangian structure and the ALE fluid (single or multi-material) that flows through the mesh using the advection scheme to update velocity and history variables. Since the fluid flows through a fixed mesh, there is no mesh distortion that allows the limitations of constrained based method to be overcome (see [48, 140]) as a result of mesh great deformation. The Euler-Lagrange method also guarantees the conservation of momentum and energy, whereas constrained based method fails to conserve the total energy.

The coupling forces between the fluid and the structure are similar to contact forces and are defined by:

$$F = k.d, \tag{2.139}$$

where F is the force applied to both the master and slave nodes in opposite directions, k is the contact stiffness, and d is a depth penetration that is incrementally updated at each time-step. Indeed if at time t^0 we have $d^0 = 0$, then at time $t^{n+1} = t^n + \Delta t$ the penetration vector is updated by:

$$\overrightarrow{d}^{n+1} = \overrightarrow{d}^n + \left(\overrightarrow{v}_s - \overrightarrow{v}_f \right) \Delta t, \tag{2.140}$$

where $\overrightarrow{v}_s$ is the structure velocity at the slave node and $\overrightarrow{v}_f$ is the fluid velocity at the fluid node. A sketch of the coupling method is shown in Fig.2.13.

There exist many definitions for the stiffness k, and a good value of the stiffness should reduce the loss in energy conservation and prevent fluid leakage through the structure. In this study, the stiffness used is given in [166] and is defined as:

$$k = p_f \frac{KA^2}{V},$$

(2.141)

where K is the bulk modulus of the fluid element in the coupling that contains the slave node, V is the volume of the fluid element that contains the master node, A is the average area of the structure elements connected to the master node, and $0 \leq p_f \leq 1$ is a scalar factor introduced to avoid numerical instabilities.

Figure 2.13 – Euler-Lagrange FSI coupling method in one step.
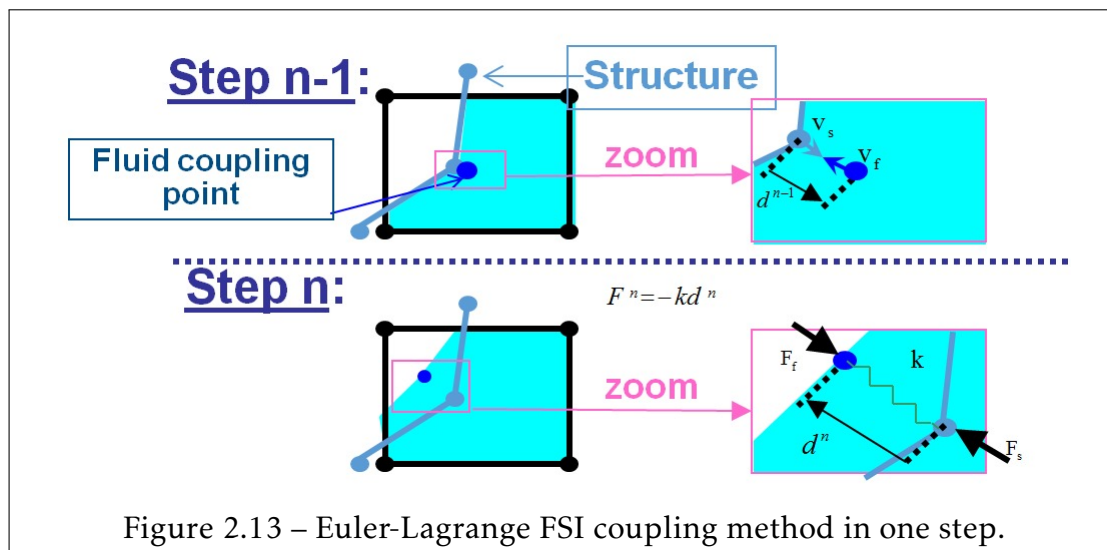
2.6   Numerical Applications

2.6.1   Validation and benchmark test cases

One dimension cavitation shock tube   The one-dimensional cavitation was proposed by [84] and is defined as follows:

- The computational domain is a $1m$ long tube $\Omega = [-0.5, 0.5]$.

- The tube is filled with water at initial temperature of $T = 293K$ and pressure of $10e + 8$. the initial liquid water saturation density, liquid water density and internal energy are computed using Eq.1.119, Eq.1.123 and Eq.1.124.

- The initial velocity is:

$$v(x,0) = \begin{cases} -100m/s & if \quad x \in= [-0.5, 0[\\[2em] 100m/s & if \quad x \in=]0, 0.5] \\[2em] 0m/s & if \quad x = 0 \end{cases}$$

- The problems is solved for 800 elements.

The two streams move in the opposite directions, which yields left and right rarefaction waves. An expansion area is generated at the center of the tube that forms a cavitation bubble. The results are plotted in red in Fig. 2.14, Fig.2.15, Fig.2.16 and Fig.2.17 and the reference results from [84] are plotted in blue. Although the same phase change model is used for both simulations, different numerical schemes are used to solve the governing equations.

Figure 2.14 – Pressure along the tube where the computed solution and the reference solution [84] are in blue points and continuous red line, respectively.

Figure 2.15 – Density along the tube where the computed solution and the reference solution [84] are in blue points and continuous red line, respectively.

Figure 2.16 – Void fraction along the tube where the computed solution and the reference solution [84] are in blue points and continuous red line, respectively.

Figure 2.17 – Velocity along the tube where the computed solution and the reference solution [84] are in blue points and continuous red line, respectively.

2.6.2   Simulation of Water Hammers in nuclear power plants

This work is part of a collaboration between EDF technical unit (EDF UTO), which was in charge of the validation of the development in LS-DYNA through benchmarks and the practical use in real life applications and the university of Lille. The different LS-DYNA simulations are compared to the experimental data.

In this collaborative work, the experience gained through the years on the topic of WHs by EDF engineers and researchers was very helpful and allowed us to face the problem from a numerical point of view.

WHs are hydraulic transient phenomena. They occur when the flow conditions of a fluid contained in a pipe are locally modified (pump start-up or stop, valve closure). A shock is generated and expressed by the discontinuity of the fluid variables (density, pressure, fluid velocity).

In nuclear power plants, such WHs occur in water supply pipes. Due to the high energy and the quantity of water in motion, there is a real threat to the nuclear safety. It can be violent and can cause damage to the structure (plasticity of pipes and even the rupture of brackets supporting pipes). Those fast dynamic phenomena are of the order of 10^{-3} seconds in time, and the actual sensors on nuclear power plants are not accurate enough to capture pressure waves. Through numerical simulation, one can attain a better understanding of these complex phenomena.

Under a certain configuration, the fluid may be put in tension and the local pressure may fall below the saturated vapor pressure, which gives arise to cavitation, small liquid-free zones ("bubbles" or "voids"). Such phenomena occur mostly near industrial apparels such as pumps, propellers, impellers and control valves. The rapid collapse of cavitation produces strong shock waves that may harm the interacting structure. Using the FSI capabilities of LS-DYNA, two test cases with experimental data will be investigated using the Cut-Off and HEM phase-change models presented in chapter1.

Theory of WHs A brief description of the theory and the main equations are presented in this paragraph to understand the phenomenon. For further details, explanations and numerical methods applied to solve WHs, see [131, 41, 40, 7, 20].

The valve slam WH, also called "the classic WH", has always been the heart of the research on WHs. Simpson is one of the early pioneers who worked actively on the experimentation of these phenomena. We simulate here the "Simpson's experience" [131] performed in 1986 on the classic WH, which provides a good validation test case. It contains a complex physic to be modeled: shock wave propagation, cavitation and fluid-structure interaction. A sketch of the experimental setup and the theoretical behavior is given in Fig.2.18 [20].
Joukowsky and Allievi gave the basis on WH's classical theory through their theoretical analysis. Considering 1D normal pressure wave, the rise of pressure ΔP is given by the Joukowsky's equation:

$$\Delta P = -\rho c_L \Delta V \tag{2.142}$$

Where c_L, ΔP, ΔV and ρ denote the pressure wave speed, the rise of pressure, the change of the fluid's velocity and the fluid's density, respectively.
The pressure wave speed is estimated from Korteweg's equation:

$$c_L = \sqrt{\frac{K/\rho}{1 + (K/E).(D/e)}} \tag{2.143}$$

Where K, ρ, E, D and e denote the fluid's bulk modulus, the fluid's density, the young modulus of the structure, the inner diameter and the pipe's thickness, respectively.

Let us denote by τ the time period for a pressure wave to travel back and forth between the valve and the water reservoir. The time period is given by $\tau = 2.L/c_L$, where L is the length of the elastic pipe.
We assume a zero prescribed velocity at the closed valve as well as a pressure boundary condition at the outlet of the water reservoir. Thus according to
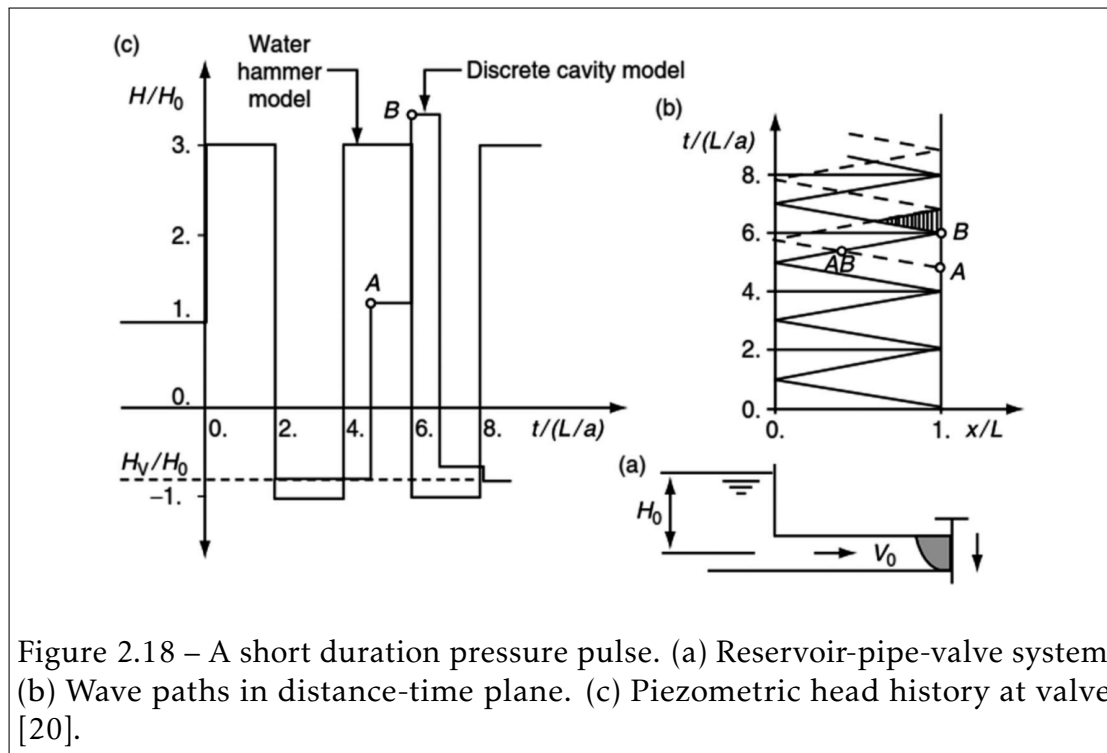
Figure 2.18 – A short duration pressure pulse. (a) Reservoir-pipe-valve system. (b) Wave paths in distance-time plane. (c) Piezometric head history at valve [20].

Eq.2.142, the change in pressure always occurs at the closed valve, and the change in velocity always occurs at the reservoir-pipe junction.

Let us denote by P_r, t_0, P_0 and V_0 the pressure in the reservoir, the valve closure time, the initial pressure and initial velocity at t_0, respectively. At time, $t = t_0$, a pressure wave $P = P_1 = P_0 + \Delta P$ is generated at the closed valve (imposed $V = 0$, but not modeled) and is propagated from the valve to the reservoir at the pressure wave velocity c_L.

According to the Eq.2.142, when the pressure wave reaches the reservoir at time $t_0 + \tau/2$, it is reflected due to the prescribed constant pressure at the reservoir ($P = P_r$), and thus travels from the reservoir to the valve leaving water at pressure $P = P_r$ and $V = -V_0$.

When the pressure wave reaches back, the valve is at time $t = t_0 + \tau$, and the confined water in the pipe is entirely at pressure $P = P_r$. The change of velocity from $-V_0$ to 0 generates a pressure drop in the cylinder at the maintained reservoir location. The pressure drop ΔP is given by $P = P_2 = P_r - \Delta P$.

Recalling that $P_0 = P_r$, thus we have $P_2 < P_0$ and the pressure drops below the initial pressure. Depending on the saturation pressure P_{sat}, two scenarios are possible:

- **Case 1:** $P_2 > P_{sat}$

    - At time $t = t_0 + \frac{3\tau}{2}$, the pressure wave reaches the reservoir and is once again reflected to the valve. This time $V = V_2 = V_0$ and $V_2 > 0$ (because $P_r > P_2$).

    - The pressure wave $P = P_r$ reaches the valve at time $t = t_0 + 2.\tau$. The entire water is at pressure $P = P_r$, and the new overpressure is $P = P_r + \Delta P_1$, it takes us back to the previous step at time $t = t_0 + \tau$.

    - Neglecting the dissipative (numerical and physical at the FSI interface) a periodical cycle is obtained.

- **Case 2:** $P_2 < P_{sat}$

    The pressure at the valve drops to the water vapor pressure and the new pressure wave is propagated at the liquid vapor pressure. In order to use Eq.2.142 we must know the new wave propagation velocity depending on the vapor-water mixture.

    To proceed in the analytical study, we add the two following hypothesis:

    - Only a vapor pocket at the valve is formed during the propagation of the pressure wave from the valve to the reservoir. We suppose that the size of the vapor pocket is very small in comparison to the pipe's length. This assumption allows us to consider the pure phase "Cut-off model" that will be considered as a reference solution to validate the "HEM phase-change model".

    - The vapor pocket imposes its pressure to the pressure wave as the reservoir acts as a fixed pressure boundary condition.

Including the two previous assumptions, we are able to complete Eq.2.142. According to Mostowsky(1929), unlike case 1 the reversed direction veloc-

ity is no more imposed to be zero, but instead, decreases to:

$$V_0 - \Delta V_{cv} = V_0 - \frac{g(P_r + P_{sat})}{c_L} \tag{2.144}$$

Now we have a system of two reservoirs (reservoir-vapor cavity). The pressure wave is reflected by the vapor cavity until the vapor pocket collapses at time $t = t_1$, accelerating the water to the closed and giving arise to a new WH. The new rise of pressure is less than the first one, but the superposition of the two pressure waves gives a greater rise of pressure. In Fig.2.18, the superposition of pressures occurs at time $t = t_0 + 3\tau$ which corresponds to the point B.

Simpson's experiment   In Simpson's experience [131], WH due to the rapid closure of the valve described in the previous paragraph are investigated. The problem's parameters are given in Tab.2.1. FEM model are shown in Fig.2.19 and Fig.2.20.

Fluid parameters		Structure parameters	
Pressure (bar)	3.419	Inside Diameter (mm)	19.05
Temperature (K)	296.46	Thickness (mm)	1.6
Density (kg/m^3)	997.55352	Length (m)	36
Initial Velocity (m/s)	0.4	Density ($kg/m3$)	8960
		Elastic Modulus (GPa)	120

Table 2.1 – Fluid's material model parameters
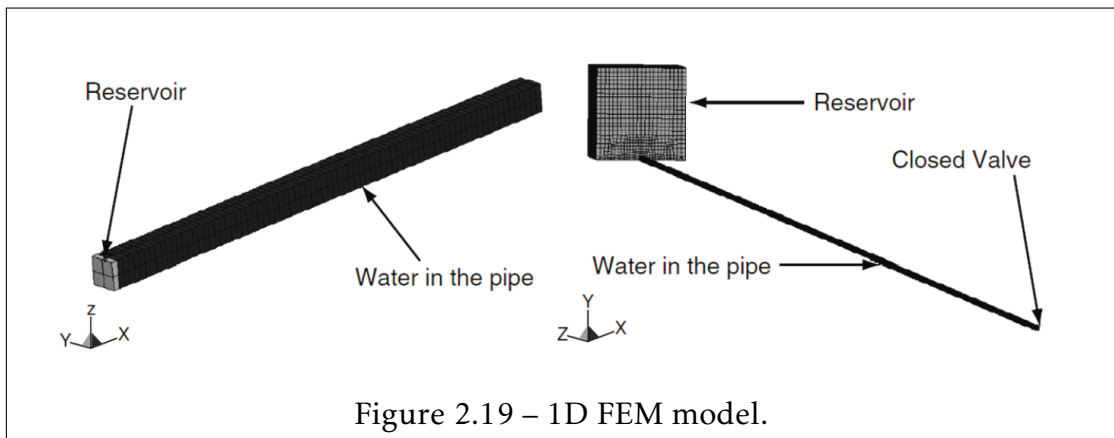
Figure 2.19 – 1D FEM model.
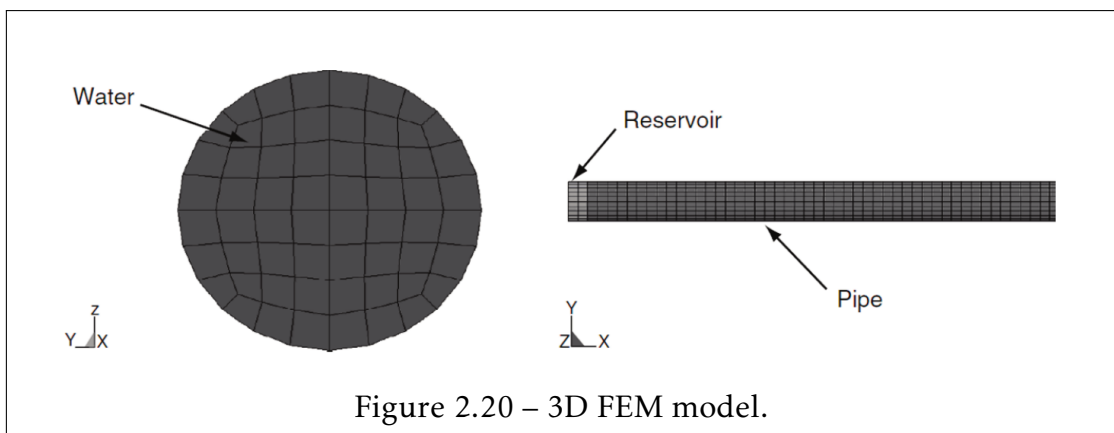
Figure 2.20 – 3D FEM model.

The main objectives of the experimental investigations are:

- To demonstrate that considering FSI coupling is mandatory. The short duration of the pressure pulse and its impact on the structure are investigated.

- To study the cavitation effects that are generated near the valve and the condensation of the vapor that leads to a second shock wave.

A first one dimensional simulation is performed in order to highlight the consequences of neglecting the elasticity of the pipe and the FSI (over-estimation of the pressure peak, over-estimation of the celerity of the pressure wave, ...) and the validation of the HEM phase change model compared to the cut-off model. Since only a small amount of void fraction is created at the end of the pipe (closed valve), the hypothesis given in the theoretical paragraph (for analytical

study) justifies the use of a pure phase Cut-off model and the validation of both the Cut-off and HEM phase change cavitation models for the valve slam WH. A second three dimensional simulation is performed using the HEM phase-change cavitation models and the LS-DYNA fluid-structure interaction coupling algorithm. The obtained results will be validated in comparison with the experiment.

The numerical setups for the two simulations are the following:

- **1-D simulation without FSI:**

    - HEM phase change and Cut-off models are considered.

    - Fluid part is modeled by 67600 Eight-node solid hexahedron ALE elements.

    - The flow is constrained to be 1D setting the velocity in Y-direction and Z-direction to be equal to zero.

    - The closed valve is modeled imposing the velocity to be zero at the valve location.

- **3-D simulation with FSI:**

    - Only the HEM phase change model is considered.

    - Fluid part is modeled by 216000 Eight-node solid hexahedron ALE elements.

    - Structure part is modeled by 86400 Belytschko-Lin-Tsay Lagrangian shell elements.

    - Lagrangian coupling where common nodes between the ALE and Lagrangian parts are merged. The radial expansion of the pipe is small enough so that it does not perform ALE re-meshing to the moving boundary.

Results for the first 1-D simulation are shown in Fig.2.21 in terms of the pressure time history of the fluid at the closed valve. The blue, green and red curves represent the LS-DYNA HEM phase-change results, the LS-DYNA cut-off results

and the experimental results, respectively.

In consideration that the 1D simulation is equivalent to a 3D simulation with a stiff pipe (very high modulus or pipe's thickness), the pressure wave velocity c_L is equal to the water speed of sound in accordance to Eq.2.143. It can be seen from Eq.2.143 that c_L decreases when the Young Modulus and/or the pipe's thickness decreases. The more stiff the pipe, the more c_L is close to the water speed of sound. From replacing in Eq.2.142 c_L by the speed of sound and comparing the LS-DYNA results to the experimental result in Fig.2.21, it can be seen that the pressure peak and the wave speed propagation are over-estimated when neglecting the elasticity of the pipe and fluid-structure coupling effects.

However, 1-D cases are pertinent to compare different solutions for the same problems. A good correlation is observed between "cut-off" and "HEM phase-change" models. The oscillations occurring at time t = 0.12s when the vapor pocket located at the valve condensates, can be explained by the huge change in speed of sound in the fluid mixture when the void fraction is very small as a result of the Wallis Eq.1.130 (see Fig.1.10).

The results from the second 3-D simulation are shown in Fig.2.22 and Fig.2.23 in terms of pressure and void fraction time history of the fluid at the closed valve, respectively. The blue, green and red curves represent the LS-DYNA HEM phase-change results, the WAHA code [41] (1-D simulation with correction of the wave propagation speed by Eq.2.143) and the experimental results, respectively. Pressure values from LS-DYNA simulation have been compared to both experimental data and WAHA code results. The mono-phase WH is well simulated according to the first pressure peak in Fig.2.22, while the second one is slightly over-estimated due to the brutal condensation of the water steam pocket. When the FSI is considered the pressure wave propagation speed is recovered well. Fig.2.23 shows through the void fraction that the different stages of the phase change phenomenon has been modeled: its generation as liquid water turns to saturated vapor ($0 < \alpha < 1$) followed by its condensation as saturated vapor turns to liquid water ($\alpha = 0$).

Figure 2.21 – Pressure time history at the closed valve for the 1-D simulation. The blue, green and red curves represent the LS-DYNA HEM phase-change results, the LS-DYNA cut-off results and the experimental results, respectively.

Figure 2.22 – Pressure time history at the closed valve for the 3-D simulation. The blue, green and red curves represent the LS-DYNA HEM phase-change results, the reference Waha code for WHs simulation results and the experimental results, respectively.

Figure 2.23 – Void Fraction computation using the LS-DYNA HEM phase-change model at the valve for the 3-D simulation.

Cold Water Hammer Test Facilities (CWHTF)   The Cold Water Hammer Test Facility (CWHTF) was designed and built at Forschungszentrum Rossendorf for the WAHALoad European project [41] to study the FSI effects during the WH phenomenon that result from the sudden condensation of a saturated vapor pocket.

This test case was chosen to investigate the performance of the presented phase change model for the simulation of WH in which classical pure phase model such as the cut-off model cannot be used due to initial presence of a non-negligible amount of saturated vapor pocket. The investigation of the collapse of the vapor is programmed in two phases:

1. Replace the water vapor by vacuum (density is considered but pressure is zero) and simulate the projection of the water column on the rigid plate.

2. Model the collapse of the bubble using the HEM phase change model.

At the current stage, the first phase is completed and the results are presented in this thesis and the second phase is on going. We are still in an early phase of the validation of the model and further investigation needs to be done on simple test cases before confronting this complex industrial case.

The experiment consist of a water tank, a equipped pipe, a closed end (rigid disk) and a valve in the middle of the pipe that separates the pipe into two parts: a left part linked to the water tank and a right part linked to a rigid disk at the closed end (see Fig.2.24). The full CWHTF experimental setup is modeled by:

- A water tank with a stiff wall. A $1bar$ constant pressure is imposed at the junction between the left part of the pipe and the water tank.

- An isotropic linear elastic pipe.

- A valve opens at time $t = 0.2s$. The valve opening is fast ($21ms$) and can be considered an instantaneous opening (EDF R&D performs internal investigations that confirm that the kinematic of the valve opening has a neglecting impact on the dynamic response of the structure).

- Two supports at the pipe-tank junction and at the valve.

- A rigid disk.

The parameters for the CWHTF experimental setup are given in Tab.2.2, Tab.2.3 and Tab.2.4.
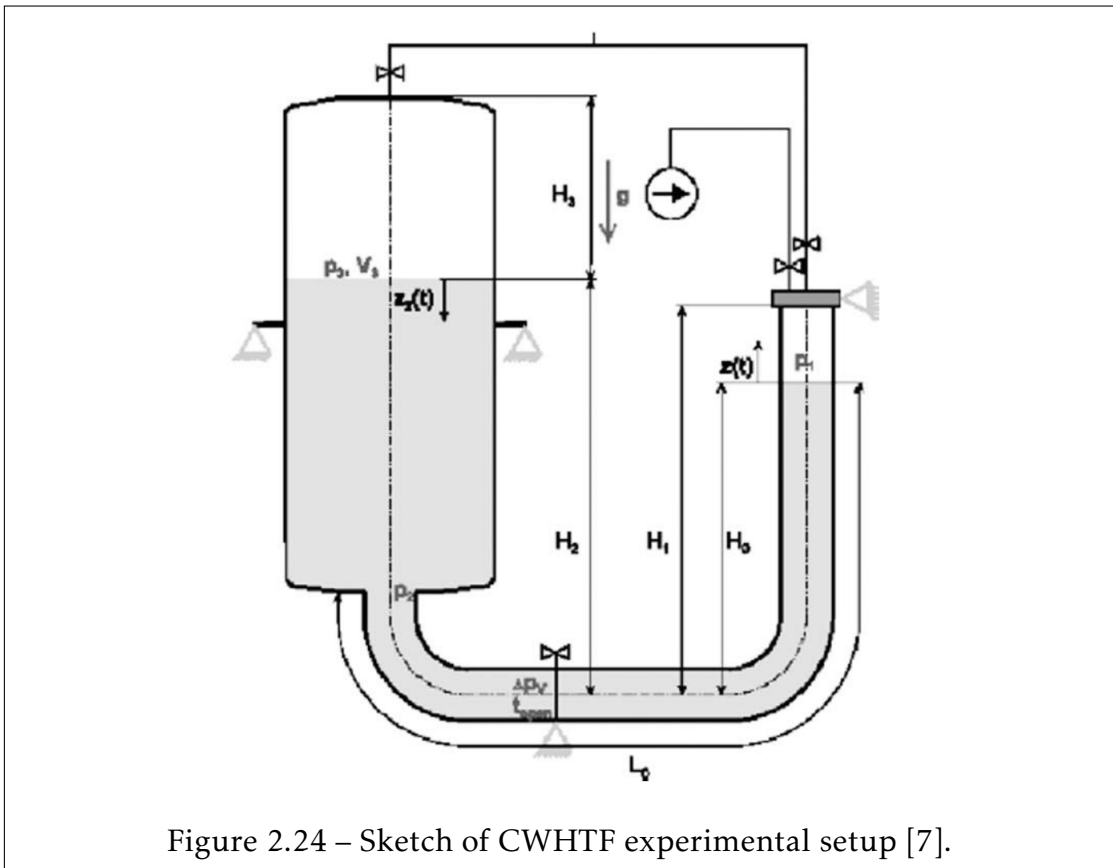
Figure 2.24 – Sketch of CWHTF experimental setup [7].

Parameters	Pipe	Tank
External diameter(mm)	219	800
Shell thicnkess (mm)	6	6
Curvature radius at elbows (mm)	306	-
Pipe's total length (m)	3.285	-
Height (m)	-	1.2
$H_1 - H_0$ (m)	0.3	
Internal volume ($litres$)	124	750
Design pressure (bar)	60	10
Plastic pressure (bar)	90	-
Failure pressure (bar)	226	-

Table 2.2 – Geometry and design properties for the structure parts

Parameters	Steel	Water
Young modulus E (GPa)	200	-
Poisson ratio	0.3	-
Density ($kg/m3$)	7850	997
Speed of sound (m/s)	5856	1450
Temperature (K)	-	293.16
Saturation pressure (bar)	-	$29e^{-3}$
Pressure in the tank	-	1

Table 2.3 – Material model parameters

Stiffness	K_x^{S1}	K_y^{S1}	$K_{rot_z}^{S1}$	K_x^{21}	K_y^{S2}	$K_{rot_z}^{S2}$
Values (N/m)	$2.35e+7$	$1e+4$	$1e+2$	$1e+9$	$2.27e+7$	$1e+3$

Table 2.4 – Stiffness of the support $S1$ at the pipe-tank junction and the support $S2$ at the opening valve

At the beginning, the valve is opened and the pipe is entirely filled with cold water at atmospheric pressure. Then, the valve is closed and the remaining air is evacuated in the right part, such that only liquid water and steam at saturation remain at the right part of the valve. Thus far, the water tank and the left part of the pipe are at atmospheric pressure and the right part of the pipe composed of liquid water at the saturation pressure and above a vapor pocket at saturation pressure closed to the end. At time $t = 0.2sec$, the valve is opened and due to the difference in pressure between the left and the right part, the liquid water is accelerated in the direction of the closed end, compressing the vapor pocket until it collapses and triggers the WH.

The finite element model is shown in Fig.2.25 in which the blue, green, red and yellow elements represent the water tank, the liquid water at atmospheric pressure, the saturated liquid water and the vacuum (replacing saturated water steam), respectively. The structure is modeled by BLT shell elements. A Lagrangian coupling is used in combination with the ALE re-meshing algorithm

due the displacement of the Lagrangian rigid disk. Since the saturated vapor is replaced by vacuum material (pressure is zero). The value of the saturated pressure $29e^{-3}bar$ is removed from the fluid pressure.

The results from the first CWHTF simulation are shown in Fig.2.26 and Fig.2.27 in terms of pressure time history of the fluid at the pressure sensor (located at $100mm$ from rigid disk) and in Fig.2.28 in terms of the vertical displacement (x-direction in Fig.2.25) time history of the rigid disk. No experimental data are provided for the structural displacements.

Pressure results are in good agreement with experimental data for the first pressure peak, but slightly delayed for the second one. To highlight the FSI effects, the vertical and horizontal displacements amplitude of the rigid disk are shown in Fig.2.28 and Fig.2.29, respectively.

The HEM model results are shown in Fig.2.30. The results are qualitatively acceptable but are not good enough to be validated. The first WH, due to the impact on the rigid plate, appears sooner than the experiment and is over-estimated.

Figure 2.25 – FEM of CWHTF experimental setup: blue, green, red and yellow elements represent the water tank, the liquid water at atmospheric pressure, the liquid water at the saturation pressure and the initial saturated water steam, respectively.

Figure 2.26 – Pressure time history at the sensor located at 100 mm from the rigid disk.

Figure 2.27 – Local zoom on the first peak of pressure.

Figure 2.28 – Vertical displacement of the rigid disk.

Figure 2.29 – Horizontal displacement of the rigid disk.

Figure 2.30 – HEM model pressure time history at the sensor located at 100 mm from the rigid disk.
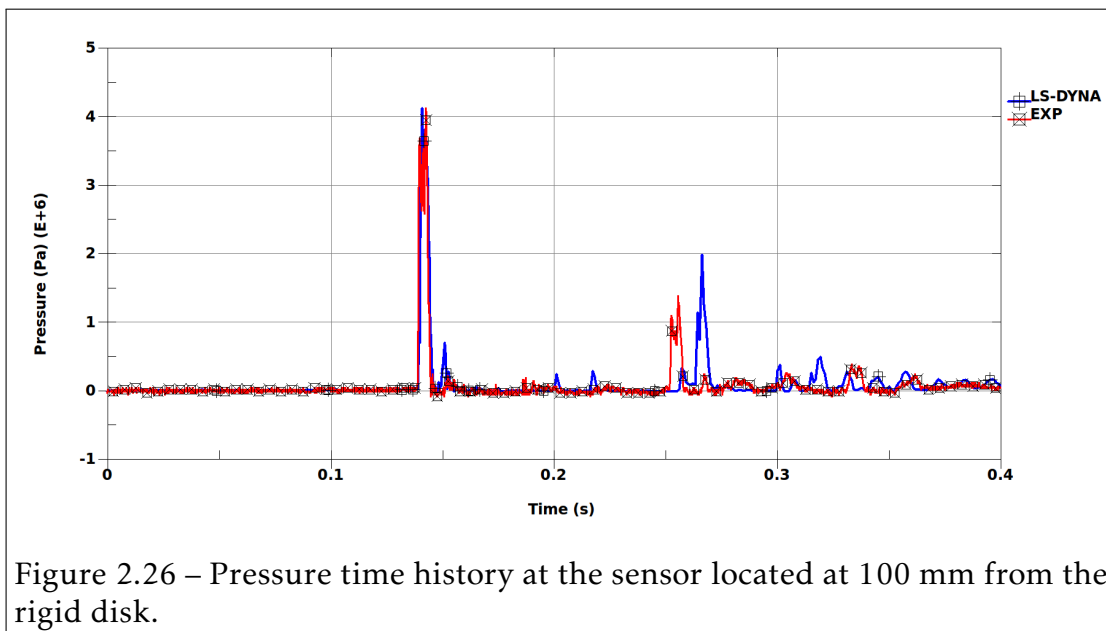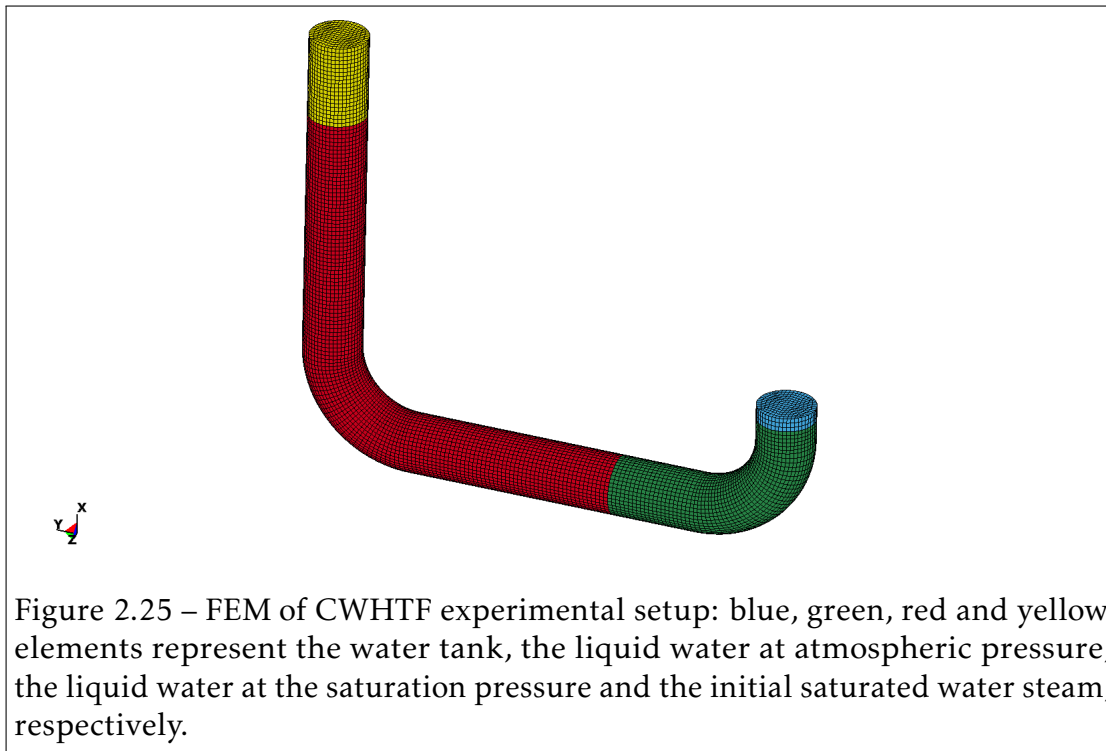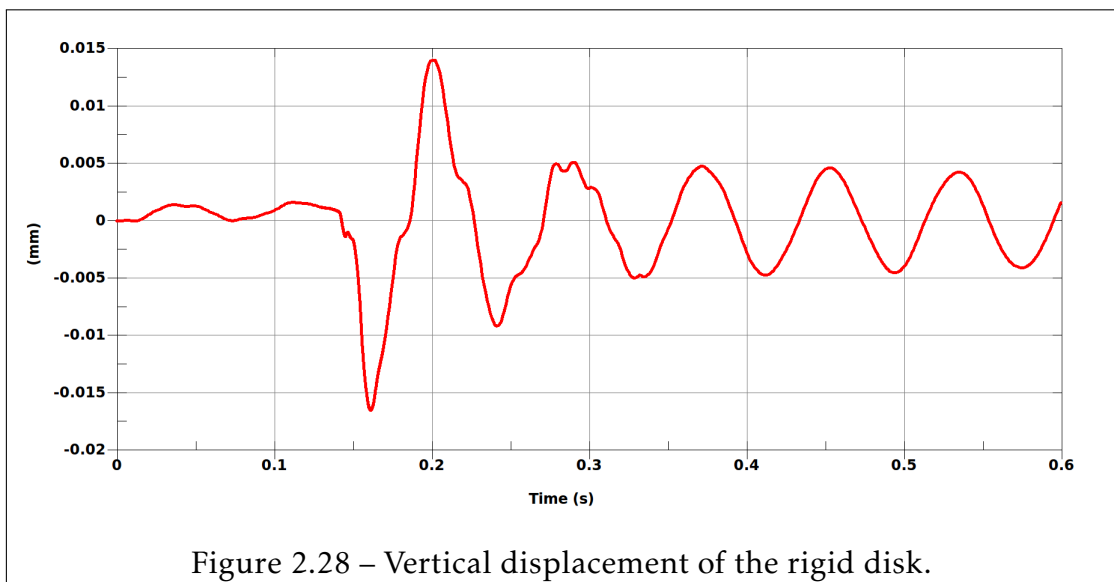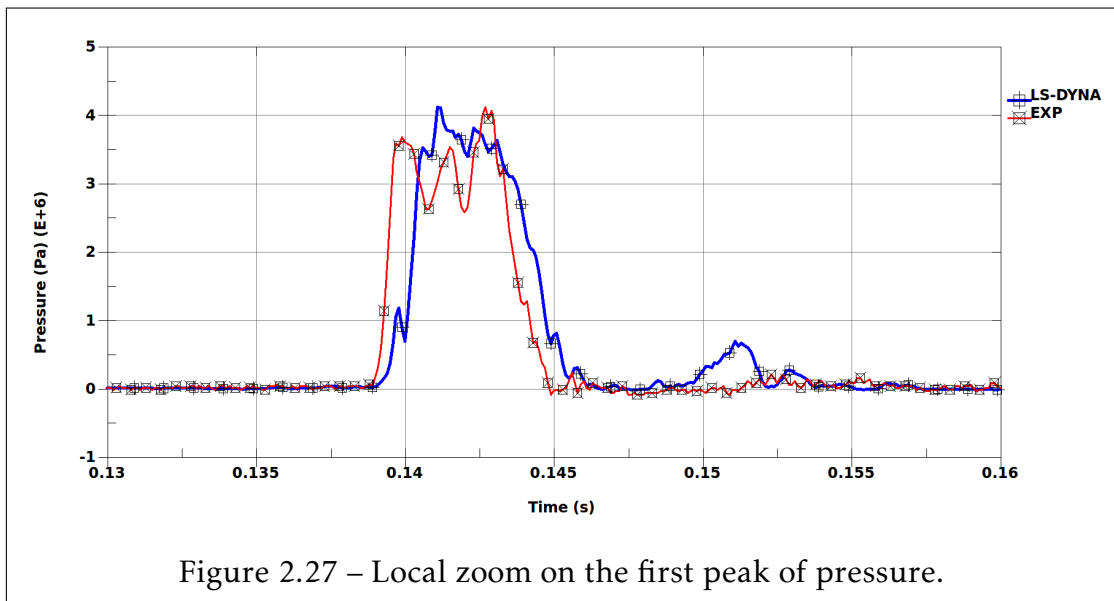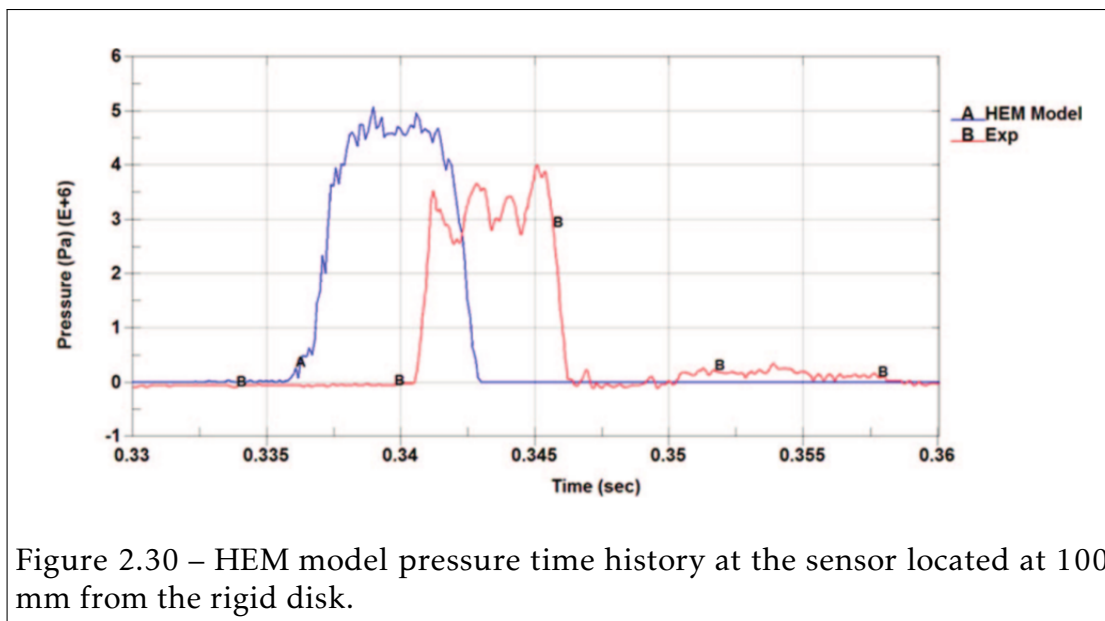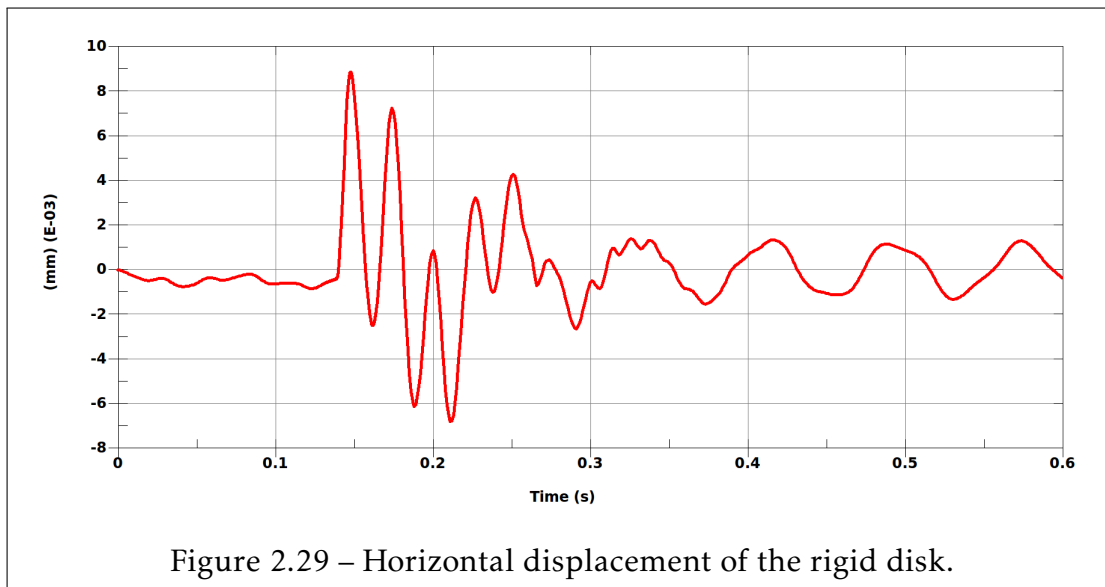
2.6.3   Simulation of CFRP shell structures subjected to water shock waves

This work is part of a collaboration between the University of Narvik (NUC - Norway), which was in charge of the experimental tasks, and the University of Lille (USTL - France) which was in charge of the numerical tasks. The project was sponsored by the Norwegian Research Council under the **PETROMAKS grant no 200585**.

The work attempts to recapture the results presented in the study [66]. A different numerical methodology is used to model the same behavior. However, the; current study involves not only structure modeling as provided in the earlier study but also fluid behavior using multi-material ALE methods. The focus of this study is to validate the numerical simulation of ALE and FSI using LS-DYNA to be able to model the complex fluid behavior in a shock tube and the structural stresses generated in a adjoin CFRP test sample.

Shock tube experimental setup:

Installation   The shock tube consists of two main parts: the driver section and the driven section (Fig.2.31) [66, 55, 62]. The driver section contains compressed air, which is released into the driven section in order to create a shock wave. This is generally triggered with a burst plate or high speed valve. In the given experiments, the shock wave is generated in water by releasing high pressure air through high speed valve. The shock wave propagates through water in the driven section [66, 55, 62]. At the end of the driven section, a T-section is fitted that enables a secure mount of a test specimen (Fig.2.32).

In these experiments, three pressure sensors (one static and two piezoelectric sensors) are placed at three different places in the shock tube to capture the propagation of the shock wave. The test specimen is mounted with a rosette strain gage in order to capture the strains created by the shock wave impact (Fig.2.32).

Figure 2.31 – The shock tube: Driver section to the right (containing compressed air) and driven section to the left containing uncompressed water and a T-section for mounting the test specimen.[66]
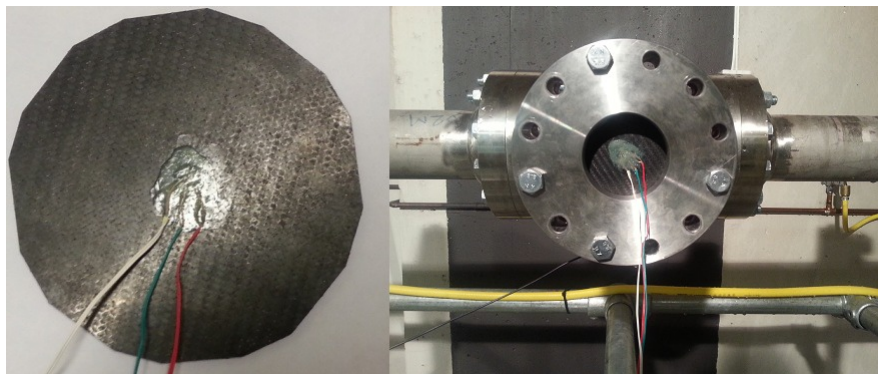
Figure 2.32 – Left: Rosette strain gage attached to a CFRP specimen using cyanide-based glue. The three-colored wires are connections to three strain gages at 45o angles. The strain gage is placed in the geometric center of the plate. Right: The T-section. The CFRP specimen is mounted in the T-section where the three-colored wires are connection of the rosette strain gage.[66, 62]

Experimental results   It was found that strain profile is a linear mapping of driven pressure profile, as can be seen from Fig.2.33. The experiments are

repeated on the same test piece to confirm the validity of the results. Repeated experiments proved that CFRP test pieces can withstand shock pressure (Fig.2.33). However, it is worth noting that applied shock pressure in these experiments did not result in strains for more than 25% of strain limit for rupture. It can be deduced that material behaves linearly under shock pressure.
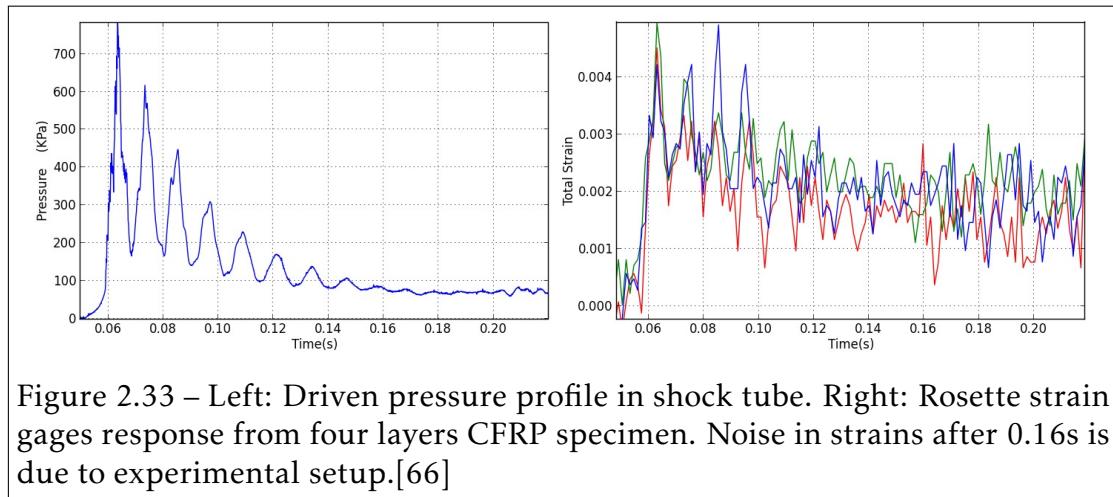
Figure 2.33 – Left: Driven pressure profile in shock tube. Right: Rosette strain gages response from four layers CFRP specimen. Noise in strains after 0.16s is due to experimental setup.[66]

Shock tube numerical setup:

In this work mutliphysics numerical analysis was carried out on a CFRP test piece subjected to shock wave attached to a tube. This setup has been commonly used to test material under dynamic loading [66]. In the previous work [66, 63], FSI effects were not considered and were lacking in the total response of the structures due to the damping effects of the water on the CFRP displacement as it goes in its vibrations modes exposed to the cyclic pressure loading (Fig.2.33).

To complete the previous work, the T-section of the shock tube, including the water contained inside the tube and the test piece are modeled using LS-DYNA. The simulation starts when the shock wave reaches the entrance of the section, the time history of the pressure is recorded by the experimental setup as an input pressure (in the green FEM elements in Fig.2.35).

Here, the physics of the shock wave in the fluid mediums and the composite shell material were strongly coupled using the advanced Euler-Lagrange FSI method described in subsection 2.5.2. The pipe walls are stiff enough to not consider the FSI effects between the water and the stiff pipe. However due to the fluid's viscosity and the effects at the boundary, stick boundary conditions are used at the fluid-pipe interface by constraining the fluid's velocity to be zero in order to model the friction at the pipe's wall. The CAD model of the simulated setup (see Fig.2.31) is shown in Fig.2.34. A sketch of the testing section (see Fig.2.32) describes the fluid structure interface in Fig.2.35.

Figure 2.34 – Shock Tube ALE Model: red shows the pressurized section with air, blue shows the section filled with water at 1 atm and the T-section is the testing section of the shock tube where material sample is placed.
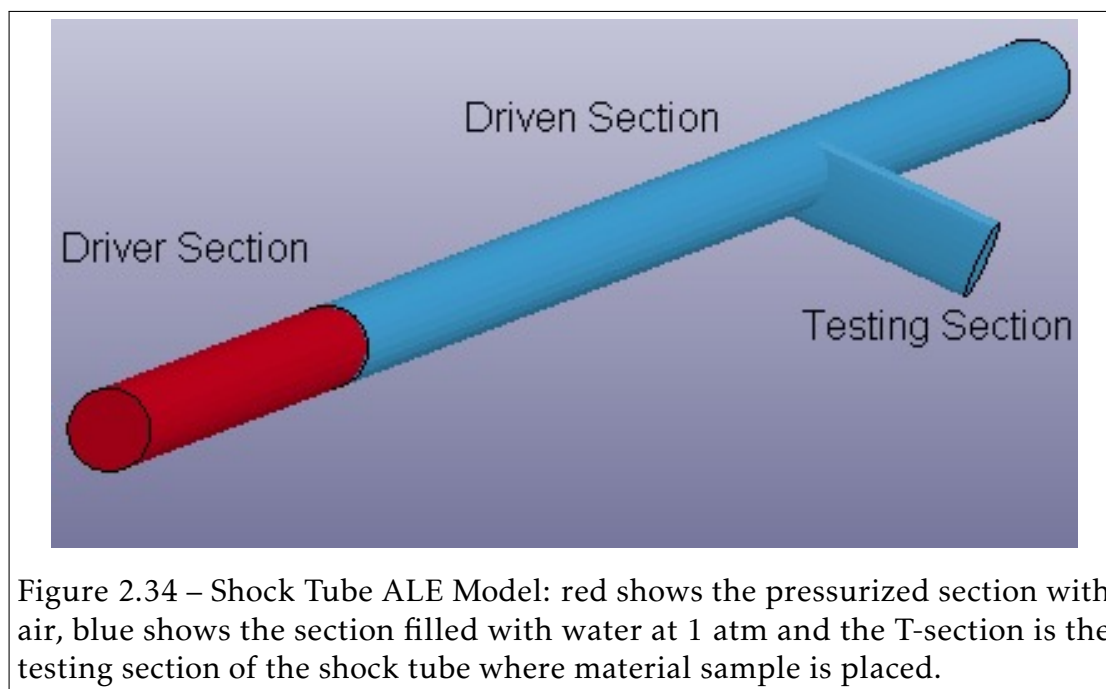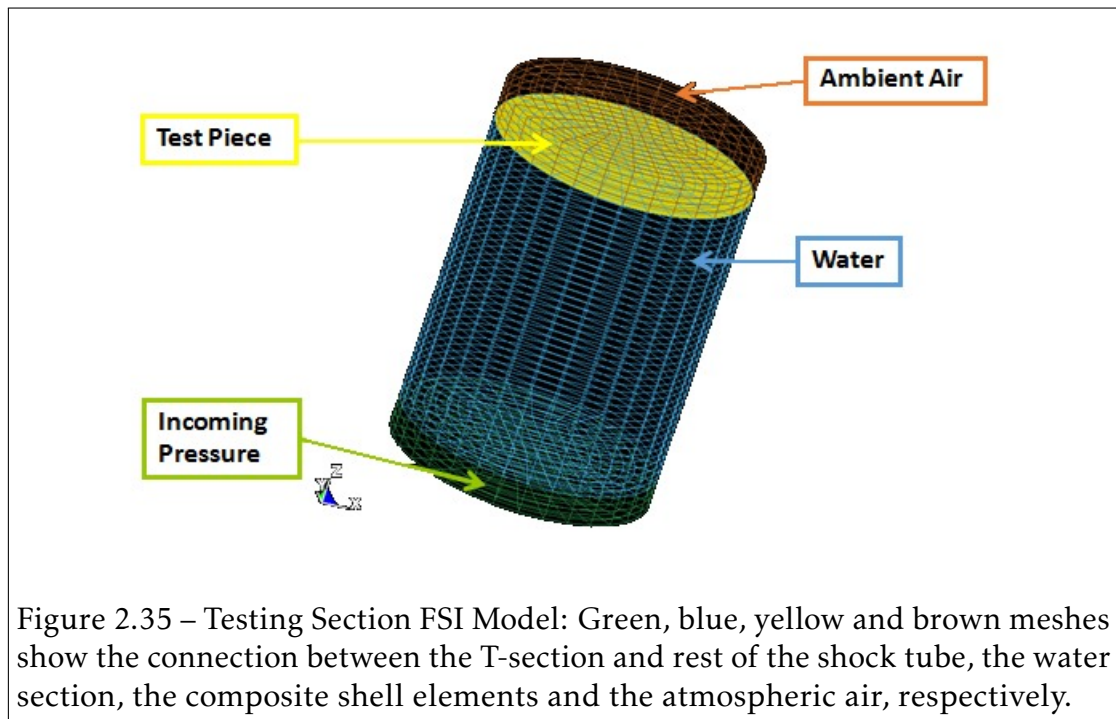
Figure 2.35 – Testing Section FSI Model: Green, blue, yellow and brown meshes show the connection between the T-section and rest of the shock tube, the water section, the composite shell elements and the atmospheric air, respectively.

Constitutive material models   Both fluids (water and air) are considered to be Newtonian and compressible (Eq.1.83). For the pressure computation, the Ideal gas EOS and Mïe-Gruneisen EOS are used for the air and the fluid, respectively. The different material models parameters are given in Tab.2.5, Tab.2.6 and Tab.2.7.

The carbon fiber reinforced polymers (CFRP) quasi-isotropic shell structure is modeled by a linear isotropic material (Eq.1.69). The homogenization strategy was proposed by Narvik University, where the input material model parameters were obtained through experimental MTS® study [66]. Validation of the homogenized parameters for the numerical study were also provided in [66] against experimental data using the commercial FEM code ANSYS [9].

The same equations and parameters were used in this paper to simulate the dynamic behavior of the composite shell CFRP subjected to water shock waves. The parameters are given in Tab.2.8.

Material	Density (kg/m^3)	Dynamic Viscosity
Air	1.28	$1.983e^{-5}$
Water	998	$1e^{-3}$

Table 2.5 – Fluid's material model parameters

Material	γ	e^{int}
Air	1.4	2.5e+5

Table 2.6 – Ideal gas EOS parameters

Material	s	Speed of sound $c(m/s^1)$	$V0 = \frac{1}{1+\nu}$
Water	1.9	1480	0.999954272

Table 2.7 – Mie-Gruneisen EOS parameters

Material (Homog.)	Density (kg/m^3)	Young Modulus	Poisson ratio
CFRP	1760	$2.6751e^{10}$	0.2

Table 2.8 – Composite shell's material model parameters

Comparison between experimental and numerical results:

The shock tube was modeled using the ALE method. The shock wave was generated by the interaction of high pressure air with water at atmospheric pressure. This interaction generated a shock wave that propagates along the tube filled with water.

Experimental and numerical results are compared in terms of X-strain at the location of the strain gages at the top of the composite shell on its center, as shown in Fig.2.32. Fig.2.37 shows the comparison between the strongly coupled numerical simulation and the experimental simulation results.

Results from the shock tube experimental setup and the numerical simulations using LS-DYNA proved that under dynamic loading, CFRP E720 composite

behaves linearly. A good agreement between strongly coupled simulations and experimental results confirm that FSI capabilities of LS-DYNA can be used to model deforming structures under dynamic loading especially in the cases discussed. Numerical simulations might help to reduce the total number of experimental tests that can be money and time consuming.

Figure 2.36 – Left: Visualization of the shock front propagation in the shock tube T-section. Right: Vizualisation of the CFRP Von-Mises stresses.

Figure 2.37 – Comparison between experimental (in red) and numerical (in blue) X-strain results taken at the top of the composite shell at its center.
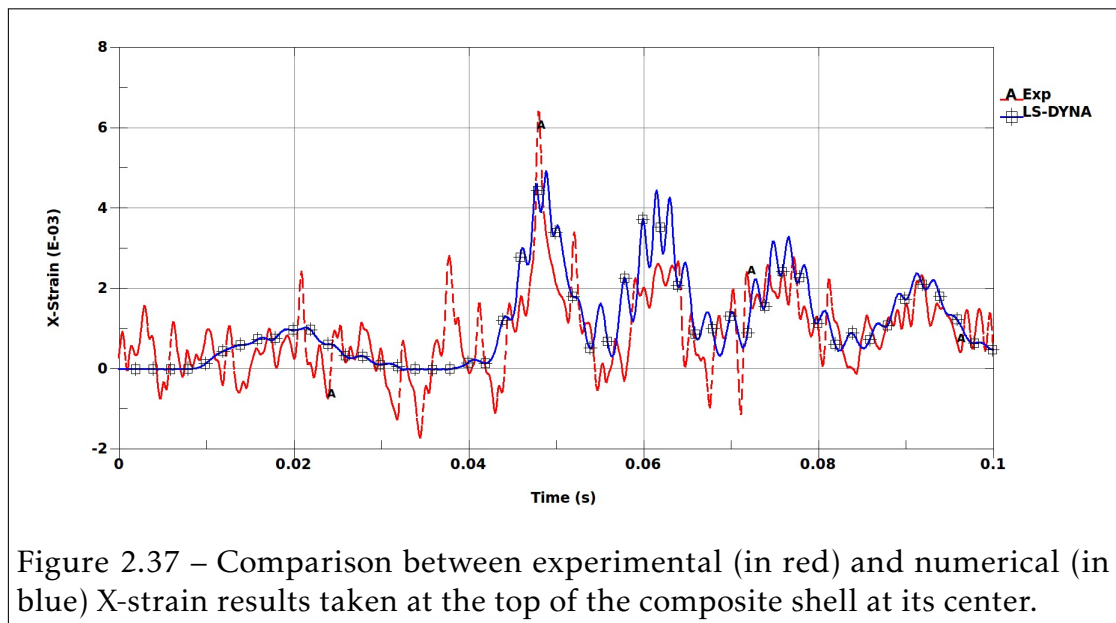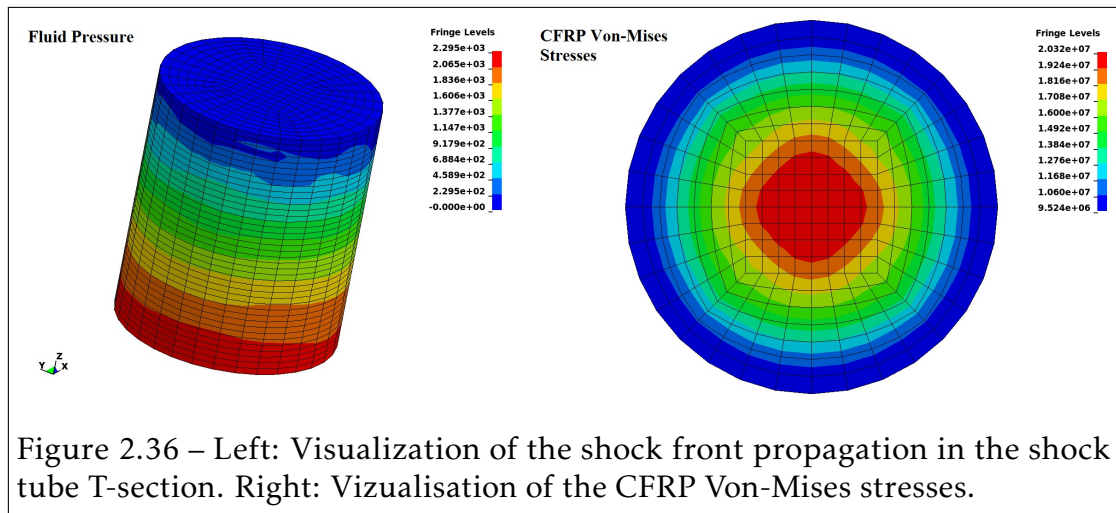
Smoothed Particle Hydrodynamics Method

3.1 Standard Weakly-Compressible SPH (WC-SPH) method

In this section, we present the basics and concepts of the standard WC-SPH method which is the most commonly used and present in many commercial codes [4, 48, 118]. For a more detailed lecture on the theoretical aspects of the method and the major challenges (convergence, numerical stability, boundary condition and adaptivity), there are many books on the SPH method that one can cite [83, 151, 152].

3.1.1 Basic concepts of the standard SPH formulation

Let us consider a scalar function f of the position vector $\overrightarrow{x}$ defined on a domain Ω. The function $f(\overrightarrow{x})$ can be defined by the following convolution product:

$$f(\overrightarrow{x}) = \int_{\Omega} f(\overrightarrow{y}) \delta(\overrightarrow{x} - \overrightarrow{y}) d\overrightarrow{y}, \tag{3.1}$$

where the Dirac delta function $\delta(\vec{x} - \vec{y})$ is defined by:

$$\delta(\vec{x} - \vec{y}) = \begin{cases} 1 & if \quad \vec{x} = \vec{y} \\ 0 & if \quad \vec{x} \neq \vec{y} \end{cases} \tag{3.2}$$

The first step in deriving the discret SPH schemes is to approximate the Dirac delta function by the kernel function $W\left(\vec{x} - \vec{y}, h\right)$ (illustrated in Fig.3.1) that satisfies the three following conditions:

- **Delta function property:**

$$\lim_{h \to 0} W\left(\vec{x} - \vec{y}, h\right) = \delta(\vec{x} - \vec{y}) \tag{3.3}$$

- **Unity condition:**

$$\int_\Omega W\left(\vec{x} - \vec{y}, h\right) = 1 \tag{3.4}$$

- **Compact condition:**

$$W\left(\vec{x} - \vec{y}, h\right) = 0 \quad if \quad | \vec{x} - \vec{y} | \geq \kappa h \tag{3.5}$$

Where κ is a constant related to the kernel function. The value of κ is discussed later.

The kernel approximation of the function f is finally obtained by replacing the kernel function in Eq.3.1:

$$< f(\vec{x}) > = \int_\Omega f(\vec{y}) W\left(\vec{x} - \vec{y}, h\right) d\vec{y} \tag{3.6}$$

The kernel approximation is second order accurate [83]. Through the substitution of the function $f(\vec{x})$ by its spatial derivative, we have:

$$< \nabla_{\vec{x}}.f(\vec{x}) > = \int_\Omega [\nabla_{\vec{x}}.f(\vec{x})] W\left(\vec{x} - \vec{y}, h\right) d\vec{y} \tag{3.7}$$

Figure 3.1 – Kernel function scaled to $\kappa = 2$ and its support domain.
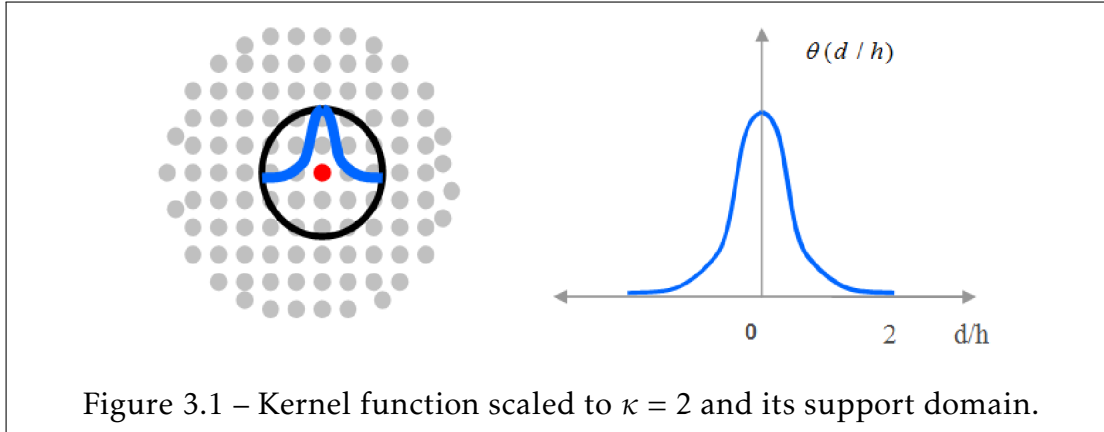
Considering that:

$$< \nabla_{\overrightarrow{x}} . f(\overrightarrow{x}) > = \int_{\Omega} \nabla_{\overrightarrow{x}} . [f(\overrightarrow{x}) W(\overrightarrow{x} - \overrightarrow{y}, h)] d\overrightarrow{y} - \int_{\Omega} f(\overrightarrow{x}) . \nabla_{\overrightarrow{x}} W(\overrightarrow{x} - \overrightarrow{y}, h) d\overrightarrow{y}$$
$$(3.8)$$

And through the use of the divergence theorem on the first integral in the right hand-side, we have:

$$< \nabla_{\overrightarrow{x}} . f(\overrightarrow{x}) > = \underbrace{\int_{\Gamma} f(\overrightarrow{x}) W(\overrightarrow{x} - \overrightarrow{y}, h) . \overrightarrow{n} \Gamma}_{= 0 \text{ due to Eq.3.5}} - \int_{\Omega} f(\overrightarrow{x}) . \nabla_{\overrightarrow{x}} W(\overrightarrow{x} - \overrightarrow{y}, h) d\overrightarrow{y} \quad (3.9)$$

The kernel approximation of the derivative $< \nabla_{\overrightarrow{x}} . f(\overrightarrow{x}) >$ is given by:

$$< \nabla_{\overrightarrow{x}} . f(\overrightarrow{x}) > \approx - \int_{\Omega} f(\overrightarrow{x}) . \nabla_{\overrightarrow{x}} W(\overrightarrow{x} - \overrightarrow{y}, h) d\overrightarrow{y} \qquad (3.10)$$

The approximate sign $\approx$ in Eq.3.10 is used because the first integral in the right hand-side in Eq.3.9 is zero only if the compact support domain of the kernel function is included in the computation domain Ω.

Particle approximation

Unlike FEM models, the SPH discrete model is mesh-less, composed of discrete particles i (see Fig.3.2) of discrete mass m_i.
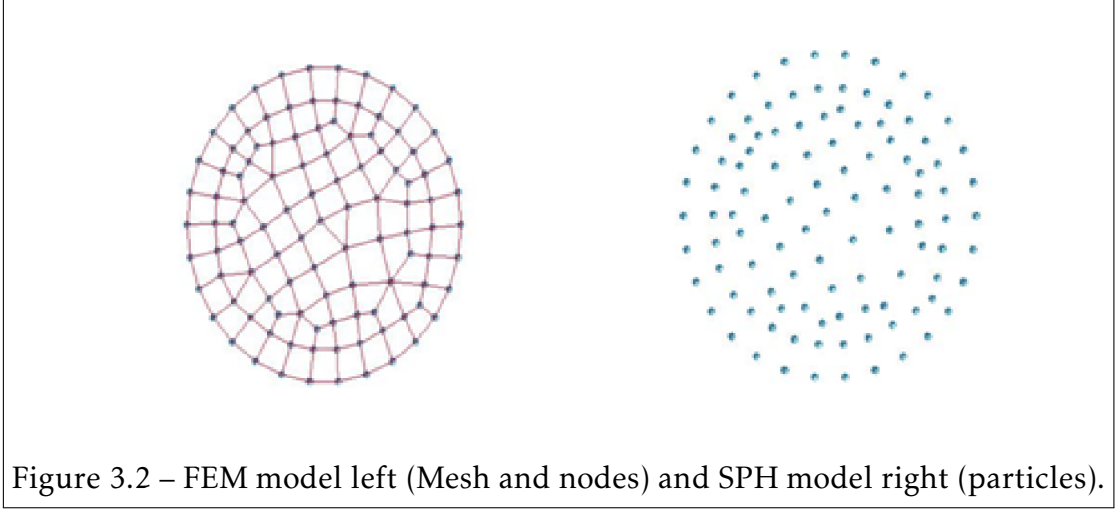
Figure 3.2 – FEM model left (Mesh and nodes) and SPH model right (particles).

Let i be a SPH particle located at the position $\overrightarrow{x_i}$. We denote by D_i the set of N_i particles contained in the kernel support domain centered at particle i. The particle approximation of the continuum equation Eq.3.5 is obtained by replacing the continuous integral with a discrete Riemann sum:

$$f^{sph}(\overrightarrow{x_i}) = \sum_{j \in D_i} \frac{m_j}{\rho_j} f(\overrightarrow{x_j}) W\left(\overrightarrow{x_i} - \overrightarrow{x_j}, h\right), \qquad (3.11)$$

where V_j is the neighbor particle j volume given by $V_j = \frac{m_j}{\rho_j}$.

For numerical purposes, kernel functions with compact support domain (Eq.3.5) are considered making the sum in Eq.3.11 finite.

Let $q = \frac{|\overrightarrow{x_i} - \overrightarrow{x_j}|}{h}$. The following kernel functions are used in the literature:

• Gaussian function truncated at three smoothing length distance ($\kappa = 3$):

$$W\left(\overrightarrow{x_i} - \overrightarrow{x_j}, h\right) = \begin{cases} \frac{\alpha_d}{h^d} exp\left[-q^2\right] & q \leq 3 \\ 0 & q > 3 \end{cases}, \qquad (3.12)$$

where $d = 1, 2, 3$ is the dimension of the problem and α_d is a normalization factor given by $1/\sqrt{\pi}, 1/\pi$ and $1/(\pi\sqrt{\pi})$ for 1-D, 2-D and 3-D problems, respectively.

- M_4 cubic spline function truncated at two smoothing length distance ($\kappa = 2$):

$$W\left(\overrightarrow{x_i} - \overrightarrow{x_j}, h\right) = \frac{\alpha_d}{h^d} \begin{cases} \frac{1}{4}(2-q)^3 - (1-q)^3 & 0 \leq q < 1 \\ \frac{1}{4}(2-q)^3 & 1 \leq q < 2 \\ 0 & q \geq 2 \end{cases}, \qquad (3.13)$$

where $d = 1, 2, 3$ is the dimension of the problem and α_d is a normalization factor given by $2/3, 10/(7\pi)$ and $1/(\pi)$ for 1-D, 2-D and 3-D problems, respectively.

- M_5 quartic spline function truncated at two smoothing length distance ($\kappa = 2.5$):

$$W\left(\overrightarrow{x_i} - \overrightarrow{x_j}, h\right) = \frac{\alpha_d}{h^d} \begin{cases} (\frac{5}{2}-q)^4 - 5(\frac{3}{2}-q)^4 + 10(\frac{1}{2}-q)^4 & if \quad 0 \leq q < \frac{1}{2} \\ (\frac{5}{2}-q)^4 - 5(\frac{3}{2}-q)^4 & if \quad \frac{1}{2} \leq q < \frac{3}{2} \\ (\frac{5}{2}-q)^4 & if \quad \frac{3}{2} \leq q < \frac{5}{2} \\ 0 & if \quad q \geq \frac{5}{2} \end{cases},$$
$$(3.14)$$

where $d = 1, 2, 3$ is the dimension of the problem and α_d is a normalization factor given by $1/24, 96/(1199\pi)$ and $1/(20\pi)$ for 1-D, 2-D and 3-D problems, respectively.

- M_6 quintic spline function truncated at two smoothing length distance ($\kappa = 3$):

$$W\left(\overrightarrow{x_i} - \overrightarrow{x_j}, h\right) = \frac{\alpha_d}{h^d} \begin{cases} (3-q)^5 - 6(2-q)^5 + 15(1-q)^5 & if \quad 0 \leq q < 1 \\ (3-q)^5 - 6(2-q)^5 & if \quad 1 \leq q < 2 \\ (3-q)^5 & if \quad 2 \leq q < 3 \\ 0 & if \quad q \geq 3 \end{cases},$$
$$(3.15)$$

where $d = 1, 2, 3$ is the dimension of the problem and α_d is a normalization factor given by $1/120, 7/(478\pi)$ and $1/(120\pi)$ for 1-D, 2-D and 3-D problems, respectively.

Unlike FEM, in which the shape functions are expressed in terms of one space

variable (e.g. position vector $\vec{x}$) the kernel smoothing functions are expressed in terms of two space variables : The distance $r =| \vec{x_i} - \vec{x_j} |$ and the smoothing length h. For a given particle, to increase the number of neighbors within its compact support domain by increasing the smoothing length does not improve the overall convergence.

For example, in consideration of the cubic spline kernel scaled by $\kappa = 2$, increasing the number of neighbors by increasing the value of κ to $\kappa = 3$ (or any greater value) is not a good practice. It is recommended to use higher order kernels (for example the Gaussian, the quartic or the quintic kernels) to increase the number of neighbors. The counterpart for using higher order kernels is the increase of the CPU time since more particles are involved in the neighbors search algorithm and in the SPH discrete sums. This is one of the reasons why most of the commercial software uses by default the cubic spline kernel scaled by $\kappa = 2$.

For now, let us consider that the smoothing length is constant. We adopt the following notations:

- $f_i = f(\vec{x_i})$ and $f_j = f(\vec{x_j})$

- $\vec{x_{ij}} = \vec{x_i} - \vec{x_j}$ and $\vec{x_{ji}} = \vec{x_j} - \vec{x_i}$

- $x_{ij} =\| \vec{x_{ij}} \|=\| \vec{x_{ji}} \|= x_{ji}$

- $W_{ij} = W\left(\vec{x_i} - \vec{x_j}, h \right)$

- $\nabla_i W_{ij} = \nabla_{\vec{x_i}} W_{ij} = \left(\frac{\partial W_{ij}}{\partial x_{i_1}}, \frac{\partial W_{ij}}{\partial x_{i_2}}, \frac{\partial W_{ij}}{\partial x_{i_3}} \right).$

Since the kernel is only a function of the space variable, the derivative of the kernel function is given by:

$$\nabla_i W_{ij} = \frac{\vec{x_{ij}}}{x_{ij}} \frac{\partial W_{ij}}{\partial x_{ij}} = -\frac{\vec{x_{ji}}}{x_{ij}} \frac{\partial W_{ij}}{\partial x_{ij}} \tag{3.16}$$

Substituting Eq.3.16 into Eq.3.10, we obtain a first particle approximation for the derivative of the function f:

$$\nabla_{\vec{x}}^{sph1} f = \sum_{j \in D_i} \frac{m_j}{\rho_j} f_j . \nabla_i W_{ij} \tag{3.17}$$

If f is a constant field, Eq.3.17 does not guarantee that $\nabla^{sph} f = 0$. Let us consider the following identity:

$$\nabla_{\vec{x}} f = \nabla_{\vec{x}} (1.f) - f \nabla_{\vec{x}} (1) \tag{3.18}$$

Applying the SPH approximation in Eq.3.17 to Eq.3.18, we obtain a second approximation for the derivative of f:

$$\nabla_{\vec{x}}^{sph2} f_i = \sum_{j \in D_i} \frac{m_j}{\rho_j} \left(f_j - f_i \right) . \nabla_i W_{ij} \tag{3.19}$$

This new formulation guarantees that $\nabla^{sph} f = 0$ for constant fields $f = 0$. But, the formulation in Eq.3.19 is not conservative when applied to the linear momentum equation.

A last conservative formulation was proposed by Monaghan [94]. Let us consider the following equation:

$$\frac{\nabla_{\vec{x}} f}{\rho} = \nabla_{\vec{x}} \left(\frac{f}{\rho} \right) + \frac{f}{\rho^2} \nabla_{\vec{x}} \rho \tag{3.20}$$

Applying the SPH approximation in Eq.3.17 to Eq.3.20, we obtain a third approximation for the derivative of the function f:

$$\nabla_{\vec{x}}^{sph3} f_i = \rho_i \sum_{j \in D_i} m_j \left(\frac{f_j}{\rho_j^2} + \frac{f_i}{\rho_i^2} \right) . \nabla_i W_{ij} \tag{3.21}$$

Convergence and renormalization techniques

It is well known from previous studies [149, 150, 104] that the particle approximations are exact only if the condition $\frac{h}{dx} \to \infty$ is satisfied. Numerically, it would require an infinite number of particles which is computationally impossible. It is a severe limitation to the method because the consistency and the convergence are not guaranteed and thus it affects the accuracy of the standard SPH method.

In order to improve the standard SPH solution a renormalization technique is introduced.

Even for disordered particles, the derivative of constant functions can be represented exactly (far from a boundary) by the standard SPH using the symmetrical approximation for the gradient operator in Eq.3.19. However, it is not the case for linear functions. In order to improve the accuracy of the SPH Method a re-normalized operator has been introduced by Randles and Libersky [121]. The employed technique enforces the exact representation of constant and linear functions and its first derivative through the introduction of renormalization matrices B_i (for a given particle i), which correct the first derivative of the kernel function $\nabla_i W_{ij}$ in Eq.3.17, Eq.3.19 and Eq.3.21. The re-normalized SPH formulation for the second approximation of the first derivative (Eq.3.19) of the function $\nabla_{\overrightarrow{x}}^{Ran} f_i$ is given by:

$$\nabla_{\overrightarrow{x}}^{Ran} f_i = \left[\sum_{j \in D_i} \frac{m_j}{\rho_j} (f_j - f_i).\nabla_i W_{ij} \right] : B_i \tag{3.22}$$

Enforcing the condition the exact particle approximation for the function $f = a\overrightarrow{x} + b$:

$$a : I = a \left[\sum_{j \in D_i} \frac{m_j}{\rho_j} \overrightarrow{x_{ij}}.\nabla_i W_{ij} \right] : B_i \tag{3.23}$$

We identify the matrix B_i to be:

$$B_i = \left[\sum_{j \in D_i} \frac{m_j}{\rho_j} \overrightarrow{x_{ij}}.\nabla_i W_{ij} \right]^{-1} \tag{3.24}$$

In addition to the improvement of the consistency of the method, Lanson [75] and Vila [150] have also shown that the renormalization technique relax the convergence criterion $\frac{h}{dx} \to \infty$ to be $\frac{h}{dx} \to O(1)$, which improves the global convergence of the method.

3.1.2   SPH approximation of the Euler equations

Conservation equations

In the previous subsection, we defined the basic concepts of the SPH method and three approximations for the first derivative of a function f. The second approximation satisfies the principle of action-reaction, but fails to conserve the linear momentum, whereas the third approximation is the opposite of the second. In this section, we derive the SPH discrete compressible Euler equations of conservation through the use of Eq.3.19 and Eq.3.21. In LS-DYNA, the WC-SPH approximation for the Euler equations is given by:

- *WC-SPH conservation of mass:*

$$\frac{d\rho_i}{dt} = \rho_i \sum_{j \in D_i} m_j \left(\overrightarrow{v_j} - \overrightarrow{v_i} \right).B_i \nabla_i W_{ij}, \tag{3.25}$$

- *WC-SPH conservation of momentum:*

$$\frac{d\overrightarrow{v_i}}{dt} = \rho_i \sum_{j \in D_i} m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) B_i \nabla_i W_{ij}, \tag{3.26}$$

- *WC-SPH conservation of the energy:*

$$\frac{de_i^{int}}{dt} = \frac{1}{2} \sum_{j \in D_i} m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij} \right) \left(\overrightarrow{v_i} - \overrightarrow{v_j} \right).B_i \nabla_i W_{ij}, \tag{3.27}$$

where B_i is the particle i renormalization matrix (Eq.3.24) and Π_{ij} is an added dissipation pressure term, similar to the bulk viscosity q in FEM, in order to treat problems involving discontinuities in the flow variables such as shock waves. The additional dissipative pressure term Π_{ij} was proposed by Monaghan and Gingold [97] and is systemically used to add stability to the numerical scheme even for weakly-compressible applications.

Monaghan and Gingold (1983) artificial viscosity

The Monaghan-type artificial viscosity Π_{ij} for WC-SPH is equivalent to the Von-Neumann and Richtmeyer bulk viscosity for FEM in the sense that it is activated only in compression for approaching particles and that it converts the kinetic energy into heat in the shock layer.

The absence of a mesh in SPH methods subjects it to potential unphysical penetrations of approaching particles from different materials. Monaghan-type viscosity [97] prevents such unphysical penetration that makes the simulation unstable, and which the majority of the time, makes the particles blow and effectively ends the simulation. Continuum and particle approaches for the derivation of the term Π_{ij} are provided in [32].

Let us denote by $c_{ij} = 1/2(c_i + c_j)$, $\rho_{ij} = 1/2(\rho_i + \rho_j)$ and $h_{ij} = 1/2(h_i + h_j)$ the mean speed of sound, the mean density and the mean smoothing length between two particles i and j, respectively. The artificial viscosity Π_{ij} is defined by:

$$
\Pi_{ij} = \begin{cases} \dfrac{-\alpha_\Pi c_{ij}\phi_{ij} + \beta_\Pi \phi_{ij}^2}{\rho_{ij}} & \vec{v_{ij}}.\vec{x_{ij}} < 0 \\[2em] 0 & \vec{v_{ij}}.\vec{x_{ij}} \geq 0 \end{cases}, \tag{3.28}
$$

where $\vec{v_{ij}} = \vec{v_i} - \vec{v_j}$, $\vec{x_{ij}} = \vec{x_i} - \vec{x_j}$ and:

$$
\phi_{ij} = \frac{h_{ij}\,\vec{v_{ij}}.\vec{x_{ij}}}{|x_{ij}|^2 + \varphi^2} \tag{3.29}
$$

α_Π and β_Π in Eq.3.28 are the linear and the quadratic coefficients, which are usually set to 1.0 and 2.0, respectively. $\varphi = 0.1 h_{ij}$ is introduced to avoid dividing by zero (machine order).

Variable smoothing length

The smoothing length h is a key parameter in the SPH method. In the absence of a mesh, the support domain D_i of a particle i that contains the interacting j particles is centered in i and has a radius of κh_i. Thus, it is entirely defined

by the smoothing length h_i of the particle i and the chosen kernel function (κ parameters). Since the method is Lagrangian and because the particles are not linked by FE mesh, a law of evolution of the smoothing is needed in order to update it such that the number of particles remains as much constant as possible and that the stability of the method is not affected.

The more a region is dense (high density) the more particles are present. Thus, a simple way to update the smoothing length is to define it as a function of the average density:

$$h = h_0 \left(\frac{\rho_0}{\rho} \right)^{1/d},$$ (3.30)

where h_0, ρ_0 and d are the initial smoothing length, the initial density and the dimension of the problem, respectively.
In LS-DYNA, the smoothing length is updated according to Benz [19] integrating in time the smoothing length as follows:

$$\frac{dh(t)}{dt} = -\frac{1}{d} \frac{h(t)}{\rho} \frac{d\rho}{dt}$$ (3.31)

The initial smoothing length h_0 is computed as follows in LS-DYNA:

1. For each particle i, compute the distance $d_i = \min_i d_{ij}$, where d_{ij} is the distance between the particles i and j.

2. Compute the distance $d_0 = \max_i d_i$.

3. Compute the initial smoothing length $h_0 = 1.23 d_0$.

Neighbors search (NS) algorithms

For a particle i of smoothing length h_i, only the particles j in a radius of κh_i interact with the particle i and are integrated in the discrete sums in Eq.3.25, Eq.3.26 and Eq.3.27. Due to the Lagrangian and mesh-less nature of the method, the neighbors search step is repeated at each time step. The neighbors search (NS) is the most CPU time consuming task. Thus, direct brute force search

methods, which is of the order of $O(N_{sph}^2)$ (where N_{sph} is the total number of SPH particles) is simply impractical.

For weakly-compressible applications, the density is almost constant and therefore it is the same for the smoothing length according to Eq.3.30 and Eq.3.31. A **bucket sort** algorithm is used in LS-DYNA [48], where the spatial domain is split into cubic boxes for 3-D problems (resp. square cells for 2-D problems) where an edge size of κh is suited for such problems with a nearly constant smoothing length. For a given particle i, the neighbors search is limited to the box in which it is located and its direct 26 neighbor boxes for 3-D problems (resp. 8 cells for 2-D) problems. The neighbors search area is truncated compared to brute force algorithm and a substantial computational time is therefore spared.
For applications with a great variation of smoothing length such us blasts in air or underwater explosions, the density variation is such that in the expansion area the particles smoothing length can be two or even three times greater than its initial value. In these particular cases, the smoothing length may cover more than one direct neighboring box and neighbors are missing in the support domain. **Tree-search** algorithms are preferred in such cases.

Boundary conditions: Ghost particle method

The ghost particle method is a robust and easy to implement method to model solid wall boundary conditions with either slip or stick boundary conditions at the wall (see [28, 35, 104]). When a particle i is close to the boundary (orthogonal distance lesser than κh_i), it is symmetrized across the boundary, which gives arise to a new ghost particle G_i. Density, pressure and internal energy values are copied from the parent particle i to the ghost child particle G_i (see Fig.3.3). The velocity of the ghost particle G_i is defined by:

- *Slip boundary conditions:*

$$\overrightarrow{v_{G_i}} = \overrightarrow{v_i} - 2.(\overrightarrow{v_i}.\overrightarrow{n})\overrightarrow{n}, \tag{3.32}$$

Figure 3.3 – Creation of ghost particles [104].
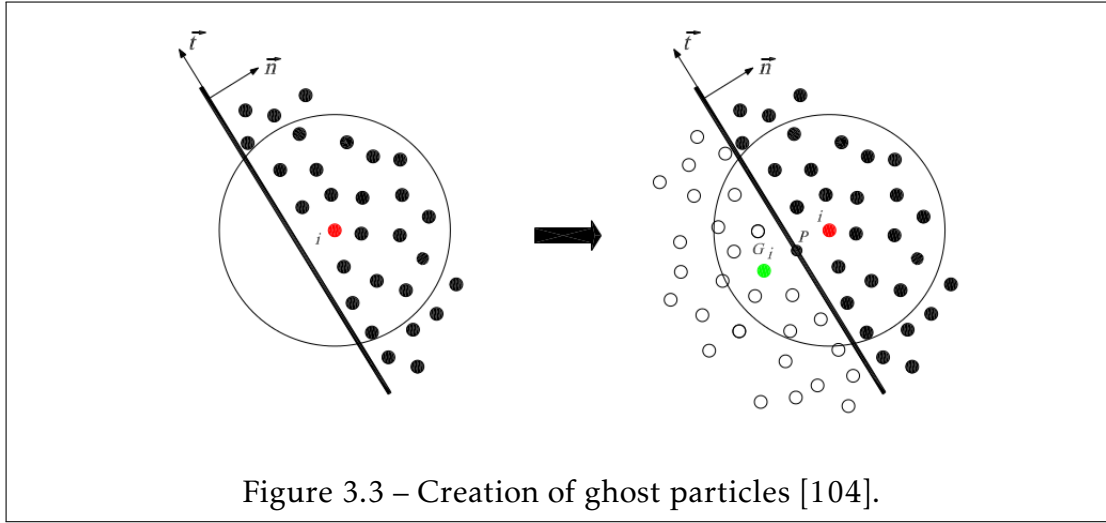
- *Slick boundary conditions:*

$$\overrightarrow{v_{G_i}} = -\overrightarrow{v_i},\qquad(3.33)$$

where $\overrightarrow{n}$ is the normal to the wall boundary.

Using the ghost particle method, mathematical consistency is restored at the boundary for particle with a truncated domain. A more general **multiple tangent boundary method** is proposed by Yildiz [160] to deal with complex boundaries.

Time integration and time step control

The explicit Leapfrog time integration method described in section 2.1 (Eq.2.14, Eq.2.15 and Eq.2.16) is also used for the evolution in time of the SPH discrete conservation variables (Eq.3.25, Eq.3.26 and Eq.3.27) and the smoothing length (Eq.3.31).

The stability CFL condition for the explicit WC-SPH scheme is given by:

$$\Delta t_{cpu} \leq \Delta t_i^{crit} = \frac{h_i}{c_i},\qquad(3.34)$$

where Δt_i^{crit} is the critical time step of the particle i. It is defined as the ratio between smoothing length h_i of the particle and speed of sound c_i.

In practice, the critical time step Δt_i^{crit} is computed for all the SPH particles ($i = 1, N_{sph}$) and the computational time step is taken as:

$$\Delta t_{cpu} = fac. \min_i(\Delta t_i^{crit}), \tag{3.35}$$

where fac is a scaling factor taken between 0 and 1.

3.1.3 Simulation of UNDEX with FSI using LS-DYNA SPH-FEM coupling

Experimental tests for explosive detonation in fluid and the impact on surrounding structures, are costly to perform. The ALE formulation is considered as a reference solution to validate the SPH formulation since it has been validated against experiments in many applications involving explosions, blast impact and shock waves [77, 12].

For this problem ALE multi-material and SPH formulation are used to solve the problem up until the physical termination time. In order to compare ALE and SPH formulations and to check the limits of the SPH formulation. In this example, we consider a simple structure in water subjected to blast loading generated through the detonation of explosive material. The FEM Structure and ALE Fluids are modeled using eight-node solid elements, whereby the SPH elements are generated at the center of the ALE elements. A sketch of the problem is illustrated in Fig.3.4.

Compressible Euler equations are solved for the fluid parts with the JWL EOS (Eq.1.100) for the explosive and MG EOS (Eq.1.106 and Eq.1.107) for the water. The structure part is modeled the with the elastic-plastic material model described in subsection 1.3.1. The initial parameters are given in Tab.3.1, Tab.3.2, Tab.3.3 and Tab.3.4 (cm, g, ms and Pa).

Density	Detonation velocity	Chapman-Jouget Pressure
1.7	0.753	0.255

Table 3.1 – LS-DYNA material model parameters for the explosive

A	B	R1	R2	ω	E0	V0
5.4094	0.0937260	4.5	1.1	0.35	0.08	0.0

Table 3.2 – LS-DYNA JWL EOS parameters for the explosive

Density	Cut-off pressure	Dynamic viscosity
1.0	-1.0E-6	0.0

Table 3.3 – LS-DYNA material model parameters for the water

C	S1	S2	S3	γ_0	a	E0	V0
0.1484	1.979	0.0	0.0	0.11	3.0	0.0	0.0

Table 3.4 – LS-DYNA MG EOS parameters for the water
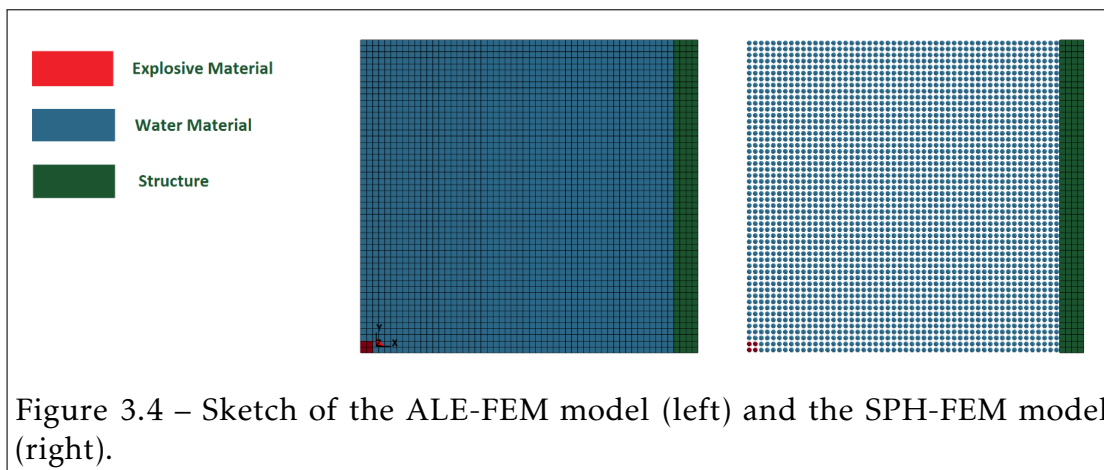

Figure 3.4 – Sketch of the ALE-FEM model (left) and the SPH-FEM model (right).

The FSI is modeled for both ALE-FEM and SPH-FEM problems using a kinematic type contact algorithm at the fluid structure interface. The kinematic constraint-based method (subsection 2.5.1) is used where constraints are imposed on the displacement and the velocity of the contact interface, as well as impenetrability condition. In addition, a smoothing algorithm is used with the ALE formulation, which constrains the nodes to move uniformly along straight lines, to overcome high mesh distortion problems preserving mesh integrity.

Slip boundary conditions are applied to the ALE elements at the top, left and

bottom boundaries, whereas the structure is fixed at the top and constrained to move in the X-direction at the bottom. The definition of proper boundary conditions for WC-SPH formulation is one of the great challenges in the SPH theory. Several techniques have been developed in order to enhance the desired conditions, to stop particle from penetrating solid boundaries and also to complete the kernel function that is truncated by the physical domain for a particle close to the boundary. The ghost particle method described in the previous subsection is used for its robustness and accuracy.

As recommended in [83], the quadratic coefficient $\beta_\Pi = 10.0$ is voluntarily chosen to be a much more bigger value than the commonly used value $\beta_\Pi = 2.0$ to prevent unphysical penetration of explosive particles in water, as the linear coefficient $\alpha_\Pi = 1.0$. This inter-penetration of particles leads to oscillations and instabilities that blow up the particle and corrupt the computation. This strategy of tuning the parameter β_Π is problem dependent and is a severe limitation to the method when applied to engineering problems. In the next subsection, we see that FC-SPH schemes overcome this limitation and offer a stable and robust SPH numerical solution for shock propagation in multiphase media.

Two simulations were performed to investigate the accuracy of the WC-SPH formulation in comparison to ALE formulation. The first simulation uses the same number of elements for both ALE and SPH methods, such that the space step dx separating two particles in the SPH case or the length of an element in the ALE case is the same. In the second simulation, the WC-SPH model is refined by two in both directions in order to improve the accuracy of the simulation and to see how does the WC-SPH solution behave compared to the ALE solution. In both simulations, WC-SPH particles are generated at the center of the ALE elements, as it is shown in Fig.3.5.

To illustrate pressure wave propagation through water material, Fig.3.6 shows the pressure fringe at time $t = 60ms$ from both ALE and non-refined WC-SPH simulations. At this time of the simulation, the structure is not deformed, since the shock wave has not yet reach the structure, and it can be seen that the WC-SPH formulation effectively qualitatively represents the physics as the shock

Figure 3.5 – SPH particles generation for the non-refined simulation (left) and the refined simulation (right).
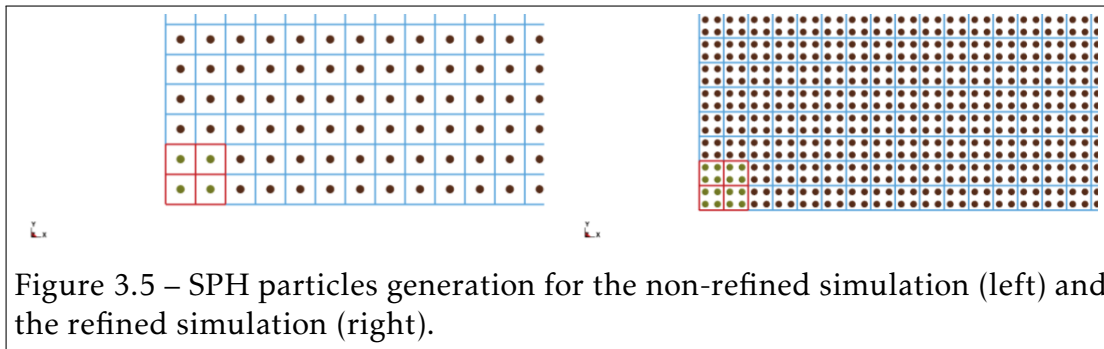
front and the expansion waves have the same shape in both ALE and non-refined WC-SPH solutions.

In order to quantitatively compare the solutions, the structure is examined where the X-displacement and the X-velocity time curve of a node on the structure are plotted in Fig.3.7 and Fig.3.8. The Von-Mises Stress time curve of an FEM element on the structure is plotted in Fig.3.9 for both ALE and non-refined WC-SPH formulations up to time $t = 130ms$. In an analysis of the ALE and WC-SPH curves, we can clearly see that is physics is well-modeled as the two solutions have the same shape, although they are not fitting in term of:
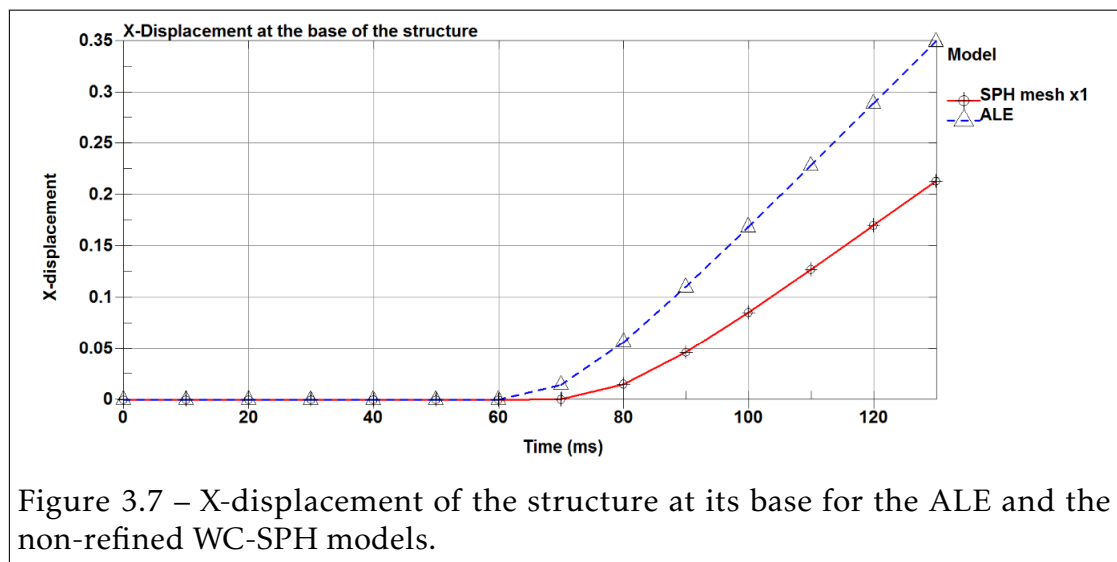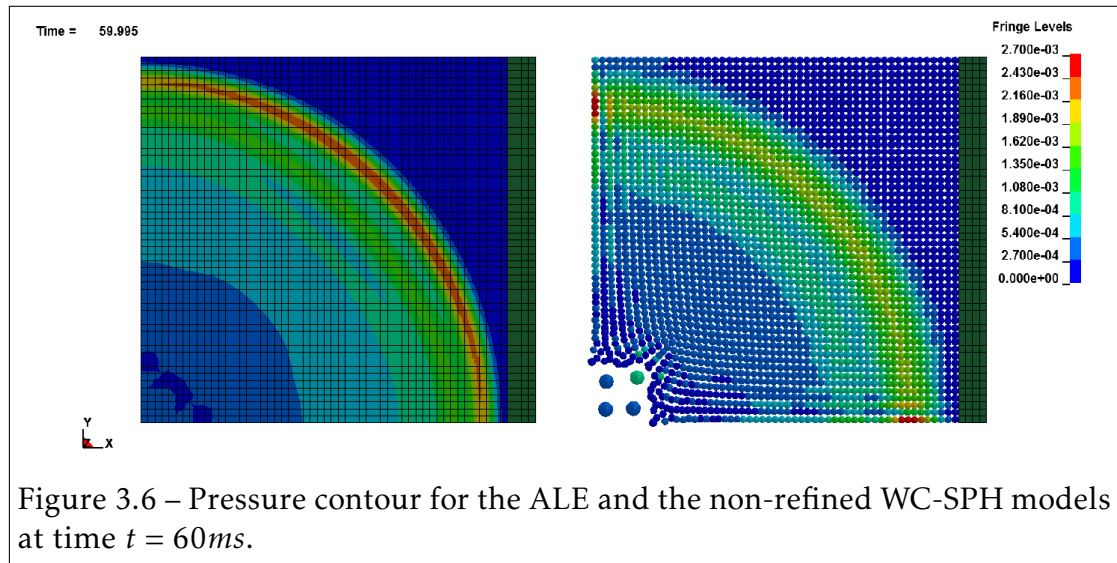
- **Impact time:** $60ms$ and $70ms$ for ALE and WC-SPH, respectively.

- **Impact power:** The maximum velocity and Von-Mises stresses are smaller in WC-SPH results than in ALE results.

For the refined WC-SPH model, the pressure fringe is plotted at time $t = 60ms$ in Fig.3.10, which shows the propagation of the shock wave in the water. Fig.3.11, Fig.3.12, Fig.3.13 and Fig.3.14 show a good correlation between the two results and that the accuracy of the method has improved refining the WC-SPH. It can be observed that in practice the WC-SPH solution is more CPU time consuming than ALE methods, as we require a smaller space step to reach the accuracy of ALE methods.

However, it is important to highlight the ability of the WC-SPH method, which due to its pure Lagrangian nature, can naturally handle complex flows involving

multiple materials. In the case of ALE formulation, it is needed to add to the mixture theory and VOF interface tracking algorithm methods that are much more complex to implement than WC-SPH methods.

Non-refined WC-SPH results

Figure 3.6 – Pressure contour for the ALE and the non-refined WC-SPH models at time $t = 60ms$.

Figure 3.7 – X-displacement of the structure at its base for the ALE and the non-refined WC-SPH models.

Figure 3.8 – X-Velocity of the structure at its base for the ALE model and the non-refined WC-SPH models.

Figure 3.9 – Von Mises stress of the structure at its center for the ALE and the non-refined WC-SPH models.

Refined WC-SPH results

Figure 3.10 – Pressure contour of the ALE model and the refined WC-SPH models at time $t = 60ms$.

Figure 3.11 – Von Mises stress of the FEM structure subjected to the shock wave in the ALE and the refined WC-SPH models at time $t = 130ms$.

Figure 3.12 – X-displacement of the structure at its base for the ALE and the refined WC-SPH models.

Figure 3.13 – X-Velocity of the structure at its base for the ALE and the refined WC-SPH models.

Figure 3.14 – Von Mises stress of the structure at the middle for ALE and refined WC-SPH models.

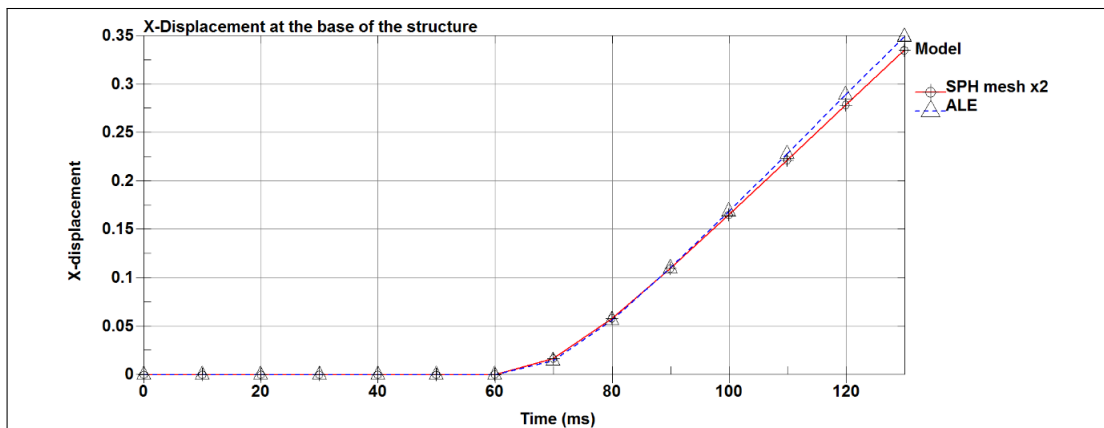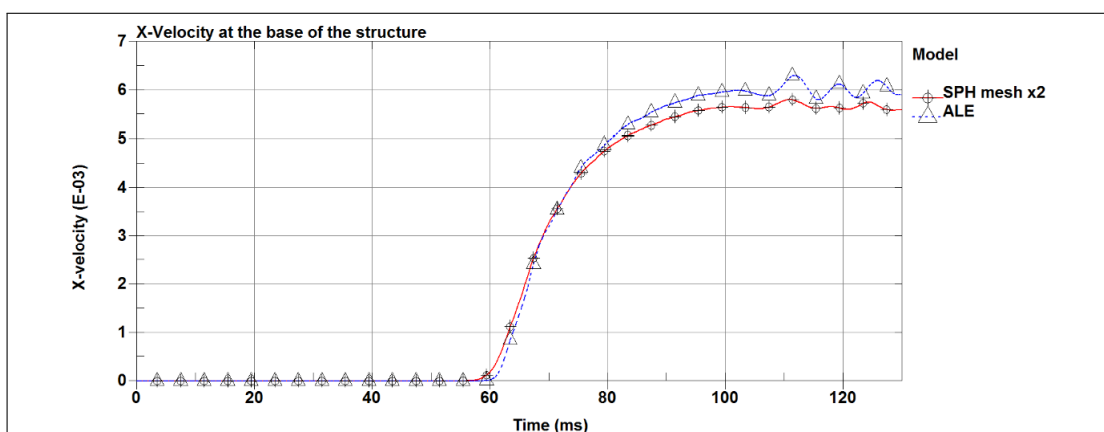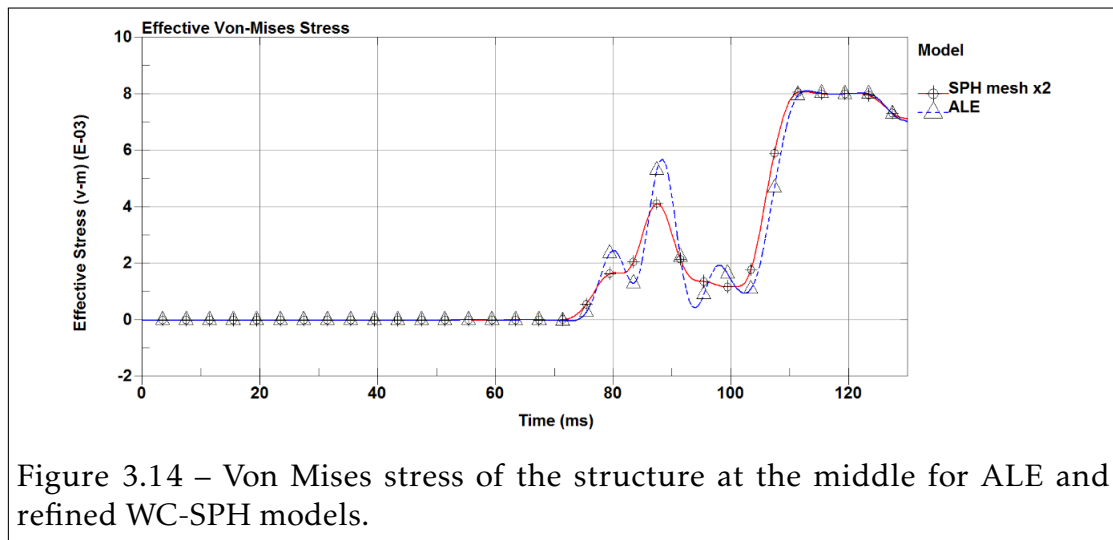### 3.1.4   Limitations of the WC-SPH method for shocks

In the absence of a FE mesh, the smoothing length h_i is the space variable that defines the support domain for a particle i. A key assumption in the WC-SPH method is the omission of the smoothing length space variable in the derivation of the kernel in Eq.3.16. For weakly compressible such as dam breaks, the density is almost constant and according to Eq.3.30 and Eq.3.31, the smoothing length is almost constant as well. So the assumptions of considering that the kernel is only a function of the space variables $\overrightarrow{x}$ stands.

In the fully compressible regime, the WC-SPH method introduces instabilities in the contact discontinuity between materials with different densities for a system of particles with non-equal masses. The higher the density ratio is between two materials in contact, the higher the amplitudes of numerical oscillations are at the contact discontinuity. In the previous simulation, the density ratio of 1.7 is acceptable. Thus, the standard WC-SPH method can solve UNDEX problems involving FSI through the use of the advanced FSI techniques implemented in the commercial code.

In order to show the limits of the method for compressible multiphase flow

problems involving shocks, let us consider the classical 1-D Sod shock tube problem with a density ration of 4.

Problem description:

Two fluids with different densities, pressures and internal energies that are initially separated are put into contact. Two waves are generated: an expansion wave traveling from the low pressure to the high pressure region and a shock wave traveling the opposite direction. The initial conditions are given in Tab.3.5.

	p	ρ	e^{int}	vx	γ
Left region	1.0	1.0	2.5	0.0	1.4
Right region	0.1795	0.25	1.795	0.0	1.4

Table 3.5 – IG EOS parameters for the left and right regions

Mesh and Geometry:

The meshes consist of $1200 \times 1 \times 1$ elements for ALE simulation and $1200 \times 1 \times 1$ particles for the WC-SPH simulation, both covering the domain:

$$\begin{aligned} -0.6 &\leq x \leq 0.6 \\ -0.01 &\leq y \leq 0.01 \\ -0.01 &\leq z \leq 0.01 \end{aligned}$$

At time $t = 0.2$, the pressure and the density profiles for both ALE and WC-SPH simulations are compared to the exact solution in Fig.3.15 and Fig.3.16, respectively. The location of the shock, the location of both the tail and head of the rarefaction of the ALE simulation match quantitatively to the exact solution. The expansion and shock waves are reproduced accurately, but instabilities are clearly highlighted in the contact discontinuity region.

The Pressure results for the similar Sod shock tube test case with a density ratio of 8 is shown in Fig.3.17. This tendency is amplified for higher density ratio is

confirmed.

Conclusion:

It is shown that the straightforward use of the WC-SPH formulation is unsat-isfactory for the simulation of multiphase problems with high density ratio. **SPH-to-SPH coupling method** [159] was developed in LS-DYNA to overcome this limitation. However, the method is not robust, as the contact stiffness is a problem depending parameter that is hard to a priori define. No robust general rule exists to define the contact stiffness, which makes the method difficult to use in an industrial context. Another important aspects to investigate, are the effects of the coefficients α_Π and β_Π in the Monaghan-type artificial viscosity. In the classical dam break example, Cueille [32] showed that the stability is not guaranteed for all α_Π and β_Π values. It was suggested to consider $\alpha_\Pi \sim 1$ and $\beta_\Pi = \alpha_\Pi/2$ but with no guarantee on the consistency of the WC-SPH results. In the next section, robust fully-compressible SPH schemes following the philoso-phy of the works from Price [113, 22, 115] are developed and validated against ALE results for multiphase flow shock problems with high density ratio (up to 200).

Figure 3.15 – Pressure along the tube length for a density ratio of 4 at time t=0.2. Exact solution is in red, ALE solution is in green with crosses and the WC-SPH solution is in blue.
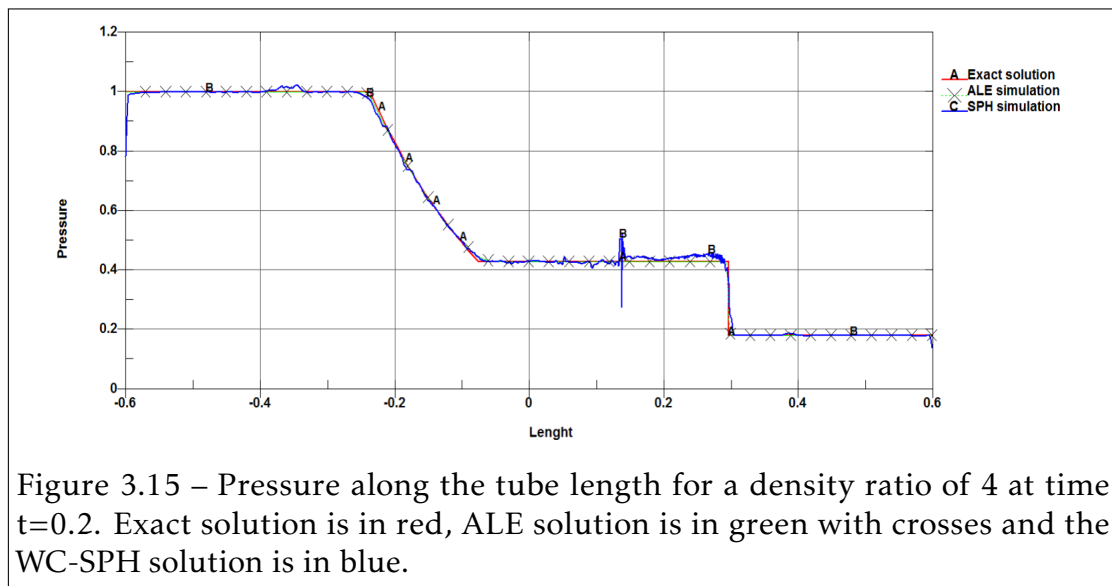
Figure 3.16 – Density along the tube length for a density ratio of 4 at time t=0.2. Exact solution is in red, ALE solution is in green with crosses and the WC-SPH solution is in blue.
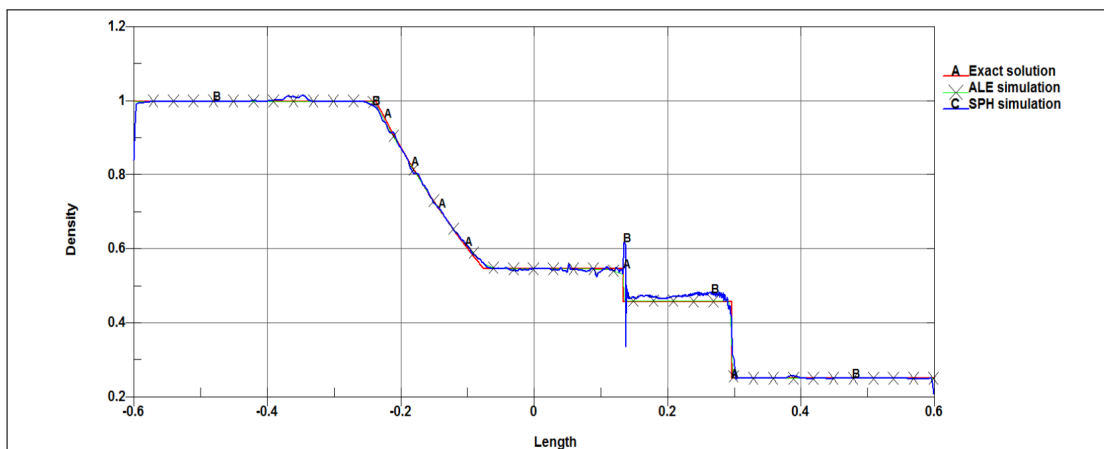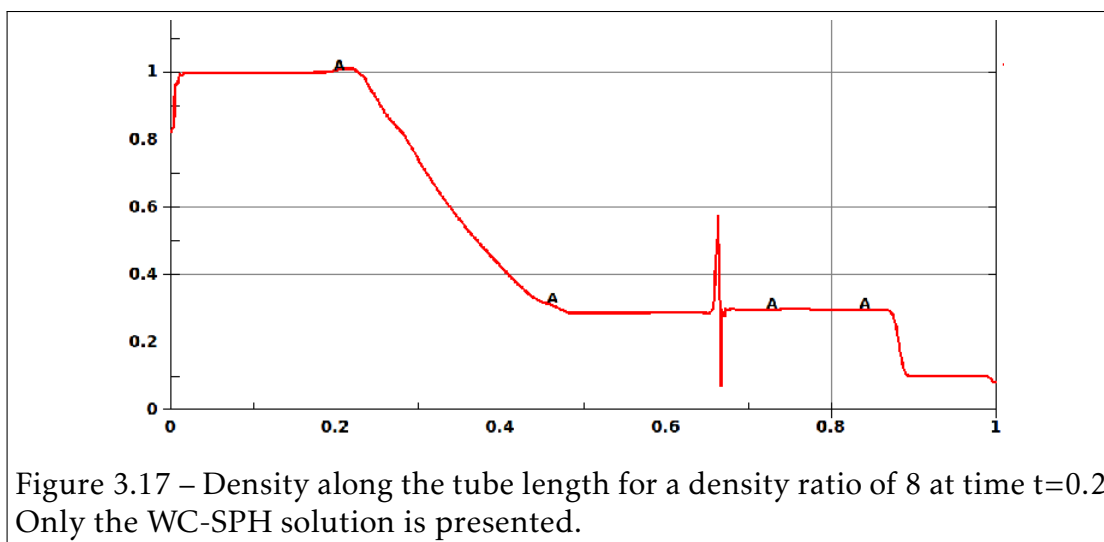

Figure 3.17 – Density along the tube length for a density ratio of 8 at time t=0.2. Only the WC-SPH solution is presented.

3.2 Fully-Compressible SPH (FC-SPH) schemes

In the previous section, the enhanced WC-SPH schemes implemented in the vast majority of commercial codes for modeling weakly compressible fluid flows are based on the following two key steps:

- Derivation of the gradient operator of the kernel considering that the smoothing length is constant (Eq.3.16).

- Correction of the gradient operator through renormalization techniques (Eq.3.24).

In this section, we present the extension of WC-SPH schemes to FC-SPH schemes: the **mass-based scheme** from Price [113] and the extended **number-density scheme** from [54] to fully-compressible regime for handling shock problems with high density ratio of the order of $100 - 200$. The two schemes were selected among a selection of schemes previously investigated by Zisis et al. [168].
It is shown through multiphase 1D and 2D shock problems, that the fully compressible number density scheme provides a robust and stable solution. An additionnal artificial dissipation term proposed by [168] is added to the mass conservation equation to increase the overall stability. For all the following test cases, we decided to keep the same input parameters and to use only the truncated Gaussian kernel to prove the robustness of the FC-SPH schemes.

3.2.1 Density estimate

In the FC-SPH formulation from Price [115], the key step is the **density estimate**. Using the Lagrangian, consistent discrete equations of motion are derived from the density estimate. In theory, the consistency of the numerical scheme implies that a good density estimate will lead to a good particle approximation. Thus, the equations of motion that are solved must be consistent with the density estimate equation.
Improving the density estimate is a major challenge in Lagrangian SPH methods. It was seen in section 2.2.1 that in Lagrangian FEM method the mass conservation equation (Eq.3.17) is solved algebraically, and thus the density computation is

trivial. The difficulty in the estimation of the SPH density is due to the absence of a FE mesh. Serrano et al. [129] and Heb and Springel [53] use dual Delaunay or Voronoi tessellation to obtain a better estimation of the density.

To overcome the limitation of the WC-SPH, we consider the smoothing length to be variable and linked to density in the fully compressible regime. The evaluation of the first derivative of the kernel function is done with respect to both space variables: the position vector $\overrightarrow{x}$ and the smoothing length h.

For a given particle i, a standard method to link the smoothing length h_i to density ρ_i is obtained by:

$$\rho_i h_i^3 = constant, \tag{3.36}$$

where the constant has a unit of mass. To introducing the mass m in the constant, the smoothing length is coupled to the density as follows:

$$h(\rho(t)) = \eta \left(\frac{\rho(t)}{m} \right)^{\frac{1}{d}} \tag{3.37}$$

where d is the dimension of the problem and η is a coupling constant with the optimal reported value $\eta = 1.2$ [95, 114].

The time derivative of the kernel function is given by:

$$\frac{dW_{ij}}{dt} = \frac{dW_{ij}(h(\rho(t)))}{dt} = (\overrightarrow{v_i} - \overrightarrow{v_j}).\nabla_i W_{ij}(h_i) + \frac{\partial W_{ij}}{\partial h_i} \frac{dh_i}{dt}, \tag{3.38}$$

where:

$$\frac{dh_i}{dt} = \frac{\partial h_i}{\partial \rho_i} \frac{d\rho_i}{dt}, \tag{3.39}$$

Mass-based scheme: m-scheme

A first integral form of the discrete SPH density is obtained by replacing the function f in Eq.3.11 by the density ρ:

$$\rho_i = \sum_{j \in D_i} m_j W_{ij}(h_i(\rho_i)) \tag{3.40}$$

Since the smoothing length h_i is itself a function of the density ρ_i, solving Eq.3.41 would require an iterative solver and can be CPU time expensive.

An alternative way is to consider the differential form of Eq.3.41. To Take the time derivative of the density in Eq.3.41, one has:

$$\frac{d\rho_i}{dt} = \sum_{j \in D_i} m_j \frac{dW_{ij}(h_i)}{dt} \tag{3.41}$$

Substituting Eq.3.38 in Eq.3.41, we finally obtain the differential form for the density estimate:

$$\frac{d\rho_i}{dt} = \frac{1}{\Omega_i} \sum_{j \in D_i} m_j (\vec{v_i} - \vec{v_j}).\nabla_i W_{ij}(h_i), \tag{3.42}$$

where

$$\Omega_i = 1 - \frac{\partial h_i}{\partial \rho_i} \sum_{j \in D_i} m_j \frac{\partial W_{ij}(h_i)}{\partial h_i} \tag{3.43}$$

The Ω-factor is a correction factor related to the compression state of the particle. We can remark that for the particular case $\Omega = 1$ Eq.3.43 is equivalent to the discrete WC-SPH mass conservation Eq.3.25.

Number-density scheme: n-scheme

An alternative integral form for density estimate in the weakly-compressible was provided by Hu and Adams [54] and is based on number-density. This new formulation takes out from the sum over the neighbors, the mass variable in order to reduce the instabilities at the contact discontinuity when modeling multiphase flows with different densities. The integral form of the number-density estimate is given by:

$$\rho_i = m_i \sum_{j \in D_i} W_{ij}(h_i(\rho_i)) \tag{3.44}$$

The differential form is obtained taking the time derivative of Eq.3.44:

$$\frac{d\rho_i}{dt} = m_i \sum_{j \in D_i} \frac{dW_{ij}(h_i)}{dt} \tag{3.45}$$

Substituting Eq.3.38 in Eq.3.45, we have:

$$\frac{d\rho_i}{dt} = \frac{m_i}{\Omega_i} \sum_{j \in D_i} (\vec{v_i} - \vec{v_j}).\nabla_i W_{ij}(h_i), \tag{3.46}$$

where:

$$\Omega_i = 1 - m_i \frac{\partial h_i}{\partial \rho_i} \sum_{j \in D_i} \frac{\partial W_{ij}(h_i)}{\partial h_i} \tag{3.47}$$

3.2.2 FC-SPH equations of motion and energy

In this subsection, we present another approach to derive the discrete SPH equations of motion based on the Lagrangian formalism that remains fully conservative. Let us consider the **Lagragian function** defined as the difference between the kinetic and the internal energies per unit mass such that:

$$\mathcal{L} = \int_\Omega \rho \left(\frac{1}{2} \vec{v}.\vec{v} - e^{int}(\rho,s) \right) d\vec{x} \tag{3.48}$$

Replacing the continuous integral in Eq.3.48 by a discrete Riemann sum, we have:

$$\mathcal{L} = \sum_j m_j \left(\frac{1}{2} \vec{v_j}.\vec{v_j} - e^{int}(\rho_j,s_j) \right), \tag{3.49}$$

where the sum j is over all the particles of the domain Ω.
Let us now consider the Euler-Lagrange equation given by:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \vec{v_i}} \right) - \frac{\partial \mathcal{L}}{\partial \vec{x_i}} = 0 \tag{3.50}$$

The first term in Eq.3.50 is given by:

$$\frac{\partial \mathcal{L}}{\partial \vec{v_i}} = \frac{\partial}{\partial \vec{v_i}} \sum_j m_j \left(\frac{1}{2} \vec{v_j}.\vec{v_j} - e^{int}(\rho_j,s_j) \right) = m_i \vec{v_i} \tag{3.51}$$

The second term in Eq.3.50 is given by:

$$\frac{\partial \mathcal{L}}{\partial \overrightarrow{x_i}} = -\sum_j \left[\frac{\partial e_j^{int}}{\partial p_j} \frac{\partial p_j}{\partial \overrightarrow{x_i}} + \frac{\partial e_j^{int}}{\partial \rho_j} \frac{\partial \rho_j}{\partial \overrightarrow{x_i}} \right] \tag{3.52}$$

$$\frac{\partial \mathcal{L}}{\partial \overrightarrow{x_i}} = -\sum_j \underbrace{\left[\frac{\partial e_j^{int}}{\partial p_j} \frac{d p_j}{d \rho_j} + \frac{\partial e_j^{int}}{\partial \rho_j} \right]}_{= \, p_j/\rho_j^2 \text{ due to Eq.1.97}} \frac{\partial \rho_j}{\partial \overrightarrow{x_i}} \tag{3.53}$$

Finally the second term is given by:

$$\frac{\partial \mathcal{L}}{\partial \overrightarrow{x_i}} = -\sum_j \frac{p_j}{\rho_j^2} \frac{\partial \rho_j}{\partial \overrightarrow{x_i}} \tag{3.54}$$

The discrete FC-SPH equations of motion are obtained combining Eq.3.50, Eq.3.51 and Eq.3.52 and the appropriate density estimate (Eq.3.40 or Eq.3.46) to compute the spatial derivative of the density.

Mass-based scheme: *m-scheme*

The spatial derivative of the density for the *m-scheme* using Eq.3.40 is given by:

$$\frac{\partial \rho_j}{\partial \overrightarrow{x_i}} = \frac{1}{\Omega_j} \sum_k m_k \nabla_j W_{jk}(h_j)(\delta_{ij} - \delta_{ik}) \tag{3.55}$$

The equation of motion for the *m-scheme* is:

$$\frac{d \overrightarrow{v_i}}{dt} = -\sum_{j \in D_i} m_j \left(\frac{p_i}{\Omega_i \rho_i^2} \nabla_i W_{ij}(h_i) + \frac{p_j}{\Omega_j \rho_j^2} \nabla_i W_{ij}(h_j) \right), \tag{3.56}$$

where Ω is defined by Eq.3.43.

Number-density scheme: *n-scheme*

The spatial derivative of the density for the *n-scheme* using Eq.3.44 is given by:

$$\frac{\partial \rho_j}{\partial \overrightarrow{x_i}} = \frac{m_j}{\Omega_j} \sum_k m_k \nabla_j W_{jk}(h_j)(\delta_{ij} - \delta_{ik}) \tag{3.57}$$

The equation of motion for the *n-scheme* is:

$$\frac{d\overrightarrow{v_i}}{dt} = -\frac{1}{m_i} \sum_{j \in D_i} \left(\frac{p_i V_i^2}{\Omega_i} \nabla_i W_{ij}(h_i) + \frac{p_j V_j^2}{\Omega_j} \nabla_i W_{ij}(h_j) \right), \tag{3.58}$$

where Ω is defined by Eq.3.47 and V is the volume for a particle defined by $V = m/\rho$.

Energy equation

Combining the conservation equations of mass and energy, we have:

$$\frac{de_i^{int}}{dt} = \frac{p_i}{\rho_i^2} \frac{d\rho_i}{dt} \tag{3.59}$$

Once the FC-SPH discrete time derivative of the density is computed (Eq.3.42 or Eq.3.46), the time derivative of the internal energy is obtained algebraically by solving Eq.3.59.

3.2.3   Monaghan (1997) Artificial viscosity and time step control

Artificial viscosity compensates for errors caused by the subtle assumption of a differentiable Lagrangian function [114, 115]. A generic form of artificial diffusion, based on the Riemann solvers for each conserved variable appears in [96, 95, 114].

The artificial viscosity terms added to the momentum and energy equations are given by:

$$\frac{d\overrightarrow{v_i}}{dt}\bigg|_{\text{diss}} = \begin{cases} \sum_j \frac{m_j}{\overline{\rho_{ij}}}\left(\alpha_v v_{sig,v}\mathbf{v}_{ij}\hat{\mathbf{x}}_{ij}\cdot\overline{\nabla_i W_{ij}}\right) & if \quad \mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij} < 0 \\ \\ 0 & if \quad \mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij} \geq 0 \end{cases} \tag{3.60}$$

$$\frac{de_i^{int}}{dt}\bigg|_{\text{diss}} = \begin{cases} -\sum_j \frac{m_j}{\overline{\rho_{ij}}}\left(\frac{1}{2}\alpha_v v_{sig,v}(\mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij})^2 - \alpha_e v_{sig,e}e_{ij}\right)\hat{\mathbf{x}}_{ij}\cdot\overline{\nabla_i W_{ij}} & if \quad \mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij} < 0 \\ \\ 0 & if \quad \mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij} \geq 0 \end{cases}$$

$$\tag{3.61}$$

Where

$$\mathbf{v}_{ij} = \overrightarrow{v_i} - \overrightarrow{v_j} \tag{3.62}$$

$$\hat{\mathbf{x}}_{ij} = \frac{(\overrightarrow{x_i} - \overrightarrow{x_j})}{\| \overrightarrow{x_i} - \overrightarrow{x_j} \|} \tag{3.63}$$

$$e_{ij} = e_i - e_j \tag{3.64}$$

$$\overline{\rho_{ij}} = \frac{1}{2}(\rho_i + \rho_j) \tag{3.65}$$

$$v_{\text{sig},\{v,e\}} = \sqrt{c_i^2 + \beta_{\{v,e\}}(\mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij})^2} + \sqrt{c_j^2 + \beta_{\{v,e\}}(\mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij})^2} - (\mathbf{v}_{ij}\cdot\hat{\mathbf{x}}_{ij}) \tag{3.66}$$

$$\overline{\nabla_i W_{ij}} = \frac{1}{2}\left(\nabla_i W_{ij}(h_i) + \nabla_i W_{ij}(h_j)\right) \tag{3.67}$$

The parameters $\alpha_{\{v,e\}}$ are equivalent to the linear and quadratic Monaghan-type artificial viscosity parameters α_Π and β_Π that are used for the WC-SPH scheme. The optimal parameters are chosen as $\alpha_e = 0.5$, $\beta_e = 1.0$, $\alpha_v = 1.0$ and $\beta_v = 2.0$. For problems with a large density ratio, the differential form of mass conservation introduces spurious blips at the contact discontinuity. One possible way to treat this issue is to solve the integral form of mass conservation (Eq.3.40 or Eq.3.44) with the drawback of increasing the overall CPU time by iteratively solving the continuous density integral equation (Eq.3.40 or Eq.3.44). Another way to treat this issue is to solve the differential form of the discrete FC-SPH mass conservation equation (Eq.3.42 or Eq.3.46) adding an artificial dissipative term to the mass equation [167, 169, 168, 172] in order to capture shocks in density

and to smear out instabilities in the distribution of pressure across contact discontinuities. The necessity for an artificial term in the evolution of energy is relaxed.

The diffusive mass flux term added in the equation of mass conservation is given by:

$$\left.\frac{d\rho_i}{dt}\right|_{\text{diss}} = \sum_j m_j \frac{\alpha_\rho(p_i - p_j)}{\overline{\rho}_{ij} v_{sig,\rho}} \hat{\mathbf{x}}_{ij} \cdot \overline{\nabla_i W_{ij}}, \tag{3.68}$$

where $v_{sig,\rho}$ is obtained replacing $\{v, e\}$ by ρ into Eq.3.68. The parameters $\alpha_\rho = 0.5$, $\beta_\rho = 1.0$, $\alpha_v = 1.0$ and $\beta_v = 2.0$ will remain unchanged for all the test cases in order to assess the robustness of the FC-SPH schemes.

The stability CFL condition for the explicit FC-SPH scheme is given by:

$$\Delta t_{cpu} \leq \Delta t_i^{crit} = \frac{h_i}{c_i + \parallel \overrightarrow{v_i} \parallel}, \tag{3.69}$$

where Δt_i^{crit} is the critical time step of the particle i. It is defined as the ratio between the smoothing length h_i of the particle and the sum between the speed of sound c_i and norm of the velocity $\parallel \overrightarrow{v_i} \parallel$. The final time step Δt_{cpu} is chosen such as:

$$\Delta t_{cpu} = fac.\min_i(\Delta t_i^{crit}) \tag{3.70}$$

Where fac is a scaling factor taken between 0 and 1.

- **Smoothing length definition**

$$h_i = \eta \left(\frac{\rho_i}{m_i} \right)^{\frac{1}{d}}$$

- **Mass scheme: *m-scheme***

  – Conservation of mass

$$\frac{d\rho_i}{dt} = \frac{1}{\Omega_i} \sum_{j \in D_i} m_j (\overrightarrow{v_i} - \overrightarrow{v_j}) . \nabla_i W_{ij}(h_i)$$

$$\Omega_i = 1 - \frac{\partial h_i}{\partial \rho_i} \sum_{j \in D_i} m_j \frac{\partial W_{ij}(h_i)}{\partial h_i}$$

  – Conservation of linear momentum

$$\frac{d\overrightarrow{v_i}}{dt} = -\sum_{j \in D_i} m_j \left(\frac{p_i}{\Omega_i \rho_i^2} \nabla_i W_{ij}(h_i) + \frac{p_j}{\Omega_j \rho_j^2} \nabla_i W_{ij}(h_j) \right)$$

- **Number density scheme: *n-scheme***

  – Conservation of mass

$$\frac{d\rho_i}{dt} = \frac{m_i}{\Omega_i} \sum_{j \in D_i} (\overrightarrow{v_i} - \overrightarrow{v_j}) . \nabla_i W_{ij}(h_i)$$

$$\Omega_i = 1 - m_i \frac{\partial h_i}{\partial \rho_i} \sum_{j \in D_i} \frac{\partial W_{ij}(h_i)}{\partial h_i}$$

  – Conservation of linear momentum

$$\frac{d\overrightarrow{v_i}}{dt} = -\frac{1}{m_i} \sum_{j \in D_i} \left(\frac{p_i V_i^2}{\Omega_i} \nabla_i W_{ij}(h_i) + \frac{p_j V_j^2}{\Omega_j} \nabla_i W_{ij}(h_j) \right)$$

- **Conservation of energy**

$$\frac{de_i^{int}}{dt} = \frac{p_i}{\rho_i^2} \frac{d\rho_i}{dt}$$

- **Artificial dissipation terms (if $\mathbf{v}_{ij} \cdot \hat{\mathbf{x}}_{ij} < 0$)**

$$\left. \frac{d\rho_i}{dt} \right|_{\text{diss}} = \sum_j m_j \frac{\alpha_\rho (p_i - p_j)}{\overline{\rho}_{ij} v_{sig,\rho}} \hat{\mathbf{x}}_{ij} \cdot \overline{\nabla_i W_{ij}}$$

$$\left. \frac{d\overrightarrow{v_i}}{dt} \right|_{\text{diss}} = \sum_j \frac{m_j}{\overline{\rho}_{ij}} \left(\alpha_v v_{sig,v} \mathbf{v}_{ij} \hat{\mathbf{x}}_{ij} \cdot \overline{\nabla_i W_{ij}} \right)$$

3.2.4   Comparison of FC-SPH schemes

Three 1-D shock-tube tests with different density ratios are studied to assess the FC-SPH schemes. In order to show the robustness of the FC-SPH, we use for all test cases:

- The truncated Gaussian kernel function (Eq.3.12).

- Artificial dissipation in mass and momentum with the parameters $\alpha_\rho = 0.5$, $\beta_\rho = 1.0$, $\alpha_v = 1.0$ and $\beta_v = 2.0$.

- In order to achieve equal "volumes" for the SPH system, the unit domain is divided in 1600 intervals of equal length Δx_o and one particle is placed in the middle of each interval. The mass of each particle is assigned according to the initial density distribution, as: $m_i = \rho_{o,i}\Delta x_o$.

- The SG EOS (Eq.1.105) for two first test cases. For the last case, the air is modeled with the SG EOS and the water is modeled with the Tait EOS.

The first test is the classical shock-tube benchmark referred to as the **Air-Air shock tube** with a density ratio of 8, which has been addressed since the early development of SPH [97] and its solution within the standard variational SPH framework is extensively examined in [114].
The second test was proposed by Saurel et al. [125] and involves a multiphase medium with high pressure liquid water in the left region and low pressure air in the right region, with discontinuous initial density distribution and fluid parameters. It is referred to as **Water-Air shock tube** with a density ratio of 20.
The third and last test involves a multiphase medium with high pressure air in the left region and low pressure water in the right region, and also includes discontinuous initial density distribution in fluid parameters as well. It is similar to Narvik's shock tube and referred to as the **Air-Water shock tube** with a density ratio of 200.
They are both Riemann problems and admit exact solutions, which can be obtained through the procedure described in [112]. The initial parameters for the three problems are given in Tab.3.6 and Tab.3.7.

	Air-Air	Water-Air	Air-Water
x_l	0.5	0.7	1.0
x_r	0.5	0.3	1.0

Table 3.6 – Shock tube lengths

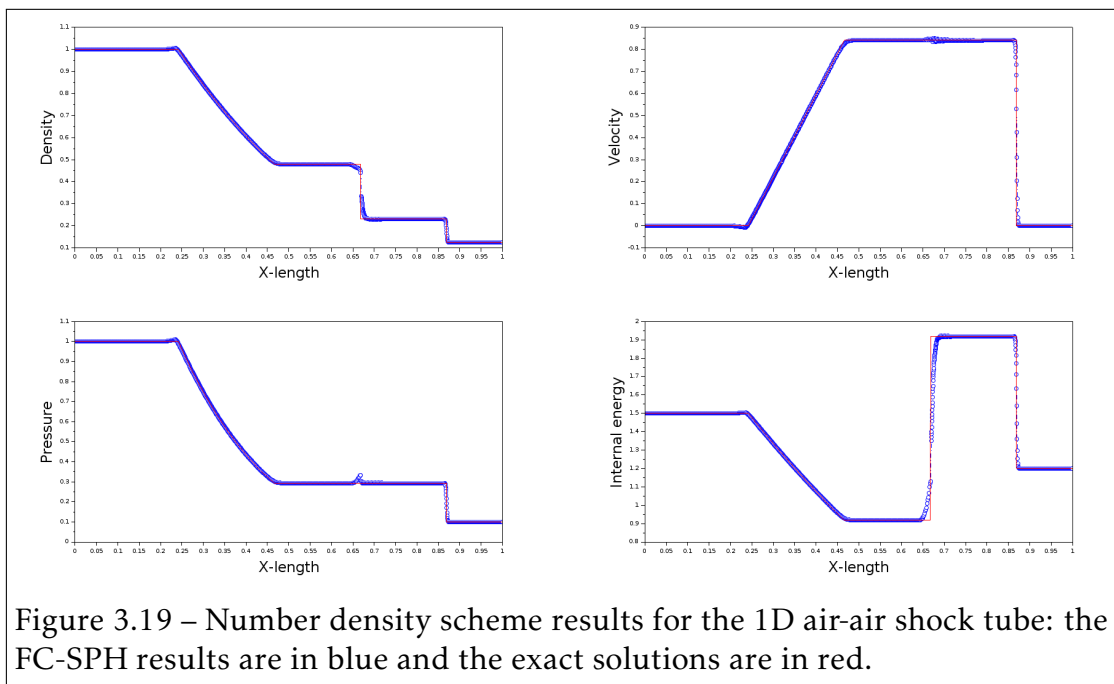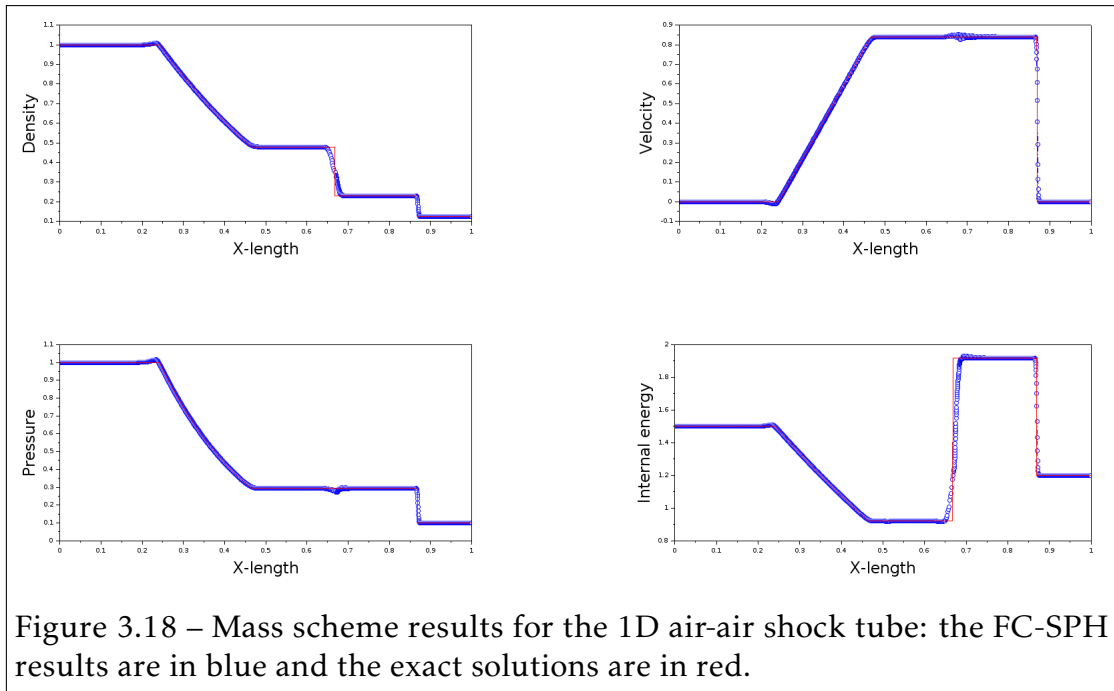|	p_0	ρ_0	vx	γ	π	c
Left Air Right Air	1.0 0.1	1.0 0.125	0.0 0.0	5./3. 5./3.	0.0 0.0	- -
Left Water Right Air	1.e+9 1.e+7	1000 50	0.0 0.0	4.4 1.4	6.e+8 0.0	- -
Left Air Right Water	94.33e+5 0	5.65087 1000	0.0 0.0	1.4 7.5	0.0 -	- 1480.

Table 3.7 – Initial data and parameters

Results and discussions:

In Fig.3.18 and Fig.3.19 the *m-scheme* and the *n-scheme* solutions at $t = 0.2$ are presented for the Air-Air shock tube, respectively. A constant time step of $\Delta t = 10^{-4}$ is used. Both schemes are in agreement with the exact result, except at the contact discontinuity where a small hump is present in the pressure variable and spurious oscillations in velocity. However, the hump is small in comparison to the WC-SPH solution and the results have consequently been improved (see Fig.3.17).
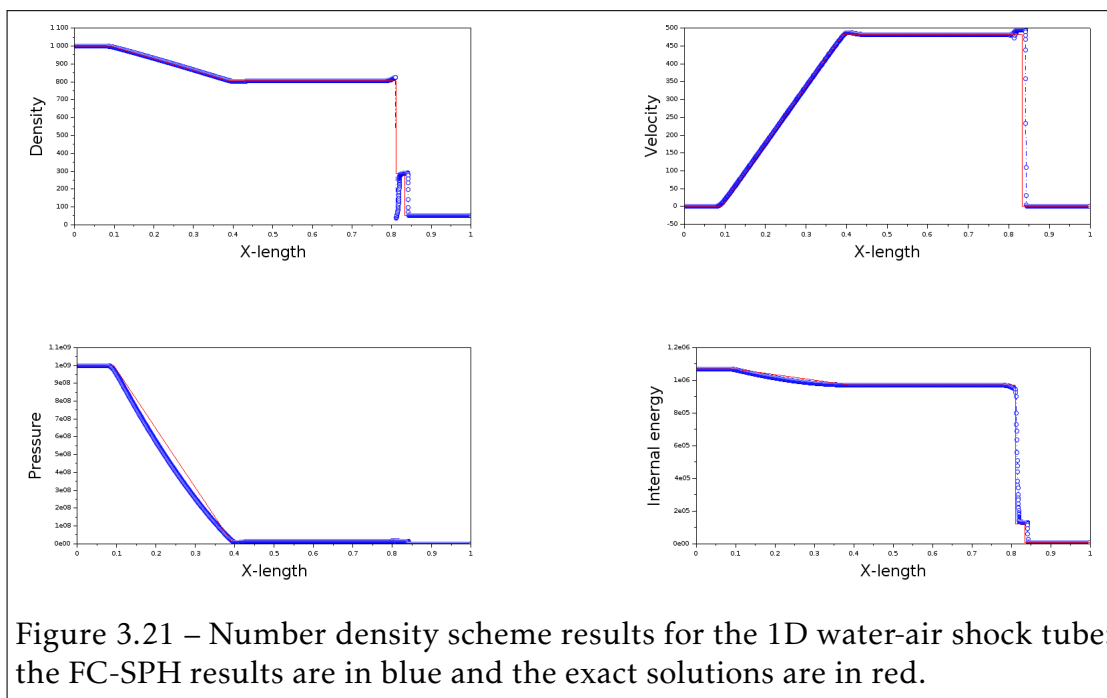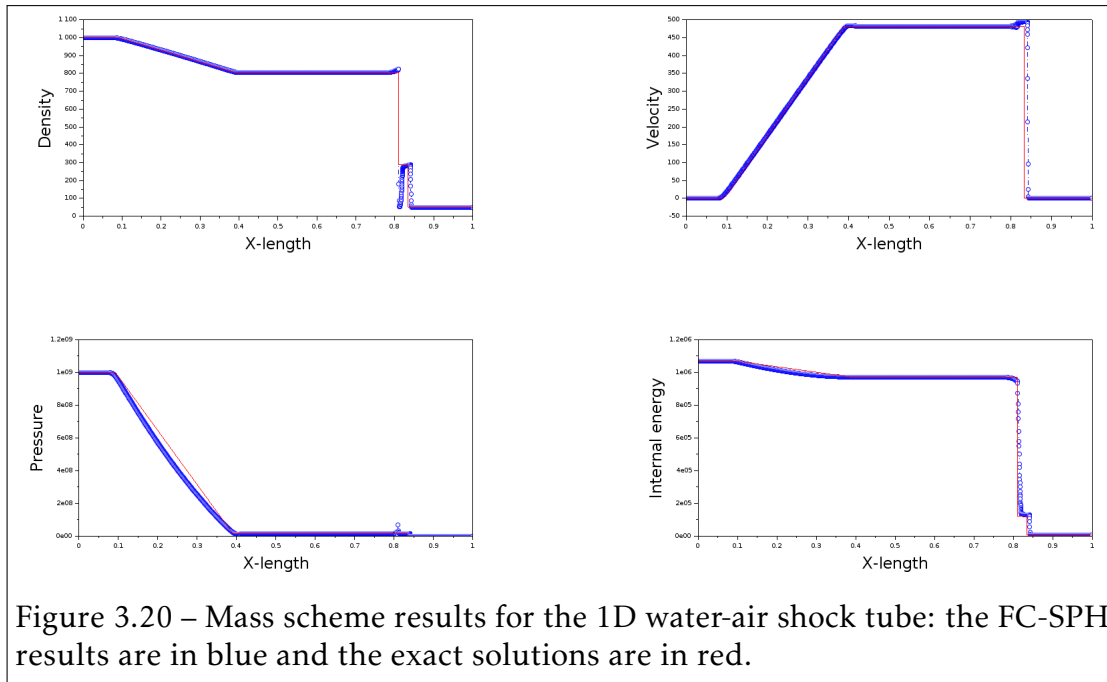
In Fig.3.20 and Fig.3.21 the *m-scheme* and the *n-scheme* solutions at $t = 0.0023$ are presented for the Water-Air shock tube, respectively. A constant time step of $\Delta t = 10^{-7}$ is used. The tendency shown in the previous Air-Air test case is confirmed in this case. Spurious oscillations developed in the contact discontinuity. However, the shock waves and rarefaction are solved accurately and the spurious oscillations do not affect the global accuracy far from the contact discontinuity. In the last test case, only the *n-scheme* managed to solve the problem. In Fig.3.22, the *n-scheme* solution at $t = 0.025$ is presented for the Air-Water shock. A constant time step of $\Delta t = 10^{-8}$ is used. It was observed that the *m-scheme* was unable to solve the strong shock problem and to capture the contact discontinuity; the solution blows up. Fig.3.20 and Fig.3.21 the *m-scheme* and the *n-scheme* solutions at $t = 0.0023$ are presented for the Water-Air shock tube, respectively. A constant time step of $\Delta t = 10^{-7}$ is used.

From the previous test cases, we can conclude that the *n-scheme* performs better than the *m-scheme* to model shock problems in fully-compressible multiphase flows. It is also important to remark that the FC-SPH schemes are very diffusive. For the next 2-D validation test cases, we retain the *n-scheme*. In absence of an exact solution, the FC-SPH results are compared to the reference ALE results.
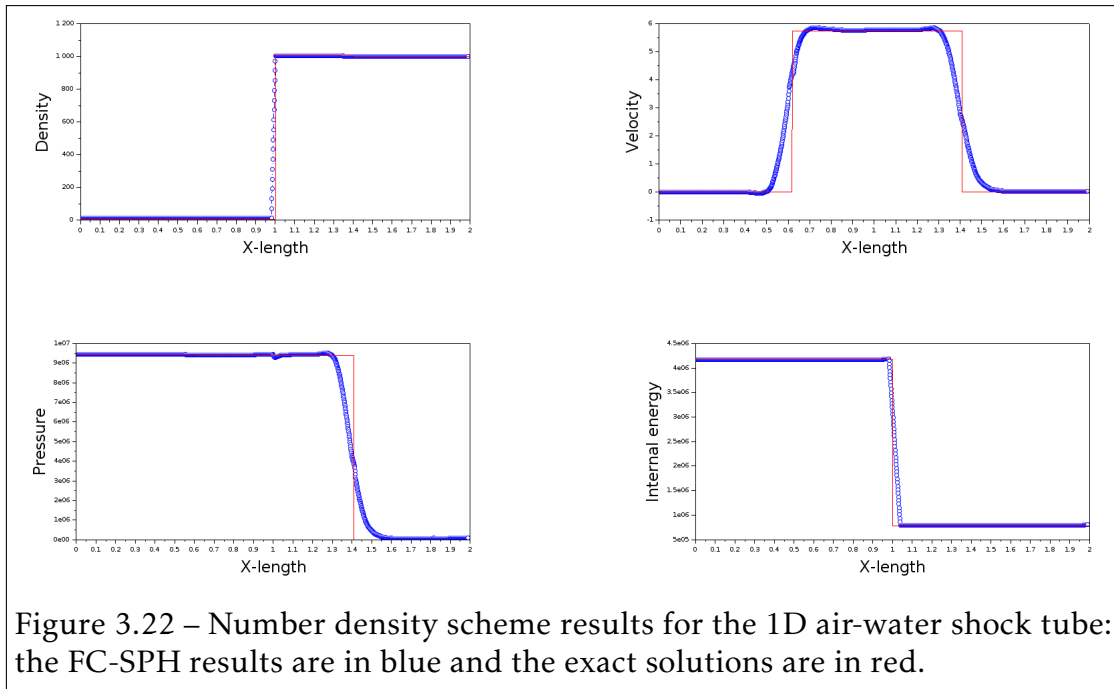
1D Air-Air (density ratio of 8)

Figure 3.18 – Mass scheme results for the 1D air-air shock tube: the FC-SPH results are in blue and the exact solutions are in red.

Figure 3.19 – Number density scheme results for the 1D air-air shock tube: the FC-SPH results are in blue and the exact solutions are in red.

1D Water-Air (density ratio of 20)

Figure 3.20 – Mass scheme results for the 1D water-air shock tube: the FC-SPH results are in blue and the exact solutions are in red.

Figure 3.21 – Number density scheme results for the 1D water-air shock tube: the FC-SPH results are in blue and the exact solutions are in red.

1D Air-Water (density ratio $\sim$ 200)

Figure 3.22 – Number density scheme results for the 1D air-water shock tube: the FC-SPH results are in blue and the exact solutions are in red.

3.3   Assessment: 2-D Validation and Comparison with ALE solution

In order the assess the **FC-SPH** *n-scheme*, we study the three following 2-D multiphase shock problems:

1. A square **Air-Air shock-chamber** defined on
   $\Omega = \{|x| \leq 2.5 \cap |y| \leq 2.5\}$ is modeled with high pressure air in its center defined on $\Omega_{high} = \{|x| \leq 0.5 \cap |y| \leq 0.5\}$ and low pressure air around defined on $\Omega_{low} = \Omega \setminus \Omega_{high}$. The ALE and FC-SPH simulations are composed of the same number of elements: 250,000 each.
   Considering the symmetry of the problem, only one-fourth of the problem is modeled. Ghost particles with slip boundary conditions are used to model the solid walls. Both fluids are modeled with the SG EOS, and the parameters are the same ones used for the 1-D test case in Tab.3.7.

2. The **UNDEX shock-chamber** test case presented in section 3.1.3 is revisited with the FC-SPH *n-scheme*. The FSI feature has not been yet implemented in the in-house SPH code developed throughout this thesis. As a result, the coupled structure in Fig.3.4 is replaced by a solid wall modeled with ghost particles. A square **Explosive-Water shock-chamber** defined on $\Omega = \{|x| \leq 13 \cap |y| \leq 13\}$ is modeled with a high pressure explosive in its center defined on $\Omega_{high} = \{|x| \leq 0.5 \cap |y| \leq 0.5\}$ and a low pressure water around defined on $\Omega_{low} = \Omega \setminus \Omega_{high}$. The ALE and FC-SPH simulations are composed of the same number of elements: 250,000 each.
   Considering the symmetry of the problem, only one-fourth of the problem is modeled. Ghost particles with slip boundary conditions are used to model the solid walls. JWL EOS and the MG EOS are used to model the explosive and water materials, respectively. The physical parameters are given in Tab.3.1, Tab.3.2, Tab.3.3 and Tab.3.4.

3. The **2D UNDEX near a Planar Wall** present in [156], in which three numerical simulations are considered: The reference ALE model, the coarse FC-SPH model and the refined FC-SPH model composed of 26,000 finite

elements, 26,000 and 156,000 equi-spaced SPH particles, respectively. The two different SPH resolutions were chosen in order to compare the FC-SPH solution to the ALE one in term of accuracy and to highlight, that in practice, we need much more SPH particles to reach the accuracy of ALE methods at the expense of an considerable increase of the overall CPU time. The computational domain is defined on $\Omega = \{| \, x \, | \leq 6.0m \cap -6.0m \leq y \leq 3.0m\}$, in which the planar wall is located at the top boundary $y = 3m$. The explosive is modeled by a high pressure gas defined on
$\Omega_{high} = \left\{(x, y), r = \sqrt{x^2 + y^2} \leq 1\right\}$ through the use of the IG EOS, in which $\gamma = 2.0$, $\rho = 1270 kg.m^{-3}$ and $p = 8.29e + 8Pa..$ While the water is defined on $\Omega_{low} = \Omega \setminus \Omega_{high}$ through the use of the Tait EOS, in which $\gamma = 7.15$, $\rho = 1000 kg.m^{-3}$, $p = 1.e + 5Pa.$ and the cut-off pressure pc is set to the saturation pressure $pc = 230Pa$.

The simulation is run up to time $t = 6.0ms$ and the time history results (pressure and density) are taken at the point $P(0.0, 3.0)$. The different stages of the simulation are the following:

- At time $t = 1.5ms$, the shock generated by the explosion reached the solid planar wall at the top of the computational domain and it is reflected (see Fig.3.30).

- At time $t = 2.0ms$, the reflected shock wave has reached the gas bubble and it gave arise to a rarefaction wave that propagates toward the solid planar wall (see Fig.3.31).

- At time $t = 3.0ms$, the rarefaction wave is reflected on the solid planar wall and it gives arise to a low pressure region near the wall (see Fig.3.32).

- At time $t = 4.0ms$, cavitation appears in the low region (see Fig.3.33).

- At time $t = 5.0ms$, the cavitation in the low region collapses and it gives arise to a compression wave propagating toward the gas bubble (see Fig.3.34).

The results for the **Air-Air shock-chamber**, the **UNDEX shock-chamber** and the **UNDEX near a planar wall** test cases are presented in subsections 3.3.1, 3.3.2

and 3.3.3, respectively. From both test cases, it can be seen that the FC-SPH *n-scheme* succeeds in resolving the two problems. However, it can be seen in both the **UNDEX shock-chamber** and the **UNDEX near a planar wall** test cases that the *n-scheme* is much more diffusive than the ALE methods.

One solution to limit the diffused region is to refine the number of particles at the cost of increasing the overall CPU time, as shown in the third test case. Another possible solution to limit the diffusion is to use switches [100, 95, 114] in the artificial dissipation terms to limit the dissipative effects away from discontinuities.
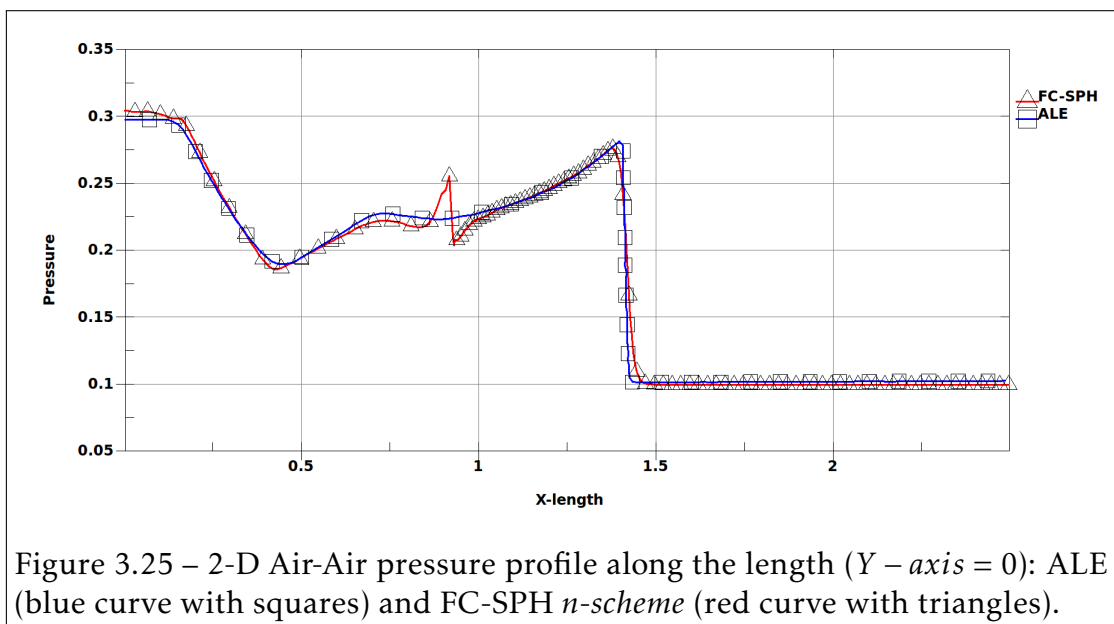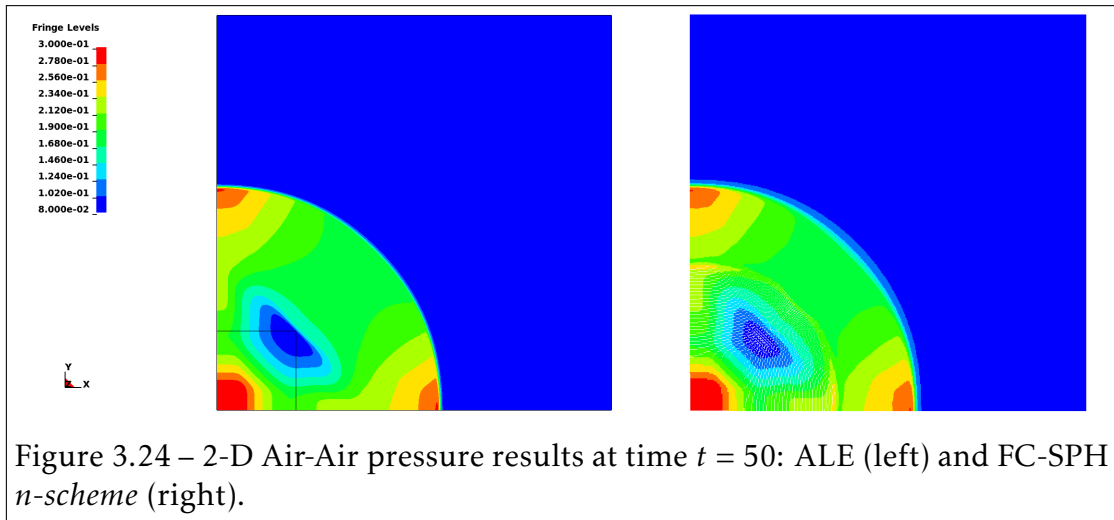
3.3.1   Air-Air shock-chamber results
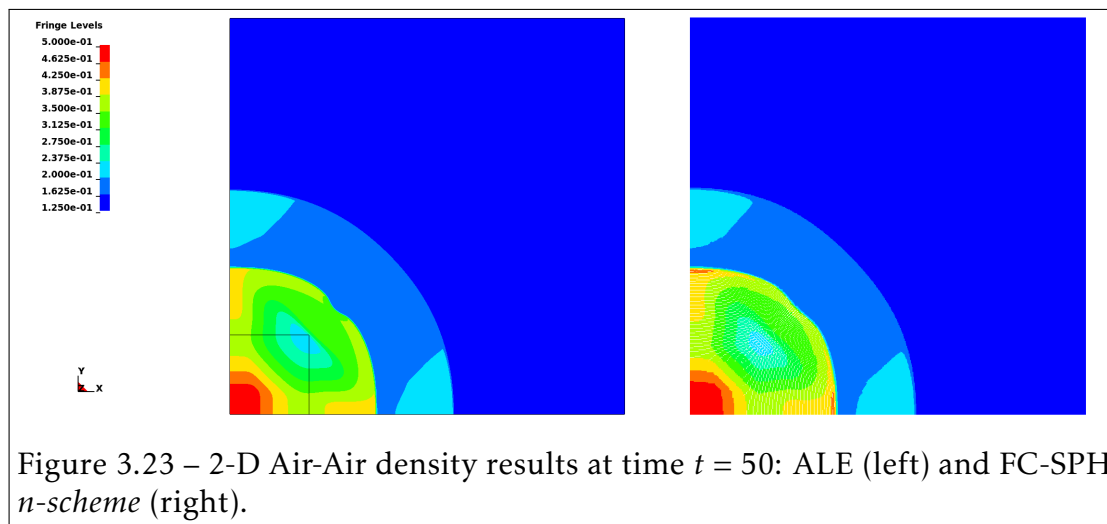
Figure 3.23 – 2-D Air-Air density results at time $t = 50$: ALE (left) and FC-SPH *n-scheme* (right).

Figure 3.24 – 2-D Air-Air pressure results at time $t = 50$: ALE (left) and FC-SPH *n-scheme* (right).

Figure 3.25 – 2-D Air-Air pressure profile along the length ($Y - axis = 0$): ALE (blue curve with squares) and FC-SPH *n-scheme* (red curve with triangles).
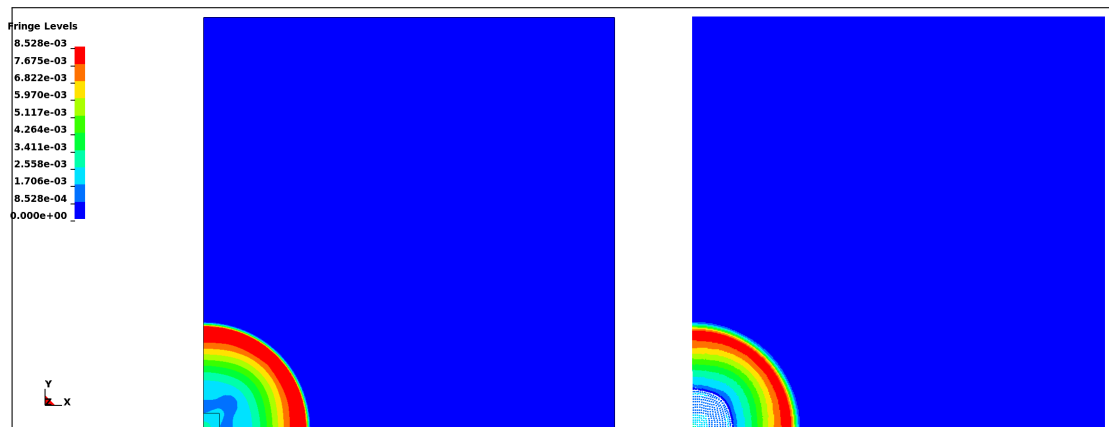
3.3.2   2D UNDEX results

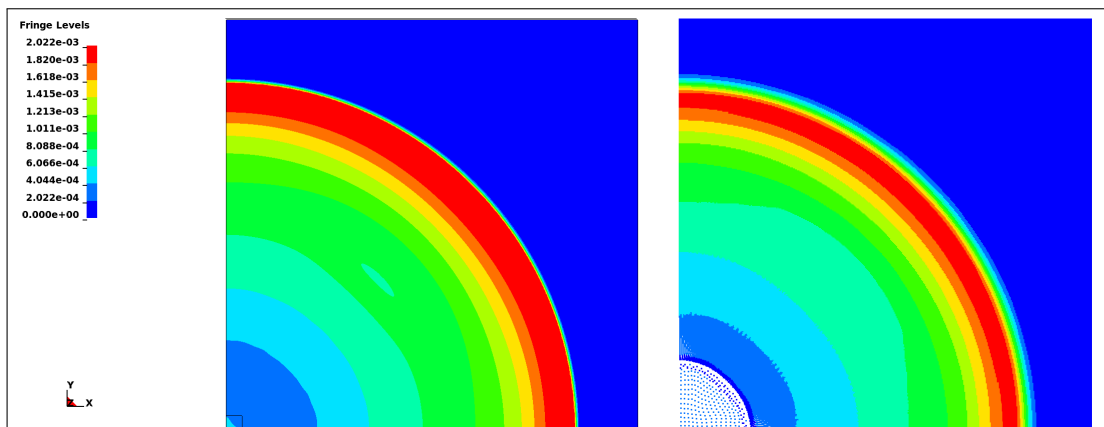Figure 3.26 – UNDEX pressure results at time $t = 10ms$: ALE (left) and FC-SPH *n-scheme* (right).

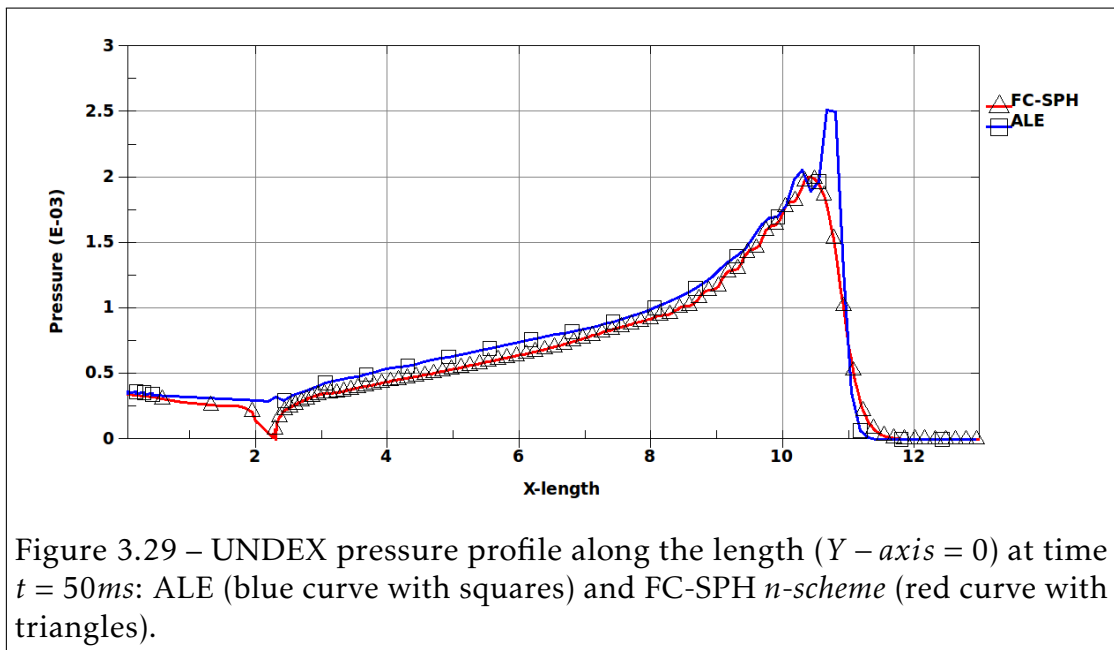Figure 3.27 – UNDEX pressure results at time $t = 50ms$: ALE (left) and FC-SPH *n-scheme* (right).

Figure 3.28 – UNDEX pressure results at time $t = 80ms$: ALE (left) and FC-SPH *n-scheme* (right).

Figure 3.29 – UNDEX pressure profile along the length ($Y - axis = 0$) at time $t = 50ms$: ALE (blue curve with squares) and FC-SPH *n-scheme* (red curve with triangles).
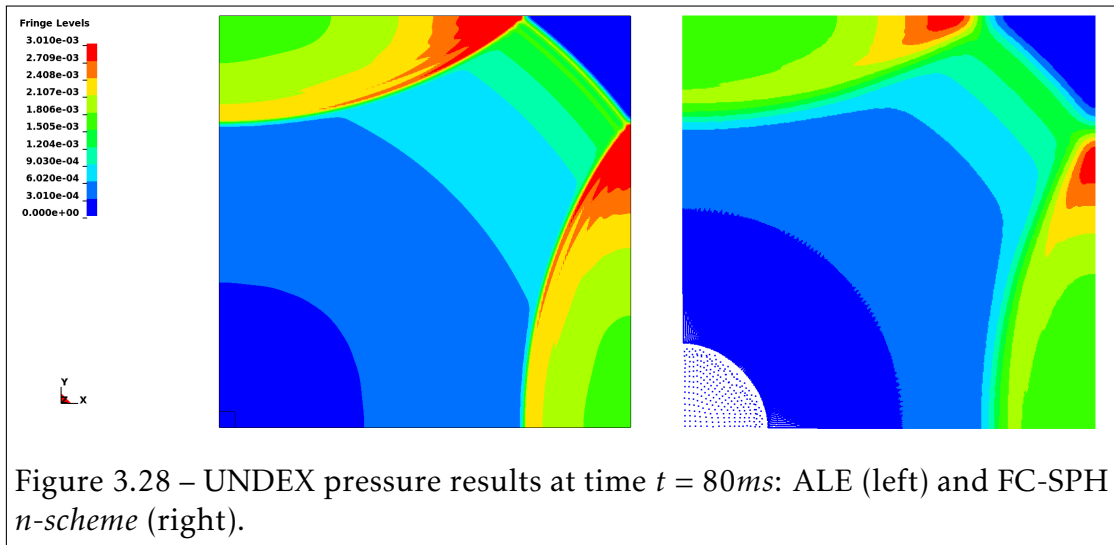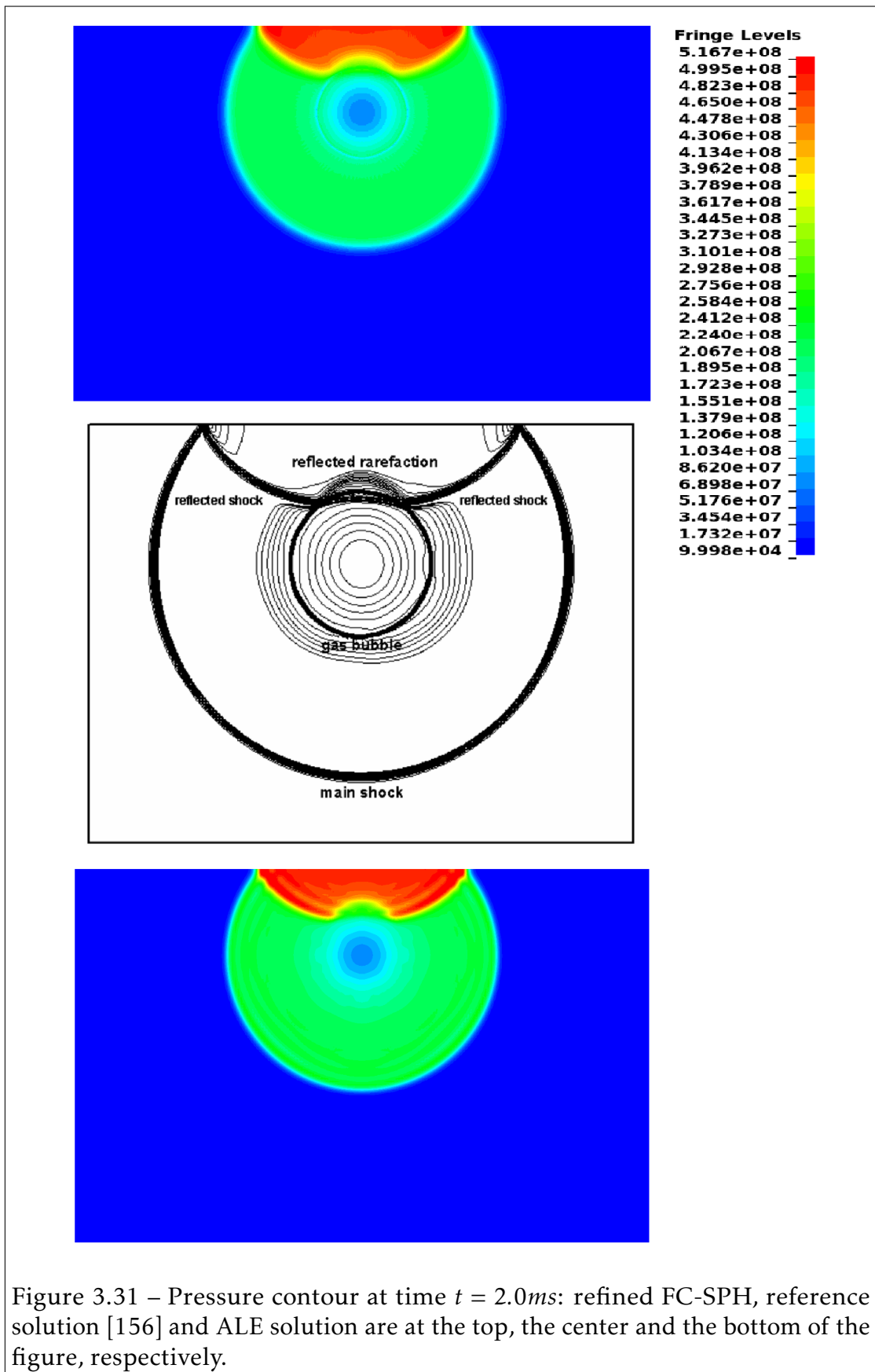
3.3.3   2D UNDEX near a Planar Wall [156] results
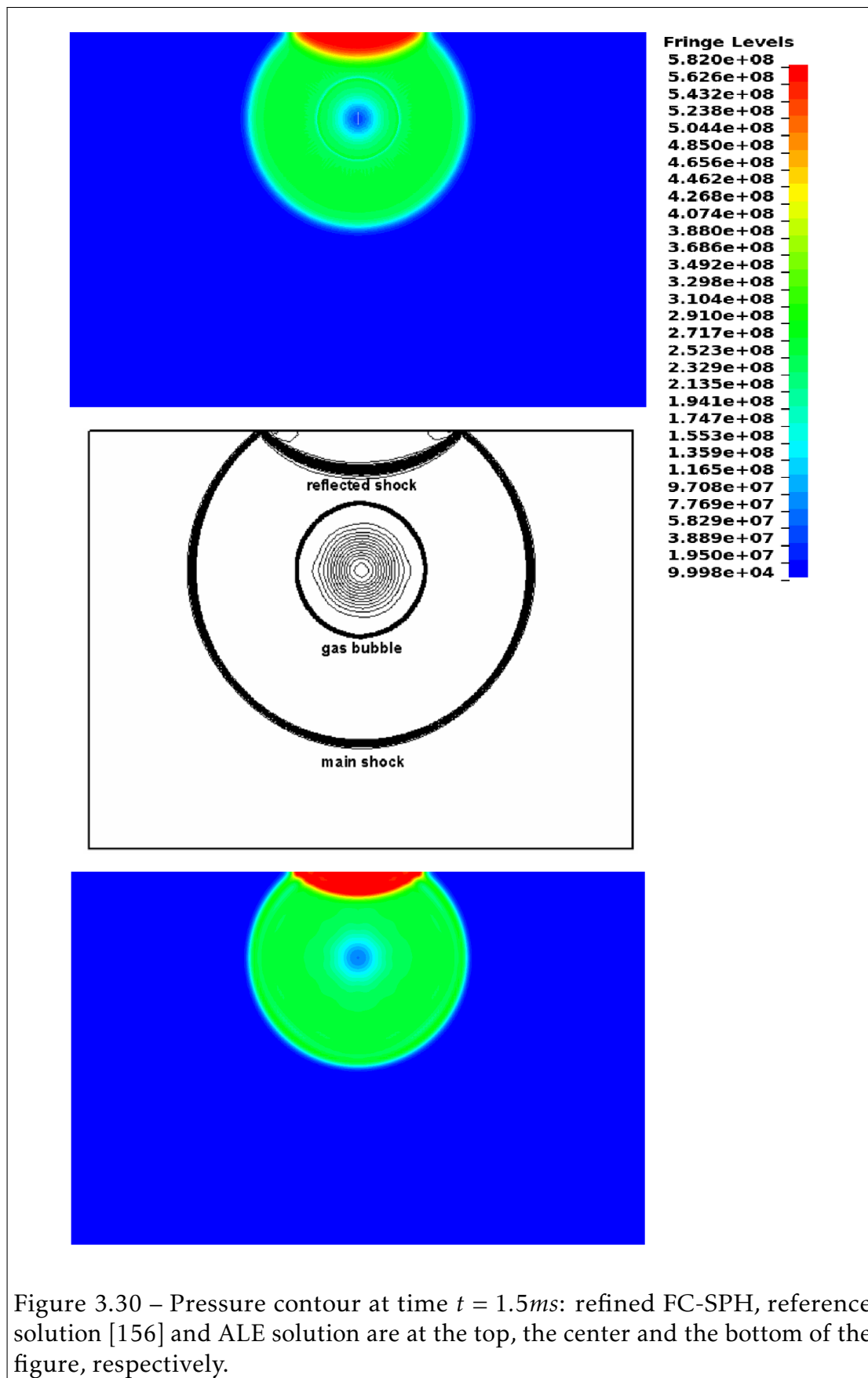
Figure 3.30 – Pressure contour at time $t = 1.5ms$: refined FC-SPH, reference solution [156] and ALE solution are at the top, the center and the bottom of the figure, respectively.
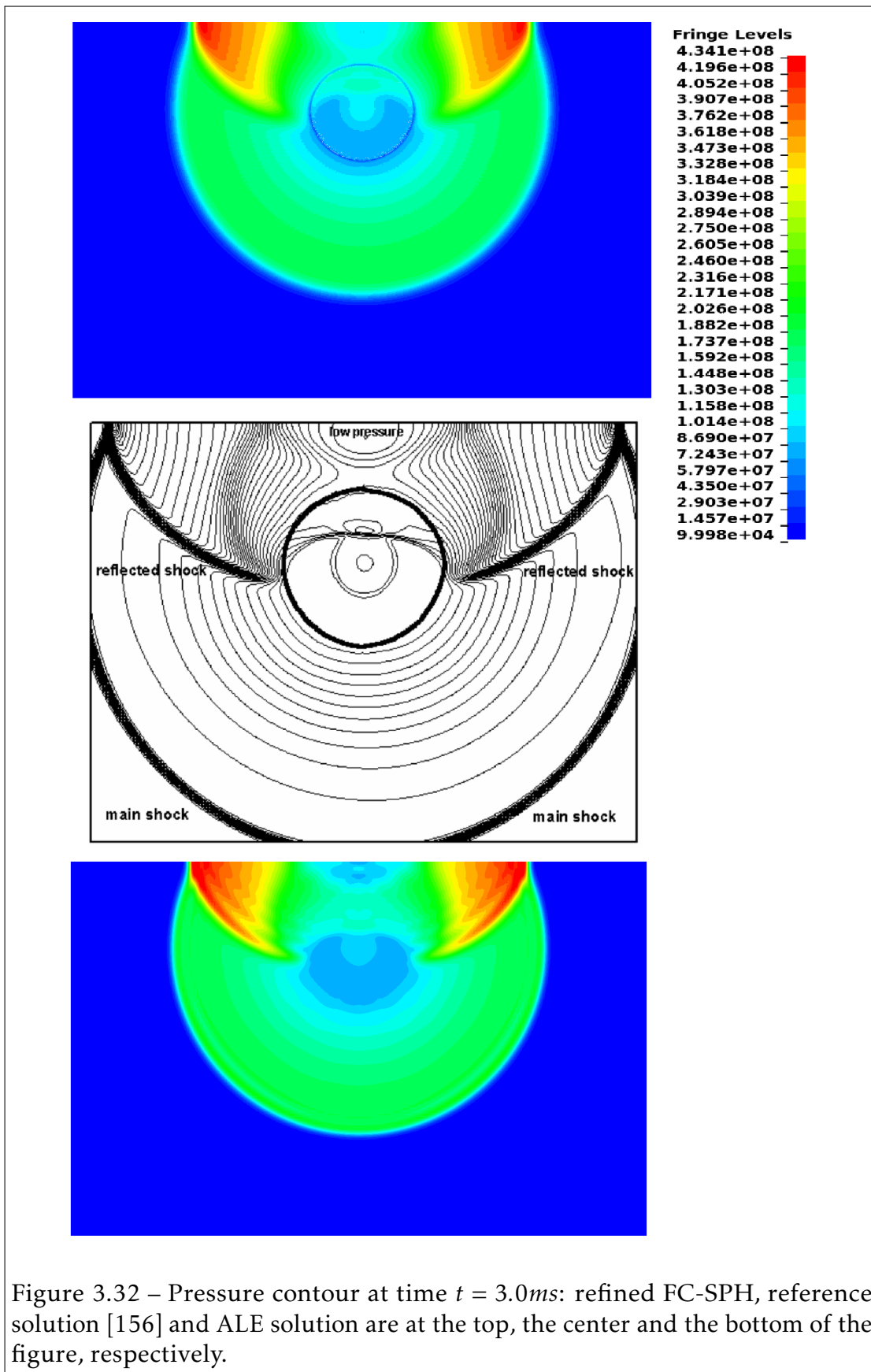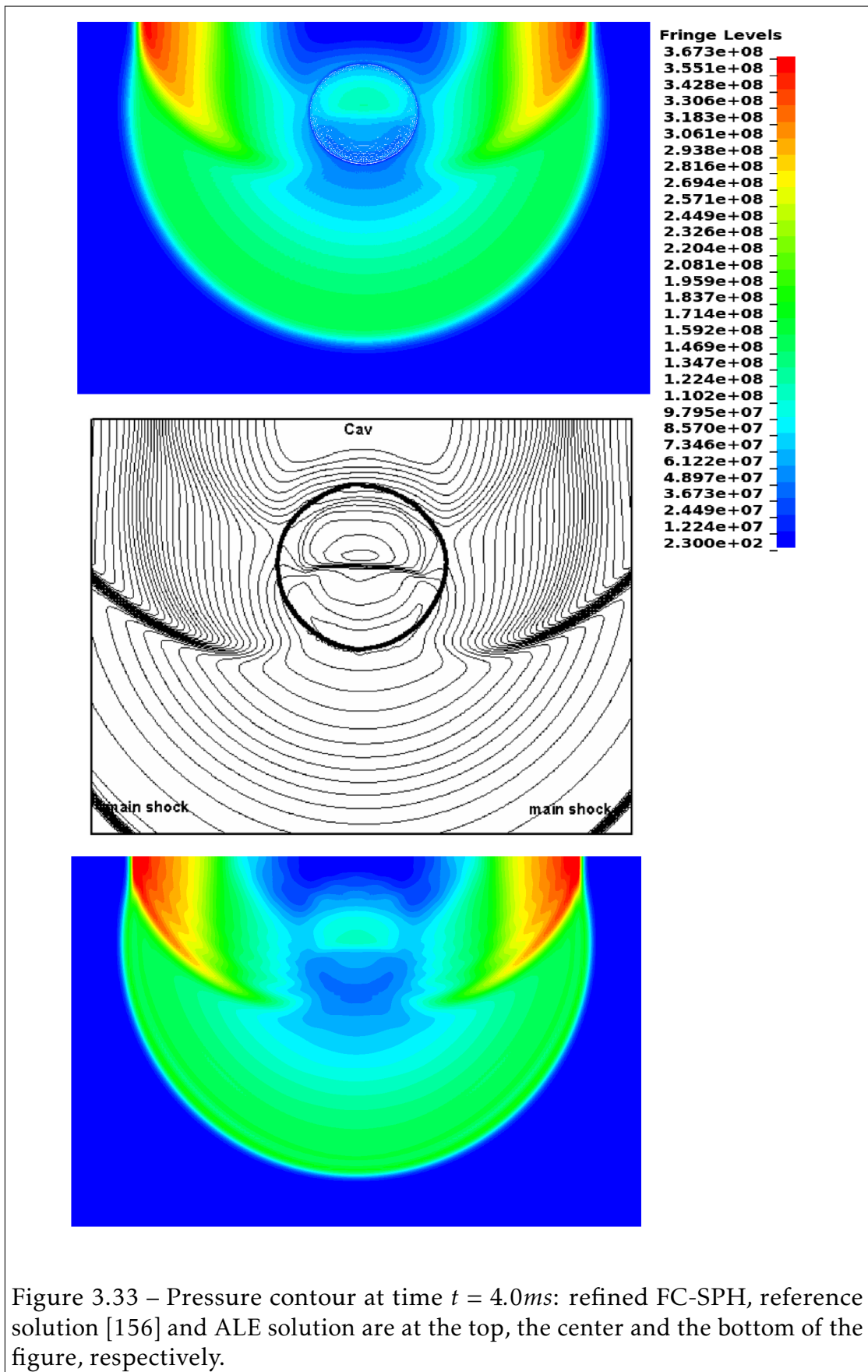
Figure 3.31 – Pressure contour at time $t = 2.0ms$: refined FC-SPH, reference solution [156] and ALE solution are at the top, the center and the bottom of the figure, respectively.

Figure 3.32 – Pressure contour at time $t = 3.0ms$: refined FC-SPH, reference solution [156] and ALE solution are at the top, the center and the bottom of the figure, respectively.
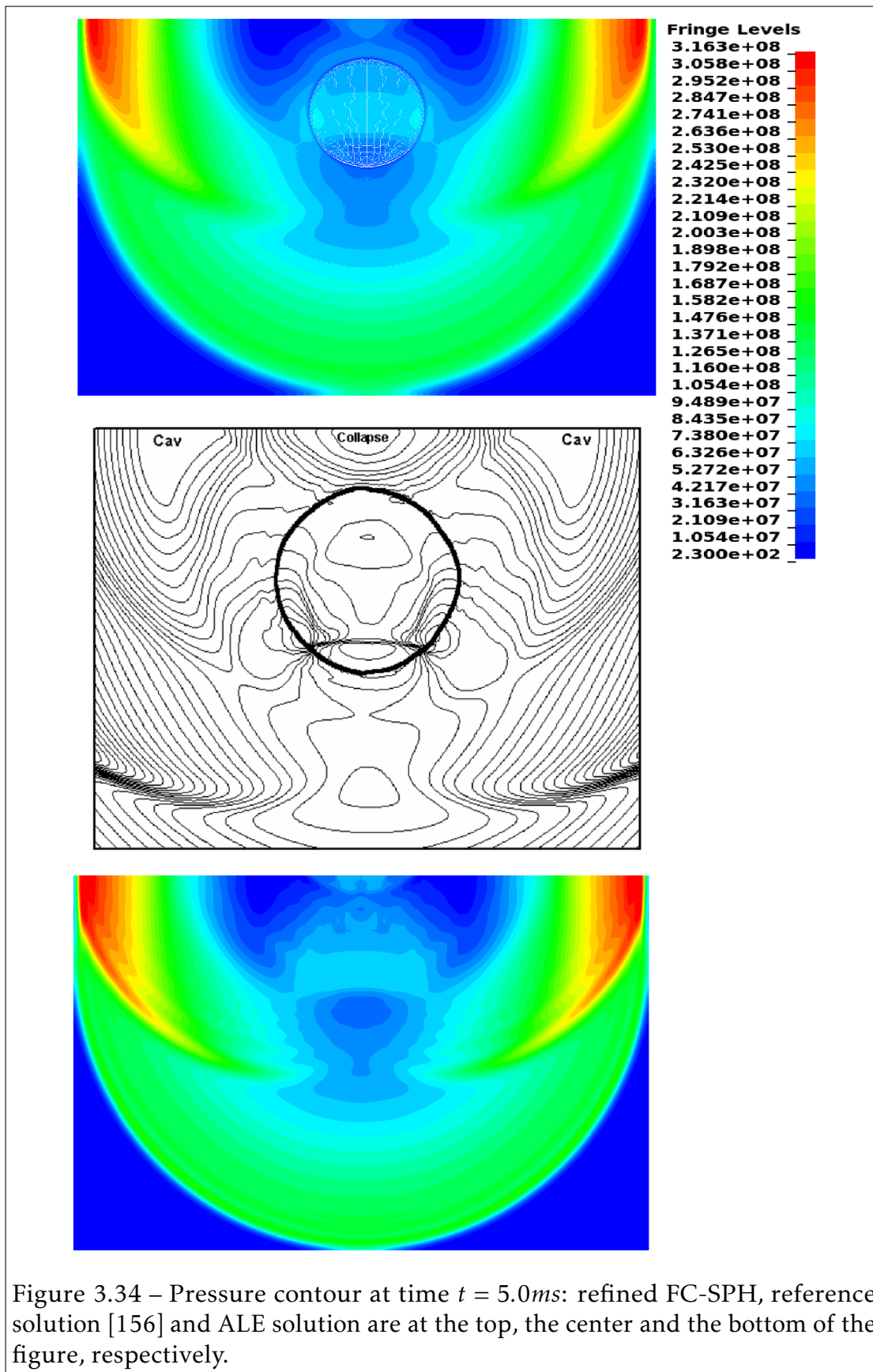
Figure 3.33 – Pressure contour at time $t = 4.0ms$: refined FC-SPH, reference solution [156] and ALE solution are at the top, the center and the bottom of the figure, respectively.

Figure 3.34 – Pressure contour at time $t = 5.0ms$: refined FC-SPH, reference solution [156] and ALE solution are at the top, the center and the bottom of the figure, respectively.
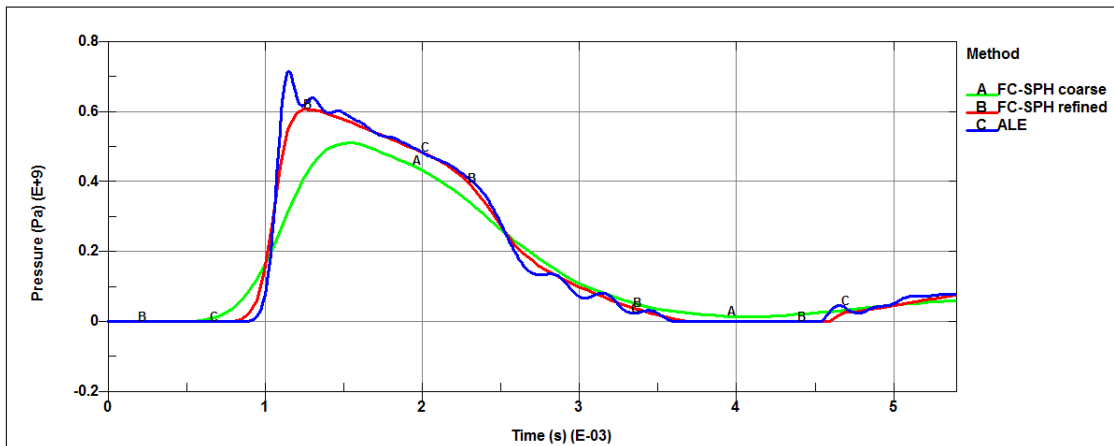
Figure 3.35 – Pressure time history at the center of the solid planar wall
$P(0.0, 3.0)$: the coarsed FC-SPH, refined FC-SPH and ALE solutions are plotted
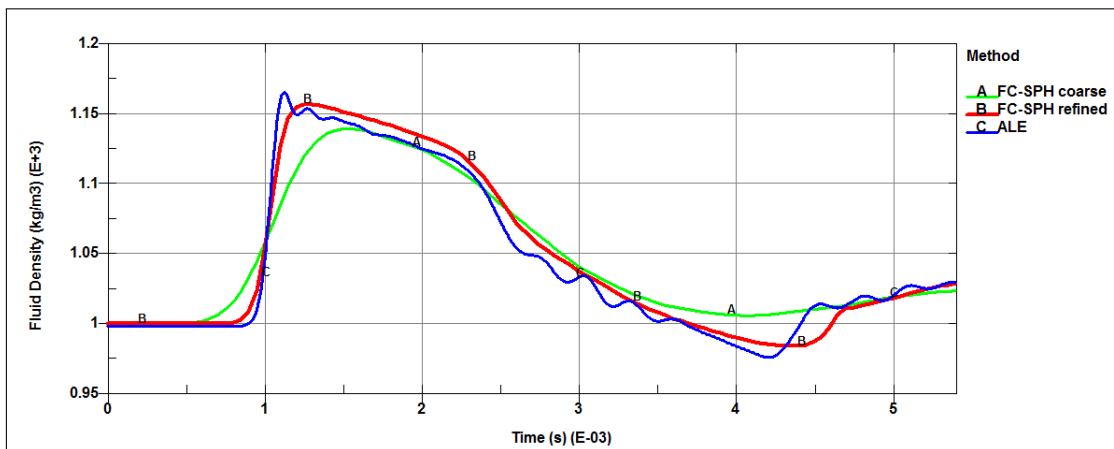in green, red and blue, respectively.

Figure 3.36 – Density time history at the center of the solid planar wall
$P(0.0, 3.0)$: the coarsed FC-SPH, refined FC-SPH and ALE solutions are plotted
in green, red and blue, respectively.

3.4   In-house SPH Code

Due to the limitations of the WC-SPH formulation implemented in most of
the commercial codes to simulate compressible multiphase flows, the need to
develop consistent FC-SPH formulations was necessary to solve high density
ratios problems in the fully-compressible regime.

Although there are many open source codes [83, 139, 25, 115], we chose to write
a new code "from scratch".

The code developed during my thesis was designed to be a compromise between
performance and **readability**, so that engineers, researchers and students could
easily modify the source codes and adapt it to their needs.

The majority of the codes developed in our research team are written in **Fortran**
programing language. So naturally the code was written in **Fortran 90** into
modular units.

Neighbors search (NS) is one of the most demanding tasks in term of CPU time.
Brute force algorithms are omitted due to their numerical complexity $O(N^2)$.
Among the different strategies of NS, two strategies are popular in mesh-less
methods:

- **Linked-list algorithm** more suited for WC-SPH methods [139].

- **Tree-search algorithm** more suited for FC-SPH methods [33].

The in-house SPH code is parallelized using **Message Passing Interface** (MPI)
and a **Data Parallel Model** (particle based decomposition) in which the particles
are assigned to a specific MPI rank. For a given particle i, since the summations
involve data from neighbors j, if i and j are not assigned to the same core, then
MPI communications are needed to gather the missing j data. The purpose
of the parallel implementation is to boost the computation and to be able to
solve academic problems in an acceptable real time duration. In the following
subsections, we describe the structure of the serial code and the tree-search
library [60] used for neighbors search.

### 3.4.1   Neighbors Search Algorithm: KD-Tree2 [60]

The KD-Tree stands for *"k-dimensional tree"* where *k* is the dimension of the problem. The KD-Tree is a data structure invented by [18] to store a set of points in a k-dimensional space in a balanced binary tree (see Fig.3.37).

In the thesis work, we used the Fortran library **KD-Tree2** provided by [60] which is easy to incorporate in existing codes. The library competes in terms of performance (CPU time only, memory not tested) with the existing ANN library [101] for the nearest neighbor search (personal tests on regular and irregular random data in 2-D). We briefly present the **KD-Tree2** library (for more details on the KD-trees see [18, 60, 81]).

For a given set of points in space, the nearest neighbors search is performed in two steps:

- **Building Kd-Tree:** The Kd-Tree organizes space to accelerate data processing. The data are structured in a binary tree (see [44]).

- **Radix neighbors search:** Let us consider two particles i and j and the Euclidean distance between two particles i and j defined by $d_2(i,j) = \sqrt{\sum_k (x_{i_k} - x_{j_k})^2}$. In FC-SPH discrete form of the linear momentum equation, for a given particle i, two terms involving the space derivative of the kernel function are present under the sum: one function of h_i and the other function of the neighbors h_j. Thus, when we compute the neighbors of i within its support domain D_i of radius κh_i, we need to include the particle j such that $i \in D_j$.

  To avoid missing neighbors in the discrete sum, for each query particle i the radix search distance r is chosen to be $r = \max_k h_k$. A first reduced selection of neighbors is computed, then it is limited to the only particles j that has a contribution in the sum by performing a brute force search on the reduced set of neighbors.

Figure 3.37 – Kd-Tree structure [44].
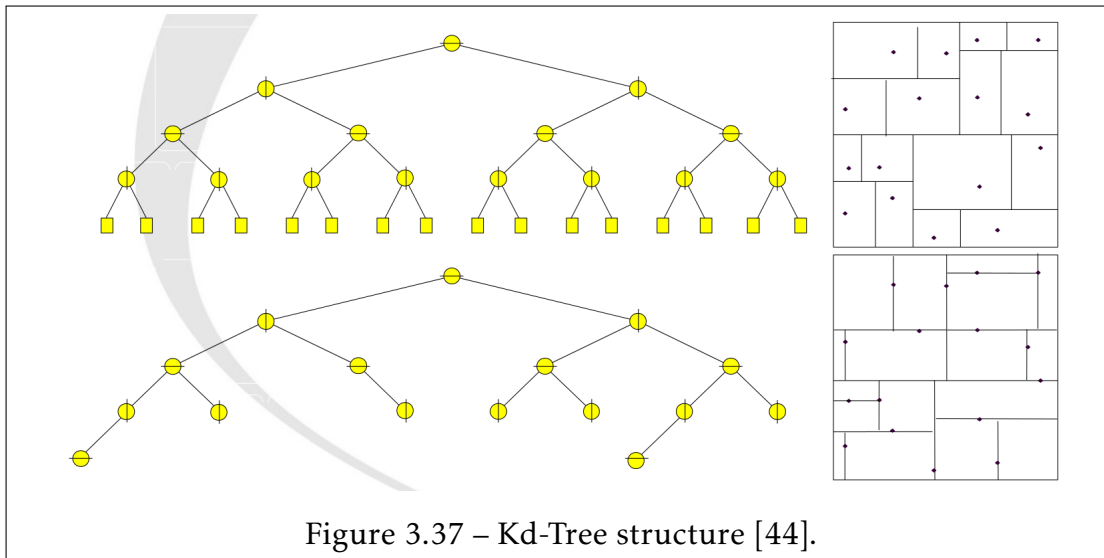
3.4.2 Structure of the serial code

The code is composed of a main program file **sph.f90** and the following modules:

- **Time integration module:** *time_integration.f90*.
  The module contains the different time integration schemes. Up until now, only the Verlet time integration scheme (section 3.1) is implemented.

- **Solver module:** *solver.f90*.
  The module contains the FC-SPH, WC-SPH and FE shell solvers. The BLT and CST shells [48] are implemented but not yet coupled to the SPH solvers. The accelerations and the differential form of the SPH internal energy is computed at the end of the solver module after the computations of the density time derivative and the pressure.

- **Kernel module:** *kernel.f90*.
  The module contains the definition of the different kernel functions and their spatial derivatives (with respect to the position vector $\overrightarrow{x}$ and the smoothing length h) and the smoothing length update equations.

- **Neighbors search module:** *neighbor.f90*.
  The module contains the Kd-tree search [60], the neighbors search and the ghost creations subroutines.

- **Density module:** *density.f* 90.
  The module contains the integral and differential forms evaluation of the SPH density for both the *m-scheme* and *n-scheme*.

- **Internal forces module:** *internal.f* 90.
  The module contains the material models subroutine, the pressure and speed of sound computations subroutine and the evaluation of the SPH internal forces for both the *m-scheme* and *n-scheme*.

- **External forces module:** *external.f* 90.
  The module contains the load body forces subroutine. The contact forces subroutine is not yet implemented.

- **Artificial forces module:** *artificial.f* 90.
  The module contains the first Monaghan-type (1983) [97] artificial dissipation for the WC-SPH subroutine and Monaghan (1997) [96] artificial dissipation terms for the FC-SPH subroutine.

- **Output module:** *output.f* 90.
  The module contains the output subroutines for three post-processing softwares: Scilab, Paraview (inspired from [139]) and LS-Prepost (following the LS-DYNA database manual [47]). The subroutine for LS-DYNA outputs is given in Appendix C.
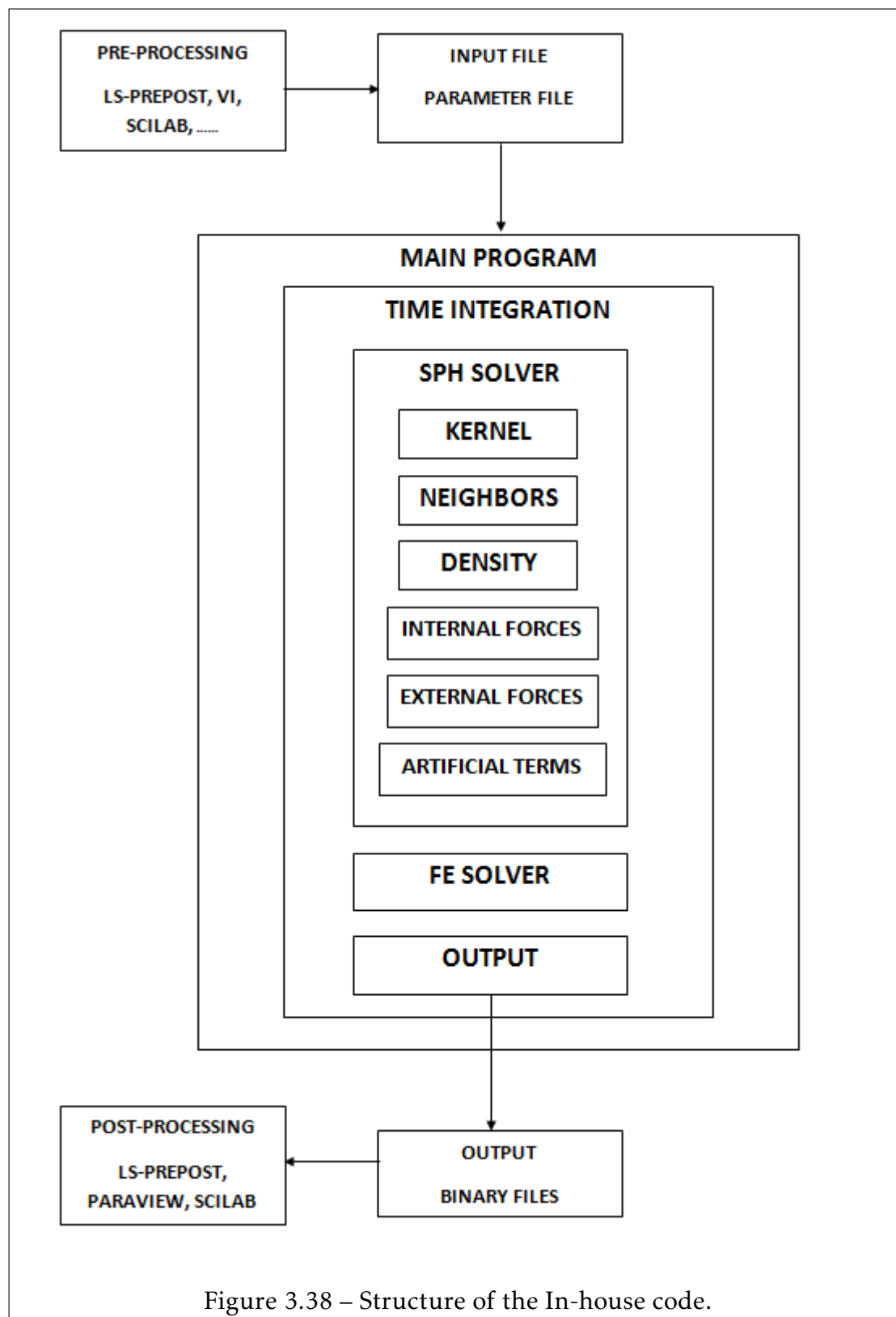
Figure 3.38 – Structure of the In-house code.

Conclusion and Perspective

Conlusion

The art of scientific computing consists in the development and the use of applied mathematical models to solve scientific problems. Numerical methods provide approached solutions to the physics by solving the developed models on computers. Over the last decades, the performance of computers has grown exponentially, and most of the laboratories have their own High Performance Computing (HPC) center. To model complex problems, the scientist needs to also consider the computational resources as an input parameter, since the size of the model directly affects the computational time, the memory storage, and also the accuracy of the solution. Finally, one needs to find the best compromise between the accuracy of the solution and the needs in terms of computation(CPU time, memory ...).
In this thesis, ALE and SPH methods have been investigated to solve FSI problems involving shock wave propagation in multiphase flows and its impact on structure for the purpose of solving a complex industrial problem. Indeed, numerical methods can be included in shape design optimization with shape optimal design techniques (see [136]) and material optimization (see [37]). Once simulations are validated by test results, they can be used as design tools for the improvement of the system structure that is involved.

The ALE and FSI methods, in conjunction with the HEM phase change model have been successfully applied to industrial and academic problems in chapter 2. In the collaboration project with EDF two types of WHs with phase change are

considered:

- The Simpson's [131] classical valve slam WH where the vapor water is generated at the closed valve due to the pressure waves reflections. This problem is very common in nuclear industry, during closing valve for security issues.

- The experimental CWHTF designed for the investigation of the sudden collapse of a water vapor pocket and the induced shock wave.

The implemented HEM model accurately solved the one-dimensional cavitation shock tube test and the first FSI coupled Simpson's WH.

However, in the CWHTF test case few differences were observed between the HEM model results and the reference experimental data in terms of shock waves time of birth and the pressure peak amplitude. The CWHTF industrial problem is very complex, including both structural (supports, vibration, damping), fluid (phase change, collapse of the bubble, shock wave generation and propagation) and FSI (damping effects of the fluid on structure, fluid pressure impact on the structure) coupled physics to be solved. In view of the complex nature of the problem, we can conclude that the obtained results are qualitatively satisfactory but further investigations of the uncoupled and coupled problems need to be done in order to quantitatively improve the convergence to the final solution.

In the collaboration project with Narvik University (Norway), the results from the shock tube experimental setup and the numerical simulations using LS-DYNA proved that under dynamic loading, CFRP E720 composite behaves linearly. A good agreement between strongly coupled simulations and experimental results confirm that FSI capabilities of LS-DYNA can be used to model deforming structures under dynamic loading. In the scope of the project, numerical simulations may help to reduce the total number of experimental tests, which can be money and time consuming.

An alternative method to multi-material ALE formulation to solve shock waves propagation in multiphase flows is the Lagrangian SPH method. A comparison between ALE and SPH methods as well as their limitations for modeling shock

wave propagation in inhomogeneous material is presented in this thesis. For shocks problems such as UNDEX, multi-material ALE methods are commonly solved in defense industry. For the last decade, SPH methods are increasing in accuracy numerical stability, and the use of SPH method is becoming more common in industry to solve fluid structure coupling problems. For instance, in aerospace, in which bird impacts on aircraft are very common and cause significant safety threats to commercial and military aircraft.

To test the performance of the WC-SPH method, we first compared ALE and WC-SPH methods implemented in LS-DYNA for the 2D UNDEX problem using similar mesh size. Each ALE element is replaced by an SPH particle at the element center. It has been observed that using same mesh size for both methods, numerical results, displacement, velocity and Von Mises stress on the structure, are underestimated with the SPH method. When refining the SPH particles, in which each ALE element is replaced by 4 SPH particles in two dimensions and 8 particles in three dimensions, WC-SPH numerical results show good correlation with the ALE simulation results: in terms of displacement, velocity and Von Mises stress on the structure. However, the WC-SPH methods showed severe limitations at modeling shock wave propagation in inhomogeneous media with high density ratio through classical 1-D shock tube.

To overcome the limitations of the WC-SPH method, an in-house SPH code has been developed including the FC-SPH schemes based on the work of Price [113, 114, 22, 115] and Zisis [167, 168, 172]. The extension of the *n-scheme* [54] to the fully-compressible regime has allowed us to extend the scope of the SPH method to shock wave problems in inhomogeneous media with a high density ratio.

The major drawback of the FC-SPH schemes is their diffusivity compared to ALE methods. A possible strategy to reduce diffusion in the FC-SPH method is to add switches to the artificial dissipative terms [100]. Another strategy to reduce the diffused region is to refine the number of particles at the cost of increasing the overall CPU time, which is not to be neglected when the method is applied to real-life engineering problems.

Perspective

The presented thesis works gave arise to additional future works in both ALE and SPH methods in order to be applied to more general engineering applications involving shock waves in multiphase media, with our without phase change.

Among the possible future works that can improve of the presented ALE method, we can cite :

- **Additionnal validation of the HEM models:** The presented HEM model [124] failed to quantitatively validate the method for the CWHTF test case, and in particular, the collapse of the initial saturated vapor bubble. Further investigations are needed on the different HEM models [124, 126, 84, 157], and also in the use of the multi-material ALE formulations combined with HEM models.

- **Investigation of other types of WHs:** We presented in this thesis only two types of WHs. Among the different types of WHs, one can cite: the *"Water Canon"* WH, the *"Column junction"* WH, the *"Steam/Water Counter Flow"* WH, and more.

Among the possible future works that can improve of the developed in-house FC-SPH code, we can cite:

- **Implementation of the Fluid Structure Coupling:** *SPH-to-SPH* coupling, in which the structure is modeled by SPH and *SPH-to-FEM*, in which the Structure is modeled by FEM.

- **Implementation of the Deviatoric Stresses.**

- **Optimization of the in-house FC-SPH code:** It was shown that, in practice, the FC-SPH methods are less accurate than the ALE method. One straightforward way to increase the accuracy of the method is to increase the number of particles in the simulation, which implies a considerable increase of the overall computational time.

In High Performance Computing (HPC), both *MPI* and *OpenMP* models have advantages and drawbacks. To get the best of both models, one needs to consider hybrid modeling by mixing *MPI* and *OpenMP*. The strategy for the evolution of the code is to combine a domain decomposition using the *MPI* model by splitting the computational domain in smaller domains which are allocated to each computational nodes, and to perform on each subdomain a nearest-neighbor search using the *OpenMP* shared memory model.

A first *C++* version of the in-house FC-SPH code has been developed in order to use the *OpenMP* model with the *C++* version of the KD-TREE2 [60]. The optimization of the *OpenMP* version is ongoing before the consideration of the domain decomposition through the use of the *MPI* model.

Bibliography

[1] H. U. Mair A. B. Wardlaw. "Spherical solutions of an underwater explosion bubble". In: *Shock and Vibration* 5 (1998), pp. 89–102.

[2] J. A. Luton A. B. Wardlaw. "Fluid-structure interaction mechanisms for close-in explosions". In: *Shock and Vibration* 7 (2000), pp. 265–275.

[3] J. E. Van Aanhold, G. J. Meijer, and P.P.M Lemmen. "Underwater shock response analysis of a floating vessel". In: *Shock and Vibration* 5 (1998), pp. 53–59.

[4] ABAQUS. "Abaqus 6.13 Theory Manual". In: v6.13 (2013).

[5] V. Ahuja, V. Hosangadi, and S. Arunajatesan. "Simulation of cavitation flows using hybrid unstructured meshes". In: *Journal of Fluids Engineering* 40 (2001), pp. 123–331.

[6] A. Alia and M. Souli. "High explosive simulation using multi-material formulations". In: *Applied Thermal Engineering* 26 (2005), pp. 1032–1042.

[7] E. Alstadt, H. Carl, and R. Weiss. *Fluid-Structure Interaction Investigations for Pipelines*. Tech. rep. Technical Report FZR-393. Forschungszentrum, Rossendorf, 2003.

[8] A. Amsden and C. Hirt. *YAQUI : An arbitrary Lagrangian-Eulerian computer program for fluid flow at all speeds*. Tech. rep. Los Alamos Scientific Laboratory, LA-5100, 1973.

[9] ANSYS. "Theory Reference, in Structures, Static Analysis". In: *Academic Research* release 14.0 (2014).

[10] N. Aquelet, M. Souli, and L. Olovsson. "Euler–Lagrange coupling with damping effects: Application to slamming problems". In: *Computer Methods in Applied Mechanics and Engineering* 195.1—3 (2005), pp. 110–132.

[11] F. Dubois B. Després. *Systèmes hyperboliques de lois de conservation: application à la dynamique des gaz*. Editions Ecole Polytechnique, 2005. ISBN: 978-2-7302-1253-3.

[12] G. Barras et al. "Numerical simulation of underwater explosions using ALE method. The pulsating bubble phenomena". In: *Ocean Engineering* 41 (2012), pp. 53–66.

[13] T. Belytschko, J. Lin, and C. Tsay. "Explicit algorithms for the nonlinear dynamics of shells". In: *Computer Methods in Applied Mechanics and Engineering* 42 (1984), pp. 225–251.

[14] T. Belytschko and M. O. Neal. "Contact-impact by the pinball algorithm with penalty, projection, and Lagrangian methods". In: *Proceedings of the symposium on computational techniques for impact, penetration, and perforation of solids AMD, New York ASME* 103 (1989), pp. 97–140.

[15] T. Belytschko et al. *Nonlinear Finite Elements for Continua and Structures*. 2nd ed. Chichester, New York, Weinheim: J. Wiley & sons, 2013. ISBN: 978-1-118-63270-3.

[16] D.J. Benson. "An efficient, accurate, simple ALE method for nonlinear finite element programs". In: *Computer Methods in Applied Mechanics and Engineering* 72 (1989), pp. 305–350.

[17] D.J. Benson. "Computational Methods in Lagrangian and Eulerian Hydrocodes". In: *Computer Methods in Applied Mechanics and Engineering* 99.2-3 (1992), pp. 235–394.

[18] J.L. Bentley. "Multidimensional binary search trees used for associated searching". In: *In Communications of the ACM* 18 (1975), pp. 509–517.

[19] W. Benz. "Smoothed particle hydrodynamics: a review". In: *NATO Workshop, Les Arcs FRANCE* (1989).

[20] A. Bergant, A. Simpson, and A. S. Tijsseling. "Water hammer with column separation: A historical review". In: *Journal of Fluids and Structures* 22.2 (2006), pp. 135–171.

[21] L. Bergerat. "Développement d'une méthode numérique compressible pour la simulation de la cavitation en géométrie complexe". PhD thesis. Arts et Métiers ParisTech, France, 2012.

[22] S. Borve and D. J. Price. "Hydrodynamical instabilities in compressible fluids using SPH". In: *Proceedings of the 4th SPHERIC workshop, May 26-29, Hannover* (2009).

[23] A. Le Bot. *Foundation of Statistical Energy Analysis in Vibroacoustics*. OUP Oxford, 2015.

[24] D. M. Causon and C. G. Mingham. "On the Numerical Modelling of Cavitation and Violent Wave Interaction". In: *Proceedings of the Twenty-first International Offshore and Polar Engineering Conference, Hawaii June 2011, USA* (2011).

[25] J. M. Cherfils, G. Pinon, and E. Rivoalen. "JOSEPHINE: A parallel SPH code for free-surface flows". In: *Computer Physics Communications* 183.7 (2012), pp. 1468–1480.

[26] A. J. Chorin, T. J. R. Hughesand M. F. McCracken, and J. E. Marsden. "Product formulas and numerical algorithms". In: *Communications on Pure and Applied Mathematics* 31.2 (1978), pp. 205–256. ISSN: 1097-0312. URL: http://dx.doi.org/10.1002/cpa.3160310205.

[27] S. Clerc. "Numerical Simulation of the Homogeneous Equilibrium Model for Two-Phase Flows". In: *Journal of Computational Physics* 161 (2000), pp. 354–375.

[28] A. Colagrossi and M. Landrini. "Numerical simulation of interfacial flows by smoothed particle hydrodynamics". In: *Journal of computational Physics* (2015), pp. 191–448.

[29] R. Courant, K. Friedrichs, and H. Lewy. "On the partial difference equations of mathematical physics". In: *IBM Journal of Reasearch and Development* 11.2 (1967), pp. 215–234.

[30] R. Courant, K. Friedrichs, and H. Lewy. "Über die partiellen Differenzengleichungen der mathematischen Physik". German. In: *Mathematische Annalen* 100.1 (1928), pp. 32–74. ISSN: 0025-5831.

[31] F. Crouzet et al. "Validation of a two-fluid model on unsteady liquid–vapor water flows". In: *Computers & Fluids* 119.2 (2015), pp. 131–142.

[32] P. V. Cueille. "Modélisation par Smoothed Particle Hydrodynamic des phénomènes de diffusion présents dans un écoulement". PhD thesis. Université de Toulouse 3 INSA, France, Dec. 2005.

[33] R. Davé, J. Dubinski, and L. Hernquist. "Parallel TreeSPH". In: *New Astronomy* 2.13 (1997), pp. 277–297.

[34] B. M. Dobratz and P. C. Crawford. *LLNL Explosives Handbook: Properties of Chemical Explosives and Explosive Simulants*. Report UCRL-5299 Rev.2. University of California: Lawrence Livermore National Laboratory, University of California; Springfield (USA), Jan. 1985.

[35] M. Doring. "Développement d'une méthode SPH pour les applications à surface libre en hydrodynamique". PhD thesis. Ecole Centrale de Nantes, France, 2006.

[36]   D. Enright et al. "A Hybrid Particle Level Set Method for Improved Interface Capturing". In: *Journal of Computational Physics* 183 (2002), pp. 83–116.

[37]   F. Erchiqui, M. Souli, and R. Ben Yedder. "Non isothermal finite-element analysis of thermoforming of polyethylene terephthalate sheet: Incomplete effect of the forming stage". In: *Polymer Engineering and Science* 47 (2007), pp. 2129–2144.

[38]   D. P. Flanagan and T. Belytschko. "A Uniform Strain Hexahedron and Quadrilateral and Orthogonal Hourglass Control". In: *International Journal for Numerical Methods in Engineering* 17 (1981), pp. 679–706.

[39]   *FP7 European Project SAFUEL.* http://www.safuel-fp7.eu.

[40]   J. Gale and I. Tiselj. "Water Hammer in elastic pipes". In: *International Conference Nuclear Energy for New Europe* (2002).

[41]   M. Giot et al. *Two-Phase Flow Water Hammer Transients and Induced Loads on Materials and Structures of Nuclear Power Plants.* Tech. rep. WA-HALoads, contract FIKS-CT-2000-00106. 2000.

[42]   E. D. Giroux. *HEMP USER'S MANUAL.* Lawrence Livermore Laboratory. University of California; Livermore (USA), Dec. 1973.

[43]   S. K. Godunov. "A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations". In: *Math. Sbornik* 47.3 (1959).

[44]   X. Granier. *Lecture : Bibliothèques et Architectures pour le Calcul, la Modélisation et la Sunthèse d'Images.*

[45]   N. Grenier et al. "An Hamiltonian interface SPH formulation for multi-fluid and free surface flows". In: *Journal of Computational Physics* 228.22 (2009), pp. 8380–8393.

[46]   H. Guillard and M. Labois. "NUMERICAL MODELLING OF COMPRESSIBLE TWO-PHASE FLOWS". In: *ECCOMAS CFD 2006, TU Delft, The Nederlands* (2006).

[47]   J. O. Hallquist. *LS-DYNA Database Binary Output files.* LSTC, Livermore Software Technology Corporation. 7374 Las Positas Road; Livermore CA 94551 (USA), 2011.

[48]   J. O. Hallquist. *LS-DYNA Theory Manual.* LSTC, Livermore Software Technology Corporation. 7374 Las Positas Road; Livermore CA 94551 (USA), Aug. 2015.

[49] F. H. Harlow and A. A. Amsden. "On finite-difference approximations and entropy conditions for shocks". In: *Journal of Computational Physics* 18 (1975), pp. 440–464.

[50] A. Harten, J. M. Hyman, and P. D. Lax. "On finite-difference approximations and entropy conditions for shocks". In: *Communications on Pure and Applied Mathematics* 29 (1976), pp. 297–322.

[51] A. Haufe, K. Schweizerhof, and P. DuBois. *Properties & Limits : Review of Shell Element Formulations*. Mar. 2013.

[52] A. Haufe, K. Weimar, and U. Göhner. "Advanced Airbags simulation using Fluid-Structure-Interaction and the Eulerian Method in LS-DYNA". In: *LS-DYNA Anwenderforum, Bamberg* (2004).

[53] S. Heb and V. Springel. "Particle hydrodynamics with tessellation techniques". In: *MNRAS* 406 (2010), pp. 2289–2311.

[54] X. Y. Hu and N. A. Adams. "A multi-phase SPH method for macroscopic and mesoscopic flows". In: *Journal of Computational Physics* 213 (2006), pp. 844–861.

[55] H. Ji et al. "Design of water shock tube for testing shell materials". In: *World Journal of Engineering* 11.1 (2014), pp. 50–60.

[56] D. Joahansson. "Statistical Energy Analysis software Development and implementation of an open source code in Matlab/Octave". PhD thesis. CHALMERS University of Technology, Göteborg Sweden, 2010.

[57] H. Jones and A. R. Miller. "Detonation of Solid Explosives: The Equilibrium Conditions in the Detonation Wave-Front and the Adiabatic Expansion of the Products of Detonation". In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 194.1039 (1948), pp. 480–507.

[58] N. Jones and T. Wierzblcki. "Structural Aspects of Ship Collisions". In: *Structural Crashworthiness, Edited by N. Jones and T. Wierzblcki, Butterworths, London* (1983), pp. 308–337.

[59] M. Kempeneers. "Eléments finis statiquement admissibles et estimation d'erreur par analyse duale". PhD thesis. Allée de la Découverte, 12 Sart Tilman - 4000 Liège - Belgique: Université de Liège Faculté des Sciences Appliquées, Dec. 2005.

[60] M.B Kennel. "KDTREE 2: Fortran 95 and C++ software to efficiently search for near neighbors in a multi-dimensional Euclidean space". In: (2004).

[61]   F. Khatami, E. T. A. van der Weide, and H. W. M. Hoeijmakers. "Numerical Issues in Higher-Order accurate simulations of flows with vortex cavitations". In: *WCCM XI, ECCM V, ECFD VI, 20-25 July 2014 Barcelona Spain* (2014).

[62]   H.A. Khawaja, J. Kapaya, and M. Moatamedi. "Shock Tube Detail overview of equipment and instruments in the shock tube experimental setup". In: *Lambert Academic Publishing* (2015).

[63]   H.A. Khawaja and M. Moatamedi. "Multiphysics Investigation of Composite Shell Structures Subjected to Water Shock Wave Impact in Petroleum Industry". In: *Materials Science Forum* (2013).

[64]   H.A. KHAWAJA et al. "Experimental and numerical study of pressure in a shock tube". In: *Journal of Pressure Vessel Technology* (2015).

[65]   H.A. KHAWAJA et al. "Multiphysics Analysis of Shell Structures Subjected to Water Shock Wave". In: *NAFEMS European Conference: Multiphysics Simulation 2012, October 2012 Frankfurt (GERMANY)*. 2012.

[66]   H.A. Khawaja et al. "Study of CFRP Shell Structures under Dynamic Loading in Shock Tube Setup". In: *Journal of Structures* (2014).

[67]   B. C. Khoo and J. G. Zheng. "The numerical simulation of unsteady cavitation evolution induced by pressure wave". In: *International Journal of Modern Physics: Conference Series* 34 (2014).

[68]   C. Klug et al. "Experimental Sphere in Sphere Testing for the Validation of a Numerical Cerebrospinal Fluid Model". In: *2013 IRCOBI Conference Proceedings in Gothenburg, Sweden* (2013).

[69]   D. Kosloff and G. A. Frazier. "Treatment of Hourglass Patterns in Low Order Finite Element Codes". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 2 (1978), pp. 57–72.

[70]   R. D. Krieg and S. W. Key. "Implementation of a Time Dependent Plasticity Theory into Structural Computer Programs". In: *Constitutive Equations in Viscoplasticity: Computational and Engineering Aspects, eds* 20 (1976).

[71]   M. Labois. "Modélisation des déséquilibres mécaniques dans les écoulements diphasiques : Approches par relaxation et par modèle réduit". PhD thesis. Université de Provence (Aix-Marseille I), France, 2008.

[72]   R. Landshoff. *A numerical method for treating fluid flow in the presence of shocks*. Tech. rep. LA-1930. Los Alamos National Laboratory Report, 1955.

[73]   R. S. Langley and K. H. Heron. "A Wave Intensity Technique for Analysis of High Frequency Vibrations". In: *Journal of Sound and Vibration* 159.3 (1992), pp. 483–502.

[74]   R. S. Langley and K. H. Heron. "Elastic Wave Transmission Through Plate/Beam Junction". In: *Journal of Sound and Vibration* 143.2 (1990), pp. 241–253.

[75]   N. Lanson. "Etude des méthodes particulaires renormalisées. Applications aux problèmes de dynamique rapide". PhD thesis. Université de Toulouse 3 INSA, France, 2008.

[76]   P. D. Lax. "Weak solutions of nonlinear hyperbolic equations and their numerical computation". In: *Communications on Pure and Applied Mathematics* 7 (1954), pp. 159–193.

[77]   J. Leblanc and A. Shukla. "Dynamic response and damage evolution in composite materials subjected to underwater explosive loading: An experimental and computational study". In: *Composite Structures* 92 (2010), pp. 2421–2430.

[78]   E. L. Lee, H. C. Horning, and J. W. Kury. *Adiabatic expansion of high explosive detonation product*. Tech. rep. REPORT UCRL-50422. Radiation Laboratory Livermore, 1968.

[79]   B. van Leer. "Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection". In: *Journal of Computational Physics* 23.3 (1977), pp. 276–299.

[80]   B. van Leer. "Towards the ultimate conservative difference scheme. V. A second order sequel to Godunov's method." In: *Journal of Computational Physics* 32.1 (1979), pp. 101–136.

[81]   C. Lemaire. "Triangulation de Delaunay et arbres multidimensionnels". PhD thesis. Université Jean Monnet et Ecole des Mines de Saint-Etienne, FRANCE, 2013.

[82]   P. J. Linstrom and W. G. Mallard. *NIST Chemistry WebBook, NIST Standard Reference Database Number 69*. National Institute of Standards and Technology, Gaithersburg MD, 20899, 2015. URL: http://webbook.nist.gov.

[83]   G. R. Liu and M. B. Liu. *Smoothed Particle Hydrodynamics: A Meshfree Particle Method*. World Scientific, 2003. ISBN: 0-520-04008-2.

[84]   T. G. Liu, B. C. Khoo, and W.F. Xie. "Isentropic one-fluid modelling of unsteady cavitating flow". In: *Journal of Computational Physics* 201 (2004), pp. 80–108.

[85] J. Lopez and J. Hernandez. "Analytical and geometrical tools or 3D volume of fluid methods in general grids". In: *Journal of Computational Physics* 227 (2008), pp. 5939–5948.

[86] L. B. Lucy. "A numerical approach to the testing of fission hypothesis". In: *The Astronomical Journal* 82 (1977), pp. 1013–1024.

[87] R. H. Lyon and R. G. DeJong. *Theory and Application of Statistical Energy Analysis, Second Edition*. Butterworth-Heinemann, 2 edition, 1994.

[88] S.P. Marsh. *LASL shock Hugoniot data*. Berkeley and Los Angeles, California: University of California Press, 1980. ISBN: 0-520-04008-2.

[89] R. MESSAHEL, M. MOATAMEDI, and M. SOULI. "FSI in Petroleum and Nuclear industries". In: *MULTIPHYSICS 2011 CONFERENCE, Barcelona (SPAIN)*. 2011.

[90] R. Messahel and M. Souli. "SPH and ALE Formulations for Fluid Structure Coupling". In: *Computer Modeling in Engineering and Sciences* 96 (2013), pp. 435–455.

[91] R. MESSAHEL et al. "Numerical and experimental investigations of water hammers in nuclear industry". In: *The International Journal of Multiphysics* 9 (2015), pp. 21–36.

[92] R. MESSAHEL et al. "Numerical Investigation of cavitation effects in nuclear power plant pipes". In: *MULTIPHYSICS 2013 CONFERENCE, Amsterdam (THE NETHERLANDS)*. 2013.

[93] R. Messahel et al. "Fluid-structure interaction for water hammers effects in petroleum and nuclear plants". In: *The International Journal of Multiphysics* 5 (2011), pp. 377–386.

[94] J. J. Monaghan. "Smoothed Particle Hydrodynamics". In: *Annual review Astronomy and Astrophysics* 30 (1992), pp. 543–574.

[95] J. J. Monaghan. "Smoothed particle hydrodynamics". In: *Reports on Progress in Physics* 68.8 (2005).

[96] J. J. Monaghan. "SPH and Riemann Solvers". In: *Journal of Computational Physics* 136.2 (1997), pp. 298–307.

[97] J. J. Monaghan and R. A. Gingold. "Shock Simulation by particle method SPH". In: *Journal of Computational Physics* 52 (1983), pp. 374–389.

[98] J. J. Monaghan and A. Kocharyan. "SPH simulation of multi-phase flow". In: *Computer Physics Communications* 87 (1995), pp. 225–235.

[99] J. J. Monaghan and A. Rafiee. "A simple SPH algorithm for multi-fluid flow with high density ratios". In: *International Journal for Numerical Methods in Fluids* 71.5 (2013), pp. 83–100.

[100] J. P. Morris and J. J. Monaghan. "A switch to reduce SPH viscosity". In: *Journal of Computational Physics* 136 (1997), pp. 41–50.

[101] D. Mount and S. Arya. *ANN: Approximate Nearest Neighbors.* `https://www.cs.umd.edu/~mount/ANN/`.

[102] J. Von Neumann and R.D. Richtmeyer. "A method for the numerical calculation of hydrodynamical shocks". In: *Journal of Applied Physics* 21 (1950), pp. 232–257.

[103] W. F. Noh and P. Woodward. *SLIC (Simple Line Interface Calculation).* Tech. rep. UCRL-77651. Livermore, California, USA: Lawrence Livermore Laboratory, University of California, 1976.

[104] G. Oger. "Aspects théoriques de la méthode SPH et applications a l'hydrodynamique a surface". PhD thesis. Ecole Centrale de Nantes, France, 2006.

[105] L. Olovsson. *LS-DYNA Training class in ALE and fluid-structure interaction.* Sept. 2006.

[106] S. Osher and J. A. Sethian. "Front Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi". In: *Journal of Computational Physics* 79 (1988), pp. 12–49.

[107] M. Otsuka et al. "Study on development of vessel for shock pressure treatment for food". In: *International Journal of Multiphysics* 1 (2007), pp. 69–84.

[108] Z. Ozdemir, Y.M. Fahjan, and M. Souli. "Numerical evaluation of non-linear response of broad cylindrical steel tanks under multidimensional earthquake motion". In: *Earthquake Spectra* 28.1 (2012), pp. 217–238.

[109] H. Paillère, C. Corre, and J. R. Garcia Cascalesi. "On the Extension of the AUSM+ Scheme to Compressible Two-Fluids Models". In: *Journal of Computers and Fluids* 32.6 (2003), pp. 891–916.

[110] I. Pericevic and M. Moatamedi. "Finite element analysis of footwear and ground interaction". In: *Strain* 41.3 (2005), pp. 113–115.

[111] S. P. van der Pijl et al. "A mass-conserving Level-Set method for modelling of multi-phase flows". In: *International Journal for Numerical Methods in Fluids* 47.4 (2004), pp. 339–361.

[112] L. R. Plumerault. "Numerical modelling of aerated-water wave impacts on a coastal structure". PhD thesis. Université de Pau et des Pays de l'Adour, FRANCE, 2009.

[113] D. J. Price. "Magnetic fields in Astrophysics". PhD thesis. Institute of Astronomy, Madingley Rd Cambridge, UK, 2004.

[114]   D. J. Price. "Modelling discontinuities and kelvin-helmholtz instabilities in SPH". In: *Journal of Computational Physics* 227.24 (2008), pp. 10040–10057.

[115]   D. J. Price. "Smoothed Particle Hydrodynamics and Magnetohydrodynamics". In: *Journal of Computational Physics* 231.3 (2012), pp. 759–794.

[116]   K. Puri and P. Ramachandran. "Approximate Riemann solvers for Godunov SPH (GSPH)". In: *Journal of Computational Physics* 270 (2014), pp. 432–458.

[117]   *PySPH: a Python-based SPH framework*. https://bitbucket.org/pysph/pysph.

[118]   RADIOSS. "Radioss Theory Manual v10.0, ALE, CFD, SPH". In: PART 3 (2009).

[119]   RADIOSS. "Radioss Theory Manual v10.0, Large Displacement Finite Element Analysis". In: PART 1 (2009).

[120]   RADIOSS. "Radioss Theory Manual v10.0, Large Displacement Finite Element Analysis". In: PART 2 (2009).

[121]   P. W. Randles and L. D. Libersky. "Smoothed Particle Hydrodynamics: Some recent improvements and applications". In: 139 (1996), pp. 375–408.

[122]   S. Rosswog. "Astrophysical Smooth Particle Hydrodynamics". In: *New Astronomy Review* 53 (2009).

[123]   E. Sarradj. *FreeSEA a Statistical Energy Analysis Freeware*. http://www.free-sea.de/.

[124]   R. Saurel, J. P. Cocchi, and P. B. Butler. "Numerical Study of Cavitation in the Wake of a Hyper-velocity Underwater Projectile". In: *Computer Methods in Applied Mechanics and Engineering* 15.4 (1999), pp. 513–522.

[125]   R. Saurel, F. Petitpas, and R. A. Berry. "Simple and Efficient Relaxation for Interfaces Separating Compressible Fluids, Cavitating Flows and Shocks in Multiphase Mixtures". In: *Journal of Computational Physics* 228 (2009), pp. 1678–1712.

[126]   D. P. Schmidt, C. J. Rutland, and M. L. Corradini. "A fully compressible, two-dimensional model of small, high speed, cavitating nozzles". In: *Atomization Sprays* 9 (1999), pp. 255–276.

[127]   E. Schmidt and U. Grigull. *Properties of Water and Steam in SI-Units*. 4th ed. Springer-Verlag Berlin Heidelberg New York, R. Oldenbourg München, 1989.

[128]   I. Senocak and W. Shyy. "A pressure-based method for turbulent cavitating flow computations". In: *Journal of Computational Physics* 176 (2002), pp. 363–383.

[129]   M. Serrano, P. Espanol, and I. Zuniga. "Voronoi Fluid Particle Model for Euler Equations". In: *Journal of Statistical Physics* 121 (2005), pp. 133–147.

[130]   I. H. Sezal. "Compressible Dynamics of Cavitating 3-D Multi-Phase Flows". PhD thesis. Technische Universität München, Germany, Mar. 2009.

[131]   A. R. Simpson. "Large water hammer pressures due to column separation in sloping pipes (transient,cavitation". PhD thesis. University of Michigan, USA, 1986.

[132]   G. A. Sod. "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.

[133]   M. Souli. *LS-DYNA advance course in ALE and Fluid-Structure Coupling*. Mar. 2010.

[134]   M. Souli and D.J. Benson. *Arbitrary Lagrangian Eulerian and Fluid-Structure Interaction: Numerical Simulation*. Wiley-ISTE, 2010. ISBN: 978-1-84821-131-5.

[135]   M. Souli and J. Gabrys. "Fluid Structure Interaction for Bird Impact Problem: Experimental and Numerical Investigation". In: *Computer Modeling in Engineering & Sciences* 2137.1 (2012), pp. 1–16.

[136]   M. Souli and J.P. Zolesio. "Shape Derivative of Discretized Problems". In: *Computer Methods in Applied Mechanics and Engineering* 108 (1993), pp. 187–199.

[137]   M. Souli et al. "A New Methodology for Fuel Mass Computation in an operating Aircraft". In: *MULTIPHYSICS 2015 CONFERENCE, LONDON*. 2015.

[138]   M. Souli et al. *Statistical Energy Acoustic for High Frequencies Analysis*. 2015.

[139]   *SPHysics*. https://wiki.manchester.ac.uk/sphysics/index.php/SPHYSICS_Home_Page.

[140]   Z. Sun, D. Howard, and M. Moatamedi. "Application of the penalty coupling method for the analysis of blood vessels". In: *European Journal of Computational Mechanics* 16.3–4 (2007), pp. 537–548.

[141] M. Sussman and E. G. Puckett. "A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows". In: *Journal of Computational Physics* 162 (2000), pp. 301–337.

[142] H. S. Tang and D. A. Huang. "A second-order accurate capturing scheme for 1D inviscid flows of gas and water with vacuum zones". In: *Journal of Computational Physics* 128 (1996), pp. 301–318.

[143] L. M. Taylor and D. P. Flanagan. *PRONTO 3D A Three-Dimensional Transient Solid Dynamics Program*. Tech. rep. Sandia Report: SAND87-1912, UC-32. Sandia Laboratory, 1989.

[144] P. A. Thompson. *compressible Fluid Dynamics*. McGraw-Hill, 1972.

[145] P. Tomov et al. "Numerical Modeling of aerated cavitating using compressible Homogeneous Equilibrium Model". In: *SimHydro 2014, Sophia Antipolis, France* (2014).

[146] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics : a practical introduction*. Berlin, New York: Springer, 1997. ISBN: 3-540-61676-4. URL: http://opac.inria.fr/record=b1093563.

[147] S. Venkateswaran et al. "Computation of multiphase mixture flows with compressible effects". In: *Journal of Computational Physics* 180 (2002), pp. 54–77.

[148] R. Vignjevic, J. Reveles, and J. Campbell. "SPH in a Total Lagrangian Formalism". In: *Computer Modelling in Engineering and Science* 14 (2006), pp. 181–198.

[149] J. P. Vila. "On particle weighted method and smoothed particle hydrodynamics". In: *Mathematical Models and Method in Applied Science* 9 (1999), pp. 161–209.

[150] J. P. Vila. "SPH renormalized hybrid methods for conservation laws: applications to free surface flows". In: *Mesh free Methods for Partial Differential Equations II* 43 (2005). Lecture Notes in Computational Science and Engineering. Springer.

[151] D. Violeau. *Computational Fluid Dynamics*. CRC Press, 2011.

[152] D. Violeau. *Fluid Mechanics and the SPH Method*. Oxford University Press, 2012.

[153] W. Wagner. *Properties of Water and Steam: The Industrial Standard IAPWS-IF97 for the Thermodynamic Properties and Supplementary Equations for Other Properties in SI-Units*. Springer-Verlag, 1998.

[154] G. B. Wallis. *One-dimensional two-phase flow*. McGraw-Hill, 1969.

[155]   M. L. Wilkins. *The equation of state of PBX 9404 and LX04-01*. Tech. rep. REPORT UCRL-7322(Rev.1). Radiation Laboratory Livermore, 1964.

[156]   W. Xie. "A NUMERICAL SIMULATION OF UNDERWATER SHOCK-CAVITATION-STRUCTURE INTERACTION". PhD thesis. National University of Singapore, Singapore, 2005.

[157]   W.F. Xie, T. G. Liu, and B. C. Khoo. "Application of a one-fluid model for large scale homogeneous unsteady cavitation : The modified Schmidt model". In: *Computers and Fluids* 35 (2006), pp. 1177–1192.

[158]   W.F. Xie, T. G. Liu, and B. C. Khoo. "The simulation of cavitating flows induced by underwater shock and free surface interaction". In: *Applied Numerical Mathematics* 57 (2007), pp. 734–745.

[159]   J. Xu and J. Wang. "Interaction Methods for the SPH Parts". In: *13th International LS-DYNA Conference* (2014).

[160]   M. Yildiz, R. A. Rook, and A. Suleman. "SPH with the multiple boundary tangent method". In: *International Journal for Numerical Methods in Engineering* 77.10 (2009), pp. 1416–1438.

[161]   D. L. Youngs. *An interface tracking method for a 3D Eulerian hydridynamics code*. Tech. rep. AWRE/44/92/3. Berkshire, United Kingdom: Atomic Weapon Research Establishment, Aldermaston, 1987.

[162]   D. L. Youngs. *Time-dependent multi-material flow with large fluid distortion*. K.W. Mortron and M. J. Baines. Academic Press, New York (USA): Numerical Methods for Fluids Dynamics, ED Morton, 1982.

[163]   B. Zakrissona, B. Wikmana, and H. Häggbladb. "Numerical simulations of blast loads and structural deformation from near-field explosions in air". In: *International Journal of Impact Engineering* 38.7 (2011), pp. 597–612.

[164]   L. Zhang, R. Makwana, and S. Sharma. "Brain Response to Primary Blast Wave Using Validated Finite Element Models of Human Head and Advanced Combat Helmet". In: *frontiers in NEUROLOGY* 19.1—12 (2013), pp. 88–100.

[165]   Z. Zhang, H. Qiang, and W. Gao. "Coupling of smoothed particle hydrodynamics and finite element method for impact dynamics simulation". In: *Engineering Structures* 33 (2011), pp. 255–264.

[166]   Z.H. Zhong. "Finite Element Procedures For Contact-Impact Problems". In: *Oxford Science Publications* (1993).

[167] I. Zisis, B. Linden, and C. Giannopapa. "Towards an SPH algorithm for shocks through layered materials". In: *Proceedings of the ASME 2013 Pressure Vessels and Piping Conference* (2013).

[168] I. Zisis et al. "On the derivation of SPH schemes for shocks through inhomogeneous media". In: *The International Journal of Multiphysics* 9.2 (2015).

[169] I. Zisis et al. "Smoothed Particle Hydrodynamics for Hypervelocity Impacts into Inhomogeneous Materials". In: *Proceedings of the ASME 2014 Pressure Vessels and Piping Conference* (2014).

[170] I. Zisis et al. "SPH for shocks through inhomogeneous media". In: *GDRI-IFS Conference on SPH and Particular Methods for : Fluids and Fluid Structure Interaction, January 2015 Lille University (FRANCE)*. 2015.

[171] *Two phase benchmarks for SPH compressible scheme*. 2014.

[172] I. Zisis et al. "Validation of robust SPH schemes for fully compressible multiphase flows". In: *The International Journal of Multiphysics* 9.3 (2015).

SAFUEL-The SAfer FUEL system FP7 project: WP4.3 Hydrodynamic Sensing

A.1 Context

Most commercial aircrafts (A/C) have their fuel tank located within the wing. The shape of the tank has a complex geometry, and thus some techniques are required to accurately measure the fuel mass inside the tank.
Several techniques have been developed for measurement of the fuel height. Capacitance gauging has been used for decades in aeronautic. The industry has almost universally accepted this method of gauging as the way to gauge fuel quantity accurately. The success of capacitance gauging systems is mainly related to their compatibility and longevity in the relative hostile aircraft environment. In this appendix, we present the developed methodology based on fuel pressure gauging for fuel mass computation in aircraft fuel tanks. This work was achieved in collaboration with Zodiac Aerospace and Airbus.

To validate our methodology of fuel mass computation, a computational method using FEM is performed to model a fuel tank filling phase. Using numerical results, pressure values can be extracted from any location of the simulated fuel tank. These pressure values will be used as input data for the validation of developed software package that computes the fuel mass based on pressure measurements and a finite element mesh.

In this appendix, we first present the hydrodynamic equations governing the

dynamic of the fluid during the filling process, where the flow is assumed incompressible and the fuel a viscous fluid. To compute accurately the free surface or the interface between the fuel and the surrounding air in the tank, numerical interface reconstruction techniques are performed.

Secondly, we present the numerical simulation of an on-ground real-life fuel tank refueling phase. The numerical simulation will provide the total fuel mass that will be considered as a reference solution and the input pressure measurements needed for the fuel mass computation developed software in absence of experimental data.

Finally, the method used in the software package, for the fuel mass computation based on pressure measurements and an embedded FE mesh using a convex polygon volume intersection algorithm [85].

A.2   Hydrodynamic Equations and Level Set Method

To solve the fuel tank refueling problem the fuel can be considered incompressible or nearly incompressible. The choice between this two physical assumptions will lead to a different numerical algorithm, explicit time integration scheme for nearly incompressible flow and implicit time integration scheme for incompressible flow, and thus two possible numerical strategies have been used for computing fuel mass in an operating aircraft.

Fuel dynamic behavior inside the tank can be modeled using classical CFD equations to predict fluid velocity and pressure at any location inside the tank:

$$\rho \frac{d\vec{v}}{dt} = -\nabla_{\vec{x}} p + \mu \left(\nabla^2_{\vec{x}} \vec{v} \right) + \frac{1}{3} \mu \left(\nabla_{\vec{x}} \left(\nabla_{\vec{x}} . \vec{v} \right) \right) + \rho \, \vec{b}, \qquad (A.1)$$

$$\nabla_{\vec{x}} . \vec{v} = 0, \qquad (A.2)$$

where Eq.A.1 expresses the conservation of momentum and Eq.A.2 the conservation of mass for incompressible flow.

To simulate fuel filling process inside the tank, inflow velocity, computed for inflow rate provides by Zodiac aerospace, is prescribed at a local opening area. Eq.A.1 governs the flow velocity and pressure inside a fluid domain given physical boundary conditions. To accurately define the fluid level at each time during the refueling process, Eq.A.1 need to be completed by level set equation:

$$\frac{\partial \alpha}{\partial t} + \vec{v} . \nabla_{\vec{x}} \alpha = 0 \qquad (A.3)$$

The basis of the Level Set methods has been proposed by Osher and Sethian [106]; and describe in detail in [134]. The interface function α defines the distance to

the fluid interface. The zero level curve of the continuous function α defines by the location of the fluid interface. Eq.A.3 is known as a transport equation that describes the evolution of the free surface or zero iso-surface.

Numerical computation of Eq.A.1, Eq.A.2 and Eq.A.3 can induce mass loss in under-resolved regions, this is the main drawback of level set methods. To improve mass conservation, different extensions of the level set method have been developed, such as the particle level set [36] and a coupling between VOF and level set [141, 111].

A.3 Simulation of Aircraft tank refueling

A.3.1 Problem description

In this work, we are interested in the hydrodynamic effects of the fluid during tank refueling and more precisely in the influence of the dynamics effects on the pressure field and how does it evolve in time for a variation of volume compared to the hydrostatic solution.

Two set of simulations will be performed for a given pitch and roll angle and results will be compared in term of volume-pressure cross plot. A first set of hydrostatic simulations using both LS-DYNA and ZODIAC's analytical tool will be performed in order to validate the LS-DYNA software for simulating hydrostatic pressure when the fuel is at rest but also to have qualitative results and to control the convergence of the results during the computation (crucial step as we need to follow a simulation that takes several days, and one cannot afford to wait for the final results). A second simulation of the full hydrodynamic process respecting the following input parameters:

- Real time simulation: 900 s. Simulation starts at t=0 s and ends at 900s (97% of total volume is filled).

- Ambient temperature: 20 C^o.

- Gravity Load: 9.81 m/s^2 following Z-axis.

- Fuel Tank is fixed in space (no motion).

- Initially filled with air, the tank is filled with fuel introduced at constant pressure (3400 $mbar$).

- The structure is considered to be rigid.

- The tank is filled with Jet-A1 kerosene type :

- Density: 807 kg/m^3.

- Kinematic viscosity: 1.65 cSt.

- Gaz pressure: 1013 $mbar$.

- $Outside pressure$: 1013 $mbar$.

- Constant inlet volume flow rate: 3.34 10^{-3} m^3/s.

Results are expressed in term of volume-pressure cross plots at a given sensor location. The CAD model of the fuel Tank has been done by ZODIAC. It includes all the details: Baffles, mice holes, central opening on baffles. Only the vent system is not included in the geometry. The CAD model of the fuel tank is shown in Fig.A.1, Fig.A.2 and Fig.A.3 local zooms are shown at the inlet and outlet locations, respectively.

Figure A.1 – Fuel Tank CAD model.

Figure A.2 – Local zoom on the Fuel Tank Inlet.
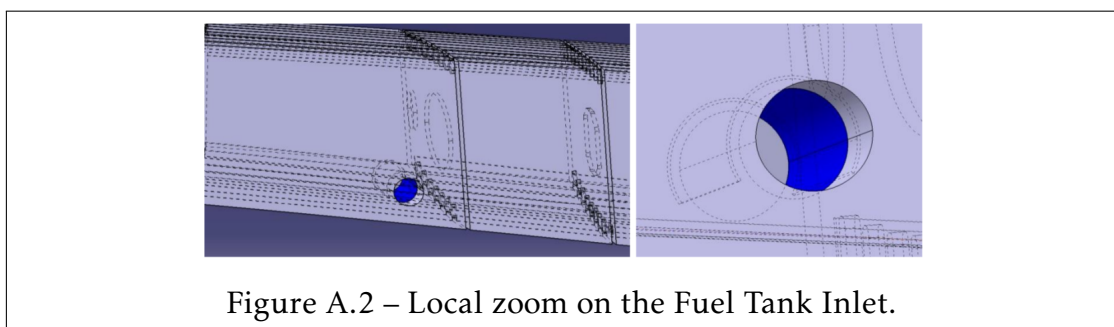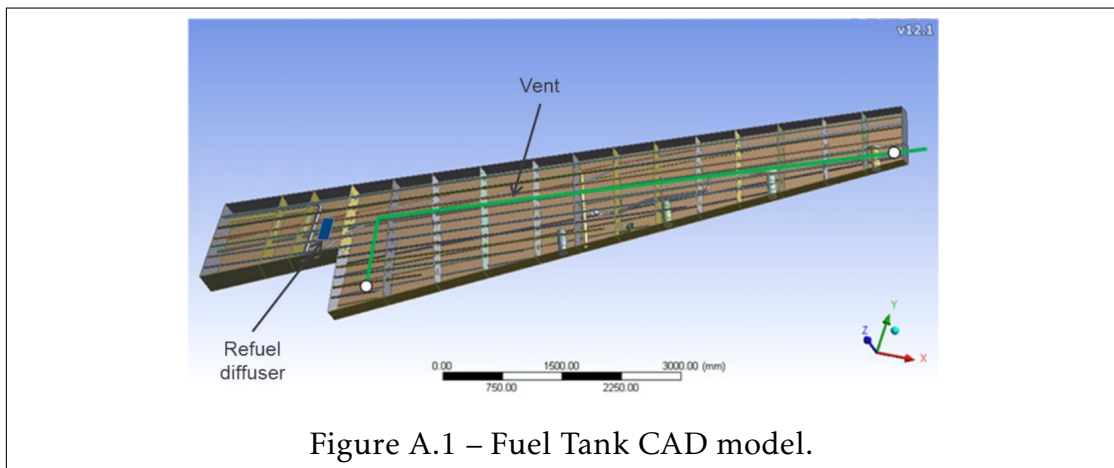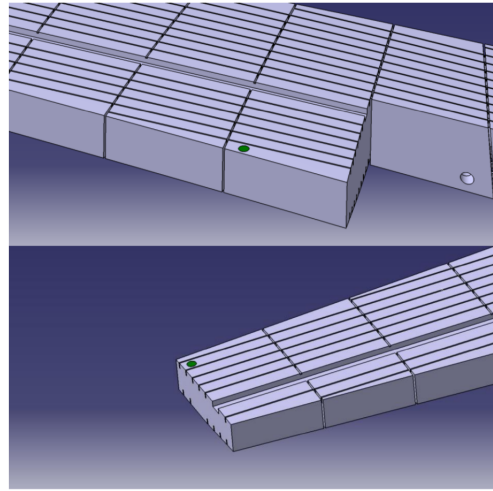
Figure A.3 – Local zoom on the Fuel Tank Outlets.

A.3.2   Numerical Simulation

The fuel tank CAD model provided by ZODIAC was modeled excluding the mice holes on the bottom and the top of the baffles. Adding the mice holes would have required smaller elements size and thus may have increased considerably the overall CPU time. Reminding that the main concern of this study is the validation of Fuel mass computation methodology against numerical simulation results in absence of experimental data, the effects of the mice holes can be neglected. The numerical model is composed of 4.536.706 tetrahedron elements. Fig.A.4 shows the numerical model including baffles with holes separating the different compartments. A local zoom on the mesh is shown in Fig.A.5 at the inlet location.

Hydrostatic simulation

When the fuel is at rest, we have only one plane fuel free surface in the whole fuel tank and a hydrostatic pressure profile is obtained. Hydrostatic pressure for each material can be simulated in LS-DYNA based on an element depth and with respect to the "reference" pressure P_{top} at the top of the fuel.
For one multi-material formulation that contains two fluids (fuel and air), the hydrostatic pressure is defined by:

$$P = P_{top} + \sum_{i=1}^{Nmat=2} \rho_i g h_i \qquad (A.4)$$

Figure A.4 – Numerical model of the Fuel Tank model : Baffles and holes.

Figure A.5 – Local zoom of the mesh at the Fuel Tank Inlet location.
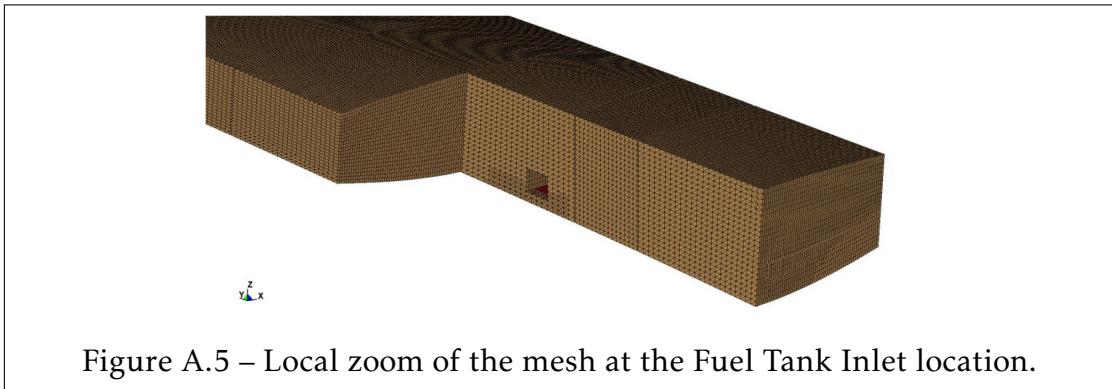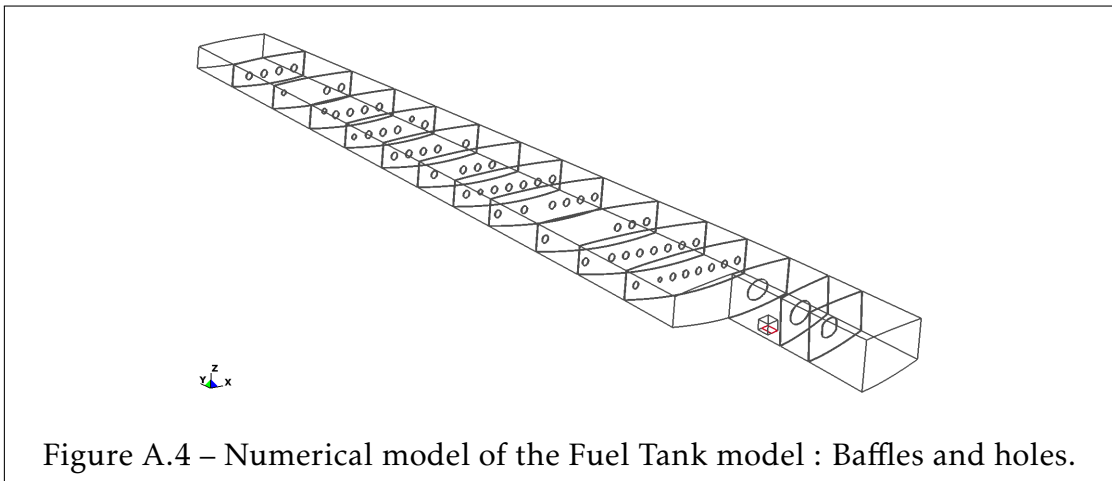
For given pitch and roll angles (θ_x, θ_y), the hydrostatic simulation is repeated for variable height of fuel to obtain the reference volume-pressure cross plot. To do so, an automatic script has been programmed in Linux OS, the different steps of the program are:

- Initialization of the parameters: Fluids properties, minimum Z-coordinate Z_{min} of the tank, maximum Z_{max} Z-coordinate of the tank, the desired number of height levels N_{sub} and the height increment $\Delta h = (Z_{max} - Z_{min})/N_{sub}$.

- For $i = 1$ to N_{sub}

  - Compute the height $h_i = Z_{min} + \Delta h$ for step i.

  - Initialize the volume fraction by cutting the element with a $X - Y$ plane located at position h_i in Z-axis.

  - For each multi-material, compute in LS-DYNA the hydrostatic pressure using Eq.A.4.

  – Copy the mass for the material fuel and convert it to volume using the density.

  – Copy the pressure at the location of the sensor (see Fig.A.6).

  – Since the pressure is computed at the center of an element in LS-DYNA (one- point quadrature rule), we compute the analytic hydrostatic pressure at the center of the element (and not at its base).

  – Store in two different files the gathered results (LS-DYNA and analytic pressure).

Relative pressure contour plots are shown for 25%, 50%, 75% and 100% of the total height ($Z_{max} - Z_{min}$) in Fig.A.7. LS-DYNA and analytic hydrostatic relative pressure are shown in Fig.A.8.

Figure A.6 – Sensor at the maximum pressure location.
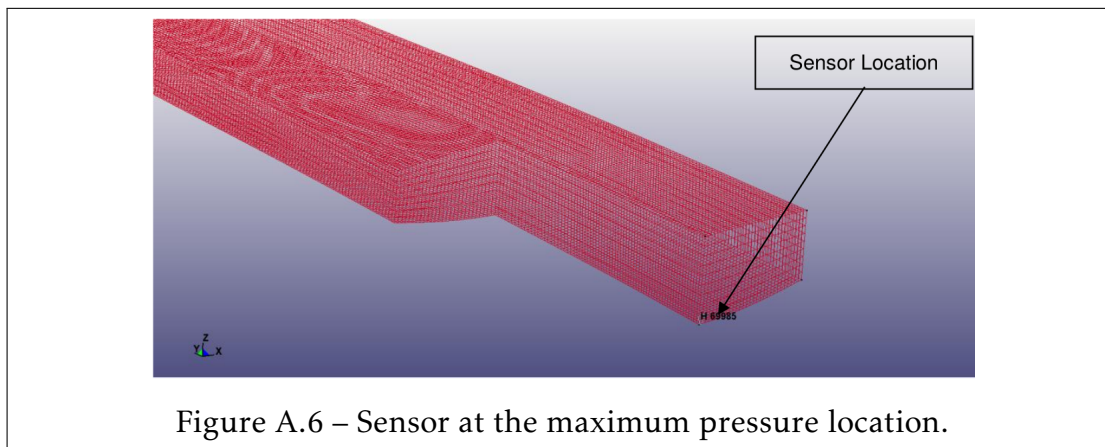
Hydrodynamic simulation

Between the hydrostatic and hydrodynamic analysis, we can notice the two following changes:

  • Free Surface is no more one perfectly flat free surface in the whole Fuel tank. Thus, at a given time the fuel height might be different from one compartment to another as the fuel flows through the holes of the baffles.

  • A priori, the hydrodynamic pressure should not be neglected. The total pressure is the sum of the hydrostatic and the hydrodynamic pressures.

After post-processing the hydrodynamic numerical results, it was observed that the refueling process is so slow (~15 minutes in real time) that the obtained total pressure is mainly hydrostatic.
Although, the hydrostatic pressure is dominant, it can be seen from Fig.A.9

Figure A.7 – Jet A-1 fuel hydrostatic pressure contour for various heights (25%, 50%, 75% and 100%).

Figure A.8 – Hydrostatic Volume-Pressure (Relative Pressure) Cross Plot where the pressure is the maximum (Sensor A).
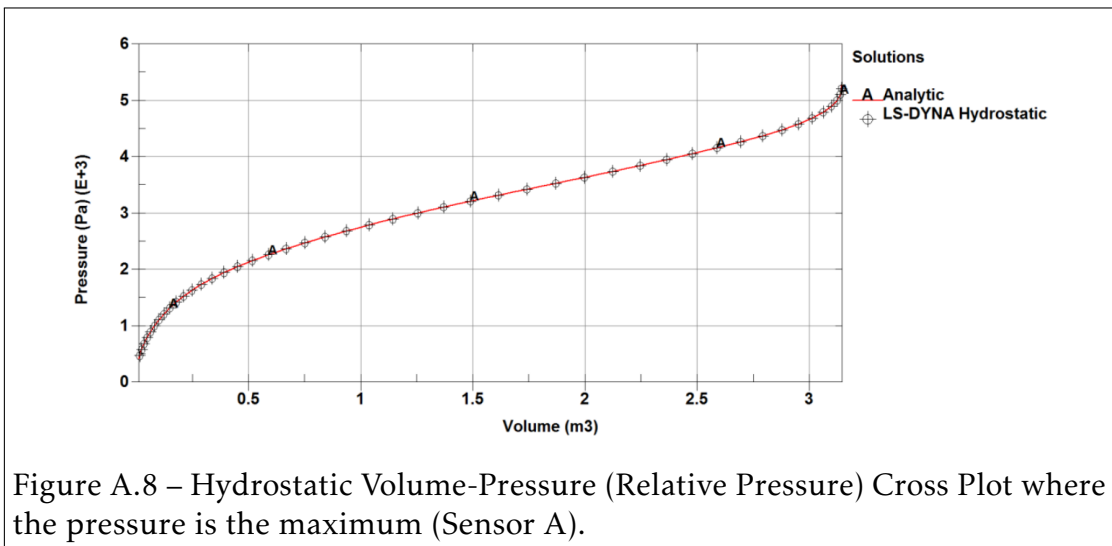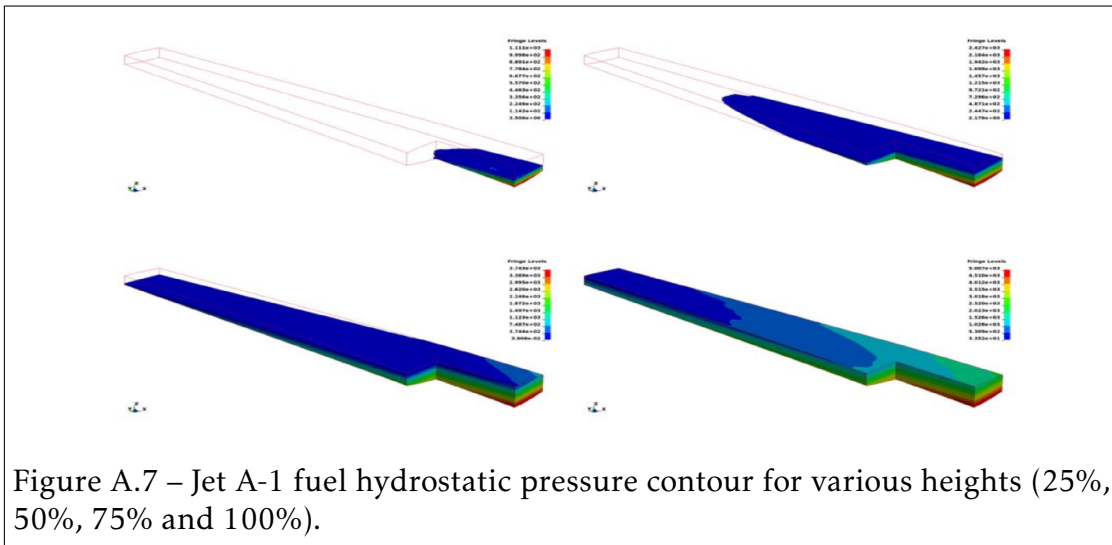
that in each compartment the fuel height is not the same. The pressure at the sensor location (see Fig.A.6) for both hydrostatic and hydrodynamic simulation is shown in Fig.A.13.

Let us comment the different stages highlighted in Fig.**??**:

1. At time $t1 = 80s$, the fuel reaches the sensor A location.

2. Between $t1$ and $t2 = 120s$, the first compartment is filled linearly.

3. Between $t2$ and $t3 = 200s$, the fuel level of the four first compartment is the same. Thus, the increase of the total height is slower (change of slope in Fig.A.13).

4. Few time step later, the fuel is leaking from the fourth to the fifth compartment. Thus, the fuel height is increasing slower. This explains the short decrease in Fig.A.13 just before the linear increase between $t3$ and $t4 = 400s$.

5. The different stages are repeated sd the fuel tank is getting filled.

At time $t5 = 660s$, $t6 = 800s$ and $t7 = 870s$ numerical noise is observed in the pressure variable when the fuel tank is almost filled.
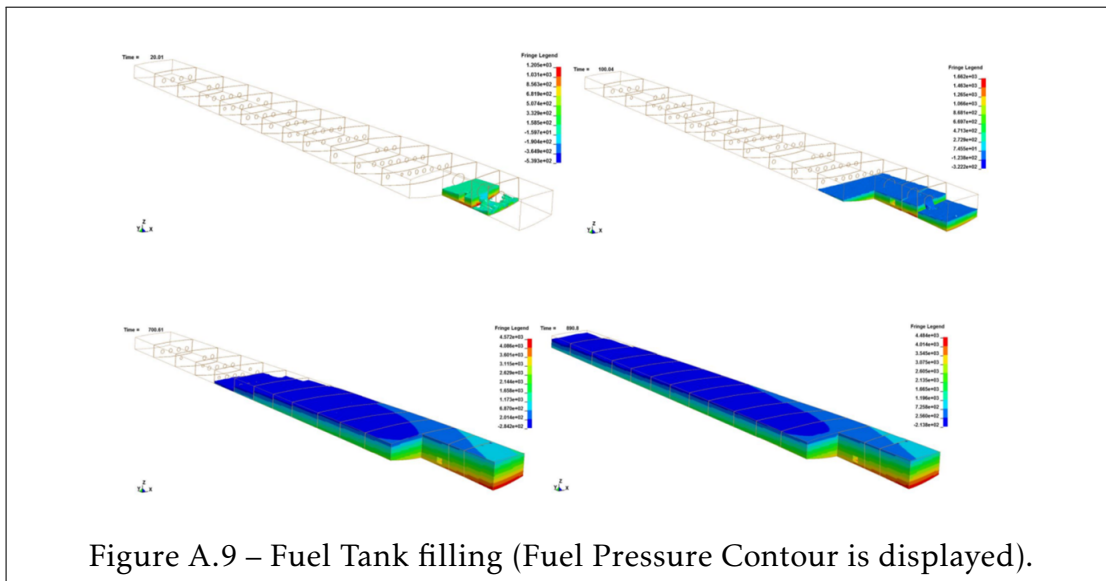
Figure A.9 – Fuel Tank filling (Fuel Pressure Contour is displayed).
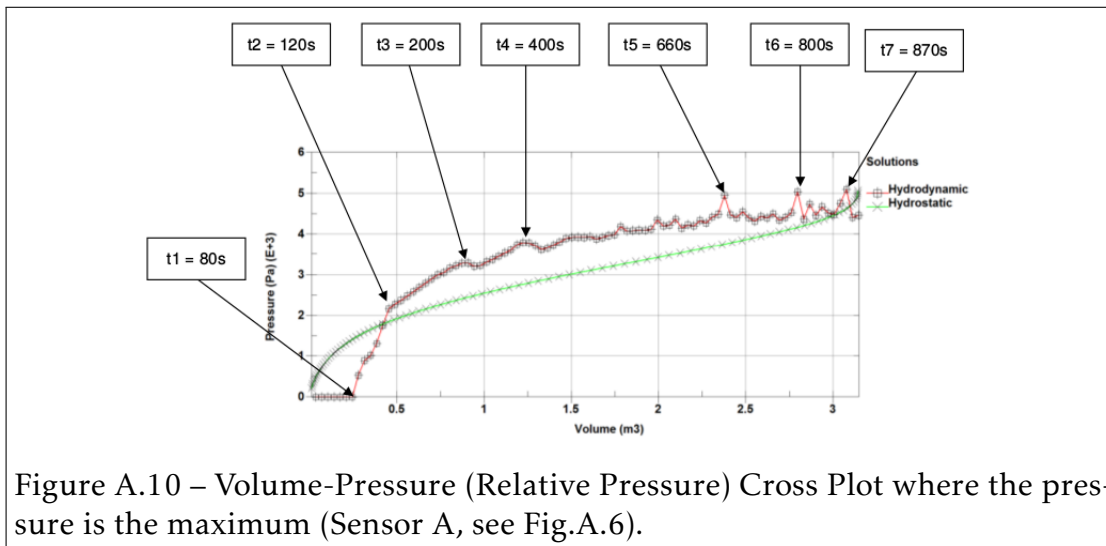
Figure A.10 – Volume-Pressure (Relative Pressure) Cross Plot where the pressure is the maximum (Sensor A, see Fig.A.6).

A.4   Fuel Mass computation based on pressure gauging

A.4.1   Physical Assumptions

From the different numerical simulations presented in the previous subsection, we can assume the following assumptions:

- The free surface in each compartment can be approximated by a linear flat surface, but cannot be assumed to be one linear flat surface for the whole tank. This assumption has been observed during the hydrodynamic simulation (see Fig.A.9) of the fuel tank filling phase. Thus, the fuel height in the tank varies from one compartment to another.

- Hydrostatic pressure is dominant, and thus we can neglect hydrodynamic pressure due to fuel motion.

In order to fulfill the first assumption, we need to have at least one flooded pressure gauge in each compartment, and thus the fuel mass needs to be computed separately on each compartment and summed up on all compartments afterward.
Considering the second assumption, for a given pressure p and density ρ (obtained by gauging) the fuel height level h_k within each compartment k is computed according to the simple hydrostatic pressure formula:

$$p = \rho g h_k \qquad\qquad (A.5)$$

A.4.2   Fuel Mass computation

For given pitch and roll angles, this algorithm computes the fuel mass within the tank based on a finite element mesh (composed of eight nodes solid elements) and advanced geometrical tools that compute, for each element, the intersecting volume between the finite element and the free surface plane limiting the fuel level.

Why FE mesh ?
Using the divergence-flux theorem the computation of the volume V of a domain A can be reduced to a surface integral computation :

$$V = \int\int\int_A dV = \frac{1}{3}\int\int_{\partial A}(x,y,z).\overrightarrow{n}\,dS, \qquad\qquad (A.6)$$

where ∂A is the boundary of the domain A and $\vec{n}$ is the unit normal vector of dS facing to the outer domain of A.

In order to use Eq.A.6 to compute the volume, one may know the parametric equation of the boundary surface in order to compute plane tangent to surface and the unit normal vector.

Since the surface of the fuel tank is curved and not flat, the parametric non-linear equation of the boundary surface is unknown. To tackle the complexity of the normal vectors computation, a finite element mesh composed of eight node solid elements (see Fig.A.11) is used as a first order linear polynomial interpolation of the boundary surface. Each finite element is composed of six flat faces.

Considering the physical assumption that the fuel free surface is flat on each compartment, we can use the finite element mesh to compute the fuel volume inside each compartment. Indeed, the finite element mesh is a partition of the fuel tank. By processing through the finite elements, we can deduce if an element is below, upper or intersected by the free surface plane.

The equation of the plane corresponding to the fuel level, can be solely described by a unit vector normal $\vec{n}$ to the plane and a point $\vec{x_p}$ located on the plane.

The point $\vec{x_p}$ located on the free surface is determined using Eq.A.5 that gives us the distance from a pressure probe to the free surface based on the pressure and the density (obtained by gauging).

In summary, the main advantages of using a FE mesh are the followings :

- Computation of the unit normal vectors.

- Computation of the surface integral.

- Computation of the intersected elements by the free surface plane to determine the fuel volume.
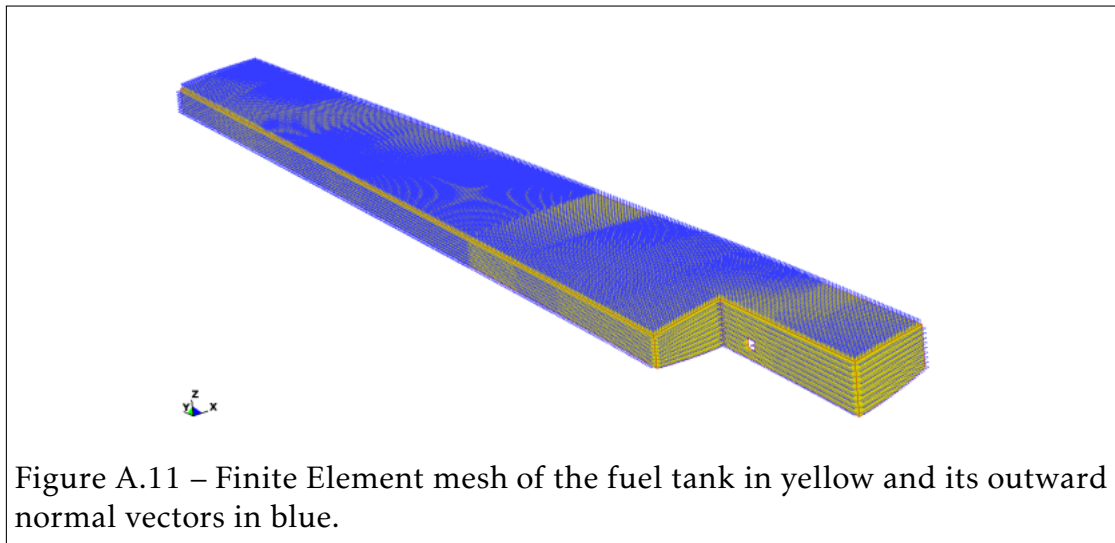
Figure A.11 – Finite Element mesh of the fuel tank in yellow and its outward normal vectors in blue.
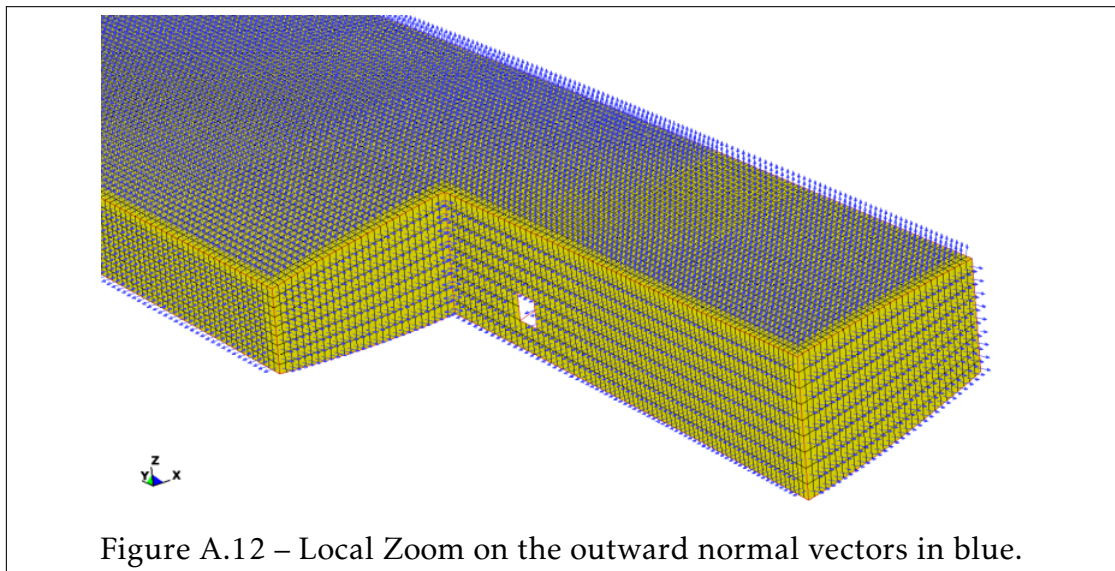
Figure A.12 – Local Zoom on the outward normal vectors in blue.

Volume Intersecting algorithm [85]

In this paper, we used the analytical and geometrical tools developed by Lopez and Hernandez [85] for 3D volume of fluid methods (VOF) in general grids. The VOF method consists in constructing the discontinuous material interface within the element using its volume and the volume of the surrounding neighbors. Such methods are implemented in many commercial software. The technique implemented in LS-DYNA [48] is the **Piecewise Linear Interface**

Construction (PLIC) proposed by Youngs (see [162]) and [161] for 2D and 3D problems, respectively). The PLIC method follows the early **Simple Line Interface Calculation** (SLIC) method proposed by Noh and Woodward [103].

The PLIC method approximate the interface by a linear plane(segment in 2D and planes in 3D). This approach is adopted in this paper where the linear plan represents the fuel free surface in each compartment. The parametric equation of the plane P at the free surface interface is defined by :

$$\vec{x_p}.\vec{n} + C = 0, \tag{A.7}$$

where :

1. The normal $\vec{n}$ to the plane is given by $\vec{n} = (0,0,1)$ (Load body in Z-direction).

2. The position vector $\vec{x_p}$ located on the free surface plane is computed from the gauged pressure and Eq.A.5.

3. The constant C is given by $C = -\vec{x_p}.\vec{n}$.

Let us consider the eight-node hexahedron polygon of six faces and the intersecting plane P shown in Fig.A.13. For a given node i, the signed distance ϕ_i from the node i to the plane P is defined by :

$$\phi_i = \vec{x_i}.\vec{n} + C \tag{A.8}$$

If the signed distance ϕ_i is positive then the normal vector $\vec{n}$ points to the node i and the node i is above the free surface (because $\vec{n} = (0,0,1)$). Else if the signed distance ϕ_i is negative then the node i is below the free surface. Thus:

- If the signed distance is positive for all the nodes of the eight-node hexahedron element then the element is at the top of the free surface and it is not filled with fuel.

- Else if the signed distance is negative for all the nodes of the eight-node hexahedron element then the element is below the free surface and it is totally filled with fuel.

- Else, the eight-node hexahedron element is intersected by the free surface plan and it is partially filled with fuel.

In the case where the eight-node hexahedron element is intersected by the plan P, the volume filled with fuel is determined by constructing two new polygons

and computing their volumes using Eq.A.5.

The nodes of intersection are created along the edges of the parent element if the signed distances ϕ_i and ϕ_j of the two nodes i and j edge are of opposite signs. Thus the new node k position vector is given by:

$$\overrightarrow{x_k} = \frac{\phi_i}{\phi_j - \phi_i}(\overrightarrow{x_j} - \overrightarrow{x_i}) \tag{A.9}$$
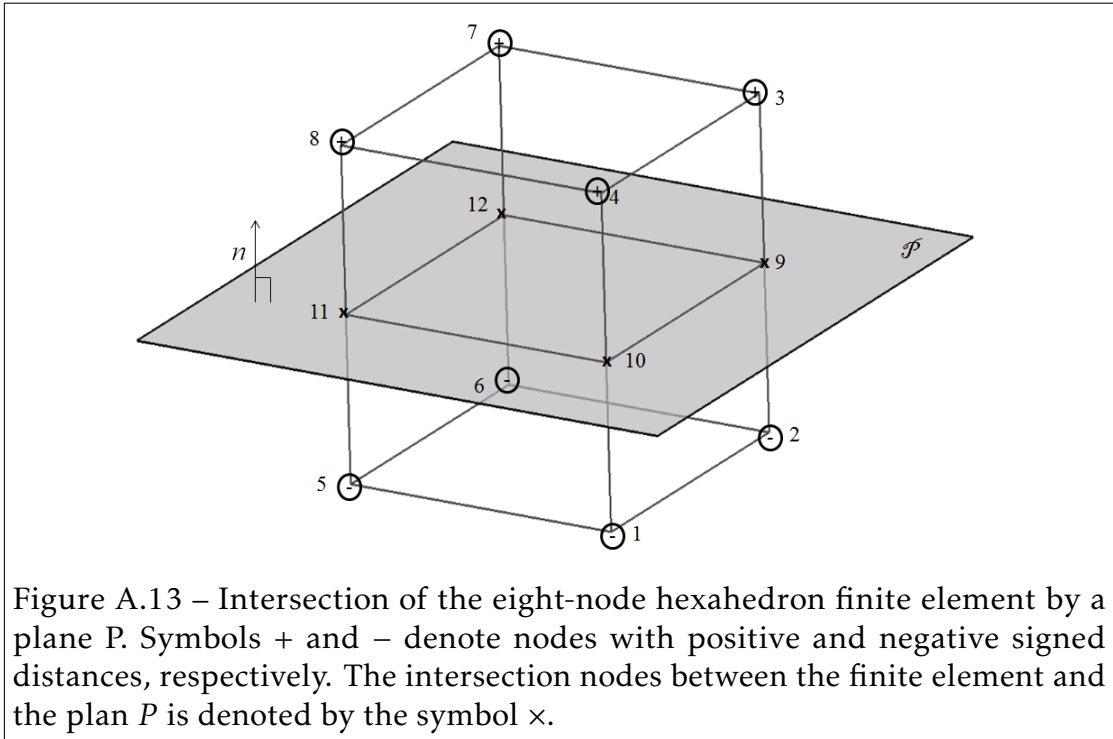
Figure A.13 – Intersection of the eight-node hexahedron finite element by a plane P. Symbols + and − denote nodes with positive and negative signed distances, respectively. The intersection nodes between the finite element and the plan P is denoted by the symbol ×.

A.4.3   Algorithm description

As assumed, the fuel free surface level is different from one compartment to another (observed from numerical simulation, see Fig.A.9). Thus, each compartment needs to be processed separately. The computed mass in each compartment is summed up at the end to obtain the total fuel mass.

Let us consider the compartment k denoted by $Comp_k$. For each probe k_i located in the compartment k, we convert the measured pressure P_{k_i} to a local fuel height h_{k_i} using Eq.A.5. From all the computed local fuel heights h_{k_i}, we determine the

averaged fuel height H_k of the compartment k.

Considering the average height H_k, the position vector of the point $\vec{x}_p$ located on the fuel free surface is determined, then the parametric equation of the fuel free surface plan P using Eq.A.7.

Once the parametric equation of the plan P is obtained, we sequentially loop over all the finite element El_k within the compartment $Comp_k$ and compute the fuel volume within the elements using the intersecting volume procedure described earlier.

Finally the volume of fuel contained in the compartment $Comp_k$ is the sum of the fuel volume in each element El_k of $Comp_k$. The volume of fuel within $Comp_k$ is converted to mass $Mass_k$ using the fuel density.

The process for the computation of the fuel mass is described in the Fig.A.14:
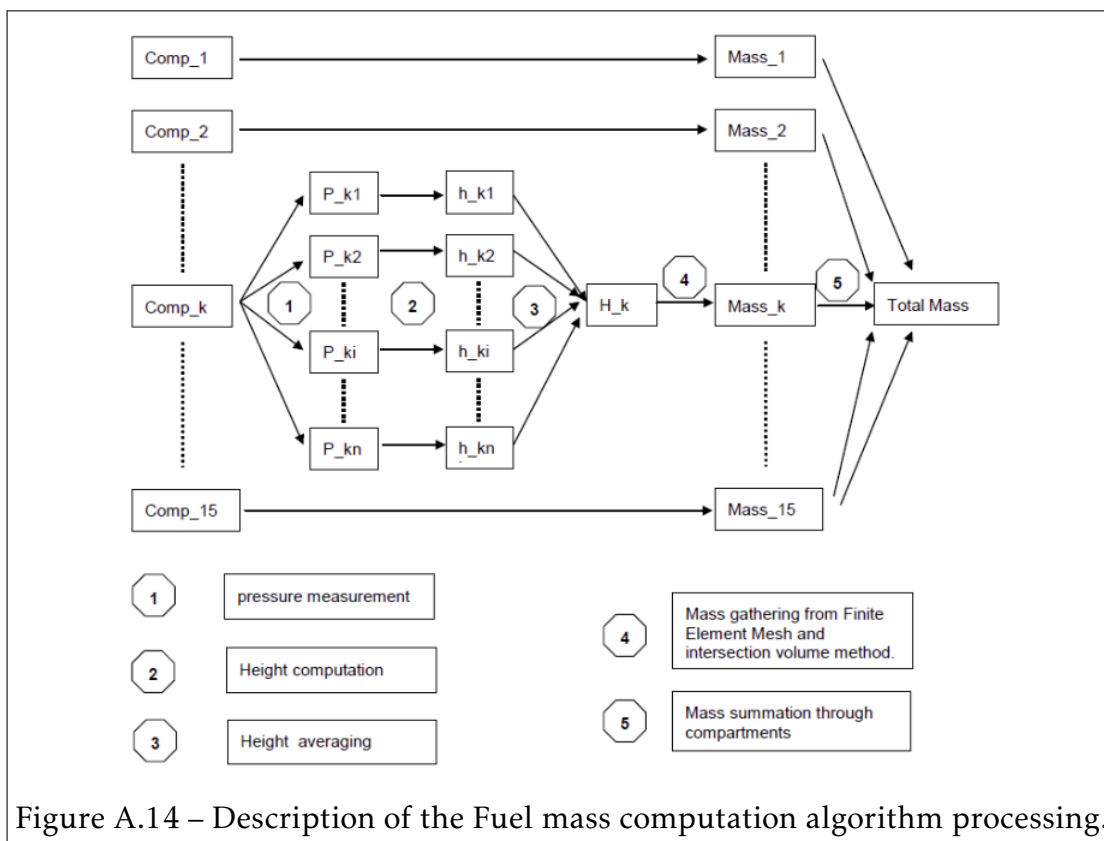
Figure A.14 – Description of the Fuel mass computation algorithm processing.

A.4.4   Density computation vs density probing

The described algorithm computes fuel masses in two steps :

- Computation of fuel volume using the **Volume Intersecting algorithm**[85].

- Conversion of fuel volume into fuel mass using the fuel density.

In the actual aircraft gauging systems, the density is provided by a density probing system. Thus the density is assumed to be known and it is an input parameter to the **fuel mass computation algorithm**. Considering the physical assumptions and Eq.A.5, we can take advantage of pressure gauging to determine the density instead of using density probes.

Let us consider two pressure probes i and j located in the same compartment $Comp_k$. We define by $h_{i,j} = Z_j - Z_i$ the vertical distance (Z-direction) between the two probes i and j and by $\Delta P_{i,j} = P_i - P_j$ the pressure difference. For constant density (incompressible fluid), the density is given by :

$$\rho = \frac{\Delta P_{i,j}}{g h_{i,j}} \tag{A.10}$$

We denote by $nprb_k$ the number of probes present in the compartment $Comp_k$. There is a combination of $n = \frac{(nprb_k)(nprb_k - 1)}{2}$, i and j pairs of probes for the computation of the density. For constant density, the n pairs must verify Eq.A.5 and thus:

$$Y = \begin{bmatrix} \Delta P_{1,2} \\ \Delta P_{1,3} \\ \vdots \\ \Delta P_{nprb_k - 1, nprb_k} \end{bmatrix} = \beta \begin{bmatrix} h_{1,2} \\ h_{1,3} \\ \vdots \\ h_{nprb_k - 1, nprb_k} \end{bmatrix} = \beta X \tag{A.11}$$

Where $\beta = \rho g$.

Three methods are tested to compute the density (i.e β in Eq.A.11) :

1. **Maximum height $h_{i,j}$:**
   In this approach, we consider the maximum distance distance between two flooded probes to compute the density :

$$\beta = \rho g = Y(k)/X(k) \tag{A.12}$$

Where the index k of the vectors X and Y is taken such that :

$$X(k) = \max_{i,j}(h_{i,j}) \qquad (A.13)$$

2. **Linear regression** :

$$\beta = \rho g = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad (A.14)$$

Where $\bar{x} = \frac{1}{n}x_i$ and $\bar{y} = \frac{1}{n}y_i$
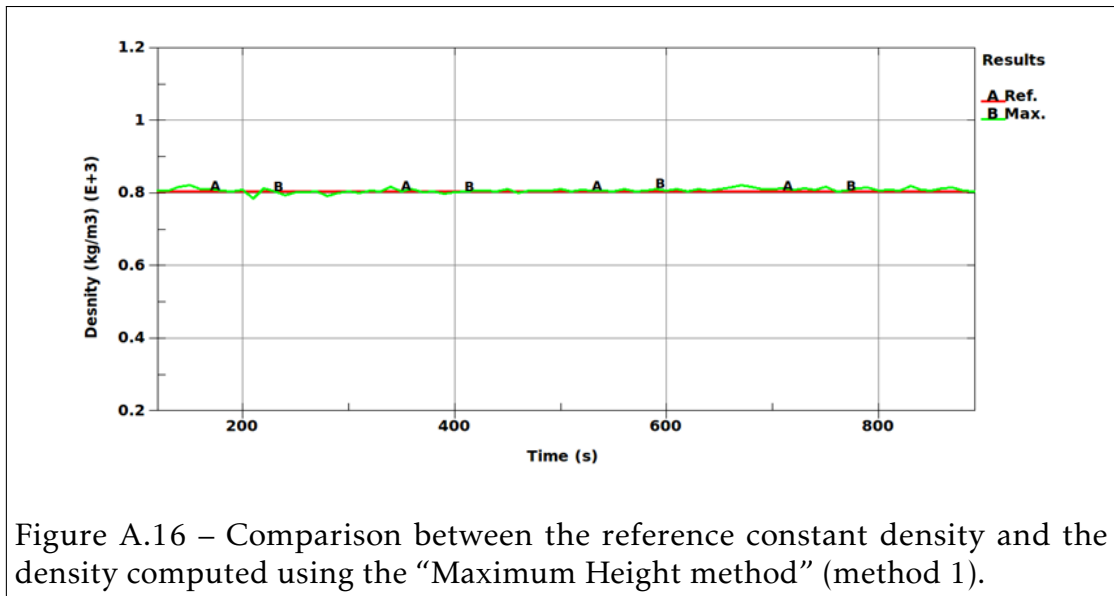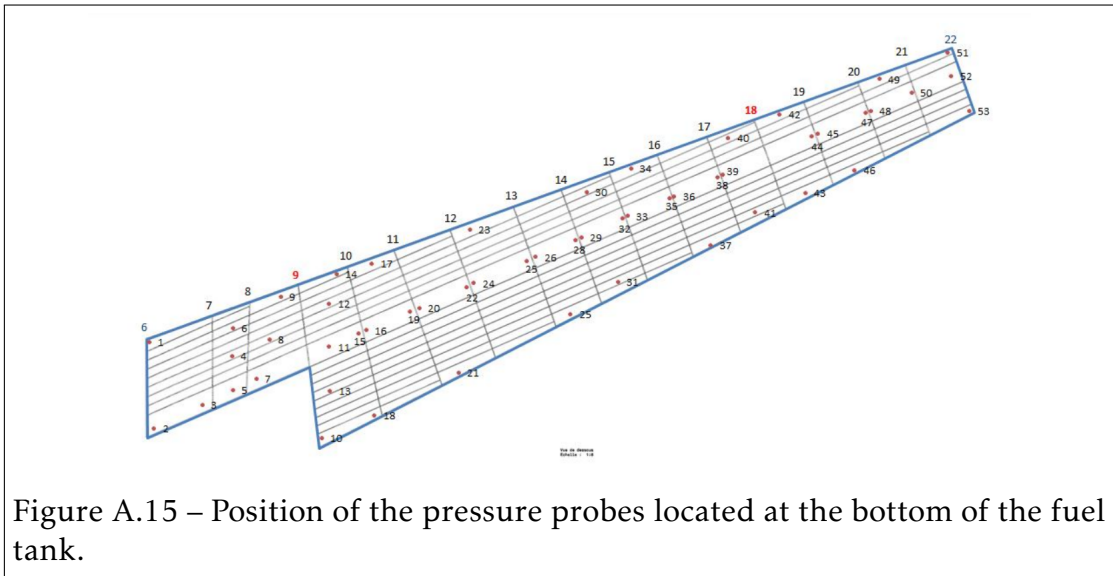
3. **Orthogonal regression**:

$$\beta = \rho g = \frac{\sum_{i=1}^{n}(x_i^2 - y_i^2) + \sqrt{(\sum_{i=1}^{n}(x_i^2 - y_i^2))^2 + 4(\sum_{i=1}^{n}x_i y_i)^2}}{2\sum_{i=1}^{n}(x_i y_i)} \qquad (A.15)$$

A.4.5 Validation

In absence of experimental data, the numerical simulation of the hydrodynamic fuel tank filling phase is considered instead. Input pressure records from experimental gauging are replaced by pressure time history at the nodes located at the expected position of the pressure probes (see Fig.A.15).

Using the presented methodology based on pressure gauging and input pressure records from the numerical simulation, the fuel mass is computed according to the algorithm described in the previous subsection and the different density computation methods.

The computed fuel mass is compared to the reference fuel mass from the numerical simulation for the validation of the methodology. The results for density computations are shown in Figs.A.16,A.17,A.18 and for the fuel mass computation in Fig.A.19. Good correlations are observed between the computed and reference fuel masses.

Figure A.15 – Position of the pressure probes located at the bottom of the fuel tank.

Figure A.16 – Comparison between the reference constant density and the density computed using the "Maximum Height method" (method 1).
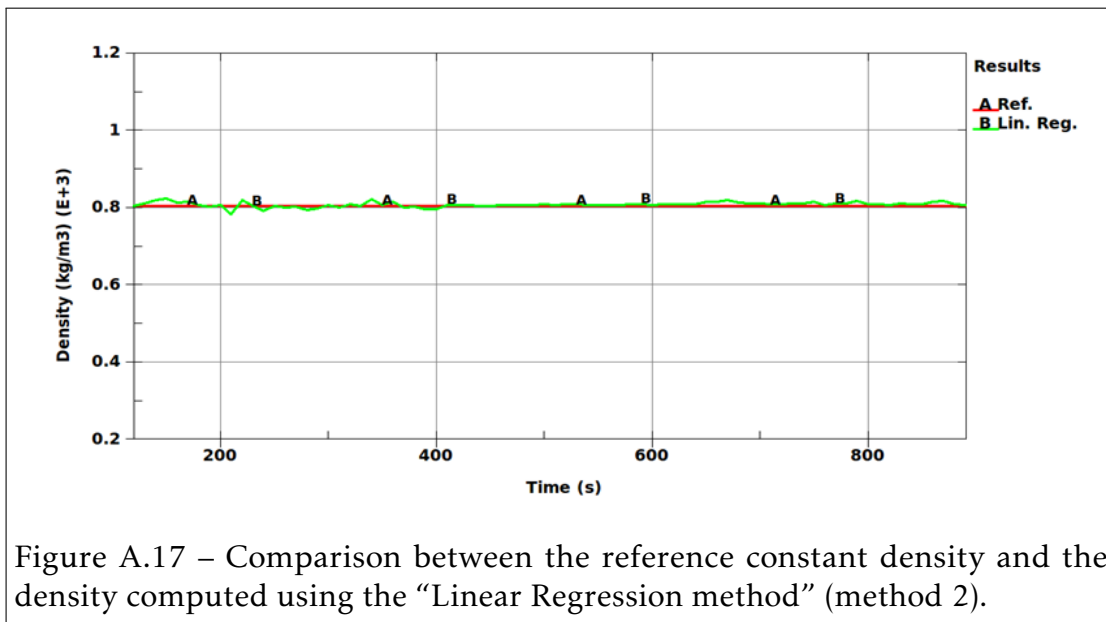
Figure A.17 – Comparison between the reference constant density and the density computed using the "Linear Regression method" (method 2).
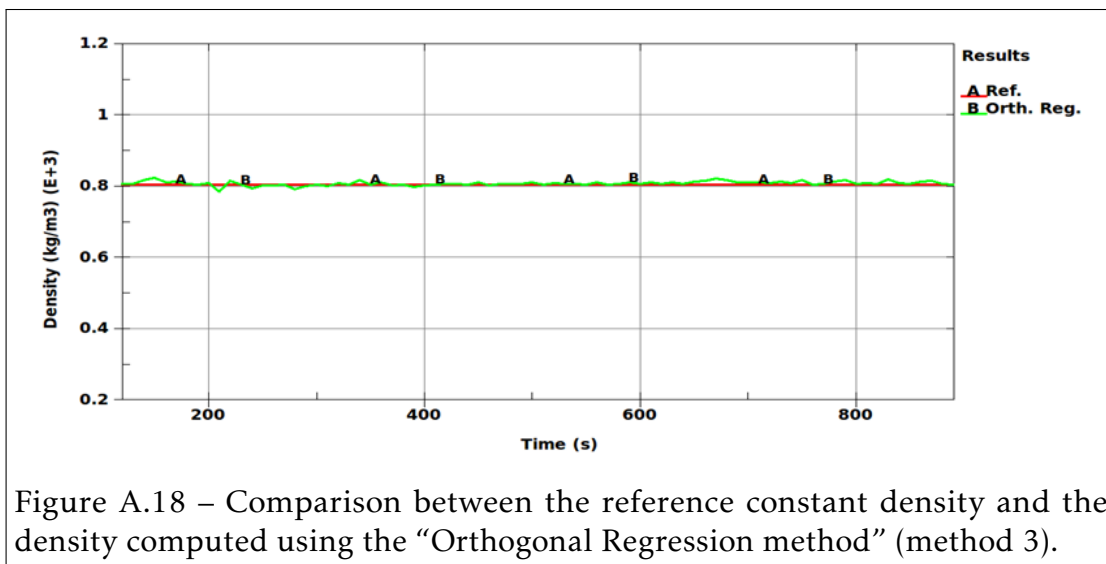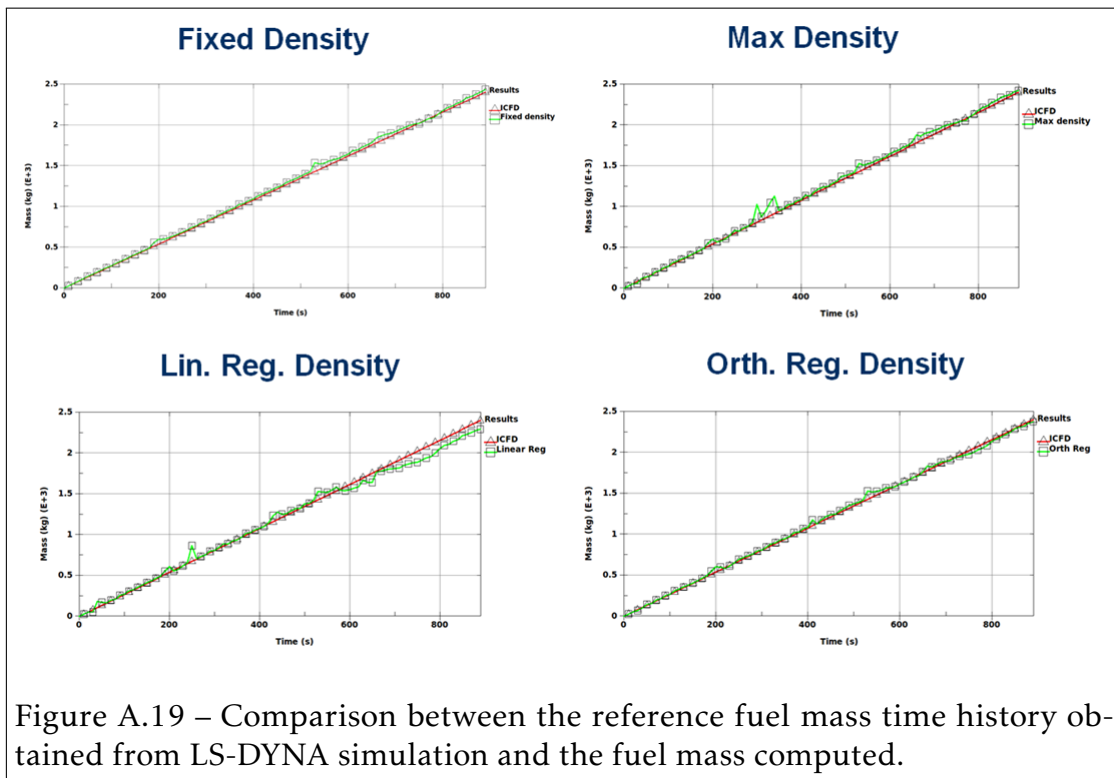

Figure A.18 – Comparison between the reference constant density and the density computed using the "Orthogonal Regression method" (method 3).

Figure A.19 – Comparison between the reference fuel mass time history obtained from LS-DYNA simulation and the fuel mass computed.

A.5 conclusion

In this appendix, a new numerical methodology is developed to predict fuel mass inside an operation aircraft. This method is only based on pressure measurements in each compartment of fuel tank. Unlike other experimental methods that require fuel density to measure the total fuel mass, the developed method computes fuel density based on pressure measurements and the height of the fuel level inside the compartment fuel tank.

To validate the methodology a numerical simulation is performed to provide pressures, that will be used as input parameters in the software package that provides fuel mass inside the tank. A computational method using CFD capabilities is performed to model a fuel filling tank up to 900 seconds. During the simulation pressure time history at different gage locations are stored and used as input data for the software package. Mass fuel is computed at each pressure record provided by the numerical simulation. We observe a good correlation between the reference density and the density computed out of pressure gages. We also observe good correlation between the numerical simulation reference fuel mass and the computed fuel mass based on pressure gauging.

Appendix B

Statistical Energy Analysis Method

B.1 Introduction

For high frequency analysis, Statistical energy analysis (SEA) has proved to be a promising approach to the calculation of sound transmission in complex structures. In automotive industry and also in civil engineering, most of noise transmission is due to high-frequency structural vibrations, in which the characteristic wavelength is small compared to the dimensions of the structure. For these applications classical methods of structural analysis, such as the finite element method (FEM), and Boundary Elements Method (BEM), cannot be used due to the large number of degrees of freedom required to model structural deformation. Statistical Energy Analysis (SEA) considers the vibrations of the structure in terms of elastic waves which propagate through the structure and are partially reflected and partially transmitted at structural connections. For the last few year there has been an increase in the application of SEA techniques to study noise transmission in motor vehicles.

In this chapter, we present the basic concepts of the SEA code that was developed during the thesis work. We also provided validation against the free sea code **FreeSEA** [123] developed by the professor Sarradj and an application to a simplified vehicle model.
The vehicle body is modeled as a collection of connected flat plates and cavities for predictive purposes. Each plate supports bending waves, longitudinal and transverse shear in-plane waves and assumed damping effects. The effects of curved plates and the transmission of waves through curves plates and beams are not included in the analysis. Numerical results in term of acoustic pressure inside the cavity, are in good agreement compared to analysis using other software published in the literature.

Statistical energy analysis is used to predict sound and vibration transmission of elastic waves which propagate through the structure that are partially reflected and partially transmitted into connected structure though structure disconti- nuity or line connection. In SEA method systems are considered to be divided into subsystems that are linearly coupled and exchange energy through resonant vibration modes. Subsystems can be plates, beams or acoustic rooms that have modes which are similar in nature, and where the primary variable of interest is energy. The theory of structural vibration transmission has been developed over many years, and detailed theory of SEA is described by Lyon and Dejong [87] and Le Bot [23]. The earliest calculations were carried out by Langley et al. [74, 73]. In SEA computation the system can be modeled by the intersection of a number of semi-infinite plates which meet at a single line junction where there must be continuity of displacement, slope and equilibrium of moments and forces. Structural vibration transmission calculations are carried out as part of the calculation of sound and vibration transmission through structures. If two or more plates meet together to form a line junction then the properties of the junction can be given in terms of the transmission coefficient, defined as the ratio of the power transmitted across the junction to the power incident. The power may be transmitted by a transmission wave type that is different from the incident wave in which case there will be energy conversion from one wave type to another.

Such applications are to be found in automotive industry as well as civil, naval and aerospace engineering for noise transmission in buildings, ships, aircraft and other structures as described in [56]. In this chapter, the plates are assumed to be isotropic, thin and flat, and to meet at a continuous line junction.

B.2   SEA Equations

The averaged energy conservation equation between the total number of subsystem is expressed as a function of the *coupling loss factors* (CLFs) η and the modal density n. For a given subsystem i, we can identify two types of CLFs:

- The **damping loss factor** $\eta_{i,d}$ that linearly relates the dissipated power $P_{i,d}$ to the total vibrational energy E_i of the modes at frequency $f = \omega/2\pi$ by:

$$P_{i,d} = \omega\eta_{i,d}E_i \tag{B.1}$$

- The **transmitted loss factor** $\eta_{i,j}$ that linearly relates the transmitted power $P_{i,j}$ from subsystems i to j to the total vibrational energy E_i of the modes at frequency $f = \omega/2\pi$ by:

$$P_{i,j} = \omega\eta_{i,j}E_i \tag{B.2}$$

Finally, The total coupling loss factor $\eta_{i,tot}$ for a subsystem i is given by:

$$\eta_{i,tot} = \eta_{i,d} + \sum_{i\neq j}\eta_{i,j} \tag{B.3}$$

For the particular of a model composed of two subsystems i and j (see Fig.B.1), the averaged energy conservation equation is given by:

$$\begin{aligned} P_{i,inp} + P_{j,i} &= P_{i,d} + P_{i,j} \\ \underbrace{P_{j,inp} + P_{i,j}}_{IN} &= \underbrace{P_{j,d} + P_{j,i}}_{OUT} \end{aligned} \tag{B.4}$$

Where $P_{i,inp}$ denotes the input power. Through the substitution of Eq.B.1 and Eq.B.2 into Eq.B.4, we have:

$$\begin{aligned} P_{i,inp} &= \omega\left(\eta_{i,d}E_i + \eta_{i,j}E_i - \eta_{j,i}E_j\right) \\ P_{j,inp} &= \omega\left(\eta_{i,d}E_j + \eta_{j,i}E_j - \eta_{i,j}E_i\right) \end{aligned} \tag{B.5}$$

Finally, the matrix form for the averaged energy conservation equation is given by:

$$\omega\begin{bmatrix} \eta_{i,tot} & -\eta_{j,i} \\ -\eta_{i,j} & \eta_{j,tot} \end{bmatrix}\begin{bmatrix} E_i \\ E_j \end{bmatrix} = \begin{bmatrix} P_{i,inp} \\ P_{j,inp} \end{bmatrix} \tag{B.6}$$

For a SEA model composed of *nsub*, the linear system in Eq.B.6, can be general-

Figure B.1 – Representation of a two subsystems SEA model.
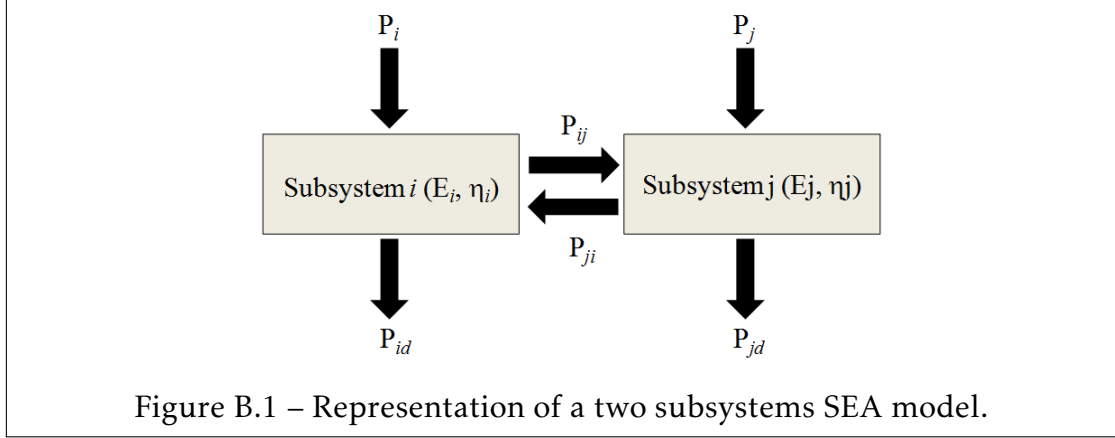
ized to:

$$
\omega
\begin{bmatrix}
\eta_{1,tot} & -\eta_{2,1} & \cdots & -\eta_{nsub,1} \\
-\eta_{1,2} & \eta_{2,tot} & \cdots & -\eta_{nsub,2} \\
\vdots & \vdots & \ddots & \vdots \\
-\eta_{1,nsub} & -\eta_{2,nsub} & \cdots & \eta_{nsub,tot}
\end{bmatrix}
\begin{bmatrix}
E_i \\
E_2 \\
\vdots \\
E_{nsub}
\end{bmatrix}
=
\begin{bmatrix}
P_{1,inp} \\
P_{2,inp} \\
\vdots \\
P_{nsub,inp}
\end{bmatrix}
\tag{B.7}
$$

Two types of subsystem were considered during this thesis work : semi-infinite plates and acoustic cavities. The energies for both subsystems are given by:

- Acoustic subsystem i:

$$
E_i = \frac{p_i^2 V_i}{\rho_i c_i},
\tag{B.8}
$$

where p_i, V_i, ρ_i and c_i denote respectively the sound pressure averaged in the acoustic room, the volume, the density and the wave speed, respectively.

- Structural subsystem i:

$$
E_i = m_i v_i^2,
\tag{B.9}
$$

where m_i and v_i are the mass of the subsystem and the velocity average the plate's surface, respectively.

Three types of connections are considered for the computation of the CLFs: plate-plate line junctions, cavity-plate (resonant coupling) and cavity-cavity (non-resonant coupling) area junctions.

The SEA parameters defined to solve Eq.B.7 are provided in [74, 73, 87, 56, 23].

B.3   Validation and Applications

For comparison to examples that been modeled using other software [123], in this section we consider the following three different test cases that have been modeled by the Free-SEA code [123]. The results will be compared to the ones published by the author.

B.3.1   Three plates connection

This example consists of three plates which are connected at a common line, see Fig.B.2. The three plates are both modeled as a linear elastic material with the following properties:

- Density: $\rho = 7800 kg/m^3$.

- Young modulus: $E = 210.e + 9Pa$.

- Poisson ratio: $\nu = 0.3$.

Each of the plates has a size of 1x1 m. Plates 1 and 3 are 2mm thick, while plate 2 is 3 mm thick. The Plate 1 is connected to plate 2 by an angle of 90 degrees and to plate 3 by an angle of 210 degrees. All Plates have constant damping of 0.01. Plate 1 is excited by a an input power $P_{1,inp} = 1Watt$ (in bending wave).

To goal is to calculate the mean velocity amplitude of the bending waves at all three plates. The results are shown in Fig.B.3.
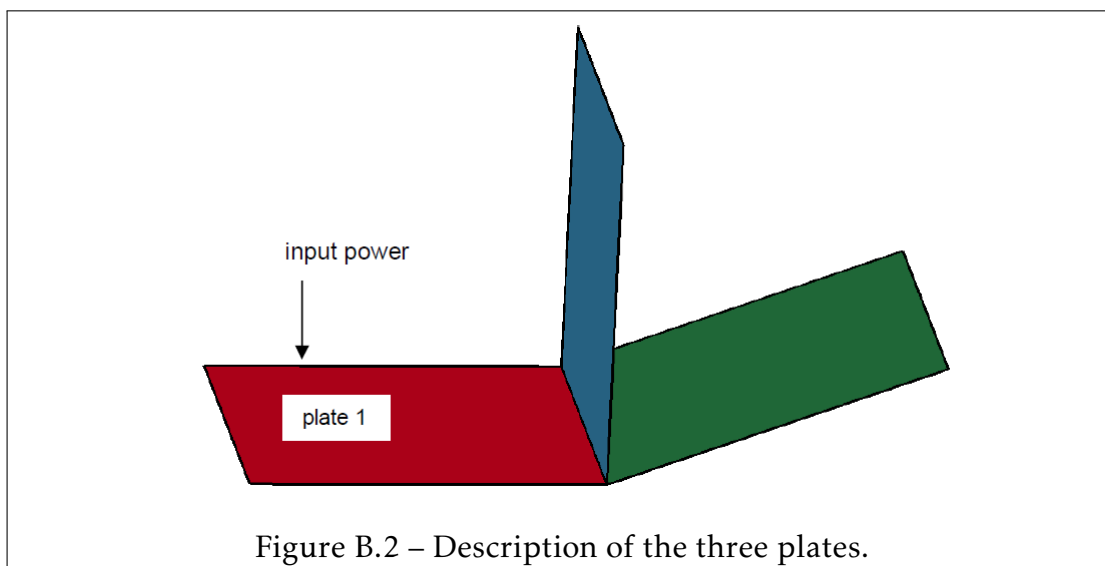
Figure B.2 – Description of the three plates.

Figure B.3 – Comparison of the mean velocity (Log-Log scale) of plates 1 and 2: Free Sea results are in green with triangle symbols and the in-house SEA code results are in red with crossed circle symbols.
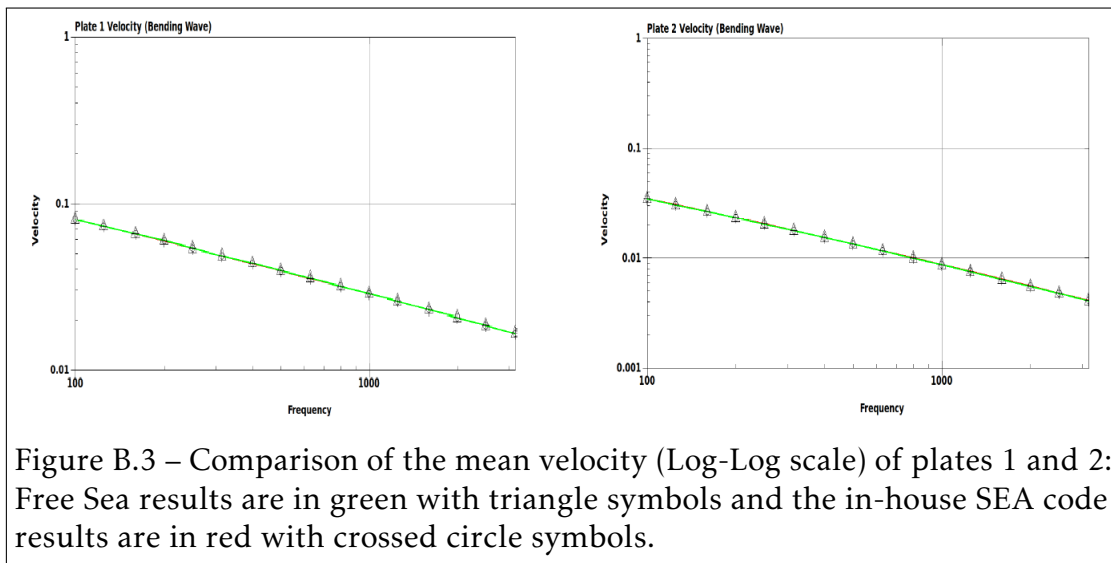
B.3.2   Sound transmission between two rooms

This example consists of thirteen SEA subsystems: eleven plates (walls) and two acoustic rooms, see Fig.B.4.

The eleven plates are all modeled by a linear elastic concrete material, which parameters are the following:

- The thickness is $0.19m$.

- The area is $13.95m^2$.

- Density: $\rho = 2300kg/m^3$.

- Young modulus: $E = 28.e + 9Pa$.

- Poisson ratio: $\nu = 0.3$.

The two rooms are modeled with ambient air material. An acoustic source of input power equal to 1 Watt is applied to room 1.

To goal is to calculate the acoustic sound pressure in both rooms. The results are shown in Fig.B.5.

Figure B.4 – Problem description for the two rooms.

Figure B.5 – Comparison of acoustic pressures (Log-Log scale) in both rooms 1 and 2: Free Sea results are in green and light blue with triangle symbols and the in-house SEA code results are in red and dark blue with crossed circle and square symbols.
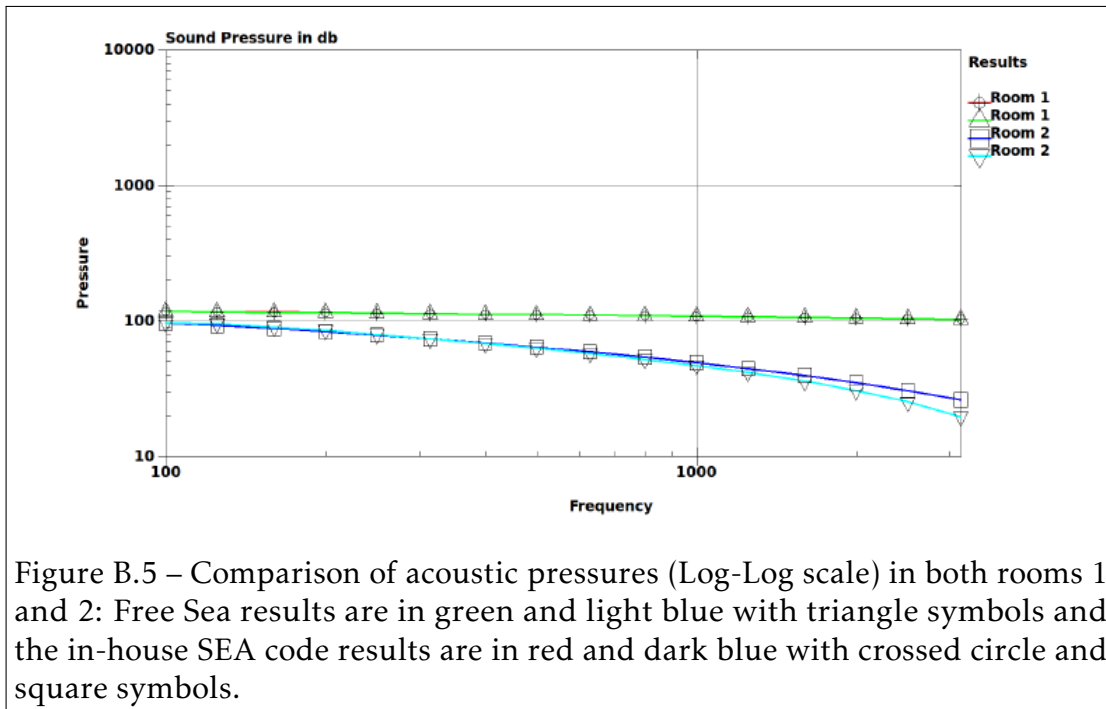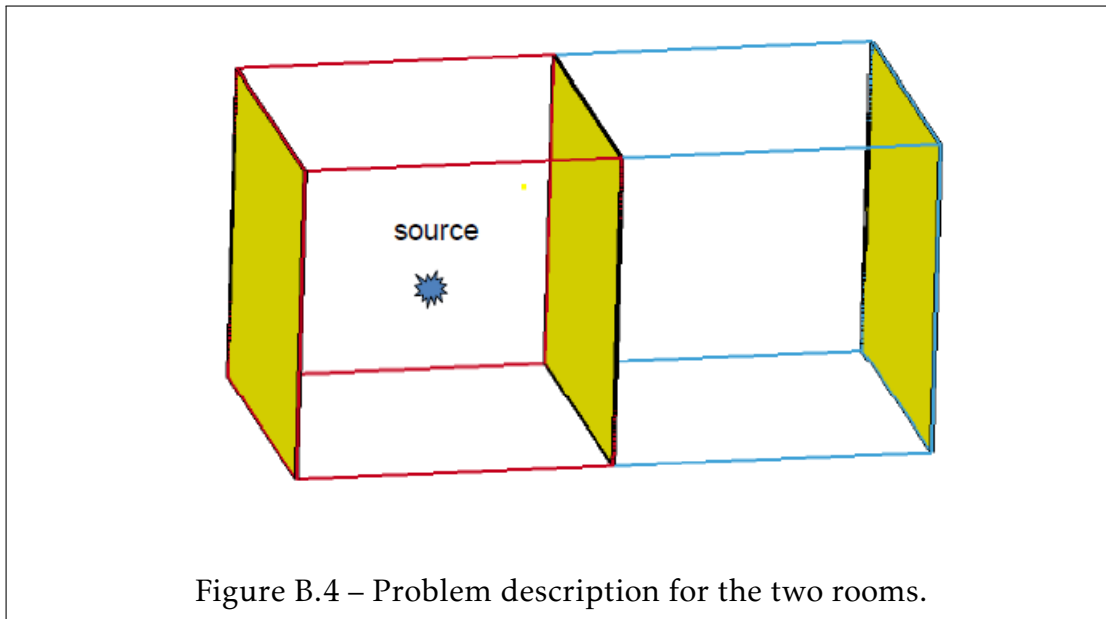
B.3.3   Sound transmission inside a car

The car example consists of twenty-six SEA subsystems: eighteen steel plates, six glass plates and two acoustic rooms, see Fig.B.6.

The steel and glass plates are both modeled by a linear elastic material, which properties are the following:

- Steel density: $\rho_s = 7800 kg/m^3$.

- Steel Young modulus: $E_s = 210.e + 9Pa$.

- Steel Poisson ratio: $\nu_s = 0.3$.

- Glass density: $\rho_g = 2500 kg/m^3$.

- Glass Young modulus: $E_g = 60.e + 9Pa$.

- Glass Poisson ratio: $\nu_g = 0.2$.

A real car would have some sort of a frame, pillars and stiffeners. However, to test and validate the in-house SEA code, we build the simple SEA model shown in Fig.B.6.

The input power is applied in bending wave on the fender left and right front parts of the car as described in Fig.B.6.

The acoustic sound pressure in dB units in the passenger compartment results for both the in-house SEA and FreeSEA codes are shown in Fig.B.7.
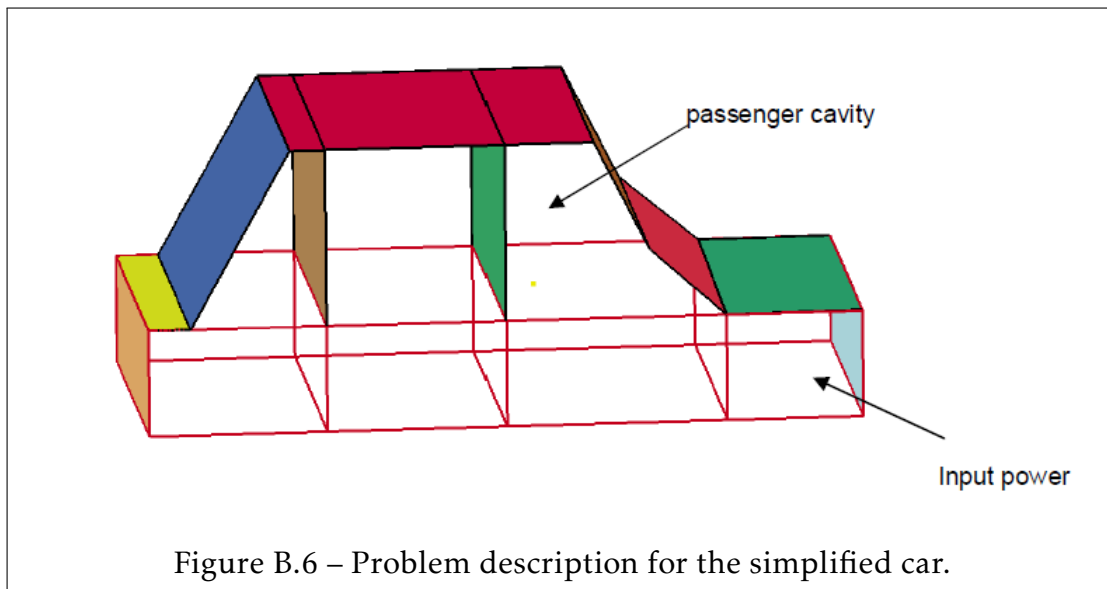
Figure B.6 – Problem description for the simplified car.
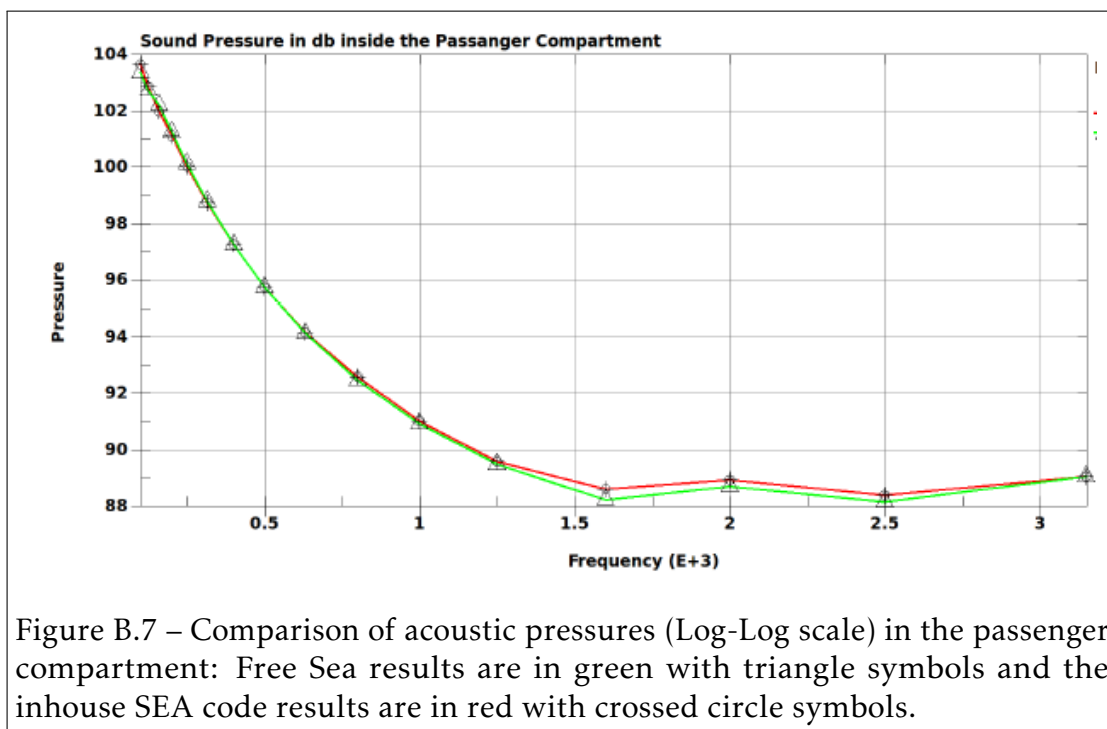
Figure B.7 – Comparison of acoustic pressures (Log-Log scale) in the passenger compartment: Free Sea results are in green with triangle symbols and the inhouse SEA code results are in red with crossed circle symbols.

Appendix C

D3plot Outputs

```fortran
SUBROUTINE WRITE2DYNA(                                         &
&            time,x_sph,vx_sph,dvx_sph,sig_sph,mass_sph,   &
&            rho_sph,p_sph,  u_sph,hsml_sph,eps_sph,        &
&            plastic_sph,itype_sph,countiac,nsph,ntotal,   &
&            nvirt,npart_sph,ngrab,pointer_dyna)

IMPLICIT NONE

INCLUDE '../run_directory/param.inc'

integer, intent(in)            :: nsph,ntotal,nvirt,npart_sph
double precision, intent(in) :: time

! SPH ARRAYS

integer, dimension(mem*nsph), intent(in)          ::   &
&    itype_sph,countiac
double precision, dimension(mem*nsph), intent(in) ::   &
&    mass_sph,rho_sph,p_sph,u_sph,hsml_sph,plastic_sph
double precision, dimension(3,mem*nsph), intent(in) :: &
&    x_sph,vx_sph,dvx_sph
double precision, dimension(6,mem*nsph), intent(in) :: &
&    sig_sph,eps_sph

! SHELL ARRAYS
```

```fortran
27
28 integer, parameter                          :: nnele = 1
29 integer, parameter                          :: nnnode = 1
30 integer, parameter                          :: npart_shell = 0
31 integer, dimension(5,nnele)          :: elenod
32 double precision, dimension(nnele)     :: tz_shell
33 double precision, dimension(6,nnnode) :: x_shell
34 double precision, dimension(6,nnnode) :: vx_shell
35 double precision, dimension(6,nnnode) :: dvx_shell
36 double precision, dimension(8,nnele)   :: fr_shell
37 double precision, dimension(6,5,nnele) :: sig_shell
38
39 integer, intent(inout) :: pointer_dyna,ngrab
40
41 character (LEN=8) :: RELNUM, SUPP
42 character (LEN=8), dimension(10) :: TITLE
43 integer(kind=8)   :: i_tmp, ii, kk
44 double precision  :: r_tmp,EOF
45 character (LEN=8) :: name2
46 character (LEN=9) :: name3
47 integer(kind=8)    :: I,J,JJ,FILENO,K,FILETYPE,SOURCE_VERSION
48 double precision   :: RUNTIME, VERSION,THICKNESS
49 integer(kind=8)    :: JUNK,NDIM,NGLBV,NPOSITION,NFILES
50 integer(kind=8)    :: NUMNP,NMMAT
51 integer(kind=8)    :: IT, IU, IV, IA
52 integer(kind=8)    :: NEIPH, NEIPS
53 integer(kind=8)    :: MAXINT,NGSPH,NUMMAT,NUMFLUID,ID3PROP
54 integer(kind=8)    :: NPEFG,NVEFG,IDTDT,NUMRBS
55
56 !NODE AND ELMENT NUMBERING
57
58 integer(kind=8)    :: ICODE,NEL8,NUMMAT8,NV3D,NUMBRE
59 integer(kind=8)    :: NEL2,NUMMAT2,NV1D,NEL4,NUMMAT4,NV2D
60 integer(kind=8)    :: NMSPH,NGPSPH,NARBS,NELT,NUMMATT,NV3DT
61 integer, dimension(4)  :: IOSHL
62 integer(kind=8)    :: IALEMAT,NCFDV1,NCFDV2,NADAPT,MATTYP
63
64 !SPH ELEMENT FLAG
65
66 integer(kind=8), dimension(11)  :: ISPHFG
```

```fortran
67
68  !FLUID MATERIAL ID DATA
69
70  integer(kind=8), allocatable, dimension(:)  :: FLUIDID
71
72  !SPH NODE AND MATERIAL LIST
73
74  integer(kind=8), allocatable, dimension(:,:) :: NUMSPH
75
76  !MATERIAL TYPE DATA
77
78  integer(kind=8), allocatable, dimension(:)  :: IRBTYP
79
80  !GEOMETRY DATA
81
82  integer(kind=8), allocatable, dimension(:,:)  :: IX8,IX82,IXT,IX2,IX4
83
84  ! Read State Data
85
86  double precision :: CURRENT_TIME,KE,IE,TE
87  double precision :: X_GLB,Y_GLB,Z_GLB,EXT_W,SCALE_K
88
89  double precision,allocatable,dimension(:)::X_MAT,Y_MAT,Z_MAT
90  double precision,allocatable,dimension(:)::IE_MAT,KE_MAT,MASS_MAT
91
92  ! Read  Element Data
93
94  integer(kind=8)::ISTRN,Max_size,global_size,shell_size,sph_size
95
96  ! PART TITLE Data
97
98  integer(kind=8) :: NTYPE, NUMPROP, IDP
99  character (LEN=8), dimension(9) :: PTITLE, HEAD
100 integer(kind=8) :: SUCCESS, DEBUG
101 integer :: factor, nele, nnode
102
103 ! SHELL NOT IMPLEMENTED YET !!!
104
105 elenod    = 0.d0
106 tz_shell  = 0.d0
```

```fortran
107 x_shell   = 0.d0; vx_shell = 0.d0 ; dvx_shell = 0.d0
108 fr_shell  = 0.d0
109 sig_shell = 0.d0
110
111 DEBUG = 0
112 EOF = -999999.0
113
114 nele  = 0
115 nnode = 0
116
117 IF ((ngrab.NE.1).OR.(pointer_dyna.NE.0)) GO TO 111
118
119 !OPEN THE LSDYNA FILE
120
121 ! IFORT VERSION
122 !OPEN(UNIT=12,FILE='d3plot',FORM='BINARY',  &
123 !&    ACCESS='DIRECT',recl=8)
124
125 ! GFORTRAN AND PGF90 VERSION
126 OPEN(UNIT=12,FILE='d3plot',FORM='UNFORMATTED',  &
127 &     ACCESS='DIRECT',recl=8)
128
129 TITLE(1)="SPH_LML ";TITLE(2)="CODE RES";TITLE(3)="ULTS    ";
130 DO i = 4,10
131 TITLE(i)="        "
132 ENDDO
133
134 ii=0 ;
135 !Read Title
136 DO i=1,10
137 ii = ii + 1
138 write(12,rec=ii) TITLE(i)
139 ENDDO
140
141 !Read computer id, code name, version
142 RUNTIME=duration
143 ii = ii + 1
144 write(12,rec=ii) RUNTIME
145 FILETYPE = 1
146
```

```
147  ii = ii + 1
148  write(12,rec=ii) FILETYPE
149  SOURCE_VERSION=971065543
150  ii = ii + 1
151  write(12,rec=ii) SOURCE_VERSION
152  RELNUM="511␣"
153  ii = ii + 1
154  write(12,rec=ii) RELNUM
155  VERSION=960.0000
156  ii = ii + 1
157  write(12,rec=ii) VERSION
158  !Read NDIM, NNODES, ICODE
159  NDIM = 4
160  ii = ii + 1
161  write(12,rec=ii) NDIM
162  NUMNP = mem*nsph+nnode
163  ii = ii + 1
164  write(12,rec=ii) NUMNP
165  ICODE = 6
166  ii = ii + 1
167  write(12,rec=ii) ICODE
168
169  !Number of Global variables, IT, IU, IV, IA
170
171  NGLBV = 6 + 7*(npart_sph+npart_shell+1)
172  ii = ii + 1
173  write(12,rec=ii) NGLBV
174  IT = 0
175  ii = ii +1
176  write(12,rec=ii) IT
177  IU = 1
178  ii = ii +1
179  write(12,rec=ii) IU
180  IV = 1
181  ii = ii +1
182  write(12,rec=ii) IV
183  IA = 1
184  ii = ii +1
185  write(12,rec=ii) IA
186
```

```
187  !NEL8,NUMMAT8, BLANk, BLANK, NV3D
188
189  NEL8=0
190  ii = ii +1
191  write(12,rec=ii) NEL8
192  NUMMAT8=0
193  ii = ii +1
194  write(12,rec=ii) NUMMAT8
195  JUNK=0
196  ii = ii + 1
197  write(12,rec=ii) JUNK
198  ii = ii + 1
199  write(12,rec=ii) JUNK
200  NV3D = 7
201  ii = ii + 1
202  write(12,rec=ii) NV3D
203
204  !NEL2,NUMMAT2, BLANk, BLANK, NV1D
205
206  NEL2 = 0
207  ii = ii + 1
208  write(12,rec=ii) NEL2
209  NUMMAT2 = 0
210  ii = ii + 1
211  write(12,rec=ii) NUMMAT2
212  NV1D = 6
213  ii = ii + 1
214  write(12,rec=ii) NV1D
215
216  !NEL4,NUMMAT4, BLANk, BLANK, NV2D
217
218  NEL4 = nele
219  ii = ii + 1
220  write(12,rec=ii) NEL4
221  NUMMAT4 = npart_shell
222  ii = ii + 1
223  write(12,rec=ii) NUMMAT4
224  NV2D = 33
225  ii = ii + 1
226  write(12,rec=ii) NV2D
```

```
227
228  !NEIPH
229  NEIPH = 0
230  ii = ii + 1
231  write(12,rec=ii) NEIPH
232
233  !NEIPS,MAXINT
234  NEIPS = 0
235  ii = ii + 1
236  write(12,rec=ii) NEIPS
237  MAXINT = -10003
238  ii = ii + 1
239  write(12,rec=ii) MAXINT
240  NMSPH = mem*nsph
241  ii = ii + 1
242  write(12,rec=ii) NMSPH
243  NGSPH = npart_sph + 1
244  ii = ii + 1
245  write(12,rec=ii) NGSPH
246  NARBS = 0
247  ii = ii + 1
248  write(12,rec=ii) NARBS
249
250  !NELT,NUMMATT,NV3D
251  NELT = 0
252  ii = ii + 1
253  write(12,rec=ii) NELT
254  NUMMAT = 0
255  ii = ii + 1
256  write(12,rec=ii) NUMMATT
257  NV3D = 21
258  ii = ii + 1
259  write(12,rec=ii) NV3D
260
261  !IOSH
262
263  DO i=1,4
264  IOSHL(i) =1000
265  ENDDO
266
```

```
267  ii = ii + 1
268  write(12,rec=ii) IOSHL(1)
269  ii = ii + 1
270  write(12,rec=ii) IOSHL(2)
271  ii = ii + 1
272  write(12,rec=ii) IOSHL(3)
273  ii = ii + 1
274  write(12,rec=ii) IOSHL(4)
275
276  !IALEMAT,NCFDV1,NCFDV2,NADAPT,NMMAT,NUMFLUID
277
278  IALEMAT = 0
279  ii = ii + 1
280  write(12,rec=ii) IALEMAT
281  NCFDV1 = 0
282  ii = ii + 1
283  write(12,rec=ii) NCFDV1
284  NCFDV2 = 0
285  ii = ii + 1
286  write(12,rec=ii) NCFDV2
287  NADAPT = 0
288  ii = ii + 1
289  write(12,rec=ii) NADAPT
290  NMMAT = npart_shell + npart_sph + 1
291  ii = ii + 1
292  write(12,rec=ii) NMMAT
293  NUMFLUID = 0
294  ii = ii + 1
295  write(12,rec=ii) NUMFLUID
296
297  !ID3PROP,NPEFG,NVEFG,IDTDT
298
299  ID3PROP = 0
300  ii = ii + 1
301  write(12,rec=ii) ID3PROP
302  NPEFG = 0
303  ii = ii + 1
304  write(12,rec=ii) NPEFG
305  NVEFG = 0
306  ii = ii + 1
```

```
307 write(12,rec=ii) NVEFG
308 IDTDT = 0
309 ii = ii + 1
310 write(12,rec=ii) IDTDT
311
312 !BLANK,....
313
314 DO i=58,64
315 ii = ii + 1
316 write(12,rec=ii) JUNK
317 ENDDO
318
319 !SPH ELEMENT Data Flag
320
321 IF (NMSPH.GT.0) THEN
322 ! WRITE(*,*)"=============================="
323 ! WRITE(*,*)"Reading SPH Element Flag  Data"
324 ! WRITE(*,*)"=============================="
325
326 ISPHFG(1)=11
327 ISPHFG(2)=1
328 ISPHFG(3)=1
329 ISPHFG(4)= 6
330 ISPHFG(5)=1
331 ISPHFG(6)=1
332 ISPHFG(7)= 1
333 ISPHFG(8)=1
334 ISPHFG(9)=6
335 ISPHFG(10)=1
336 ISPHFG(11)=1
337
338 DO i=1,11
339 ii=ii+1
340 write(12,rec=ii) ISPHFG(i)
341 ENDDO
342
343 ENDIF
344
345 ! WRITE(*,*)"====================="
346 ! WRITE(*,*)"Reading Geometry Data"
```

```
347  ! WRITE(*,*)"===================="
348
349  ! Nodal Coordinates
350
351  DO k = 1,nsph
352  ii=ii+1
353  write(12,rec=ii)   x_sph(1,k)
354  ii=ii+1
355  write(12,rec=ii)   x_sph(2,k)
356  ii=ii+1
357  write(12,rec=ii)   x_sph(3,k)
358  ENDDO
359
360  DO k = nsph+1,mem*nsph
361  ii=ii+1
362  write(12,rec=ii)   x_sph(1,k)
363  ii=ii+1
364  write(12,rec=ii)   x_sph(2,k)
365  ii=ii+1
366  write(12,rec=ii)   x_sph(3,k)
367  ENDDO
368
369  DO k = 1,nnode
370  ii=ii+1
371  write(12,rec=ii)   x_shell(1,k)
372  ii=ii+1
373  write(12,rec=ii)   x_shell(2,k)
374  ii=ii+1
375  write(12,rec=ii)   x_shell(3,k)
376  ENDDO
377
378  !SHELL CONNECTIVITY
379
380  IF (nnode.GT.0) THEN
381  ALLOCATE(IX4(5,NEL4))
382  DO i = 1, NEL4
383  DO k=1,4
384  IX4(k,i)=elenod(k,i)
385  ENDDO
386  ENDDO
```

```
387 IX4(5,1:NEL4)=3
388 DO i = 1, NEL4
389 DO k= 1,4
390 ii=ii+1
391 write(12,rec=ii) IX4(k,i)
392 ENDDO
393 ii=ii+1
394 write(12,rec=ii) IX4(5,i)
395 ENDDO
396 DEALLOCATE(IX4)
397 ENDIF
398
399 ! WRITE(*,*)"==============================="
400 ! WRITE(*,*)"   SPH NODE AND MATERIAL LIST   "
401 ! WRITE(*,*)"==============================="
402
403 IF(NMSPH.GT.0) THEN
404
405 k = 0
406 DO i = 1, nsph
407 k = k + 1
408 ii=ii+1
409 write(12,rec=ii) k
410 ii=ii+1
411 write(12,rec=ii) itype_sph(i)
412 !write(12,rec=ii) 1
413 ENDDO
414 DO i = nsph+1, mem*nsph
415 k = k + 1
416 ii=ii+1
417 write(12,rec=ii) k
418 ii=ii+1
419 write(12,rec=ii) npart_sph+1
420 ENDDO
421 ENDIF
422
423 ii=ii+1
424 write(12,rec=ii) EOF
425
426 NTYPE = 90001
```

```
427  ii=ii+1
428  write(12,rec=ii) NTYPE
429  NUMPROP = npart_shell+npart_sph+1
430  ii=ii+1
431  write(12,rec=ii) NUMPROP
432  DO i=1,NUMPROP
433  IDP= i
434  ii=ii+1
435  write(12,rec=ii) IDP
436  DO k=1,9
437  PTITLE(k)="_____"
438  ii=ii+1
439  write(12,rec=ii) PTITLE(k)
440  ENDDO
441  ENDDO
442
443  NTYPE = 90000
444  ii=ii+1
445  write(12,rec=ii) NTYPE
446
447  DO i=1,9
448  HEAD(i)="_____"
449  ii=ii+1
450  write(12,rec=ii) HEAD(i)
451  ENDDO
452
453  ii=ii+1
454  write(12,rec=ii)    EOF
455
456  DO WHILE (mod(ii,512).ne.0)
457  ii=ii+1
458  write(12,rec=ii)    r_tmp
459  ENDDO
460
461  CLOSE(12)
462
463  111   NMMAT = npart_shell+npart_sph+1
464  NEL4  = nele
465  NUMNP = mem*nsph+nnode
466  NMSPH = mem*nsph
```

```fortran
467
468 ! Maximum word for a file
469 factor = 1000
470 max_size = factor * 7 * 512 * 512
471 global_size = 8+NMMAT*6+NUMNP*9
472 shell_size = 34 * NEL4
473 sph_size   = 21 * NMSPH
474
475 FILENO = ngrab
476
477 IF (FILENO.LE.99) THEN
478
479 write(SUPP,'(i2.2)') FILENO
480 name2='d3plot'//SUPP
481 ! WRITE(*,*)"==============================="
482 ! WRITE(*,*)"Writing State Data in "//name2
483 ! WRITE(*,*)"==============================="
484
485 IF (pointer_dyna.EQ.0) THEN
486 ! IFORT
487 !OPEN(UNIT=12+FILENO,FILE=name2,FORM='BINARY',      &
488 !&     ACCESS='DIRECT',recl=8)
489 ! GFORTRAN or PGF90
490 OPEN(UNIT=12+FILENO,FILE=name2,FORM='UNFORMATTED', &
491 &     ACCESS='DIRECT',recl=8)
492 ELSE
493 ! IFORT
494 !OPEN(UNIT=12+FILENO,FILE=name2,FORM='BINARY',      &
495 !&     ACCESS='DIRECT',STATUS='OLD',recl=8)
496 ! GFORTRAN or PGF90
497 OPEN(UNIT=12+FILENO,FILE=name2,FORM='UNFORMATTED', &
498 &     ACCESS='DIRECT',STATUS='OLD'recl=8)
499 ENDIF
500
501 ELSEIF ((FILENO.GT.99).AND.(FILENO.LE.999)) THEN
502 write(SUPP,'(i3.3)') FILENO
503 name3='d3plot'//SUPP
504 ! WRITE(*,*)"==============================="
505 ! WRITE(*,*)"Writing State Data in "//name3
506 ! WRITE(*,*)"==============================="
```

```fortran
507
508  IF (pointer_dyna.EQ.0) THEN
509  ! IFORT
510  !OPEN(UNIT=12+FILENO,FILE=name3,FORM='BINARY',       &
511  !&    ACCESS='DIRECT',recl=8)
512  ! GFORTRAN or PGF90
513  OPEN(UNIT=12+FILENO,FILE=name3,FORM='UNFORMATTED', &
514  &     ACCESS='DIRECT',recl=8)
515  ELSE
516  ! IFORT
517  !OPEN(UNIT=12+FILENO,FILE=name3,FORM='BINARY',       &
518  !&    ACCESS='DIRECT',STATUS='OLD',recl=8)
519  ! GFORTRAN or PGF90
520  OPEN(UNIT=12+FILENO,FILE=name3,FORM='UNFORMATTED', &
521  &     ACCESS='DIRECT',STATUS='OLD'recl=8)
522  ENDIF
523
524  ENDIF
525
526  IF((pointer_dyna+global_size+shell_size+sph_size+1) &
527  &    .GT.max_size)THEN
528
529  ii=pointer_dyna + 1
530  write(12+FILENO,rec=ii)    EOF
531  DO WHILE (mod(ii,512).ne.0)
532  ii=ii+1
533  r_tmp=0.d0
534  write(12+FILENO,rec=ii)    r_tmp
535  ENDDO
536  CLOSE(12+FILENO)
537  ngrab = ngrab + 1
538  pointer_dyna = 0
539  GO TO 111
540
541  ENDIF
542
543  ! Reinitialisation of the rec position
544  ii=pointer_dyna
545
546  ! Trash  double precision variable r_tmp
```

```
547  r_tmp=0.d0
548
549  ALLOCATE(IE_MAT(NMMAT))
550  ALLOCATE(KE_MAT(NMMAT))
551  ALLOCATE(X_MAT(NMMAT),Y_MAT(NMMAT),Z_MAT(NMMAT))
552  ALLOCATE(MASS_MAT(NMMAT))
553
554  ii=ii+1
555  CURRENT_TIME = time
556  write(12+FILENO,rec=ii) CURRENT_TIME
557  ii=ii+1
558  KE = 0.d0
559  write(12+FILENO,rec=ii) KE
560  ii=ii+1
561  IE = 0.d0
562  write(12+FILENO,rec=ii) IE
563  ii=ii+1
564  TE = 0.d0
565  write(12+FILENO,rec=ii) TE
566  ii=ii+1
567  X_GLB = 0.d0
568  write(12+FILENO,rec=ii) X_GLB
569  ii=ii+1
570  Y_GLB = 0.d0
571  write(12+FILENO,rec=ii) Y_GLB
572  ii=ii+1
573  Z_GLB = 0.d0
574  write(12+FILENO,rec=ii) Z_GLB
575  DO i = 1, NMMAT
576  ii=ii+1
577  IE_MAT(i) = 0.d0
578  write(12+FILENO,rec=ii) IE_MAT(i)
579  ENDDO
580  DO i = 1, NMMAT
581  ii=ii+1
582  KE_MAT(i) = 0.d0
583  write(12+FILENO,rec=ii) KE_MAT(i)
584  ENDDO
585  DO i = 1, NMMAT
586  ii=ii+1
```

```
587 X_MAT(i) = 0.d0
588 write(12+FILENO,rec=ii) X_MAT(i)
589 ii=ii+1
590 Y_MAT(i) = 0.d0
591 write(12+FILENO,rec=ii) Y_MAT(i)
592 ii=ii+1
593 Z_MAT(i) = 0.d0
594 write(12+FILENO,rec=ii) Z_MAT(i)
595 ENDDO
596 DO i = 1, NMMAT
597 ii=ii+1
598 MASS_MAT(i) = 0.d0
599 write(12+FILENO,rec=ii) MASS_MAT(i)
600 ENDDO
601 DO i = 1, NMMAT
602 ii=ii+1
603 write(12+FILENO,rec=ii) r_tmp
604 ENDDO
605
606 DEALLOCATE(X_MAT, Y_MAT, Z_MAT, IE_MAT, KE_MAT, MASS_MAT)
607
608 ! WRITE(*,*)"================="
609 ! WRITE(*,*)"Writing Node Data"
610 ! WRITE(*,*)"================="
611
612 ! Nodal Data
613
614 ! Writting Nodal coordinates (current time)
615 DO k = 1, nsph
616 ii=ii+1
617 write(12+FILENO,rec=ii)  x_sph(1,k)
618 ii=ii+1
619 write(12+FILENO,rec=ii)  x_sph(2,k)
620 ii=ii+1
621 write(12+FILENO,rec=ii)  x_sph(3,k)
622 ENDDO
623 DO k = nsph+1, mem*nsph
624 ii=ii+1
625 write(12+FILENO,rec=ii)  x_sph(1,k)
626 ii=ii+1
```

```
627  write(12+FILENO,rec=ii)  x_sph(2,k)
628  ii=ii+1
629  write(12+FILENO,rec=ii)  x_sph(3,k)
630  ENDDO
631  DO k = 1, nnode
632  ii=ii+1
633  write(12+FILENO,rec=ii)  x_shell(1,k)
634  ii=ii+1
635  write(12+FILENO,rec=ii)  x_shell(2,k)
636  ii=ii+1
637  write(12+FILENO,rec=ii)  x_shell(3,k)
638  ENDDO
639
640  ! Writting Nodal Velocities (current time)
641  DO k = 1, nsph
642  ii=ii+1
643  write(12+FILENO,rec=ii)  vx_sph(1,k)
644  ii=ii+1
645  write(12+FILENO,rec=ii)  vx_sph(2,k)
646  ii=ii+1
647  write(12+FILENO,rec=ii)  vx_sph(3,k)
648  ENDDO
649  DO k = nsph+1, mem*nsph
650  ii=ii+1
651  write(12+FILENO,rec=ii)  vx_sph(1,k)
652  ii=ii+1
653  write(12+FILENO,rec=ii)  vx_sph(2,k)
654  ii=ii+1
655  write(12+FILENO,rec=ii)  vx_sph(3,k)
656  ENDDO
657  DO k = 1, nnode
658  ii=ii+1
659  write(12+FILENO,rec=ii)  vx_shell(1,k)
660  ii=ii+1
661  write(12+FILENO,rec=ii)  vx_shell(2,k)
662  ii=ii+1
663  write(12+FILENO,rec=ii)  vx_shell(3,k)
664  ENDDO
665
666  ! Writting Nodal Acceleration (current time)
```

```
667  DO k = 1, nsph
668  ii=ii+1
669  write(12+FILENO,rec=ii)  dvx_sph(1,k)
670  ii=ii+1
671  write(12+FILENO,rec=ii)  dvx_sph(2,k)
672  ii=ii+1
673  write(12+FILENO,rec=ii)  dvx_sph(3,k)
674  ENDDO
675  DO k = nsph+1, mem*nsph
676  ii=ii+1
677  write(12+FILENO,rec=ii)  dvx_sph(1,k)
678  ii=ii+1
679  write(12+FILENO,rec=ii)  dvx_sph(2,k)
680  ii=ii+1
681  write(12+FILENO,rec=ii)  dvx_sph(3,k)
682  ENDDO
683  DO k = 1, nnode
684  ii=ii+1
685  write(12+FILENO,rec=ii)  dvx_shell(1,k)
686  ii=ii+1
687  write(12+FILENO,rec=ii)  dvx_shell(2,k)
688  ii=ii+1
689  write(12+FILENO,rec=ii)  dvx_shell(3,k)
690  ENDDO
691
692  ! WRITE(*,*)"========================="
693  ! WRITE(*,*)"Reading Shell Element Data"
694  ! WRITE(*,*)"========================="
695
696  ! Element Data
697  ISTRN = 0
698
699  DO k = 1, NEL4 ! LOOP ON SHELL ELEMENTS
700
701  DO jj = 1,3 ! LOOP ON INTEGRATION POINT THROUGH THICKNESS
702  ii=ii+1
703  write(12+FILENO,rec=ii) sig_shell(1,jj,k)
704  ii=ii+1
705  write(12+FILENO,rec=ii) sig_shell(2,jj,k)
706  ii=ii+1
```

```
707 write(12+FILENO,rec=ii) sig_shell(6,jj,k)
708 ii=ii+1
709 write(12+FILENO,rec=ii) sig_shell(3,jj,k)
710 ii=ii+1
711 write(12+FILENO,rec=ii) sig_shell(5,jj,k)
712 ii=ii+1
713 write(12+FILENO,rec=ii) sig_shell(4,jj,k)
714 ii=ii+1
715 write(12+FILENO,rec=ii)   r_tmp
716 ENDDO ! END LOOP ON INTEGRATION POINTS THROUGH THE THICKNESS
717
718 ii=ii+1
719 write(12+FILENO,rec=ii) -fr_shell(6,k)
720 ii=ii+1
721 write(12+FILENO,rec=ii) -fr_shell(7,k)
722 ii=ii+1
723 write(12+FILENO,rec=ii) -fr_shell(8,k)
724 ii=ii+1
725 write(12+FILENO,rec=ii)  fr_shell(4,k)
726 ii=ii+1
727 write(12+FILENO,rec=ii)  fr_shell(5,k)
728 ii=ii+1
729 write(12+FILENO,rec=ii)  fr_shell(1,k)
730 ii=ii+1
731 write(12+FILENO,rec=ii)  fr_shell(2,k)
732 ii=ii+1
733 write(12+FILENO,rec=ii)  fr_shell(3,k)
734 ii=ii+1
735 write(12+FILENO,rec=ii)  tz_shell(k)
736 ii=ii+1
737 write(12+FILENO,rec=ii)   r_tmp
738 ii=ii+1
739 write(12+FILENO,rec=ii)   r_tmp
740 ii=ii+1
741 IE = r_tmp
742 write(12+FILENO,rec=ii) IE
743
744 ENDDO ! END LOOP ON SHELL ELEMENTS
745
746 DO k = 1,NEL4
```

```
747  ii=ii+1
748  r_tmp=1.d0
749  write(12+FILENO,rec=ii)    r_tmp
750  ENDDO
751
752  DO k = 1,nsph ! LOOP ON SPH ELEMENTS
753  ii=ii+1
754  write(12+FILENO,rec=ii) itype_sph(k)
755  ii=ii+1
756  write(12+FILENO,rec=ii) hsml_sph(k)
757  ii=ii+1
758  write(12+FILENO,rec=ii) p_sph(k)
759  ii=ii+1
760  write(12+FILENO,rec=ii) sig_sph(1,k)
761  ii=ii+1
762  write(12+FILENO,rec=ii) eps_sph(1,k)
763  ii=ii+1
764  write(12+FILENO,rec=ii) eps_sph(2,k)
765  ii=ii+1
766  write(12+FILENO,rec=ii) mass_sph(k) ! sig_xy(k) replaced
767  ii=ii+1
768  write(12+FILENO,rec=ii) hsml_sph(k) ! sig_yz(k) replaced
769  ii=ii+1
770  write(12+FILENO,rec=ii) u_sph(k) ! sig_zx(k) replaced
771  ii=ii+1
772  write(12+FILENO,rec=ii) rho_sph(k) ! plastic_sph(k)
773  ii=ii+1
774  write(12+FILENO,rec=ii) rho_sph(k)
775  ii=ii+1
776  write(12+FILENO,rec=ii) u_sph(k)
777  ii=ii+1
778  write(12+FILENO,rec=ii) countiac(k) ! number of neighb
779  DO  i = 1,6
780  ii=ii+1
781  write(12+FILENO,rec=ii) eps_sph(i,k)
782  ENDDO
783  ii=ii+1
784  write(12+FILENO,rec=ii) mass_sph(k)
785  ii=ii+1
786  write(12+FILENO,rec=ii) r_tmp ! ?????
```

```
787 ENDDO ! END LOOP ON SPH ELEMENTS
788
789 DO k = nsph+1,mem*nsph ! LOOP ON SPH ELEMENTS
790 ii=ii+1
791 write(12+FILENO,rec=ii) npart_sph+1
792 ii=ii+1
793 write(12+FILENO,rec=ii) hsml_sph(k)
794 ii=ii+1
795 write(12+FILENO,rec=ii) p_sph(k)
796 DO  i = 1,3
797 ii=ii+1
798 write(12+FILENO,rec=ii) sig_sph(i,k)
799 ENDDO
800 ii=ii+1
801 write(12+FILENO,rec=ii) mass_sph(k) ! sig_xy(k) replaced
802 ii=ii+1
803 write(12+FILENO,rec=ii) hsml_sph(k) ! sig_yz(k) replaced
804 ii=ii+1
805 write(12+FILENO,rec=ii) u_sph(k)    ! sig_zx(k) replaced
806 ii=ii+1
807 write(12+FILENO,rec=ii) rho_sph(k)  ! plastic_sph(k) replaced
808 ii=ii+1
809 write(12+FILENO,rec=ii) rho_sph(k)
810 ii=ii+1
811 write(12+FILENO,rec=ii) u_sph(k)
812 ii=ii+1
813 write(12+FILENO,rec=ii) countiac(k)  ! number of neighbors
814 DO  i = 1,6
815 ii=ii+1
816 write(12+FILENO,rec=ii) eps_sph(i,k)
817 ENDDO
818 ii=ii+1
819 write(12+FILENO,rec=ii) mass_sph(k)
820 ii=ii+1
821 write(12+FILENO,rec=ii) r_tmp ! ?????
822 ENDDO ! END LOOP ON SPH ELEMENTS
823
824 pointer_dyna = ii
825
826 IF (time.GE.duration) THEN
```

```
827
828 ! LAST RECORD OF A LOOP IF NOT EOF THEN
829 ! r_tmp= CURRENT_TIME AT NEXT LOOP
830
831 ii=pointer_dyna + 1
832 write(12+FILENO,rec=ii)    EOF
833 DO WHILE (mod(ii,512).ne.0)
834 ii=ii+1
835 r_tmp=0.d0
836 write(12+FILENO,rec=ii)    r_tmp
837 ENDDO
838 ENDIF
839
840 CLOSE(12+FILENO)
841
842 RETURN
843 END SUBROUTINE WRITE2DYNA
```

Contents